Aix*Marseille université
Socialement engagée

NNT/NL : 2023AIXM0355/011ED184

THÈSE DE DOCTORAT

Soutenue à Aix-Marseille Université
le 24 octobre 2023 par

Julie PARREAUX

Weighted Timed Games: Decidability, Randomness and Robustness

Discipline
Informatique

École doctorale
ED 184 Mathématiques et Informatique

Laboratoire/Partenaires de recherche
Laboratoire d'Informatique et Systèmes

Composition du jury

Orna KUPFERMAN Hebrew University, Israël	Rapporteure
Didier LIME École Centrale de Nantes - LS2N	Rapporteur
Patricia BOUYER-DECITRE CNRS - LMF	Examinatrice
Hugo GIMBERT CNRS - LaBRI	Examinateur
Jeremy SPROSTON Università degli Studi di Torino, Italie	Examinateur
Eugene ASARIN Université Paris Cité - IRIF	Président du jury
Pierre-Alain REYNIER Aix-Marseille Université - LIS	Directeur de thèse
Benjamin MONMEGE Aix-Marseille Université - LIS	Codirecteur de thèse

LIS
LABORATOIRE D'INFORMATIQUE & SYSTÈMES
UMR 7020

Affidavit

I, undersigned, Julie Parreaux, hereby declare that the work presented in this manuscript is my own work, carried out under the scientific direction of Pierre-Alain Reynier and Benjamin Monmege, in accordance with the principles of honesty, integrity and responsibility inherent to the research mission. The research work and the writing of this manuscript have been carried out in compliance with both the french national charter for Research Integrity and the Aix-Marseille University charter on the fight against plagiarism.

This work has not been submitted previously either in this country or in another country in the same or in a similar version to any other examination body.

Marseille, 23/06/2023



Cette œuvre est mise à disposition selon les termes de la Licence Creative Commons Attribution - Pas d'Utilisation Commerciale - Pas de Modification 4.0 International.

Liste de publications et participation aux conférences

Liste des publications réalisées dans le cadre du projet de thèse:

[MPR20] Benjamin Monmege, Julie Parreaux, and Pierre-Alain Reynier. "Reaching Your Goal Optimally by Playing at Random with No Memory". In: *31st International Conference on Concurrency Theory, CONCUR 2020, September 1-4, 2020, Vienna, Austria (Virtual Conference)*. Ed. by Igor Konnov and Laura Kovács. Vol. 171. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020, 26:1–26:21. DOI: 10.4230/LIPICS.CONCUR.2020.26. URL: https://doi.org/10.4230/LIPIcs.CONCUR.2020.26.

[MPR23a] Benjamin Monmege, Julie Parreaux, and Pierre-Alain Reynier. "Decidability of One-Clock Weighted Timed Games with Arbitrary Weights". Under submission in LMCS, special issue CONCUR 2022. 2023. arXiv: 2207.01608.

[MPR23b] Benjamin Monmege, Julie Parreaux, and Pierre-Alain Reynier. "Playing Stochastically in Weighted Timed Games to Emulate Memory". Under submission in LMCS. 2023. arXiv: 2105.00984.

Participation aux conférences et écoles d'été au cours de la période de thèse:

Conférences internationales avec présentation

1. CONCUR'20 (online): "Reaching Your Goal Optimally by Playing at Random with No Memory"
2. ICALP'21 (online): "Playing Stochastically in Weighted Timed Games to Emulate Memory"

3. Highlights'21 (online): "Playing Stochastically in Weighted Timed Games to Emulate Memory"

4. CONCUR'22 (online): "Decidability of One-Clock Weighted Timed Games with Arbitrary Weights"

Conférences internationales

1. Highlights'20 (online)

2. STACS'21 (online)

3. WATA'21 (online)

4. STACS'22 (online)

5. Highlights'22 (Paris)

Écoles internationales avec présentation

1. MOVEP'22 (Aalborg, Danemark): "Playing Stochastically in Weighted Timed Games to Emulate Memory"

2. Les mathématiques discrètes et la logique (Marseille): "Playing Stochastically in Weighted Timed Games to Emulate Memory"

Réunions nationales avec présentation

1. ANR TickTac 2021 (online): "Reaching Your Goal Optimally by Playing at Random with No Memory"

2. GT Vérif (Paris): "Playing Stochastically in Weighted Timed Games to Emulate Memory"

3. ANR TickTac 2022 (online): "Playing Stochastically in Weighted Timed Games to Emulate Memory"

4. ANR TickTac 2023 (Paris): "Robustness in Weighted Timed Games"

Exposé de vulgarisation scientifique

1. École des cigales printemps 2023 (Marseille): "Théorie des jeux : de la vérification à la synthèse de systèmes informatiques"

Résumé

Les systèmes logiciels sont omniprésents et leur fiabilité est souvent cruciale, surtout lorsqu'une défaillance peut entraîner une catastrophe humaine, environnementale ou même économique. Développées depuis de nombreuses années, les méthodes formelles, comme l'approche classique du «model-checking», sont utilisées pour prouver la correction des systèmes. Plus récemment, au lieu de les construire manuellement un nouveau paradigme émerge pour automatiquement synthétiser des systèmes corrects. Pour cela, nous nous appuyons sur la théorie des jeux qui fournit des outils mathématiques afin de modéliser le comportement de deux agents antagonistes. Dans ce contexte, deux joueurs, un contrôleur et un environnement, jouent sur le graphe des configurations du système dans lequel la condition gagnante pour le contrôleur induit la correction du système. De plus, via l'ajout de poids, ces jeux peuvent être étendus pour étudier les performances du système, comme sa consommation d'énergie. Dans ce travail, nous nous concentrons sur la synthèse de systèmes temps-réel efficaces avec une condition de victoire simple : atteindre une configuration correcte.

Les jeux temporisés à coûts sont des jeux à deux joueurs et à somme nulle joués dans un automate temporisé pondéré. Nous considérons l'objectif d'accessibilité optimale, dans lequel l'un des joueurs, que nous appellerons Min, veut atteindre une cible tout en minimisant le poids cumulé. Même si savoir quand Min a une stratégie (déterministe) pour garantir que la valeur obtenue est inférieure à un seuil donné, est indécidable (avec deux horloges ou plus), plusieurs conditions ont été données pour retrouver la décidabilité. L'une d'entre elles est la divergence, une autre consiste à se limiter aux jeux temporisés à coûts avec une seule horloge et contenant uniquement des poids positifs. Nous étendons d'abord cette liste en considérant les jeux temporisés à coûts à une horloge avec des poids arbitraires, en montrant que leur fonction de valeur peut être calculée en temps exponentiel (si les poids sont codés en unaire).

Ensuite, dans de tels jeux temporisés à coûts (comme dans les jeux pondérés non temporisés en présence de poids négatifs), Min peut avoir besoin d'une mémoire finie pour jouer (presque) optimalement. Il est donc tentant d'essayer d'émuler cette mémoire finie avec d'autres fonctionnalités comme en permettant aux joueurs d'utiliser des décisions stochastiques, à la fois dans le choix des transitions et des délais. Nous donnons, pour la première fois, une définition de la valeur espérée dans les jeux temporisés à coûts, surmontant plusieurs défis théoriques. Nous montrons

ensuite que, dans les jeux temporisés à coûts divergents, la valeur stochastique est bien égale à la valeur classique (déterministe), prouvant ainsi que Min peut garantir la même valeur en n'utilisant que des choix stochastiques et sans mémoire.

Enfin, les stratégies stochastiques quasi-optimales que nous synthétisons ne sont pas implémentables car elles utilisent la précision infinie des horloges dans le choix des délais ainsi qu'une connaissance précise des configurations. La robustesse est un processus connu pour encoder l'imprécision des délais dans les stratégies : elle permet au joueur Max (l'adversaire de Min) de modifier légèrement le délai choisi par Min. Dans la littérature, deux sémantiques robustes existent : la sémantique conservatrice qui vérifie la garde après la perturbation, et la sémantique excessive qui la vérifie avant. Comme pour les stratégies déterministes, déterminer si Min a une stratégie (robuste) pour garantir que la valeur est inférieure à un seuil donné, est connu pour être indécidable. En adaptant le processus d'itération de la valeur introduit par Alur et al, nous calculons la valeur (robuste) dans des jeux temporisés à coûts acycliques (ou divergents avec uniquement des poids positifs) sous une sémantique conservatrice ou excessive.

Mots clés : jeux temporisés à coûts, algorithmique de la théorie des jeux, automate temporisé, calcul de valeur, stratégies randomisées, robustesse

Abstract

Software systems are omnipresent, and their reliability is often crucial, especially when a failure can lead to a human, environmental or even economic catastrophe. Formal methods, e.g. the classical model-checking approach, developed for many years, are used to prove the correctness of a system. More recently, instead of manually building a model, another paradigm emerges to automatically synthesise a correct system. For this, we rely on game theory which provides mathematical tools to model the behaviour of two antagonistic agents. In this context, two players, a controller and an environment play on a graph of all configurations of the system in which the winning condition for the controller induces a correct system. Moreover, such games may be extended to study the performance, e.g. the energy consumption of the system, modelled by adding weights. In this work, we focus on the synthesis of efficient real-time systems under a simple condition of correctness (reach a good state).

Weighted timed games are two-player zero-sum games played in a timed automaton equipped with integer weights. We consider optimal reachability objectives, in which one of the players, whom we call Min, wants to reach a target location while minimising the cumulated weight. While knowing if Min has a (deterministic) strategy to guarantee a value lower than a given threshold is known to be undecidable (with two or more clocks), several conditions have been given to recover decidability. One of them is a property called divergence, another one is to restrict to one-clock WTGs with only non-negative weights. We first extend this list by considering one-clock WTGs with arbitrary weights and showing that their value function can be computed in exponential time (if weights are encoded in unary).

Next, in such weighted timed games (like in untimed weighted games in the presence of negative weights), Min may need finite memory to play (close to) optimally. This is thus tempting to try to emulate this finite memory with other strategic capabilities. In particular, we allow the players to use stochastic decisions, both in the choice of transitions and timing delays. We give, for the first time, a definition of the expected value in weighted timed games, overcoming several theoretical challenges. We then show that, in divergent weighted timed games, the stochastic value is indeed equal to the classical (deterministic) value, thus proving that Min can guarantee the same value while only using stochastic choices and no memory.

Finally, the almost optimal stochastic strategies we synthesise are not imple-

mentable since they use infinite precision of clocks in the choice of the delay or the knowledge about the configurations. Robustness is a known process to encode the imprecision of delays in strategies: it allows Max player (the opponent of Min) to slightly modify the delay chosen by Min. In the literature, two robust semantics exist: the conservative semantics which checks a guard after the perturbation, and the excessive semantics which checks it before. As for deterministic strategies, determining whether Min has a (robust) strategy to guarantee a value lower than a given threshold is known to be undecidable. By adapting the process of value iteration introduced by Alur et al., we compute the (robust) value in acyclic (or divergent with only non-negative weights) weighted timed games under conservative and excessive semantics.

Keywords: weighted timed games, algorithmic game theory, timed automata, value computation, stochastic strategies, robustness

Remerciements

Avant d'entrer dans le vif du sujet de cette thèse, je souhaiterais remercier l'ensemble des personnes qui m'ont accompagnée pendant ces trois années de thèse. Tout d'abord, j'aimerais remercier Pierre-Alain Reynier ainsi que Benjamin Monmege pour leur disponibilité et leurs précieux conseils pendant toute la durée de ma thèse. Ces trois années ont été plaisantes et très enrichissantes grâce à leur accompagnement toujours bienveillant.

J'aimerais également remercier l'ensemble des membres du jury qui ont accepté de s'intéresser à mes travaux. Un remerciement particulier pour Eugene Asarin pour avoir accepté de présider ce jury, Orna Kupferman et Didier Lime pour avoir accepté de relire mon manuscrit, ainsi que Patricia Bouyer-Decitre, Hugo Gimbert et Jeremy Sproston pour avoir accepté de participer à ce jury.

Un grand merci à Yann Vexes et Enrico Porreca pour avoir accepté de suivre mes travaux au travers de mon comité de suivi de thèse et à Kevin Perrot pour l'intérêt qu'il a montré pour mes travaux lors des différents séminaires que j'ai pu donner au sein du LIS. Qu'il m'ait fait confiance pour aller discuter avec des lycéennes dans le cadre des "Cigales" m'a beaucoup touchée.

Je remercie bien évidemment les membres de l'équipe MoVe du LIS pour leur accueil et leur accompagnement pendant ces trois années notamment à l'heure du déjeuner et pendant les pauses café. Un merci particulier à Guillaume, Théodore et Idriss pour avoir partagé mon bureau.

Je n'oublie pas de remercier Nathalie Bertrand pour m'avoir fait découvrir le domaine de la vérification formelle et qui, il y a quatre ans, a eu une bonne idée en me suggérant un stage de master à Marseille.

Finalement, j'aimerais remercier l'ensemble de mes proches qui s'est toujours intéressé à mes travaux même lorsque ceux-ci leur sont complètement obscurs. Cet accompagnement s'illustre lors de ma soutenance où la majorité d'entre eux ont répondu présent soit en personne (après avoir traversé une bonne partie de la France) soit devant leur ordinateur en prêtant une oreille attentive à distance.

Contents

Introduction 12

1. Weighted Timed Games 23
- 1.1. Playing in a timed automaton . . . 24
- 1.2. Reachability timed games . . . 30
- 1.3. Weighted timed games . . . 37

I. Deterministic Value 45

2. Fixpoint Characterisation of the Deterministic Value Function 52
- 2.1. Kleene's theorem is not applicable . . . 53
- 2.2. Restriction of $\mathcal{F}$ according to a strategy of Min . . . 55
- 2.3. The case of one-clock weighted timed games . . . 58
- 2.4. The case of divergent weighted timed games . . . 67

3. Switching Strategies 69
- 3.1. A memory combination of two well-chosen memoryless strategies . . . 70
- 3.2. Almost-optimal strategies in divergent weighted timed games . . . 75

4. Deterministic Value Problem in One-clock Weighted Timed Games 81
- 4.1. Closure of one-clock weighted timed games . . . 84
- 4.2. How Max can control negative cycles in the closure . . . 90
- 4.3. A partial unfolding of the closure . . . 95
- 4.4. Decidability of the value problem . . . 112
- 4.5. Implementation . . . 112

Discussion 115

II. Stochastic Value 121

5. Playing Stochastically in Weighted Timed Games 127
- 5.1. Probability measure on plays . . . 128

5.2. Expected payoff of plays . . . 138
5.3. Discrete strategies . . . 148

6. Probabilities are Useless in the Presence of Memory **151**
6.1. Max always has a deterministic best-response . . . 152
6.2. Stochastic values are bounded by the smooth deterministic ones . . . 157
6.3. Equality between the stochastic and deterministic values . . . 159

7. Trading Memory for Probabilities **162**
7.1. Memoryless value in divergent weighted timed games . . . 165
7.2. Memoryless value in shortest path games . . . 175
7.3. Characterisation of memoryless optimality in shortest path games . . . 183

Discussion **190**

III. Robust Values **198**

8. Playing Robustly in Weighted Timed Games **204**
8.1. Robust semantics in weighted timed games . . . 204
8.2. Values guaranteed by Min under robust semantics . . . 211

9. Reductions between Robust and Exact Semantics **218**
9.1. A robust semantics simulated by the exact semantics . . . 219
9.2. A conservative semantics encoded in an excessive semantics . . . 231
9.3. Applications to robust reachability problems . . . 237
9.4. Application to robust value problems . . . 240

10. Symbolic Computation of the Limit Robust Values **244**
10.1. Deterministic value function in an acyclic weighted timed game . . . 245
10.2. Example of computation of robust value functions . . . 253
10.3. Limit robust values in an acyclic weighted timed game . . . 258
10.4. An extension to the strictly non-Zeno weighted timed games . . . 270

Discussion **272**

Conclusion **278**

Bibliography **281**

Introduction

Software systems are everywhere (e.g. in personal computers, smartphones or coffee machines) and are sometimes entrusted with our safety (as in planes, trains or cars). They are designed to assist with or autonomously execute critical tasks. We can cite, for example, controlling traffic lights, driving an autonomous vehicle, managing large communication networks, helping in the decision about self-care, etc. In particular, most of them are used in *cyber-physical systems* where an embedded software system in a machine monitors and controls the real-world: it links a computer environment and a machine that evolves in the real-world. In this case, software systems are increasingly subject to real-time constraints like response times and operational deadlines, e.g. the airbag opening time. Our capability to verify their behaviour prevents some catastrophes. However, this is a challenging task since the correctness of a system can depend on tricky interactions with its environment (human users or other software systems) under execution time and response time guarantees. In particular, for the designer, it can be arduous to anticipate and consider all possible behaviours of the system.

Synthesis of efficient real-time systems

The verification and, more recently, the synthesis of different classes of software systems is a challenging question. It has motivated an extensive research effort in computer science: a series of works (see [BP22] for a summary) has been devoted to analysing, testing and certifying software systems.

From the verification to the synthesis of software systems Formal verification provides automated methods to prove that the behaviour of a system described in a mathematical idealisation satisfies a given specification (or not). Even if several techniques (like abstract interpretation [CC77] or proof assistant with Coq[1], for example) have been developed for formal verification, model-checking has an important place. It exhaustively explores the mathematical model (often an automaton) to check if all behaviours (generated by the model) satisfy some desired properties expressed with a logical formula, e.g. eventually performing a particular

[1] https://github.com/coq/coq

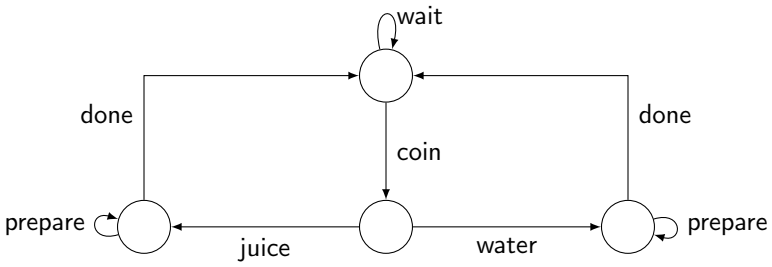


Figure 1 – An automaton modelling a drinks dispenser.

task or avoiding pre-defined errors. This problem has been extensively studied in various settings [BK08; Cla+18] (like infinite, concurrent, timed or stochastic systems), with practical tools being used for the verification of hardware and software in the industry (see [Cim01; Kar+18], for instance).

Example 1. We consider a software system that controls a drinks dispenser. Its natural model is depicted in Figure 1: the system waits until a user inserts a coin and chooses between orange juice or water. Then, the drink is produced before the system returns to its original situation when the drink is delivered. A desired property for this system may be "if a coin is inserted, then a drink (water or juice) will eventually be delivered". We note that, with this model, the property is not satisfied: the system may be blocked during the preparation. ⌟

One of the difficulties in the model-checking approach is the definition of a true-to-life model of our system before the verification. For several years, a new direction has emerged to automatically build a correct model by construction from the specification alone: we call this *synthesis* [Chu63]. This approach is based on the study of model the interaction between the system (called *controller*) and an antagonistic *environment* capturing uncontrollable events from the complex interactions with the real world, e.g. via user inputs, communication with other software systems, or uncertainty of the physical parameters of the system.

Example 2. Going back to Example 1, we can consider the presence of a user that interacts with the drink dispenser without being under the control of the dispenser. In particular, we model the behaviour of this user by an environment. Hence, the controller can choose whether to prepare or deliver the drink, but the environment is the one that decides when a coin is inserted and which drink needs to be prepared. ⌟

In this work, we rely on *game theory*[2] *on graphs* [Fij+23] to define a (mathematical) model representing the evolution of the system when the controller or the

[2]Game theory was first theorised by John Nash for economic purposes [NM47].

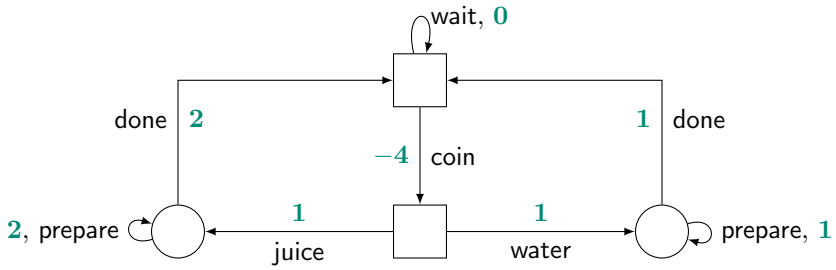


Figure 2 – A weighted extension of the game used to synthesis the drinks dispenser depicted in Figure 1.

environment applies actions. In particular, the controller is not the unique one to decide on the evolution of the system: it is deduced from the choices of both players in a turn-based fashion. We aim to automatically define a strategy (i.e. a sequence of choices) for the controller such that the evolution of the system satisfies the specification, whatever the environment does. For instance, we want to ask that a target location is eventually reached or conversely that an error location is never reached. Since the specification must hold in all possible scenarios, such a strategy finally describes a system that is necessarily correct.

Qualitative versus quantitative systems In a game on graphs under *zero-sum objectives*, *qualitative objectives* (i.e. winning conditions for the controller) express the required specification of the system, whereas the objective of the environment (the opponent) is dual. For instance, we may model the termination by the reachability of a target location, the correctness by safety from a bugged location, the liveness by repeated reachability, and even more complex properties by all ω-regular objectives [GTW02].

Under such objectives, the controller may have several winning strategies, i.e. many implementations guarantee the expected property on the system. By adding weights to the model, we select a special winning strategy regarding a qualitative objective and thus synthesise an efficient system according to the metric induced by weights (called *quantitative objectives*). For instance, we can synthesise a correct system that minimises its energy consumption with *total-payoff* objective [GZ04], minimises its mean consumption with *mean-payoff* objective [EM79], or minimises its long-term consumption with *discounted-payoff* objective [ZP96]. To model these systems, we extend well-known finite automata with a weighted function over edges: we use *weighted automata*. Weighted automata are already studied for the verification of quantitative properties [CDH10; DG07] in which the decidability of problems strongly relies on the definition of weights inside the automaton [ABK22].

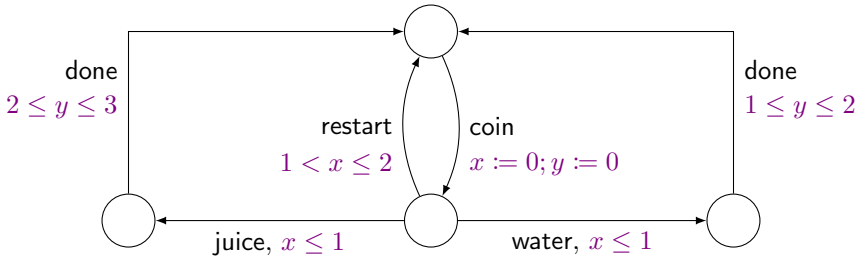


Figure 3 – A timed extension of the automaton modelled the drinks dispenser depicted in Figure 1.

Example 3. Going back to Example 2, we consider the weights modelling the cost of production of the drink (we depict it in Figure 2). In particular, since the user puts a coin before choosing its drink, this action brings back money to the system. Then, preparing water or orange juice does not induce the same price, and each delay in the preparation generates an over cost. Finally, when the system waits for a new user, it is in standby mode and consumes nothing. ⌟

Combining a quantitative and a qualitative objective enables the selection of a good strategy among the valid ones regarding the considered qualitative objective: correct and efficient systems are directly built. One of the most natural is *shortest-path games* that combine a reachability objective with a total-payoff objective [Kha+08; Bri+17]. Even if both objectives are simple, this combination brings some issues since a strategy for the controller needs memory to play optimally. We note that another combination of interest is the one between a parity objective (that can model every ω-regular objective) and a mean-payoff objective [CHJ05] in which the controller needs infinite memory.

Real-time systems Real-time systems are a class of software systems where the time elapsing between two actions is required to differentiate the good and bad behaviours of the system. Designing such systems not only involves developing correct software in the classical sense but also requires ensuring task execution times and response time guarantees. It is a difficult problem since small errors with grave consequences can stay unseen during the development and testing phase. Thus, the synthesis (i.e. the automatisation of the process) is necessary.

To model these systems, we extend well-known finite automata with real-valued variables called clocks: we use timed automata [AD94]. In particular, time elapsing is modelled by clocks that evolve at the same speed. Moreover, on each transition a timed constraint over clocks allows one to choose when it is fireable or not, i.e. when an action is realisable.

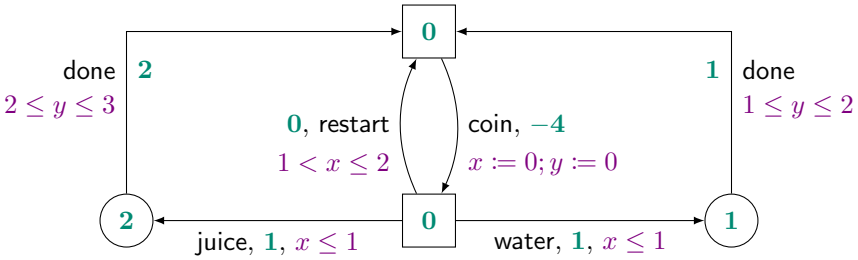


Figure 4 – A weighted timed game seen as a timed extension of Figure 2, or a quantitative game extension of Figure 3 to synthesising the drinks dispenser depicted in Figure 1.

Example 4. Going back to Example 3, we consider the time modelling of actions in our drinks dispenser (depicted in Figure 3). This model brings a more precise control of time spent for each action (by the time spent in each step before firing the transition). For instance, self-loops for preparation or waiting are replaced by constraints on time that denote the time needed for each action. With this model, we can thus control (inside the model) when an action is feasible according to the execution time of previous actions. ⌟

The verification of real-time systems consists of verifying whether all executions in the timed automaton satisfy a property. It has been proven in [AD94] that checking the reachability (i.e. reach a location), and Büchi (i.e. repeatedly reachability) are both PSPACE-complete, by partitioning the state space into *regions*. However, some more complex properties can be expressed (and checked) in the timed computation tree logic [ACD93], e.g. we could check if the actions labelled by coin and done are always separated by less than four units of time (or not).

In this thesis, we focus on the synthesis of an efficient real-time system under a reachability objective. As for finite systems, synthesis relies on a (turn-based) zero-sum two-player game played in a timed automaton. In particular, *reachability timed games* can be solved [AM99] and are EXPTIME-complete [JT07]. This qualitative objective is again too unsophisticated to model faithfully the system: we also want to select a good winning strategy for the controller. In particular, weighted extensions of timed automata [Beh+01; ALP04] and timed games [Bou+04a; ABM04] have thus been considered in this sense. There, weights are twofold: transitions are equipped with weights as in the untimed setting, but locations are also equipped with rates of weights such that the cost is then proportional to the time spent in it (that models the cost of the action realised inside the location). In particular, we study *weighted timed games* which are the weighted extension of reachability timed games with a shortest-path objective that are formally introduced in Chapter 1.

Example 5. Going back to Examples 3 and 4, we consider the weighted timed game (depicted in Figure 4). This model uses two sorts of weights: weights from Figure 2 (on transitions) that model the constant cost of the action (e.g. the glass, or the raw materials); and weights in locations that model the variable cost of the action (e.g. the consumption of the energy to realise the action) that depends on the time spent in the location. ⌟

In a game under a quantitative objective, we look after the optimal synthesis problem: we ask for the existence of a strategy for the controller such that whatever the environment does, the obtained weight is minimal. In the time setting, the optimal synthesis depends on the value of clocks: in particular, we introduce a function, called value function, which gives the weight guaranteed by the controller according to the initial *configuration*, i.e. initial location and value of clocks. Thus, in the context of weighted time games, the problem of our interest is the computation of such a value function.

Contributions of this thesis

We study the features needed by the controller to play optimally whatever the environment does. In particular, we investigate three features of the strategies for the controller: strategies using memory, i.e. the classical ones, in Part I, strategies using probabilities to remove the memory in Part II, and strategies resisting to small perturbations to deal with the precision of clocks and the immediacy of actions in Part III. In each case, we study the value associated to the class of strategies that we consider for both players.

Strategies using memory Adding weights in timed automata does not change the complexity of the resolution: the optimal reachability problem is known as PSPACE-complete [Bou+07]. However, in timed games, this problem becomes undecidable [BBR05] (instead of EXPTIME-complete for the reachability problem). Several restrictions have thus been considered to regain decidability. The first one restricts the weights allowed on cycles. The class of strictly *non-Zeno weighted timed games* with only non-negative weights has thus been introduced and studied in [Bou+04a]: this hypothesis requires that every execution of the timed automaton that follows a cycle (of locations) has a weight far from 0. More recently, an extension for arbitrary weights, called *divergent weighted timed games*, has been considered and studied in [BMR17].

Another restriction to recover the decidability is a limitation on the number of clocks. In a weighted time game without clocks, called a *shortest-path game*, we describe a finite system (where only discrete weights are allowed). In particular,

by adapting Dijkstra's algorithm the problem is in PTIME when weights are supposed non-negative [Knu77; Kha+08]. An extension to arbitrary weights has been proposed in [Bri+17], and the problem is solved in polynomial time when weights are encoded in unary.

A last restriction to keep the infinite system while limiting the clocks is restricting ourselves to one clock, for which its solving is known as PSPACE-hard [FIS20]. Solving *one-clock weighted time games* containing only non-negative weights is decidable [BBM06] with an exponential-time complexity [Rut11; HIM13]. More recently, the decidability frontier was pushed in the case of arbitrary weights [Bri+22]. However, the decidability (in exponential time) is obtained when the resets in cycles are restrained. In this context, our main contribution (Chapter 4) is the conclusion of this open question: the decidability with an exponential algorithm that computes the value function in all one-clock weighted timed games (for arbitrary weights).

Our contribution relies on the following main ingredients:

- The resolution of one-clock weighted time games is mainly based on the iteration of cyclic paths (cycle of locations with a reset corresponds to a cycle of configurations (a location with a real value for the clock). We define the value of such a cycle, that allows us to identify which player may benefit from iterating it.

- Using the classical region graph construction, we prove stronger properties on the value function: it is continuous on the closure of region intervals. This allows us to prove that the environment has an optimal memoryless strategy that avoids cycles whose value is negative.

- We introduce a partial unfolding of the game, to obtain an acyclic weighted time game, for which decidability is known. To do so, we rely on the existence of (almost-)optimal *switching* strategies for the controller that are composed of two memoryless strategies, the second one being triggered after a given pseudo-polynomial number of steps (Chapter 3). In particular, from the existence of such strategies, we limit the depth of exploration. Also, we keep track of cycles encountered and handle them according to their value. By transporting the previous result on the existence of a "smart" optimal strategy for the environment in the context of this unfolding, we show that the unfolding has the same value as the original weighted time game.

- To conclude the proof, we crucially need that the value function is obtained as a fixed point (indeed the greatest one) of an operator that was already used in many contributions before [ABM04; Bou+04a; BMR17]. We formally show this statement in Chapter 2.

Finally, based on our unfolding, we provide a prototype to compute the value function in all one-clock weighted time games. In particular, given a one-clock weighted time game, this prototype implements the value iteration process (from the fixpoint characterisation) in our unfolding.

Strategies using probabilities In weighted time games, the controller needs memory to play optimally. More precisely, the use of memory by the controller is exponential in the size of the game. This is prohibitive from an application point of view: we want to find how to avoid such complex strategies. In particular, we propose an alternative class of strategies that are more easily amenable to implementation. Thus, even if probabilities do not exist in the definition of weighted time games, it is interesting to introduce them as a feature for strategies, i.e. we allow both players to use stochastic choices in their strategies.

Our main contribution is to trade memory for randomness in some classes of weighted time games. This tradeoff has been already studied in Street or Müller games both in the untimed [CAH04] or timed setting [CHP08b]; and in multi-dimensional objectives [CRR14]: for instance, in mean-payoff parity games, if there exists a deterministic finite-memory winning strategy, then there exists a randomised memoryless almost-sure winning strategy. In our work, Chapter 7 explains how we adapt special optimal strategies (i.e. switching strategies) into memoryless strategies: switching strategies are obtained by the combination of two memoryless strategies. Intuitively, randomisation replaces the memory in the combination for switching strategies, i.e. it is used to choose between the two memoryless strategies with a well-chosen distribution of probabilities. Moreover, we prove that stochastic strategies do not increase the power of the controller, i.e. he can guarantee the same expected weight with probabilities (and memory) and without probability. In this case, we also rely on a fixpoint characterisation with a Bellman-like equation, this is discussed in Chapter 6.

Finally, in Chapter 5, the first important challenge is to analyse how to play stochastically in weighted timed games. To our knowledge, this has not been studied before. Starting from a notion of stochastic behaviours in a timed automaton considered in [Ber+14] (for the one-player setting), we propose a new class of stochastic strategies. Compared with [Ber+14], our class is larger in the sense that we allow Dirac distributions for delays, which subsumes the setting of deterministic strategies. However, to ensure that strategies yield a well-defined probability distribution on sets of executions, we need measurability properties stronger than the one considered in [Ber+14] (we actually provide an example showing that their hypothesis was not strong enough). Moreover, we propose an original definition of the expected payoff and provide some necessary conditions on the stochastic strategies for this expectation to be well-defined: the controller

not only must reach the target but must also do it quickly enough. It has to be noticed that randomisation only comes from the strategies of both players: in the presence of randomness also in the game itself, the situation becomes much more complex, even undecidable with at least two clocks [BCJ09].

Strategies resisting small perturbations Robustness is a crucial property of real-time systems: we require them to resist little errors or failures. More precisely, the behaviour of a real-time system should not change drastically if a small error occurs.

Example 6. In our drinks dispenser (depicted in Figure 4), the user may be indecisive and take this decision with some delays. In particular, if he waits one unit of time, he can not anymore choose a drink: he must restart the process (and he loses four euros). Thus, a small error in the information taking or a little delay in the decision may change a lot the behaviour of our machine. ⌟

Moreover, timed automata [AD94] provide a mathematical idealisation to design, model and verify real-time systems: it assumes that clocks have infinite precision and that actions are instantaneous. Indeed, proving that a timed automaton satisfies a property may not ensure that a real implementation satisfies the property. This robustness issue is a challenging problem for real-time systems [HS06]. Thus, alternative semantics have been proposed, so as to ensure that the verified (or synthesised) behaviour remains correct in the presence of small timing perturbations.

Example 7. In our drinks dispenser (depicted in Figure 4), we consider now that producing the drink makes more and more times (raw materials decrease, hardware system wears off). In particular, the system plays with a delay to reach $2 - 1/2^i$ to produce the i-th drink. This strategy that exploits the infinite precision of clocks can not be implemented. Thus, we use robustness to prevent this behaviour by imposing on the choices of the controller a small perturbation in the chosen delay. ⌟

Robustness was naturally studied in the context of model-checking in timed automata. For instance, under the Büchi condition, it consists in determining whether there exists an accepting infinite execution [Bus+19]. In the context of robustness, semantics considered is defined as a game that depends on some parameter $\mathfrak{p}$ representing an upper bound on the amplitude of the perturbation [CHP11]. In this game, the controller plays against an antagonistic environment that can perturb each delay using a value chosen in the interval $[-\mathfrak{p}, \mathfrak{p}]$. In Chapter 8, we extend this idea to game settings where we increase the power of the environment with the choice of perturbation. In particular, we propose a generic definition of

robust semantics and define some classical problems studied. Moreover, by exhibiting some links between semantics in weighted time games, we complete the state-of-the-art in value problems in Chapter 9.

Our main contribution consists of pushing the decidability frontier for robust value problems in the case of acyclic weighted time games and non-Zeno weighted time games. In particular, we provide an algorithm to compute the robust value (when the parameter tends to 0). Then, we apply this algorithm in an unfolding of non-Zeno weighted timed games that keeps their property when we assume that the perturbed delay (i.e. the delay after the perturbation) always satisfies the guard.

Our algorithm is a combination of two well-known technics to compute classical value on one side and the set of configurations from where the controller can guarantee that a target location is reached under robust semantics on the other side. We provide a robust-compliant extension of value iteration in acyclic games introduced in [ABM04]. In particular, by using ideas from shrinking DBM[3] [SBM11] we define the parametric value function and parametric cells[4]. Moreover, we can compute an upper bound over the parameter to guarantee the validity of our computation. The key argument is a decomposition of one computation of an operator that preserves the data structure that we have introduced. This result is presented in Chapter 10.

Structure of this thesis

This thesis is split into three independent parts, one by class of strategies. Each part starts with an introduction chapter that presents the class of strategies studied with some related works and finishes with a discussion chapter that concludes the part with a non-exhaustive list of interesting open questions. Most of the contributions in the two first parts are published in [MPR20; MPR21; MPR22] and submitted in a journal paper [MPR23a; MPR23b]. The work in the last part is not submitted yet.

This thesis contains ten chapters with three types of chapters whose reading may differ:

Preliminary chapters contain all definitions and preliminary results useful to understand the contributions. Chapter 1 contains preliminaries for the whole thesis and Chapter 8 preliminaries for the robustness part.

[3] Difference Bound Matrices (DBM) are the classical data structure used to compute configurations from which a target location can be reached in a timed automaton.

[4] Cells are the data structure used in [ABM04] to partition the regions.

Technical chapters contain technical contributions that may be used as a black box: only statements claimed at the beginning of the chapter are used in the main contributions. In particular, reading them is not necessary to understand the other contributions of this thesis. Chapters 2, 3, 5 and 9 are technical.

Contribution chapters contain a main contribution of this thesis. Chapters 4, 7, 6 and 10 are contribution chapters.

1. Weighted Timed Games

Modelling a system with real-time behaviours (under cost constraints) with finite graphs is not precise enough to incorporate subtle behaviours induced by real-time. To model the costs for electrical power consumption, we use fixed costs (e.g. contract costs) and variable costs (e.g. the energy consumed) that depend on the time a given appliance is used. The discretizaton of time is always a solution, but this induces a loss over the precision with a blow up of the model size. Thus, with the discretizaton, we may miss some interesting behaviours of the system.

Even if we consider infinite games, not all of them are relevant to us: we use a structure induced by time. In particular, we instead model such situations with *timed automata* [AD94] equipped with weights, both on transitions (to model fixed costs) and on locations (to model variable costs). In this chapter, we formally introduce (weighted) timed games under the reachability objective that we consider in this work.

Table of contents

1.1. Playing in a timed automaton . . . 24
1.1.1. Timed automaton . . . 25
1.1.2. Timed game . . . 27
1.2. Reachability timed games . . . 30
1.2.1. Semantics in reachability timed games . . . 30
1.2.2. Region abstraction . . . 33
1.2.3. Reachability objective and attractor . . . 35
1.3. Weighted timed games . . . 37
1.3.1. Deterministic value . . . 38
1.3.2. Deterministic value problem . . . 40
1.3.3. Classes of weighted timed games to recover decidability . . . 41
1.3.3.1. Shortest path games . . . 41
1.3.3.2. One-clock weighted timed games . . . 42
1.3.3.3. Divergent weighted timed games . . . 44

We denote by $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, and $\mathbb{R}$ the set of natural numbers, integers, rational, and real numbers, respectively. We let $\mathbb{R}_{\geq 0}$ be the set of non-negative real numbers and $\mathbb{R}_\infty = \mathbb{R} \cup \{-\infty, +\infty\}$. The set of vectors indexed by S with values in a set X is denoted by X^S: the components of such a vector $x \in X^S$ are written x_s or $x(s)$ depending on the context. Finally, for all $a \in \mathbb{R}_{\geq 0}$, $\lfloor a \rfloor \in \mathbb{N}$ denotes the integral part of a, and $\mathsf{fract}(a)$ $[0, 1)$ its fractional part, such that $a = \lfloor a \rfloor + \mathsf{fract}(a)$.

1.1. Playing in a timed automaton

With clocks, we extend the notion of finite automata to define timed automata that induce the desired structure for our timed games. Formally, let $\mathcal{X} = \{x_1, \ldots, x_n\}$ be a finite, non-empty set of variables called *clocks* that store time values. A *valuation* $\nu \colon \mathcal{X} \to \mathbb{R}_{\geq 0}$ is a mapping from clocks to their non-negative real value (or, equivalently, a point in the space $\mathbb{R}_{\geq 0}^{\mathcal{X}}$) such that $\nu(x_1), \ldots, \nu(x_n)$ are called the *coordinates* of ν.

Modelling real-time constraints On each clock, we consider two actions defining two operations over valuations: time elapse (with the same rate for all clocks) and reset. Formally, given a non-negative real number $t \in \mathbb{R}_{\geq 0}$, the *time elapse of delay t of ν* is the valuation $\nu + t$ defined by $(\nu + t)(x) = \nu(x) + t$, for all clocks $x \in \mathcal{X}$. The *time-successors* (resp. *time-predecessors*) of ν are all valuations $\nu + t$ (resp. $\nu - t$ in $\mathbb{R}_{\geq 0}^{\mathcal{X}}$) with $t \geq 0$ and both define the unique "diagonal" line in $\mathbb{R}_{\geq 0}^{\mathcal{X}}$ that contains ν. Moreover, if Y is a subset of $\mathcal{X}$, we define the valuation $\nu[Y \coloneqq 0]$ to be the *reset of Y* such that for all clocks $x \in \mathcal{X}$, $(\nu[Y \coloneqq 0])(x) = 0$ if $x \in Y$ and $(\nu[Y \coloneqq 0])(x) = \nu(x)$, otherwise.

Valuations are subject to guards defined by constraints on some clocks. More precisely, an *open* (resp. *closed*) *atomic constraint* over clocks $x, y \in \mathcal{X}$ refers to an affine inequality of one of these forms

- $x \bowtie \alpha$, called *non-diagonal*;
- $x - y \bowtie \alpha$, called *diagonal*;

with $\alpha \in \mathbb{Q}$ a constant, and $\bowtie \in \{>, <\}$ (resp. $\bowtie \in \{\geq, \leq\}$). Then, a *guard* g over $\mathcal{X}$ is a finite conjunction of atomic constraints over the clocks of $\mathcal{X}$. As a shorthand, we denote by $x = \alpha$ the guard $x \leq \alpha \wedge x \geq \alpha$, and by $\alpha_1 < x < \alpha_2$ the guard $x > \alpha_1 \wedge x < \alpha_2$. Finally, a guard is called *open* (resp. *closed*, *diagonal*, *non-diagonal*) if all of its atomic constraints are open (resp. closed, diagonal, non-diagonal), and we denote by $\mathsf{Guards}(\mathcal{X})$ the set of all non-diagonal guards on $\mathcal{X}$.

Let $\nu \in \mathbb{R}_{\geq 0}^{\mathcal{X}}$ be a valuation. For all atomic constraints $x \bowtie \alpha$, we say that ν *satisfies* $x \bowtie \alpha$ (and we write $\nu \models x \bowtie \alpha$) when $\nu(x) \bowtie \alpha$. We extend this notion

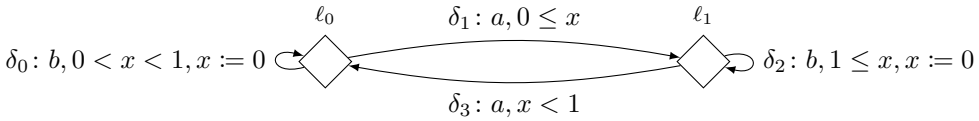


Figure 1.1 – A timed automaton over $\mathcal{X} = \{x\}$ and actions $\Sigma = \{a, b\}$ where the reset of the clock x is denoted by $x \coloneqq 0$ such that the transition δ_0 which loops in ℓ_0 refers to $(\ell_0, b, 0 < x < 1, \{x\}, \ell_0)$.

by saying that ν satisfies the guard g (and write $\nu \models g$) if ν satisfies all atomic constraints in g. A guard g is said *satisfiable* when there exists at least a valuation that satisfies g. The guard $\overline{g}$ is the closed version of a satisfiable guard g where every strict constraint of comparison operator $<$ or $>$ is replaced by its closed version $\leq$ or $\geq$. In this case, we note that the set of valuations that satisfy $\overline{g}$ is the topological closure of the set of valuations that satisfy g.

1.1.1. Timed automaton

Now, we can define a timed automaton.

Definition 1.1. A *timed automaton* is a tuple $\langle L, \mathcal{X}, \Sigma, \Delta \rangle$ where L is a finite set of locations, $\mathcal{X}$ is a finite set of clocks, Σ is a finite set of actions and $\Delta \subseteq L \times \Sigma \times \mathsf{Guards}(\mathcal{X}) \times 2^{\mathcal{X}} \times L$ is a transition relation.

An example of a timed automaton is depicted in Figure 1.1. We defined timed automata such that they only use non-diagonal guards. An equivalent model that is exponentially more succinct may be defined with diagonal guards [Bér+98].

Paths A *path* of a timed automaton is a finite or infinite sequence of consecutive transitions $\ell_0 \xrightarrow{\delta_0} \ell_1 \xrightarrow{\delta_1} \cdots$ where $\delta_0, \delta_1, \ldots \in \Delta$ and $\ell_0, \ell_1, \ldots \in L$. Since initial and intermediate locations are uniquely defined by the sequence of transitions, we sometimes denotes a path by $\delta_0 \cdot \delta_1 \cdots$. Moreover, we denote by $\pi_1 \cdot \pi_2$ the concatenation of a finite path π_1 ending in location ℓ and another path π_2 starting in location ℓ. We let $\mathsf{FPaths}_{\mathcal{A}}$ (or FPaths when the timed automaton is clear from the context) be the set of all finite paths of $\mathcal{A}$.

Given a finite path π, we let $|\pi|$ be its *length*, which is its number of transitions, and $|\pi|_\delta$ the number of occurrences of a given transition δ in π.

Example 1.2. We consider the timed automaton depicted in Figure 1.1 and

$$\pi = \ell_0 \xrightarrow{\delta_0} \ell_0 \xrightarrow{\delta_1} \ell_1 \xrightarrow{\delta_3} \ell_0 \xrightarrow{\delta_0} \ell_0$$

is a finite path of length $|\pi| = 4$. This path is also denoted by $\delta_0 \cdot \delta_1 \cdot \delta_3 \cdot \delta_0$. ⌟

Semantics of a timed automaton Given a timed automaton, its *semantics* is given by an infinite graph (also called a *configuration graph*) where a *configuration* is a location and a valuation, e.g. $(\ell_0, 0)$ or $(\ell_0, 0.5)$ in the timed automaton of Figure 1.1. In particular, we denote by $\mathsf{Conf}^{\mathcal{A}}$ (resp. Conf when the timed automaton is clear from the context) the set of configurations induced by the timed automaton $\mathcal{A}$. The configuration graph explains all the relationships between the configurations via feasible transitions. Given a configuration (ℓ, ν), a *decision* (δ, t) is a pair given by a transition $\delta = (\ell, a, g, Y, \ell')$ and a delay t, e.g. $(\delta_0, 0)$ and $(\delta_0, 0.5)$ are two decisions from the configuration $(\ell_0, 0)$ in the timed automaton of Figure 1.1. Moreover, we say that a decision (δ, t) with $\delta = (\ell, a, g, Y, \ell')$ is *feasible* from the configuration (ℓ, ν) when $\nu + t \models g$. E.g. in Figure 1.1, from $(\ell_0, 0)$, the decision $(\delta_0, 0)$ is not feasible whereas the decision $(\delta_0, 0.5)$ is.

Definition 1.3. Let $\langle L, \mathcal{X}, \Sigma, \Delta \rangle$ be a timed automaton. Its *semantics* is the system of transitions defined by the tuple $\langle \mathsf{Conf}, E \rangle$ such that $\mathsf{Conf} \subseteq L \times \mathbb{R}^{\mathcal{X}}_{\geq 0}$ is the set of configurations and E is the set of *edges* such that $(\ell, \nu) \xrightarrow{\delta, t} (\ell', \nu') \in E$ with $\delta = (\ell, a, g, Y, \ell') \in \Delta$, if $\nu + t \models g$ and $\nu' = (\nu + t)[Y \coloneqq 0]$.

Finite plays A *finite play* (called *run* in the literature) in a timed automaton consists of a finite sequence of edges from its semantics, i.e. $(\ell_0, \nu_0) \xrightarrow{\delta_0, t_0} (\ell_1, \nu_1) \xrightarrow{\delta_1, t_1} (\ell_2, \nu_2) \cdots (\ell_k, \nu_k)$. We denote by $I(\rho, \delta)$ the interval[1] of delays t such that the play ρ can be extended with the decision $\xrightarrow{\delta, t}$. This extension is sometimes denoted $\rho[\delta, t]$. Since intermediate configurations are uniquely defined by the initial configuration and the sequence of transitions and delays, we sometimes denote a play ρ by $(\ell_0, \nu_0)[\delta_0, t_0] \cdots [\delta_{k-1}, t_{k-1}]$. The concatenation of two finite plays ρ_1 ending in configuration (ℓ, ν) and ρ_2 starting in configuration (ℓ, ν), is denoted by $\rho_1 \rho_2$. We let $\mathsf{FPlays}_{\mathcal{A}}$ (resp. FPlays when the timed automaton is clear from the context) be the set of all finite plays in $\mathcal{A}$.

Example 1.4. We consider the timed automaton depicted in Figure 1.1 and

$$\rho = (\ell_0, 0) \xrightarrow{\delta_0, 0.1} (\ell_0, 0) \xrightarrow{\delta_0, 0.3} (\ell_0, 0) \xrightarrow{\delta_1, 0.2} (\ell_1, 0.2) \xrightarrow{\delta_2, 1.05} (\ell_1, 0)$$

is a finite play sometimes denoted $(\ell_0, 0)[\delta_0, 0.1][\delta_0, 0.3][\delta_1, 0.2][\delta_2, 1.05]$. Moreover, the set of delays that can extend ρ with transition δ_2 is $I(\rho, \delta_2) = [1, +\infty)$. ⌟

Given a finite play ρ, we let $|\rho|$ be its *length* which is its number of edges, and $|\rho|_\delta$ be the number of occurrences of a given transition δ in ρ. More generally, for

[1] It is an interval since guards are conjunctions of affine inequality constraints on clocks. More precisely, for each constraint given by an inequality over the valuation of one clock, adequate delays are described by a (possibly empty) interval. Then, $I(\rho, \delta)$ is defined by the intersection of these intervals.

a play ρ and a set A of transitions, we let $|\rho|_A$ be the number of occurrences of all transitions from A in ρ, i.e. $|\rho|_A = \sum_{\delta \in A} |\rho|_\delta$.

Example 1.5. We consider the finite play ρ defined in Example 1.4. Its length is $|\rho| = 4$ and by letting $A = \{\delta_1, \delta_2\}$, we have $|\rho|_A = 2$.

Each finite play ρ of a timed automaton is associated with a unique finite path π (by projecting away everything but the transitions): we say that ρ *follows* the path π. We also let $\mathsf{last}(\pi)$ and $\mathsf{last}(\rho)$ be the last location or configuration of a finite path π and play ρ. Finally, a path is said to be *maximal* if it is infinite or no feasible decision exists from all valuations in the last location of the path. A *maximal play* is then a maximal sequence of consecutive edges, i.e. a maximal play following a maximal path.

Example 1.6. We consider the finite play ρ defined in Example 1.4. It follows the finite path

$$\pi = \ell_0 \xrightarrow{\delta_0} \ell_0 \xrightarrow{\delta_0} \ell_0 \xrightarrow{\delta_1} \ell_1 \xrightarrow{\delta_2} \ell_1 \,. \qquad \lrcorner$$

1.1.2. Timed game

A *timed game* [AHK98; AM99] is a two-player extension of a timed automaton where the two players are denoted by Min and Max. More precisely, a turn-based timed game is based on a timed automaton where the set of locations is partitioned into two distinct sets (one for each player).

Definition 1.7. A (*turn-based*) *timed game* is a tuple $\mathcal{A} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, \mathcal{X}, \Sigma, \Delta \rangle$ such that $L = L_{\mathsf{Min}} \cup L_{\mathsf{Max}}$ with $L_{\mathsf{Min}} \cap L_{\mathsf{Max}} = \emptyset$ and $\langle L, \mathcal{X}, \Sigma, \Delta \rangle$ is a timed automaton.

Example 1.8. From the timed automaton depicted in Figure 1.1, we define a timed game such that $L_{\mathsf{Min}} = \{\ell_0\}$ and $L_{\mathsf{Max}} = \{\ell_1\}$. $\lrcorner$

A *one-player* timed game is a special class of timed games where all locations of a given timed game belong to a unique player, i.e. $L_{\mathsf{Min}} = \emptyset$ or $L_{\mathsf{Max}} = \emptyset$. In this case, the timed game has the same behaviour as the underlying timed automaton.

Semantics for timed games Given a timed game $\mathcal{A}$, we define its semantics by the one of the underlying timed automaton, i.e. in terms of a turn-based game played on an infinite transition system (called *game structure*) whose vertices are configurations of the timed game.

Since a timed game is a two-player game, its set of configurations is split amongst players. In particular, we denote by $\mathsf{Conf}^{\mathcal{A}}_{\mathsf{Min}}$ (resp. $\mathsf{Conf}^{\mathcal{A}}_{\mathsf{Max}}$) the set of configurations of Min (resp. Max) or $\mathsf{Conf}_{\mathsf{Min}}$ (resp. $\mathsf{Conf}_{\mathsf{Max}}$) when the timed game is clear

from the context. Intuitively, the alphabet of the game structure is $\Delta \times \mathbb{R}_{\geq 0}$ and models decisions (δ, t) encoding the delay t that the current player (given by the location) wants to spend in the current location before firing transition δ.

Formally, we define the semantics of a timed game as follows:

Definition 1.9. Let $\mathcal{A} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, \mathcal{X}, \Sigma, \Delta \rangle$ be a timed game. Its *semantics*, denoted by $[\![\mathcal{A}]\!] = \langle \mathsf{Conf}, E \rangle$, is a system of transitions over the set of configurations of $\mathcal{A}$ such that E is equal to

$$\left\{ (\ell, \nu) \xrightarrow{\delta, t} (\ell', \nu') \mid \delta = (\ell, g, Y, \ell') \in \Delta, \nu + t \models g, \text{ and } \nu' = (\nu + t)[Y := 0] \right\}.$$

We extend the notions of path and finite play from timed automata. In particular, $\mathsf{FPlays}_{\mathcal{A},\mathsf{Min}}$ (resp. $\mathsf{FPaths}_{\mathcal{A},\mathsf{Min}}$) and $\mathsf{FPlays}_{\mathcal{A},\mathsf{Max}}$ (resp. $\mathsf{FPaths}_{\mathcal{A},\mathsf{Max}}$) denote the finite plays (resp. paths) ending in a configuration (resp. location) of Min and Max. To lighten the notations, when the timed game is clear from the context, we may omit the subscript $\mathcal{A}$.

Strategies A strategy models the decisions of one of the players along plays. Formally, given $\mathcal{A} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, \mathcal{X}, \Sigma, \Delta \rangle$ a timed game, a *deterministic strategy* for Min (resp. Max) in $\mathcal{A}$ is a mapping $\sigma \colon \mathsf{FPlays}_{\mathcal{A},\mathsf{Min}} \to \Delta \times \mathbb{R}_{\geq 0}$ (resp. $\tau \colon \mathsf{FPlays}_{\mathcal{A},\mathsf{Max}} \to \Delta \times \mathbb{R}_{\geq 0}$) such that for all finite plays $\rho \in \mathsf{FPlays}_{\mathcal{A},\mathsf{Min}}$ (resp. $\rho \in \mathsf{FPlays}_{\mathcal{A},\mathsf{Max}}$) ending in configuration (ℓ, ν), there exists an edge $(\ell, \nu) \xrightarrow{\sigma(\rho)} (\ell', \nu')$ in $[\![\mathcal{A}]\!]$. We let $\mathsf{dStrat}_{\mathsf{Min}}(\mathcal{A})$ (resp. $\mathsf{dStrat}_{\mathsf{Min}}$) and $\mathsf{dStrat}_{\mathsf{Max}}(\mathcal{A})$ (resp. $\mathsf{dStrat}_{\mathsf{Max}}$) denote the set of deterministic strategies in $\mathcal{A}$ (resp. if $\mathcal{A}$ is clear from the context) for players Min and Max, respectively. In the following, we will always use letters σ and τ to differentiate from deterministic strategies of Min and Max.

Example 1.10. We consider the timed game defined in Example 1.8. We fix a deterministic strategy for Min defined, for all finite plays ρ, by

$$\sigma(\rho) = \begin{cases} (\delta_1, 1) & \text{if } \mathsf{last}(\rho) = (\ell_0, 0); \\ (\delta_0, 0) & \text{if } \mathsf{last}(\rho) = (\ell_0, \nu) \text{ with } 0 < \nu < 1; \\ (\delta_1, 1) & \text{if } \mathsf{last}(\rho) = (\ell_0, \nu) \text{ with } 1 \leq \nu. \end{cases}$$

⌟

A play or finite play $\rho = (\ell_0, \nu_0)[\delta_0, t_0][\delta_1, t_1] \cdots$ *conforms* to a deterministic strategy σ of Min if for all k such that (ℓ_k, ν_k) belongs to Min, we have that $(\delta_k, t_k) = \sigma\big((\ell_0, \nu_0)[\delta_0, t_0] \cdots [\delta_{k-1}, t_{k-1}]\big)$. An analogous definition holds for strategies of Max. Moreover, a finite path π is said to be *conforming* to a strategy σ (resp. τ) if there exists a finite play following π that is conforming to σ (resp. τ).

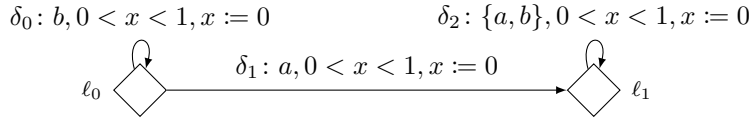


Figure 1.2 – A concurrent timed game where Min uses action a and Max uses action b.

Given two deterministic strategies σ and τ, we let $\mathsf{Play}((\ell, \nu), \sigma, \tau)$ be the *outcoming play* of σ and τ defined as the unique maximal play conforming to σ and τ and starting in configuration (ℓ, ν).

Example 1.11. We consider the timed game defined in Example 1.8. We fix a deterministic strategy for Max defined, for all finite plays ρ, by

$$\tau(\rho) = \begin{cases} (\delta_3, 0) & \text{if } \mathsf{last}(\rho) = (\ell_1, \nu) \text{ with } \nu < 1; \\ (\delta_2, 0) & \text{if } \mathsf{last}(\rho) = (\ell_0, \nu) \text{ with } 1 \leq \nu. \end{cases}$$

In particular, by considering the deterministic strategy for Min defined in Example 1.10, the outcoming play of σ and τ is

$$\mathsf{Play}((\ell_0, 0), \sigma, \tau) = \big((\ell_0, 0) \xrightarrow{\delta_1, 1} (\ell_1, 1) \xrightarrow{\delta_2, 0} (\ell_1, 0) \xrightarrow{\delta_3, 0} (\ell_0, 0)\big)^\omega . \qquad \lrcorner$$

A useful restriction over strategies is the absence of memory, i.e. if the chosen decision only depends on the last configuration of plays. A (deterministic) strategy σ of Min is *memoryless* if for all finite plays ρ, ρ' such that $\mathsf{last}(\rho) = \mathsf{last}(\rho')$, we have that $\sigma(\rho) = \sigma(\rho')$. A similar definition holds for Max.

Example 1.12. Both deterministic strategies σ (resp. τ) defined in Example 1.10 (resp. Example 1.11) are memoryless strategies. $\lrcorner$

Concurrent timed game An extension[2] of turn-based games is *concurrent timed games* [AHK98]: instead of using locations, concurrent timed games are based on timed automata where Σ is split into two distinct subsets containing the actions of each player. On each turn, both players simultaneously choose a decision from their set of actions, i.e. the chosen transition must be labelled by an action of the player. The player who chooses the shortest[3] delay wins this turn and applies his decision.

Example 1.13. We consider the concurrent timed game depicted in Figure 1.2.

[2] In the literature, timed games were first introduced like concurrent timed game.

[3] A problem may arrive when players choose the same delay. In this case, we use an external strategy, called a *scheduler*, to choose the winner of the turn.

The set of possible decisions from $(\ell_0, 0)$ for both players is of the form $\{(\mathrm{A}, t) \mid t > 0\}$ where $\mathrm{A} \in \{a, b\}$ is the action of the player. ⌟

Objectives To define a timed game, we need a timed automaton and an objective to define which player wins. In this work, we focus on *zero-sum* objectives, i.e. each play is winning for exactly one player. In particular, we focus on the reachability objective [AHK98; AM99] and its quantitative extension [ABM04; Bou+04a] with weighted timed games. However, we note that other objectives have been also considered for timed games like Büchi timed games [Bou+11], parity timed games [CHP08a], or energy and mean-payoff timed games [BCR14].

1.2. Reachability timed games

We consider a particular class of (turn-based) timed games: the *reachability timed games* [AHK98; AM99; JT07], defined as timed games with reachability objectives. Moreover, we introduce classical tools to solve such classes of games: region abstraction and attractor computation. Formally, we define a reachability timed game as follows:

Definition 1.14. A *reachability timed game* (RTG for short) is a tuple $\mathcal{A} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta \rangle$ where

- $L = L_{\mathsf{Min}} \uplus L_{\mathsf{Max}} \uplus L_T$ is a finite set of locations split between players Min and Max, and a set of *target* locations;
- $L_u \subseteq L$ is a finite set of *urgent* locations where time cannot be delayed;
- $\mathcal{X}$ is a finite set of clocks;
- $\Delta \subseteq L \times \mathsf{Guards}(\mathcal{X}) \times 2^{\mathcal{X}} \times L$ is a finite set of transitions.

We note that our definition is not usual. Indeed, the presence of urgent locations is not standard: in a timed automaton with several clocks, urgency can be modelled with an additional clock u that is reset just before entering the urgent location and with guards $u = 0$ on outgoing transitions. However, when we limit the number of clocks to one, we regain modelling capabilities by allowing for such urgent locations. In drawings, to recognise them, urgent locations will be displayed with a u inside.

1.2.1. Semantics in reachability timed games

Given an RTG $\mathcal{A} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta \rangle$, we define its semantics using the one of the timed game $\langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, \mathcal{X}, \Delta \rangle$, i.e. in terms of a game played on an infinite transition system whose vertices are configurations of $\mathcal{A}$.

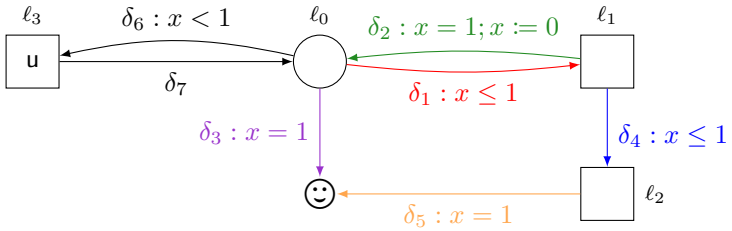


Figure 1.3 – A (one-clock) RTG with a target location is ☺ and an urgent location ℓ_3. Locations belonging to Min (resp. Max) are depicted by circles (resp. squares). Transitions that contain the reset of the clock x are labelled with $x := 0$.

More precisely, instead of only splitting the set of configurations among players, we also define *final* configurations such that their locations are in L_T. Moreover, without loss of generality, we suppose the absence of deadlocks except on target locations, i.e. for each configuration (ℓ, ν) with $\ell \in L \setminus L_T$, a feasible decision exists from it, and no transitions start in L_T. Formally, we define the semantics of an RTG as follows.

Definition 1.15. Let $\mathcal{A} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta \rangle$ be an RTG. Its *semantics*, denoted by $[\![\mathcal{A}]\!] = \langle \mathsf{Conf}^{\mathcal{A}}, E \rangle$, is the semantics of the timed game defined when target locations are also locations of Min, i.e. by $\langle L_{\mathsf{Min}} \cup L_T, L_{\mathsf{Max}}, \mathcal{X}, \emptyset, \Delta \rangle$, and E verifies that for all urgent locations ℓ, only edges $(\ell, \nu) \xrightarrow{\delta,0} (\ell', \nu')$ belong to E.

An example of RTG is depicted in Figure 1.3.

Plays and paths Since the objective of Min is to reach a target configuration, we denote by TPlays the set of finite plays ending in a target configuration. Finally, we let TPlays_ρ (resp. TPlays^n_ρ) be the subset of TPlays that contains plays starting in the last configuration of the finite play ρ (resp. and containing n edges, i.e. of length n).

A *target path* is a finite path ending in the target set L_T. We denote by TPaths the set of target paths. We let TPaths_ρ (resp. TPaths^n_ρ) be the subset of target paths that start from the last location of the finite play ρ (resp. and containing n transitions without taking into account the size of ρ).

A *cyclic path* is a finite path that starts and ends in the same location. A *cyclic play* is a finite play that starts and ends in the same configuration: it necessarily follows a cyclic path, but the reverse might not be true since some non-cyclic plays can follow a cyclic path (if they do not end in the same valuation as the one in which they start) as illustrated in the following example.

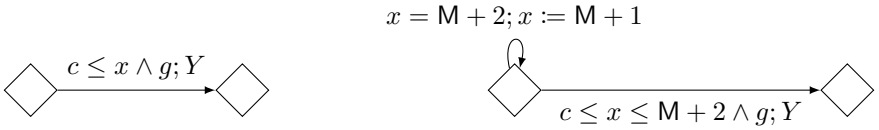


Figure 1.4 – On the left, a transition in a timed automaton where the clock x is not bounded. On the right, a bounded version given by [Beh+01] of this transition where M is the greatest constant that appears in guards of the timed automaton. We denote by $x := \mathsf{M}+1$ the operation on valuations such that the valuation $\nu[x := \mathsf{M}+1](y)$ is equal to $\mathsf{M}+1$ if $x = y$ and $\nu(y)$ otherwise.

Example 1.16. In the RTG depicted in Figure 1.3, plays that follow the cyclic path $\pi = \ell_0 \xrightarrow{\delta_1} \ell_1 \xrightarrow{\delta_2} \ell_0$ are cyclic only when they start from $(\ell_0, 0)$ (since δ_2 resets the clock). ⌟

Strategies Again we extend the definition of deterministic strategies from timed games in this context. In particular, given an RTG, a deterministic strategy is called winning for this RTG if, for all maximal plays conforming to it, plays end in a target configuration, i.e. belongs to TPlays. More formally, from the configuration (ℓ, ν), a deterministic strategy σ of Min is *winning* if, for all deterministic strategies τ of Max, $\mathsf{Play}((\ell, \nu), \sigma, \tau) \in \mathsf{TPlays}$.

Example 1.17. We consider the timed game depicted in Figure 1.3. We fix a deterministic strategy for Min defined, for all configurations (ℓ_0, ν) with $\nu \leq 1$, by

$$\sigma(\ell_0, \nu) = (\delta_3, 1 - \nu)\,.$$

We note that this deterministic strategy is winning for Min from $(\ell_0, 0)$. ⌟

Finally, we note that this class of timed games is *positionally determined*, i.e. either Min or Max wins with a memoryless strategy [JT07, Theorem 1]. We recall algorithms computing a winning strategy for the player that wins a given RTG in Section 1.2.3.

Clock bounding Seminal works in timed games [ABM04; Bou+04a] have assumed that clocks are *bounded*. This is known to be without loss of generality for (weighted) timed automata [Beh+01, Theorem 2]: it suffices to replace transitions with unbounded delays with self-loop transitions periodically resetting[4] the clocks

[4]The model uses a more permissive reset operator: the model admits the reset of a clock with a given value and not only to 0 as in our model. However, timed automata using such resets are known to be equivalent to classical timed automata [Bou+04b, Theorem 9]. This transformation does not add new behaviours to our system.

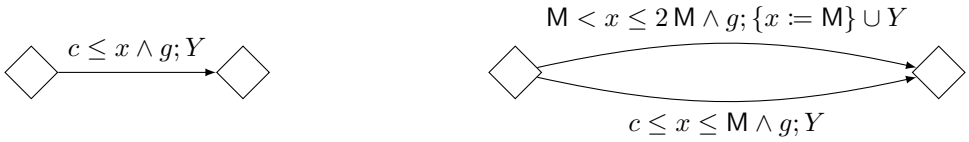


Figure 1.5 – On the left, a transition of an RTG where the clock x is not bounded. On the right, a bounded version of this transition where M is the greatest constant that appears in guard of the RTG. We denote with $\{x := \mathsf{M}\} \cup Y$ the set of reset such that x use the special reset $x := \mathsf{M}$ when $x \notin Y$, otherwise $x\mathsf{M}0$.

(Figure 1.4 illustrates this transformation). However, in timed games, this transformation increases the power of Max: he can stay in the same location forever.

Fortunately, in an RTG, we can adapt this technique to bound the clocks without loss of generality. In particular, we prefer to duplicate the transition with a guard that admits to elapse M units of time (Figure 1.5 illustrates this transformation). In this case, we do not create a new loop (so Max can not avoid the target locations), and the elapsing time for other clocks is always admited.

1.2.2. Region abstraction

To analyse a timed system, we use the crucial notion of regions defined by an equivalence relation over the valuations, as introduced in [AD94].

Definition 1.18. With respect to the set $\mathcal{X}$ of clocks and the upper bound M on the valuation of clocks, a *region* is a subset of valuations in $[0, \mathsf{M})^{\mathcal{X}}$ characterised by a pair (ι, β) where $\iota \colon \mathcal{X} \to [0, \mathsf{M}) \cap \mathbb{N}$ and β is a partition of $\mathcal{X}$ into subsets $\beta_0 \uplus \beta_1 \uplus \cdots \uplus \beta_m$ (with $m \geq 0$), where β_0 can be empty but for all $1 \leq i \leq m$, the set β_i is not empty. We denote by $\mathsf{Reg}(\mathcal{X}, \mathsf{M})$ the set of regions (that partitions the set $[0, \mathsf{M})^{\mathcal{X}}$ of valuations) bounded by M.

Example 1.19. The left of Figure 1.6 represents the 24 regions over two clocks $\{x_1, x_2\}$ bounded by the maximal constant $\mathsf{M} = 2$. The red region contains only one valuation $(1, 0)$ and is characterized by $\iota = (1, 0)$ and $\beta_0 = \{x_1, x_2\}$. The set of valuations $\{\nu \mid \nu(x_1) = 0 \wedge 1 < \nu(x_2) < 2\}$ is the green region characterized by $\iota = (0, 1)$ and $\beta_0 = \{x_1\} < \beta_1 = \{x_2\}$, whereas the set of valuations $\{\nu | 0 < \nu(x_1) < 1\}$ is the blue region characterized by $\iota = (0, 0)$ and $\beta_0 = \emptyset < \beta_1 = \{x_1, x_2\}$. Finally, the orange region defined by $\iota = (0, 0)$ and $\beta_0 = \emptyset < \beta_1 = \{x_1\} < \beta_2 = \{x_2\}$ denotes the set of valuations $\{\nu | 0 < \nu(x_1) \leq \nu(x_2) < 1\}$. ⌟

A valuation ν of $[0, \mathsf{M})^{\mathcal{X}}$ belongs to the region characterised by (ι, β) if

- for all $x \in \mathcal{X}$, $\iota(x) = \lfloor \nu(x) \rfloor$;

- for all $x \in \beta_0$, $\mathsf{fract}(\nu(x)) = 0$;
- for all $0 \leq i \leq m$, for all $x, y \in \beta_i$, $\mathsf{fract}(\nu(x)) = \mathsf{fract}(\nu(y))$;
- for all $0 \leq i < j \leq m$, for all $x \in \beta_i$ and all $y \in \beta_j$, $\mathsf{fract}(\nu(x)) < \mathsf{fract}(\nu(y))$.

A region r' is said to be a *time successor* of the region r if there exists $\nu \in r$, $\nu' \in r'$, and $t > 0$ such that $\nu' = \nu + t$. In particular, if r is a region, then the time successor of valuations in r form a finite union of regions, and the reset $r[Y \coloneqq 0]$ of $Y \subseteq \mathcal{X}$ is also a region.

Finally, an RTG can be extended with the region information: the partition into regions can be maintained throughout plays in an RTG. By forgetting about the precise valuation of clocks, we obtain a finite abstraction called the *region game* (or *region automaton*).

The set of valuations contained in a region r characterised by (ι, β) with $\beta = \beta_0 \uplus \beta_1 \uplus \cdots \uplus \beta_m$ can be described by the (diagonal) guard $g_0 \wedge g_1 \wedge \cdots \wedge g_m$ with

$$g_0 = \bigwedge_{x \in \beta_0} \big(x = \iota(x)\big), \qquad g_1 = \bigwedge_{x,y \in \beta_1} \big(0 < x - \iota(x) = y - \iota(y) < 1\big)$$

and for $i \in \{2, \ldots, m\}$,

$$g_i = \bigwedge_{x,y \in \beta_i} \big(z - \iota(z) < x - \iota(x) = y - \iota(y) < 1\big)$$

where z is any clock of β_{i-1}. To define the game of regions as a timed game, we cannot use diagonal guards. However, in the construction, we want to use these guards to constrain the time-elapsing: by firing a transition, the obtained region is a time successor region on which some clocks have been reset. Since the time-elapsing does not affect the diagonal component of the guard (the clocks evolve at the same speed), they are not discriminating and we can "forget" the diagonal atomic constraints to define the guards in the region game.

Definition 1.20. Let $\mathcal{A} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta \rangle$ be an RTG such that all clocks are bounded by M. By letting $\mathsf{Reg}(\mathcal{X}, \mathsf{M})$ be the set of regions for the set of clocks $\mathcal{X}$ bounded by M, the *region game*, is the RTG $\mathcal{R}(\mathcal{A}) = \langle Q_{\mathsf{Min}}, Q_{\mathsf{Max}}, Q_t, Q_u, \mathcal{X}, \Delta' \rangle$ whose locations, called *region states*, are $Q_{\mathsf{Min}} = L_{\mathsf{Min}} \times \mathsf{Reg}(\mathcal{X}, \mathsf{M})$, $Q_{\mathsf{Max}} = L_{\mathsf{Max}} \times \mathsf{Reg}(\mathcal{X}, \mathsf{M})$, $Q_t = L_T \times \mathsf{Reg}(\mathcal{X}, \mathsf{M})$ and $Q_u = L_u \times \mathsf{Reg}(\mathcal{X}, \mathsf{M})$, and Δ' is defined by all transitions $\big((\ell, r), g'', Y, (\ell', r')\big)$ with $(\ell, g, Y, \ell') \in \Delta$ such that the model of the guard g'' (i.e. all valuations ν such that $\nu \models g''$) represents a region r'', time successor of r such that r'' satisfies the guard g, and r' is the region obtained from r'' by resetting all clocks of Y.

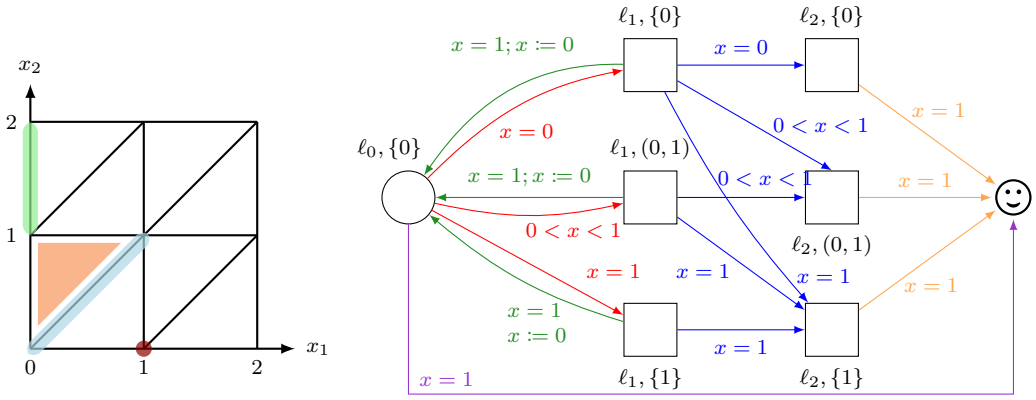


Figure 1.6 – On the left, regions for two clocks $\{x_1, x_2\}$ bounded by 2. On the right, the restriction (according to locations ℓ_0, ℓ_1, ℓ_2 and ☺) of the region game from the RTG depicted in Figure 1.3.

An example of a region game is depicted on the right in Figure 1.6. Given an RTG, the number of locations in its region game depends on the number of regions over its clock.

We call a *region path* a finite or infinite sequence of transitions in a region game, and we again denote by π such paths. A play ρ in $\mathcal{A}$ is projected on a region path π, with a similar definition as the projection on paths: we again say that ρ follows the region path π. It is important to notice that even if π is a cyclic region path (i.e. starts and ends in the same location as the region game), there may exist plays following it in $\mathcal{A}$ that are not cyclic because regions are sets of valuations.

1.2.3. Reachability objective and attractor

Knowing the region game of an RTG, we can solve it by applying another classical tool to solve a (finite) game under a reachability objective: an *attractor*. In particular, we know that:

Theorem 1.21 [JT07, Theorem 21]**.** *Deciding if there exists a winning deterministic strategy* Min *in an RTG is* EXPTIME*-complete for at least two clocks.*

The algorithm to solve an RTG consists of a computation of an attractor strategy in the region game that is done in exponential time. For the hardness result, authors used a reduction to a *countdown game*[5] that is known as EXPTIME-complete

[5] A countdown game is a (finite) weighted game such that at each turn, the first player chooses a weight in the set of weights induced by the outcoming transitions, and then the second player responds by applying a transition with this weight. From an initial node with an initial budget, solving this game asks if the first player can reach any node with an exact budget of 0.

to solve it by [JSL08, Theorem 4.5].

Attractor computation and attractor strategies Deciding the winner of classical reachability games, i.e. turn-based two-player games over a finite graph under a reachability objective, can be done in polynomial time [Tho95; GTW02] via an attractor computation. We introduce this notion for finite games, but it is easily extended to the semantics of RTG.

We consider a finite reachability game $\langle S, E\rangle$ where S is a finite set of states partitioned between players Min and Max, and $E \subseteq S \times S$. We recall the classical notion of the attractor of a player towards a set of states $S_T \subseteq S$: an *attractor computation* defines the sequence of winning states sets within i steps, $(W_i)_i \subseteq S$, by induction on $i \in \mathbb{N}$: $W_0 = S_T$, and for all $i \in \mathbb{N}$, $s \in W_{i+1}$ if and only if $s \in W_i$ or

1. s belongs to Min, and there exists an edge $s \to s'$ such that $s' \in W_i$;
2. s belongs to Max, and for all edges $s \to s'$, we have $s' \in W_i$.

It is well-known that this sequence computed as a fixpoint converges in linear time in $|S| + |E|$.

The crucial point for our work is that this computation defines a memoryless strategy, called an *attractor strategy* ensuring that a target state is reached in less than $|S|$ steps from every state. Moreover, this strategy is computable in polynomial time in $|S| + |E|$. In particular, by extending this notion to RTG, we fix:

Theorem 1.22 [Tho95]. *Let $\mathcal{A} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta\rangle$ be an RTG and an initial configuration (ℓ, ν), if* Min *has a winning strategy from (ℓ, ν), then there exists an* attractor *strategy, i.e. a winning memoryless strategy σ. Moreover, it is such that, for all τ deterministic strategies of* Max*,*

$$|\mathsf{Play}((\ell, \nu), \sigma, \tau)| \leq |L \times \mathsf{Reg}(\mathcal{X}, \mathsf{M})|\,.$$

About concurrent reachability timed games Even if we focus later on turn-based reachability timed games, we can extend the reachability problem to concurrent timed games. In this case, the problem of the existence of a winning strategy for Min is decidable, but a winning strategy for Min (unlike for RTG) may require infinite memory to control the element of surprise of the game [Alf+03] as illustrated in the following example.

Example 1.23. We consider the concurrent reachability timed game depicted in Figure 1.2 where $L_T = \{\ell_1\}$ in which Min plays with action a and Max with action b. In this game, Min wins only with an infinite memory deterministic

strategy. As long as the decision of Min is not winning, its next decision contains a delay closer to 0. Thus, a winning deterministic strategy for Min maintains an extra global value of the clock to choose the right delay when it is in ℓ_0. Formally, if the current configuration is (ℓ_0, ν) for a valuation ν, the strategy uses a new infinitely precise global clock without a reset, denoted y and chooses $(a, 1/2^{\nu(y)})$ at each turn. This strategy is winning since either Max always chooses a decision with a smaller delay and the conformed play is *time-convergent*[6], or there exists a moment when Min chooses a smaller delay than Max, and the play reaches ℓ_1. ⌟

1.3. Weighted timed games

We now define the principal object of our study, weighted timed games, as a weighted extension of RTGs. In particular, Min wants to win in the RTG and minimises a measure function encoded by a weight function that models the performance of the system. This class of games can be seen as an infinite shortest-path game where Min wants to reach a target location while minimising the weight to do it. Formally, we define a weighted timed game as follows:

Definition 1.24. A *weighted timed game* (WTG for short) is a tuple $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt}\rangle$ where $\langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta\rangle$ is its underlying RTG and $\mathsf{wt}\colon \Delta \uplus L \to \mathbb{Z}$ is a weight function.

Figure 1.7 illustrates a WTG where unlabelled edges correspond to weight 0. Again, the presence of urgent locations is also unusual but increases modelling capabilities by allowing for such urgent locations when we limit WTGs with only one clock. The weight of an urgent location is never used and will thus not be given in drawings: instead, urgent locations will be displayed with u inside (as for RTGs).

The semantics of a WTG is defined as a weighted extension of the semantics of its underlying RTG. Formally, we fix:

Definition 1.25. The *semantics* of $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt}\rangle$, denoted by $[\![\mathcal{G}]\!] = \langle \mathsf{Conf}^{\mathcal{G}}, E, \mathsf{wt}\rangle$ is a weighted transitions system such that $\langle \mathsf{Conf}^{\mathcal{G}}, E\rangle$ is the semantics of the underlying RTG, and for all configurations $(\ell, \nu) \in \mathsf{Conf}^{\mathcal{G}}$, $\mathsf{wrt}(\ell, \nu) = \mathsf{wt}(\ell)$ as well as for all edges $(\ell, \nu) \xrightarrow{\delta,t} (\ell', \nu') \in E$, $\mathsf{wrt}\big((\ell, \nu) \xrightarrow{\delta,t} (\ell', \nu')\big) = \mathsf{wt}(\delta)$.

Since both semantics are only separated by the weight function, the definition of plays, paths and deterministic strategies are easily extended to WTGs.

[6] To avoid an undesirable behaviour when Max always plays a decision with a null delay and blocks the game, we suppose that Min wins when plays reach a target location or the game produces an infinite play in which Min makes only a finite number of decisions and the global time is finite. This hypothesis is called Min *blameless* in [Alf+03].

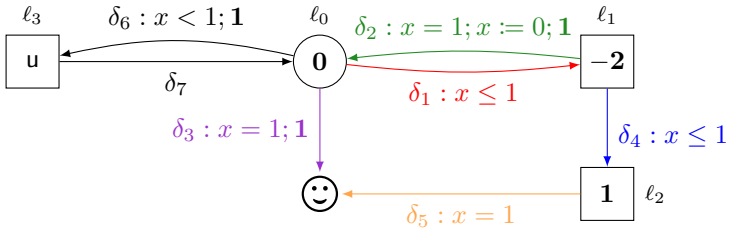


Figure 1.7 – A (one-clock) WTG associated with the RTG depicted in Figure 1.3. Its weights are depicted in bold font, and the missing ones are 0.

Given a WTG $\mathcal{G}$, we denote by W_{loc} (resp. W_{tr}, and W_{e}) the maximal weight in absolute values of locations (resp. of transitions, and edges) of $\mathcal{G}$, i.e.

$$W_{\mathsf{loc}} = \max_{\ell \in L} |\mathsf{wt}(\ell)| \qquad W_{\mathsf{tr}} = \max_{\delta \in \Delta} |\mathsf{wt}(\delta)| \qquad W_{\mathsf{e}} = \mathsf{M}\, W_{\mathsf{loc}} + W_{\mathsf{tr}}$$

where M is the bound over clocks. Moreover, we denote by $W = \max(W_{\mathsf{loc}}, W_{\mathsf{tr}})$ the maximal weight constant of $\mathcal{G}$.

1.3.1. Deterministic value

A main difference between WTGs and RTGs is the objective of Min: in a WTG, Min wants to reach a target configuration while minimising the cumulated weight up to the target location. To define the winner in this class of games, we first associate with every finite play $\rho = (\ell_0, \nu_0) \xrightarrow{\delta_0, t_0} (\ell_1, \nu_1) \xrightarrow{\delta_1, t_1} \cdots (\ell_k, \nu_k)$ its cumulated weight, taking into account both discrete and continuous costs:

$$\mathsf{wt}_\Sigma(\rho) = \sum_{i=0}^{k-1} [t_i \times \mathsf{wt}(\ell_i) + \mathsf{wt}(\delta_i)] \,.$$

A maximal play ρ (either infinite or trapped in a deadlock that is, necessarily, a target configuration) is associated with a *payoff* $\mathbf{P}(\rho)$ defined as follows: the payoff of an infinite play (meaning that it never visits a target location) is $+\infty$, while the payoff of a finite play is $\mathsf{wt}_\Sigma(\rho)$. Given a finite path π, the set of weights of the plays following π is known to be an interval of $\mathbb{R}$ that is closed when all the guards along the path are closed constraints, by [Beh+01].

A deterministic strategy $\sigma \in \mathsf{dStrat}_{\mathsf{Min}}$ guarantees a certain (deterministic) value, against all possible strategies of the opponent (a symmetric definition can be given for strategies of Max): for all configurations (ℓ, ν), we let

$$\mathsf{dVal}^\sigma_{\mathcal{G}}(\ell, \nu) = \sup_{\tau \in \mathsf{dStrat}_{\mathsf{Max}}(\mathcal{G})} \mathbf{P}\big(\mathsf{Play}((\ell, \nu), \sigma, \tau)\big) \,.$$

Then, for all configurations (ℓ, ν), we define the *deterministic value* from (ℓ, ν) under the point of view of Min by

$$\overline{\mathsf{dVal}}_{\mathcal{G}}(\ell, \nu) = \inf_{\sigma \in \mathsf{dStrat}_{\mathsf{Min}}(\mathcal{G})} \mathsf{dVal}^{\sigma}_{\mathcal{G}}(\ell, \nu)\,.$$

Similarly, we define the deterministic value from (ℓ, ν) under the point of view of Max by

$$\underline{\mathsf{dVal}}_{\mathcal{G}}(\ell, \nu) = \sup_{\tau \in \mathsf{dStrat}_{\mathsf{Max}}(\mathcal{G})} \inf_{\sigma \in \mathsf{dStrat}_{\mathsf{Min}}(\mathcal{G})} \mathbf{P}\big(\mathsf{Play}((\ell, \nu), \sigma, \tau)\big)\,.$$

Since, for all configurations (ℓ, ν), WTGs are known to be determined [Bri+15, Theorem 2.2], i.e. $\underline{\mathsf{dVal}}_{\mathcal{G}}(\ell, \nu) = \overline{\mathsf{dVal}}_{\mathcal{G}}(\ell, \nu)$, we denote by $\mathsf{dVal}_{\mathcal{G}}(\ell, \nu)$ this value. To lighten the notations, when the WTG is clear from the context, we may omit the subscript $\mathcal{G}$.

Finally, we say that a deterministic strategy σ of Min is *optimal* if, for all configurations (ℓ, ν),

$$\mathsf{dVal}^{\sigma}_{\mathcal{G}}(\ell, \nu) \leq \mathsf{dVal}_{\mathcal{G}}(\ell, \nu)\,.$$

Because of the infinite nature of the WTG, optimal deterministic strategies may not exist: for example, a player may want to let time elapse as much as possible, but with a delay $t < 1$ because of an open guard, preventing them to obtain the optimal deterministic value. We will see in Example 1.26 that this situation can happen even when all guards contain only closed constraints. Thus, we naturally extend the definition to *almost-optimal strategies*, taking into account small possible errors: for all $\varepsilon > 0$, we say that a deterministic strategy σ of Min is ε-*optimal* if, for all configurations (ℓ, ν),

$$\mathsf{dVal}^{\sigma}_{\mathcal{G}}(\ell, \nu) \leq \mathsf{dVal}_{\mathcal{G}}(\ell, \nu) + \varepsilon\,.$$

Example 1.26. We consider the WTG depicted in Figure 1.7 and we compute the deterministic value for the configuration $(\ell_0, 0)$. First, we note that Min has no interest in following the cyclic path $\ell_0 \xrightarrow{\delta_6} \ell_3 \xrightarrow{\delta_7} \ell_0$ since all plays following it have weight 1. Jumping directly to the target location via δ_3 leads to a weight of 1. But Min can do better: from valuation 0, by jumping to ℓ_1 after a delay $t \leq 1$, it leaves a choice to Max to either jump to ℓ_2 and then to the target leading to a total weight of $1 - t$, or to loop back in ℓ_0 thus closing a cyclic play of weight $-2\,(1 - t) + 1 = 2t - 1$. If t is chosen greater than $1/2$, the weight of the cyclic path is greater than 0, and Max will benefit from it by increasing the total weight. If t is chosen smaller than $1/2$, the weight of the cyclic path is negative, and Max will prefer to go to the target location to obtain a weight $1 - t$ close to 1, not very beneficial to Min. Thus, Min prefers to play just below $1/2$, for example, at $1/2 - \varepsilon$. In this case, Max will choose to go to the target with a total weight of $1/2 + \varepsilon$. The

deterministic value in configuration $(\ell_0, 0)$, is thus $\mathsf{dVal}(\ell_0, 0) = 1/2$.

Not only Min does not have an optimal strategy (but only ε-optimal ones, for every $\varepsilon > 0$), but needs memory to play ε-optimally. Min cannot play *ad libitum* transition δ_1 with a delay $1/2 - \varepsilon$: in this case, Max would prefer staying in the cycle, thus avoiding the target location. Thus, Min will play the transition δ_1 at least $1/2\varepsilon$ times so that the cumulated weight of all the cycles is below $-1/2$, in which case Min can safely use transition δ_3 still earning $1/2$ in total. ⌟

We also extend the definition of region games for WTGs from RTGs. The region game of a WTG is defined by the region game for the underlying RTG equipped with a weight function inherited from the WTG. In particular, we denote by $\mathcal{R}(\mathcal{G})$ the region game of the WTG $\mathcal{G}$. Moreover, by projecting away the region information of $\mathcal{R}(\mathcal{G})$, we simply deduce:

Lemma 1.27 ([Bus19, Lemma 8.1])**.** *Let $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt}\rangle$ be a WTG. For all $\ell \in L$, regions r and $\nu \in r$, $\mathsf{dVal}_{\mathcal{G}}(\ell, \nu) = \mathsf{dVal}_{\mathcal{R}(\mathcal{G})}((\ell, r), \nu)$.*

Clock bounding In WTGs, we do not know if we can upper bound clocks without loss of generality since techniques defined for timed automata and RTGs cannot be directly applied. Indeed, when clocks are not bounded, Max can stay longer and longer in a location with a positive weight. In this case, the deterministic value may be equal to $+\infty$ even if Min guarantees that a target location is reached. Trying to detect and avoid these situations in our turn-based case seems difficult in the presence of negative weights since the opportunities of Max crucially depend on the configurations of deterministic value $-\infty$ that Min could control afterwards. As the problem of detecting such configurations (for all WTGs) is undecidable [Bus19, Proposition 9.2], this gives additional evidence to motivate the decision to focus only on WTGs with bounded clocks.

Remark. When we consider concurrent WTGs, this problem does not hold since this new power of Max is compensated by always giving Min a chance to leave the location by playing a fresh action (see [Bou+04a] for details). ◁

1.3.2. Deterministic value problem

Solving a WTG consists computing its deterministic value function, i.e. the deterministic value for all configurations, as well as an ε-optimal deterministic strategy (for all $\varepsilon > 0$). Introduced in [BJM15], the corresponding decision problem is the *deterministic value problem*: given a WTG $\mathcal{G}$, a configuration (ℓ, ν), and a threshold $\alpha \in \mathbb{Q} \cup \{-\infty\}$, we want to know whether $\mathsf{dVal}_{\mathcal{G}}(\ell, \nu) \leq \alpha$.

Even if we restrict ourselves to only non-negative weight functions, the deterministic value problem is known undecidable by reducing the problem to the halting

problem of two-counter machines[7] that is known undecidable (by [Min67, Theorem 1]).

Theorem 1.28. *The deterministic value problem is undecidable in WTG:*

- *with at least three clocks and only non-negative weights [BBR05; BBM06];*
- *with at least two clocks and arbitrary weights [Bri+14].*

We can also define the deterministic value problem for $+\infty$, i.e. given a WTG $\mathcal{G}$, we want to know whether $\mathsf{dVal}_{\mathcal{G}}(\ell, \nu) = +\infty$ for a given configuration (ℓ, ν). This problem is purely qualitative [Bou+04a; BMR17] and can be thus be solved with the region game: indeed, $\mathsf{dVal}_{\mathcal{G}}(\ell, \nu) = +\infty$ if and only if Min has no deterministic strategies to reach a target location of L_T. This is thus a reachability objective, where weights are useless. Moreover, Max has a strategy that guarantees that no plays reach the target locations of L_T from any configuration (ℓ, ν) such that $\mathsf{dVal}_{\mathcal{G}}(\ell, \nu) = +\infty$. **We thus rule out this case by supposing, in the following, that no configurations have a deterministic value** $+\infty$: such configurations can be removed in the region game by strengthening the guards on transitions.

1.3.3. Classes of weighted timed games to recover decidability

In the following chapters, we will focus on some classes of WTGs for which we know that the deterministic value problem is decidable. In the one-player case (i.e., automata), computing the optimal value and an ε-optimal strategy for weighted timed automata is known to be PSPACE-complete [Bou+07]. However, when we focus on two-player games, two orthogonal research directions to recover decidability are studied: reducing the number of clocks or restricting cyclic paths in the region game.

1.3.3.1. Shortest path games

When we restrict the number of clocks, it is natural to study WTGs without clocks, generally called *shortest-path games* (or *min-cost reachability games* [Bri+17]). Since there are no clocks, we remove the components regarding the guard and clock reset from transitions. Moreover, we only provide the weight of transitions

[7] A two-counter machine is an abstract machine with two registers that contain only non-negative integers, and it is controlled by a list of arithmetic instructions to move inside the machine and update the value of the registers (by increment or decrement). The halting problem asks if, from an initial configuration of the machine, we can execute the whole list of instructions.

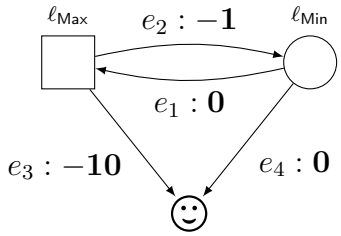


Figure 1.8 – An SPG, where Min requires memory to play optimally.

and not of locations (usually called vertices or states in this setting). To summarise, here is a self-contained definition of this class of game.

Definition 1.29. A *shortest-path game* (SPG for short) is a tuple $\langle V_{\mathsf{Max}}, V_{\mathsf{Min}}, V_T, E, \mathsf{wt}\rangle$ where $V \coloneqq V_{\mathsf{Max}} \uplus V_{\mathsf{Min}} \uplus V_T$ is a finite set of vertices partitioned into the sets V_{Max} and V_{Min} of Max and Min respectively, and a set V_T of target vertices, $E \subseteq V \setminus V_T \times V$ is a set of *directed edges*, and $\mathsf{wt}\colon E \to \mathbb{Z}$ is the *weight function*, associating an integer weight with each edge.

Example 1.30. We consider $\mathcal{G}$ to be the SPG depicted in Figure 1.8. The deterministic value of both locations (except the target location) is -10 since Min guarantees it from ℓ_{Min} by jumping 10 times into ℓ_{Max} before going to the target location ☺. For Max, we remark that jumping directly to the target ensures him the deterministic value -10. However, Min cannot achieve this optimal value by playing without memory: it results either in a weight 0 (by directly going to the target), or in a weight $+\infty$ (by always going to ℓ_{Max}) since Max has the opportunity to keep Min in the negative cycle forever, thus never reaching the target. ⌟

The deterministic value problem in shortest path games has previously been studied [Bri+17; BMR17]: both players have optimal strategies such that Max does not require memory, while Min requires *pseudo-polynomial*[8] size memory.

Theorem 1.31 [Bri+17]**.** *Let $\mathcal{G}$ be a shortest path game.*

1. *Computing all deterministic values (i.e. for all $\ell \in V$) can be done in polynomial time in $|V|$ and W.*
2. Max *has a memoryless optimal deterministic strategy.*

1.3.3.2. One-clock weighted timed games

When we consider a restriction on the number of clocks, we note that a single clock is often sufficient for modelling purposes. In particular, we want to focus

[8] It is polynomial when weight constants are encoded in unary instead of binary.

on one-clock WTGs, i.e. WTGs defined by $\langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \{x\}, \Delta, \mathsf{wt}\rangle$. Since one-clock WTGs contain only one clock x, their guards can be seen as interval constraints on x, i.e. intervals of real values with closed or open bounds that are natural numbers or $+\infty$.

Regions As there is a single clock, regions partition the set of valuations $[0, \mathsf{M})$ and their number is polynomial instead of exponential. Let $\mathcal{G}$ be a given one-clock WTG, we define regions by a construction inspired by Laroussinie, Markey, and Schnoebelen [LMS04], and we call regions of $\mathcal{G}$ the set

$$\mathsf{Reg}(\mathcal{G}, \mathsf{M}) = \{(M_i, M_{i+1}) \mid 0 \leq i \leq k-1\} \cup \{\{M_i\} \mid 0 \leq i \leq k\}$$

where $M_0 = 0 < M_1 < \cdots < M_k$ are all the endpoints of the intervals appearing in the guards of $\mathcal{G}$ (to which we add 0 if needed). As usual, if r is a region, then the time successors of valuations in r form a finite union of regions, and the reset $r[x \coloneqq 0] = \{0\}$ is also a region.

Deterministic value problem We introduce one-clock WTGs to recover the decidability of the deterministic value problem. When only non-negative weights are considered, decidability has been proven in [Bou+06] and later improved in [Rut11; HIM13] to obtain exponential time algorithms. Despite several recent works, the decidability status of one-clock WTGs with arbitrary weights was still open.

Positive results obtained for one-clock WTGs with non-negative weights are based on a reduction to so-called *simple WTGs*, where the underlying timed automata contains no guard, no reset, and the clock value along the execution exactly spans the $[0, 1]$ interval. In simple WTG, it is possible to compute (in exponential time) the whole deterministic value function starting at time 1, and going back in time until 0 [Bou+06; Rut11]. Another technique that we will not explore further in this work, consists of using the paradigm of strategy iteration [HIM13], leading to an exponential-time algorithm too. A PSPACE lower-bound is also known for related decision problems [FIS20].

A first extension to negative weights in [Bri+14] uses a different approach to obtain decidability in pseudo-polynomial time with arbitrary weights on transitions and rates on locations restricted to two values chosen in the set $\{-d, 0, d\}$, with $d \in \mathbb{N}$.

Recent works [Bri+15; Bri+22] extend the positive results of simple WTGs to arbitrary weights, yielding decidability of *reset-acyclic* one-clock WTGs, i.e. that do not contain cyclic paths with a transition with a reset. In particular, the authors prove that the deterministic value function is continuous and piecewise affine on each region and is computable in pseudo-polynomial time.

Theorem 1.32 [Bri+22, Theorem 6.5]**.** *If $\mathcal{G}$ is a reset-acyclic one-clock WTG, then for all locations ℓ, the piecewise affine mapping $\nu \mapsto \mathsf{Val}_{\mathcal{G}}(\ell, \nu)$ is computable in time polynomial in $|L|$ and W.*

In [Bri+22], this result is extended to consider cyclic paths containing reset transitions when the weight of all the plays following them is not arbitrarily close to 0 and negative.

1.3.3.3. Divergent weighted timed games

Finally, we conclude this chapter with a (large) class of WTGs that does not constrain the number of clocks to regain the decidability of the deterministic value problem. This class is obtained by enforcing a semantical property of divergence (originally called *strictly non-Zeno cost* when only dealing with non-negative weights [Bou+04a]): it asks that every play following a cyclic path in the region game has a weight far from 0. Formally, a cyclic region path π of $\mathcal{R}(\mathcal{G})$ is said to be a *positive cycle* (resp. a *negative cycle*) if every finite play ρ following π satisfies $\mathsf{wt}_\Sigma(\rho) \geq 1$ (resp. $\mathsf{wt}_\Sigma(\rho) \leq -1$).

Definition 1.33 [BMR17]**.** A WTG is *divergent* if every cyclic region path is positive or negative.

In [BMR17], it was shown that this definition is equivalent to requiring that for all strongly connected components (SCC) S of the graph of $\mathcal{R}(\mathcal{G})$, either every cycle π inside S is positive (we say that the SCC is positive), or every cycle π inside S is negative (we say that the SCC is negative). The best computability result in this setting is:

Theorem 1.34 [BMR17]**.** *Let $\mathcal{G}$ be a divergent WTG. Then, for all locations ℓ, the piecewise affine mapping $\nu \mapsto \mathsf{Val}_{\mathcal{G}}(\ell, \nu)$ is computable in triply-exponential-time.*

Towards almost-divergent WTG The class of divergent WTGs can be extended by considering cyclic region paths with a weight of exactly 0. Such a class of WTGs is called *almost-divergent WTGs* (or *non-Zeno WTGs*) in [BJM15; BMR18]. Even if the deterministic value problem is still undecidable, an approximation schema exists to compute an approximation of the deterministic value function with piecewise affine and continuous functions over regions [BMR18].

Part I.

Deterministic Value

Introduction

As illustrated in the previous chapter, the deterministic value of WTGs has already been well-studied. From the first result of undecidability [BBR05], some classes have been studied to recover the decidability like divergent WTGs or one-clock WTGs with non-negative weights (see Table I.1). Despite several recent works, a question remained open: the decidability of the deterministic value problem when we restrict to a class of one-clock WTGs with arbitrary weights.

In this part, we aim to prove that the deterministic value functions in a one-clock WTG (with arbitrary weights) can be computed in exponential time (if weights are encoded in unary and not binary). Moreover, we introduce some main ingredients, which will have other uses: a fixpoint characterisation for the deterministic value and switching strategies.

Value iteration algorithm Solving games usually uses a fixpoint iteration algorithm, e.g. attractor computation [Tho95; GTW02], strategy iterations [GS09], and value iterations [ZP96]. Thus, it is natural to look for a fixpoint characterization of the deterministic value function to recover the decidability of the deterministic value problem in some classes of WTGs. It is also the main ingredient in divergent WTGs with non-negative (and arbitrary) weights: the iteration of a given operator over an initial (continuous) well-chosen function is used. Intuitively, this operator locally defines the best decision for the current player by computing its best decision at horizon 1. By iterating this operator, the value iteration algorithm computes the best option of the current player at the bounded horizon: after the

	SPGs	one-clock	divergent	all
$\mathbb{N}$	PTIME [Knu77; Kha+08]	EXPTIME [Rut11; HIM13]	EXPTIME [Bou+04a; ABM04]	undecidable [BBR05]
$\mathbb{Z}$	pseudo-polynomial [Bri+17]	pseudo-exponential Theorem 4.1	3-EXPTIME [BMR17]	undecidable [Bri+14]

Table I.1 – Solving the deterministic value problem in weighted timed games where weights are encoded in binary. We recall that pseudo-polynomial (resp. pseudo-exponential) means that the time complexity is polynomial (resp. exponential) when weights are encoded in unary.

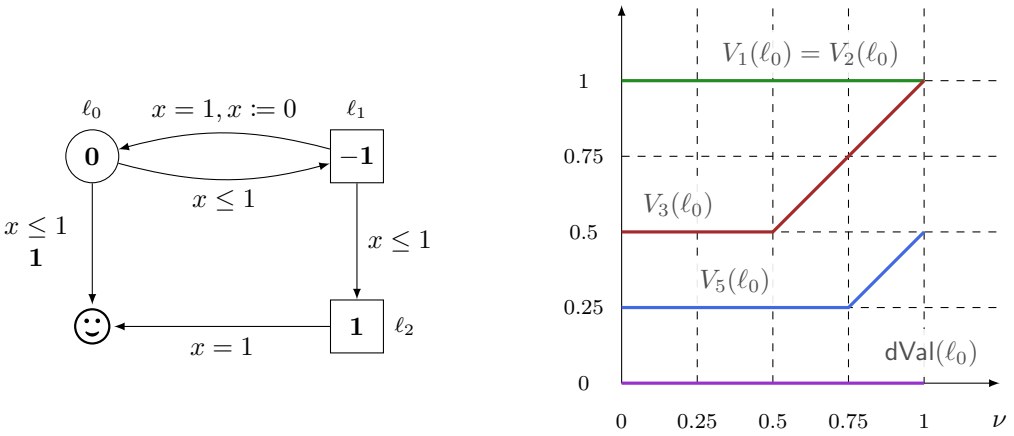


Figure I.1 – On the left is a one-clock WTG in which the value iteration algorithm does not converge in finite time. On the right, the three iterations first of the value iteration algorithm for ℓ_0 as well as the deterministic value function of ℓ_0 is the limit of the function given by the value iteration algorithm.

ith iteration, it knows the best decisions during the i first steps.

The proof of decidability for divergent WTGs is based on the finite convergence of the value iteration algorithm. However, when we consider one-clock WTGs, this algorithm may not finitely converge. In contrast to divergent WTGs, it does not provide an algorithm to compute the deterministic value function in all one-clock WTGs.

Example I.1. We consider the one-clock WTG depicted in Figure I.1, and we want to apply the value iteration algorithm to compute the deterministic value function of ℓ_0. By letting $\varepsilon > 0$ and considering an analogous reasoning as in Example 1.26, we can see that Min can use the cycle to guarantee the payoff ε instead of 1 by directly reaching the target, i.e. the deterministic value function is equal to 0.

Unfortunately, it is this behaviour that the value iteration algorithm can not pass through: it will compute each approximation (with $\varepsilon = 1/2^i$ for all $i \geq 0$) of the deterministic value function, which explains why the algorithm does not converge in finite time when we symbolically compute the deterministic value function from the value iteration algorithm. First, we note that for all $\ell \in L$, $V_0(\ell, \nu) = +\infty$ except ☺.

Now, to compute V_1 for a location, we will locally optimise the choice of the player (minimising if the location belongs to Min, maximising otherwise) according to the time spent in the location, the transition chosen and the value of V_0 obtained from the reached configuration. Let ν be a valuation such that $\nu(x) \leq 1$. In

particular, from ℓ_1, Max can not reach the target location within 1 step, so

$$V_1(\ell_1, \nu) = \max(V_0(\ell_0, \nu), V_0(\ell_2, \nu)) = +\infty\,.$$

In ℓ_2, Max must reach the target location in one step, thus (since there are no choices for Max), for all $i \geq 1$, we have

$$V_i(\ell_2, \nu) = \sup_{t \mid t+\nu \leq 1} \big[t \times 1\big] = 1 - \nu\,.$$

In ℓ_0, Min takes the best decision between going to the target location (after a possible delay) or reaching ℓ_1 where $V_0(\ell_1, \nu) = +\infty$. In particular, Min chooses to go to the target location and

$$V_1(\ell_0, \nu) = \min\Big(\inf_{t \mid t+\nu \leq 1} \big[t \times 0 + 1\big], \inf_{t \mid t+\nu \leq 1} \big[t \times 0 + \infty\big]\Big) = 1\,.$$

Next, to compute V_2, we see that only the value function of ℓ_1 needs to be updated. Indeed, the computation for ℓ_0 remains identical since the value function V_1 in ℓ_1 has not changed. In this case, we remark that Max must reach the target location by ℓ_0 or ℓ_2. In particular, by letting $\nu \leq 1$ be a valuation, we have

$$V_2(\ell_1, \nu) = \max\Big(\sup_{t \mid t+\nu = 1} \big[t \times (-1) + V_1(\ell_0, 0)\big], \sup_{t \mid t+\nu \leq 1} \big[t \times 1 + V_1(\ell_2, \nu + t)\big]\Big)\,.$$

Since $t + \nu = 1$ when $t = 1 - \nu$, we deduce that

$$V_2(\ell_1, \nu) = \max\Big(\nu, 1 - \nu\Big) = \begin{cases} 1 - \nu & \text{if } \nu \leq \frac{1}{2}; \\ \nu & \text{otherwise.} \end{cases}$$

Then, we compute the new value function in ℓ_0 where Min can choose between making once the cyclic path or going to the target. In particular, by the computation of V_1, we deduce that

$$V_3(\ell_0, \nu) = \min\Big(1, \inf_{t \mid t+\nu \leq 1} \big[t \times 0 + V_2(\ell_1, \nu + t)\big]\Big)\,.$$

To compute the infimum, we note that $\nu \mapsto V_2(\ell, \nu)$ is decreasing when $\nu \leq 1/2$, and is increasing otherwise. In particular, to minimise the value function, Min wants to wait until $1/2$ when $\nu \leq 1/2$, and directly take the transition, otherwise.

Thus, we deduce that

$$V_3(\ell_0,\nu) = \min \begin{cases} 1 & \\ \frac{1}{2} & \text{if } \nu \leq \frac{1}{2}; \\ \nu & \text{otherwise;} \end{cases} \quad \text{i.e.} \quad V_3(\ell_0,\nu) = \min = \begin{cases} \frac{1}{2} & \text{if } \nu \leq \frac{1}{2}; \\ \nu & \text{otherwise;} \end{cases}$$

as depicted in red on the right of Figure I.1.

Finally, by continuing the reasoning between ℓ_0 and ℓ_1, we will remark that $V_{2i+1}(\ell_0,\nu)$ is equal to $\frac{1}{2^i}$ over $(0, \frac{2^i-1}{2^i})$ and ν otherwise. Thus, the computed function converges to the deterministic value function but not in finite time. ⌟

Nevertheless, the value iteration algorithm induces a fixpoint characterisation of the deterministic value that gives us tools to achieve the decidability of our problem. However, the proof of the claim that the deterministic value function is indeed the greatest fixpoint of the operator[1], given in [Bou+04a; Bou16], contains flaws. Indeed, to prove that this limit is the deterministic value function, the authors suppose that the limit of the iterates of the operator is a continuous function of $\mathbb{R}_{\geq 0}$, whereas the limit of a sequence of continuous functions (i.e. functions computed by iterating the operator) may not be continuous. Thus, this fact needs to be proven: it is the object of Chapter 2 for one-clock WTGs and divergent WTGs.

Switching strategies for Min Another problem of our interest, especially when the deterministic value function is computable, is the synthesis of optimal (or almost-optimal) strategies for both players. In particular, to obtain an efficient algorithm, we look for the simplest class of strategies as possible. However, in SPGs, Min already needs memory to play optimally, as illustrated in Example 1.30.

Given an SPG, we know from [Bri+17] that Min can optimally play by following a *switching strategy* looping in the negative cycle long enough so that, no matter how, he reaches the target finally. In particular, the switching strategy (a central notion in our work) sequentially satisfies two objectives of Min (each without memory): accumulating enough weights before reaching a target location. We note that an optimal strategy with respect to the deterministic value in SPGs uses pseudo-polynomial memory.

Given a general WTG, knowing if there exists an almost-optimal switching strategy with respect to the deterministic value is an open question. Since reaching a target location can always be guaranteed by an attractor strategy (see Section 1.2.3), the main difficulty is the definition of the memoryless deterministic

[1] Fortunately, from the results established in [BMR17], even if this point is not mentioned, the necessary claim can be recovered in the case of divergent WTGs (at least in the turn-based case that we consider here, and not necessarily in the concurrent case studied in [Bou+04a]) even in presence of both negative and non-negative weights.

strategy that reaches a negative cycle. In particular, the existence of such an almost-optimal switching strategy with respect to the deterministic value implies that Min can reach a negative cycle, and iterate it while preserving a weight less than a constant.

Concerning the divergent WTGs, the first part of the switching strategy, i.e. reaching a negative cycle and staying inside, can be defined by locally taking the best decision. In particular, we use the fixpoint characterisation of the deterministic value function. We will explicitly give this precise construction in Chapter 3.

Decidability of the deterministic value for one-clock WTGs The crucial ingredient to obtain decidability for non-negative weights or reset-acyclic one-clock WTGs is to limit the number of reset transitions taken along a play. This is no longer possible in the presence of cycles of negative weights containing a reset. There, Min may need to iterate cycles for a number κ of times that depends on the desired precision ε on the deterministic value (to play ε-optimally, Min needs to cycle $O(1/\varepsilon)$ times, see Example 1.26). To rule out these annoying behaviours, we rely on three main ingredients:

- As there is a single clock, a cyclic path ending with a reset corresponds to a cyclic play. We define the deterministic value of such a cycle to identify which player may benefit from iterating it.
- Using the classical region graph construction, we prove that the deterministic value function is continuous on the closure of region intervals. From this result, we deduce that Max has an optimal memoryless deterministic strategy with respect to the deterministic value, that avoids cycles whose deterministic value is negative.
- We introduce a partial unfolding preserving the deterministic value of the one-clock WTG to obtain an acyclic (one-clock) WTG, for which decidability of the deterministic value problem is known [ABM04]. To do so, we rely on the existence of almost-optimal switching strategies for Min, allowing us to limit the depth of exploration. Also, we keep track of cycles encountered and handle them according to their deterministic value. Finally, we transport the previous result on the existence of a "smart" optimal strategy with respect to the deterministic value for Max in the context of this unfolding.

We present the detailed proofs in Chapter 4.

Tools for solving timed games In the literature, we can find some tools designed to model-check timed systems modelled by timed automata under properties expressed by real-time TCTL logic [AH91; Ost92; BMN00]. For instance, we can cite

UPPAAL[2] [BDL04] (which is the most famous model-checker over timed automata under reachability properties (notably)) KRONOS[3] [Yov97; Boz+98] (which implements a symbolic model-checking algorithm for timed automata), PAT[4] [LSD08] (which implements the compositional verification for *event-recording automata*[5]), or TCHECKER[6] [HSW12] (which checks emptiness for Büchi timed automata).

From the result of decidability in RTGs, a symbolic algorithm allowing tool development has been proposed in [Beh+07] to provide a tool for synthesis in timed games. For instance, we note the existence of UPPAAL-TIGA[7], a controller synthesis extension of UPPAAL. Some extensions have been proposed like ECDAR [Dav+10] to improve compositionality inside UPPAAL-TIGA, or UPPAAL-STRATEGO [Dav+15] to study stochastic weighted timed games (that are known as undecidable) using learning mechanisms. More recently, the extension of PRISM[8] for games has also studied turn-based probabilistic timed games that are an extension of *probabilistic timed automata*[9] in a two-player turn-based game under a reachability objective [KNP19].

In our knowledge, tools to (exactly) solve a class of WTGs where the deterministic value problem is decidable do not exist. We thus propose a first implementation to solve all one-clock WTGs (with arbitrary weights) by implementing the symbolic value iteration algorithm in this case as well as our unfolding. This implementation is presented in Section 4.5.

[2] https://uppaal.org/
[3] https://www-verimag.imag.fr/DIST-TOOLS/TEMPO/kronos/
[4] https://pat.comp.nus.edu.sg/
[5] An event-recording automaton [AFH99] is a subclass of deterministic timed automata where each action is associated with only one clock that measures the time elapsed from the last occurrence of this action.
[6] https://www.labri.fr/perso/herbrete/tchecker/
[7] https://uppaal.org/features/#tiga
[8] https://www.prismmodelchecker.org/games/
[9] A probabilistic timed automaton [GJ95; Kwi+02; Bea03] is an extension of timed automata with a probabilistic distribution over transitions (time remains deterministic).

2. Fixpoint Characterisation of the Deterministic Value Function

Some classes of WTGs for which the value is decidable are solved with a value iteration algorithm. However, except for acyclic WTGs, the known proof that claims the correctness of this approach contains flaws. In this chapter, we explain why we need to use ad-hoc proofs, and we give a proof for one-clock WTGs and divergent WTGs.

Table of contents

2.1. Kleene's theorem is not applicable 53
2.2. Restriction of $\mathcal{F}$ according to a strategy of Min 55
2.3. The case of one-clock weighted timed games 58
2.3.1. Iterates of $\mathcal{F}$ uniformly converge to the greatest fixpoint of $\mathcal{F}$ 58
2.3.2. The greatest fixpoint of $\mathcal{F}$ is equal to the value function . . 61
2.4. The case of divergent weighted timed games 67

Let $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt}\rangle$ be a WTG, its value function $\mathsf{dVal}_{\mathcal{G}} \colon L \times \mathbb{R}^{\mathcal{X}}_{\geq 0} \to \mathbb{R}_\infty$ can sometimes be characterised as a fixpoint (and even the greatest fixpoint) of the operator $\mathcal{F}$ defined as follows: for all configurations (ℓ, ν) and all mappings $X \colon L \times \mathbb{R}^{\mathcal{X}}_{\geq 0} \to \mathbb{R}_\infty$, we let

$$\mathcal{F}(X)(\ell,\nu) = \begin{cases} 0 & \text{if } \ell \in L_T; \\ \inf_{(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')} \big(\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + X(\ell',\nu')\big) & \text{if } \ell \in L_{\mathsf{Min}}; \\ \sup_{(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')} \big(\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + X(\ell',\nu')\big) & \text{if } \ell \in L_{\mathsf{Max}}. \end{cases}$$

From the results established in [BMR18] for divergent WTGs and by adapting and correcting the sketch given in [Bou16] for concurrent hybrid games with only non-negative weights in the context of one-clock WTGs with negative weights, we establish our first contribution:

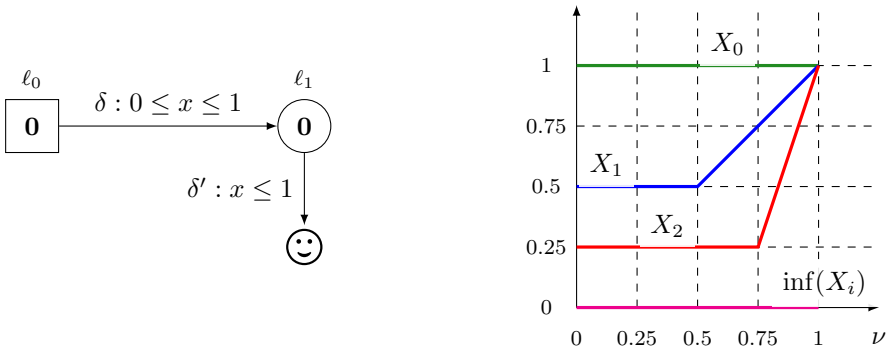


Figure 2.1 – On the left, a (one-clock) WTG in which $\mathcal{F}$ is not Scott-continuous by considering, for all locations, the non-increasing sequence of continuous functions on $(0, 1)$ that is depicted on the right.

Theorem 2.1. *The value function of a WTG is the greatest fixpoint of the operator $\mathcal{F}$ when*

1. *the WTG has at most one-clock; or*
2. *the WTG is divergent.*

We begin explaining why the natural proof can not be applied in this context that justifies the restrictions on WTGs. Then, we prove this theorem in three steps: 1) we focus on a restriction of $\mathcal{F}$ given by a (deterministic) strategy of Min; 2) we make the proof for one-clock WTGs; 3) we conclude about divergent WTGs.

2.1. Kleene's theorem is not applicable

A natural way[1] to obtain the fixpoint characterisation of the value of $\mathcal{G}$ would be to use the fixpoint theory and, more precisely, Kleene's theorem characterising the greatest fixpoint as the limit of a sequence of iterates of $\mathcal{F}$ before showing that the limit is equal to the value of $\mathcal{G}$. To be applicable, this theorem requires $\mathcal{F}$ to be *Scott-continuous* over a complete partial order[2] (CPO for short) [Tar55], i.e. monotonic and such that for all non-increasing sequences $(X_i)_{i\in\mathbb{N}}$ of elements of the CPO, $\inf_i \mathcal{F}(X_i) = \mathcal{F}(\inf_i X_i)$. Unfortunately, although the operator $\mathcal{F}$ can be shown to be monotonic (see Lemma 2.3-1), it is not necessarily Scott-continuous:

[1] In [Bri+17, Section 3.3], authors use Kleene's theorem to prove that the value is the greatest fixpoint of (the untimed version of) $\mathcal{F}$.

[2] A CPO is a partial order such that every decreasing chain of elements admits a least lower bound. An introduction to fixpoint theory is given in [Win93, Chapter 8].

Example 2.2. We consider the WTG depicted on the left of Figure 2.1. We prove that $\mathcal{F}$ is not Scott-continuous by exhibiting a non-increasing sequence $(X_i)_i$ of functions (that are continuous over regions) such that $\inf_i \mathcal{F}(X_i) > \mathcal{F}(\inf_i X_i)$. This sequence is depicted on the right of Figure 2.1 and can be defined, for all configurations (ℓ, ν) and all $i \in \mathbb{N}$, by

$$X_i(\ell,\nu) = \begin{cases} \frac{1}{2^i} & \text{if } 0 < \nu \leq \frac{2^i-1}{2^i}\,; \\ (2^i-1)\nu + (2-2^i) & \text{if } \frac{2^i-1}{2^i} < \nu < 1\,. \end{cases}$$

Indeed, by considering the location ℓ_0 of Max, for all $i \in \mathbb{N}$, and all valuations ν, we have

$$\mathcal{F}(X_i)(\ell_0,\nu) = \sup_t \big(t\,\mathsf{wt}(\ell_0) + \mathsf{wt}(\delta) + X_i(\ell_1,\nu+t)\big) = \sup_t X_i(\ell_1,\nu+t) = 1$$

and thus $\inf_i \mathcal{F}(X_i)(\ell_0,\nu) = 1$. Moreover, since $\inf_i X_i$ is the null constant function in $(0,1)$, we deduce that $\mathcal{F}(\inf_i X_i)(\ell_0,\nu) = 0$ for all configurations (ℓ_0,ν). Thus, we deduce that $\mathcal{F}(\inf_i X_i)(\ell_0,\nu) < \inf_i \mathcal{F}(X_i)(\ell_0,\nu)$. ⌟

We thus make a more pedestrian proof only using non-increasing sequences $(X_i)_i$ that *uniformly* converge over each region, i.e. the restriction of the sequence to each region uniformly converges. Our proof relies on the following technical results.

Lemma 2.3. *1. $\mathcal{F}$ is monotonous over $L \times \mathbb{R}_{\geq 0} \to \mathbb{R}_\infty$.*

2. For all $X\colon L \times \mathbb{R}_{\geq 0} \to \mathbb{R}_\infty$ and $a \geq 0$, $\mathcal{F}(X+a) \leq \mathcal{F}(X) + a$.

3. For all non-increasing sequences $(X_i)_i$ of functions $X_i\colon L \times \mathbb{R}_{\geq 0} \to \mathbb{R}_\infty$ that uniformly converge over each region, $\inf_i \mathcal{F}(X_i) = \mathcal{F}(\inf_i X_i)$.

Proof. 1. Let $X, X'\colon L \times \mathbb{R}_{\geq 0} \to \mathbb{R}_\infty$ be two functions such that $X \geq X'$ (i.e. for all configurations (ℓ,ν), $X(\ell,\nu) \geq X'(\ell,\nu)$), and (ℓ,ν) be a configuration. If $\ell \in L_T$, then $\mathcal{F}(X)(\ell,\nu) = 0 = \mathcal{F}(X')(\ell,\nu)$. Otherwise, since $X(\ell',\nu') \geq X'(\ell',\nu')$, for all edges $(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')$, we have

$$\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + X(\ell',\nu') \geq \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + X'(\ell',\nu')\,.$$

We conclude the proof by applying the infimum (resp. the supremum) over all edges in this inequality if $\ell \in L_{\mathsf{Min}}$ (resp. $\ell \in L_{\mathsf{Max}}$).

2. We consider a configuration (ℓ,ν), and we distinguish three cases. If $\ell \in L_T$, then, since $a \geq 0$, we have

$$\mathcal{F}(X+a)(\ell,\nu) = 0 = \mathcal{F}(X)(\ell,\nu) \leq \mathcal{F}(X)(\ell,\nu) + a\,.$$

Now, if $\ell \in L_{\mathsf{Min}}$, then, since a does not depend on edges, we have

$$\begin{aligned}\mathcal{F}(X+a)(\ell,\nu) &= \inf_{(\ell,\nu)\xrightarrow{\delta,t}(\ell',\nu')} \big(\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + X(\ell',\nu') + a\big) \\ &= \inf_{(\ell,\nu)\xrightarrow{\delta,t}(\ell',\nu')} \big(\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + X(\ell',\nu')\big) + a \\ &= \mathcal{F}(X)(\ell,\nu) + a\,.\end{aligned}$$

Finally, if $\ell \in L_{\mathsf{Max}}$, then, for the same reason, we have

$$\mathcal{F}(X+a)(\ell,\nu) = \mathcal{F}(X)(\ell,\nu) + a\,.$$

3. Since $\mathcal{F}$ is monotonous (item 1), we remark that for all $j \in \mathbb{N}$, we have $\mathcal{F}(X_j) \geq \mathcal{F}(\inf_i X_i)$. In particular, as this inequality holds for all $j \in \mathbb{N}$, we obtain that

$$\inf_i \mathcal{F}(X_i) \geq \mathcal{F}(\inf_i X_i)\,.$$

Conversely, let $\varepsilon > 0$ and r be a region. Since $(X_i)_i$ uniformly converges over the region r to $\inf_i X_i$, there exists $j_r \in \mathbb{N}$ such that $X_{j_r} \leq \inf_i X_i + \varepsilon$ over r. Now, since there are only a finite number of regions, we fix $j = \max_r j_r$ and for all regions r, $X_j \leq \inf_i X_i + \varepsilon$. Since $\mathcal{F}$ is monotonous and by item 2,

$$\mathcal{F}(X_j) \leq \mathcal{F}\big(\inf_i X_i + \varepsilon\big) \leq \mathcal{F}\big(\inf_i X_i\big) + \varepsilon\,.$$

In particular, we deduce that $\inf_i \mathcal{F}(X_i) \leq \mathcal{F}\big(\inf_i X_i\big) + \varepsilon$. We conclude since this inequality holds for all $\varepsilon > 0$. □

2.2. Restriction of $\mathcal{F}$ according to a strategy of Min

Before giving the fixpoint characterisation for some classes of WTGs, we turn our attention towards the value dVal^σ of a strategy σ of Min and the restriction $\mathcal{F}^\sigma$ of $\mathcal{F}$ according to this strategy by replacing the infimum for locations of Min with the choice given by σ. However, since we can not (and do not want to) suppose that σ is memoryless, we need to extend the functions that $\mathcal{F}^\sigma$ use. Formally, $\mathcal{F}^\sigma$ is a new operator over mappings $X\colon \mathsf{FPlays} \to \mathbb{R}_\infty$ such

that $\mathcal{F}^\sigma(X)(\rho)$ is equal to

$$\begin{cases} 0 & \text{if } \ell \in L_T; \\ \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + X(\rho[\delta,t]) & \text{if } \ell \in L_{\mathsf{Min}} \text{ and } \sigma(\rho) = (\delta,t); \\ \sup_{(\ell,\nu)\xrightarrow{\delta,t}(\ell',\nu')} \left[\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + X(\rho[\delta,t])\right] & \text{if } \ell \in L_{\mathsf{Max}}; \end{cases}$$

where $\mathsf{last}(\rho) = (\ell,\nu)$.

Since the value function dVal^σ has only been defined for configurations, we need to extend it over all finite plays. To do that, we define the weight of a play given by two strategies (σ and τ) from a given finite play ρ by the weight of the unique play ρ' conforming to σ and τ from the last configuration of ρ (when σ and τ are initialised by ρ), i.e.

$$\mathbf{P}(\mathsf{Play}(\rho,\sigma,\tau)) = \mathbf{P}(\rho')\,.$$

We observe that the weight of ρ is not taken into account in the weight of the play $\mathsf{Play}(\rho,\sigma,\tau)$ even if the play $\mathsf{Play}(\mathsf{last}(\rho),\sigma,\tau)$ does not describe the same play (since σ and τ may use some memory). We thus let, for all finite plays ρ,

$$\mathsf{dVal}^\sigma(\rho) = \sup_\tau \mathbf{P}(\mathsf{Play}(\rho,\sigma,\tau))\,.$$

Lemma 2.4. *Let $\mathcal{G}$ be a WTG and $\sigma \in \mathsf{dStrat}_{\mathsf{Min}}$, then $\mathsf{dVal}^\sigma \colon \mathsf{FPlays} \to \mathbb{R}_\infty$ is a fixpoint of $\mathcal{F}^\sigma$.*

Proof. Let ρ be a finite play and (ℓ,ν) its last configuration. If $\ell \in L_T$, then for all strategies τ of Max, $\mathbf{P}(\mathsf{Play}(\rho,\sigma,\tau)) = 0$. Thus, by applying the supremum over strategies of Max, we obtain that

$$\mathsf{dVal}^\sigma(\rho) = \sup_\tau \mathbf{P}(\mathsf{Play}(\rho,\sigma,\tau)) = 0 = \mathcal{F}^\sigma(\mathsf{dVal}^\sigma)(\rho)\,.$$

Now, we suppose that $\ell \in L_{\mathsf{Min}}$, and let $\sigma(\rho) = (\delta,t)$. Thus, for all strategies τ of Max, we obtain that

$$\mathbf{P}(\mathsf{Play}(\rho,\sigma,\tau)) = \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \mathbf{P}\big(\mathsf{Play}(\rho[\delta,t],\sigma,\tau)\big)\,.$$

In particular, by applying the supremum over strategies of Max, we obtain that

$$\mathsf{dVal}^\sigma(\rho) = \sup_\tau \Big[\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \mathbf{P}\big(\mathsf{Play}(\rho[\delta,t],\sigma,\tau)\big)\Big]\,.$$

We note that the decision (δ,t) given by σ depends only on ρ, which is independent

of the chosen strategy of Max. Thus, we deduce that

$$\begin{aligned}\mathsf{dVal}^{\sigma}(\rho) &= \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \sup_{\tau} \mathbf{P}\Big(\mathsf{Play}\big(\rho[\delta,t],\sigma,\tau\big)\Big)\\ &= \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \mathsf{dVal}^{\sigma}\big(\rho[\delta,t]\big) = \mathcal{F}^{\sigma}(\mathsf{dVal}^{\sigma})(\rho)\,.\end{aligned}$$

Finally, we suppose that $\ell \in L_{\mathsf{Max}}$ and we reason by double inequalities. We begin by showing that $\mathcal{F}^{\sigma}(\mathsf{dVal}^{\sigma})(\rho) \leq \mathsf{dVal}^{\sigma}(\rho)$. Let $\varepsilon > 0$, by the definition of $\mathcal{F}^{\sigma}(\mathsf{dVal}^{\sigma})(\rho)$, we obtain the existence of a feasible decision (δ, t) such that

$$\mathcal{F}^{\sigma}(\mathsf{dVal}^{\sigma})(\rho) \leq \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \mathsf{dVal}^{\sigma}\big(\rho[\delta,t]\big) + \frac{\varepsilon}{2}\,.$$

Similarly, by the definition of dVal^{σ}, there exists a strategy τ^* for Max such that

$$\mathsf{dVal}^{\sigma}\big(\rho[\delta,t]\big)\big) \leq \mathbf{P}\Big(\mathsf{Play}\big(\rho[\delta,t],\sigma,\tau^*\big)\Big) + \frac{\varepsilon}{2}\,.$$

In particular, by combining these two inequalities, we obtain

$$\mathcal{F}^{\sigma}(\mathsf{dVal}^{\sigma})(\rho) \leq \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \mathbf{P}\Big(\mathsf{Play}\big(\rho[\delta,t](\ell',\nu'),\sigma,\tau^*\big)\Big) + \varepsilon\,.$$

We consider the new strategy τ for Max defined by $\tau(\rho) = (\delta, t)$ and $\tau(\rho') = \tau^*(\rho')$, for all finite plays $\rho' \neq \rho$. In particular, since τ and τ^* make the same decision for all plays that extend ρ, we obtain that $\mathbf{P}\big(\mathsf{Play}(\rho[\delta,t],\sigma,\tau^*)\big) = \mathbf{P}\big(\mathsf{Play}(\rho[\delta,t],\sigma,\tau)\big)$. Thus, we deduce that

$$\mathcal{F}^{\sigma}(\mathsf{dVal}^{\sigma})(\rho) \leq \mathbf{P}\big(\mathsf{Play}(\rho,\sigma,\tau)\big) + \varepsilon \leq \sup_{\tau'}\,\Big(\mathbf{P}\big(\mathsf{Play}(\rho,\sigma,\tau')\big)\Big) + \varepsilon = \mathsf{dVal}^{\sigma}(\rho) + \varepsilon\,.$$

We conclude this first inequality since this holds for all $\varepsilon > 0$.

Conversely, we prove that $\mathsf{dVal}^{\sigma}(\rho) \leq \mathcal{F}^{\sigma}(\mathsf{dVal}^{\sigma})(\rho)$. Let $\varepsilon > 0$. As for the previous inequality, there exists a strategy τ^* of Max such that

$$\mathsf{dVal}^{\sigma}(\rho) - \varepsilon \leq \mathbf{P}\big(\mathsf{Play}(\rho,\sigma,\tau^*)\big)\,.$$

In particular, by letting $(\delta, t) = \tau^*(\rho)$ be the decision of τ^*, we deduce that

$$\begin{aligned}\mathsf{dVal}^{\sigma}(\rho) - \varepsilon &\leq \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \mathbf{P}\big(\mathsf{Play}(\rho[\delta,t],\sigma,\tau^*)\big)\\ &\leq \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \mathsf{dVal}^{\sigma}(\rho[\delta,t])\\ &\leq \sup_{(\ell,\nu)\xrightarrow{\delta,t}(\ell',\nu')} \big(\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \mathsf{dVal}^{\sigma}(\rho[\delta,t])\big) = \mathcal{F}^{\sigma}(\mathsf{dVal}^{\sigma})(\rho)\,.\end{aligned}$$

Again, we conclude since this inequality holds for all $\varepsilon > 0$. □

Remark. In the rest of the proof of fixpoint characterisations, we will only use the restriction of $\mathcal{F}$ to a strategy of Min. However, we note that we can obtain the analogous result for the strategies of Max by considering the infimum (instead of the supremum) for plays ending in a configuration of Min.

2.3. The case of one-clock weighted timed games

In this section, we prove the first claim of Theorem 2.1, i.e. the value function of all one-clock WTGs is the greatest fixpoint of the operator $\mathcal{F}$.

Proposition 2.5. *Let* $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \{x\}, \Delta, \mathsf{wt}\rangle$ *be a one-clock WTG. Then,*

$$\mathcal{F}(\mathsf{dVal}_{\mathcal{G}}) = \mathsf{dVal}_{\mathcal{G}}\,.$$

Moreover, for all $V\colon L \times \mathbb{R}_{\geq 0}^{\mathcal{X}} \to \mathbb{R}_{\infty}$ *fixpoints of* $\mathcal{F}$, $\mathsf{dVal}_{\mathcal{G}} \geq V$.

To prove this proposition, we adapt and correct the sketch from [Bou16] for concurrent hybrid games with only non-negative weights in the context of one-clock WTGs with negative weights. In particular, the proof is split into two parts.

1. In Section 2.3.1, we prove that the sequence of iterates of $\mathcal{F}$ (used in the value iteration algorithm of [Bou+04a]) uniformly converges via Dini's theorem by showing that they are all k-Lipschitz-continuous for the same constant.

2. In Section 2.3.2, we prove that the sequence of iterates of $\mathcal{F}$ converges to the value function. To do it, we remark that the mapping obtained after i applications of $\mathcal{F}$ is the restricted value function when we consider only plays of length i.

2.3.1. Iterates of $\mathcal{F}$ uniformly converge to the greatest fixpoint of $\mathcal{F}$

We consider the sequence $(V_i)_i$ of functions $L \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\infty}$ such that, for all $i \in \mathbb{N}$, and all configurations (ℓ, ν),

$$V_i(\ell,\nu) = \begin{cases} +\infty & \text{if } i = 0 \text{ and } \ell \notin L_T; \\ 0 & \text{if } i = 0 \text{ and } \ell \in L_T; \\ \mathcal{F}(V_{i-1})(\ell,\nu) & \text{otherwise.} \end{cases}$$

To conclude the first step of the proof, we prove that this sequence uniformly converges. Then, we can apply Lemma 2.3-3 to deduce that this sequence converges to a fixpoint of $\mathcal{F}$.

We observe thqt $\mathcal{F}$ is monotonous and $V_0 \geq \mathcal{F}(V_0)$ (since $V_0(\ell, \nu) = +\infty$, or $V_i(\ell, \nu) = 0$ for all $i \in \mathbb{N}$), thus the sequence $(V_i)_i$ is non-increasing. In particular, it (simply) converges to $\inf_i V_i$. To prove that the sequence $(V_i)_i$ uniformly converges to $\inf_i V_i$, we will use Dini's theorem: a sequence of continuous functions that (simply) converges to a continuous function, uniformly converges. The main difficulty is to prove that $\inf_i V_i$ is continuous over regions. To do it, we note that if a sequence of k-Lipschitz-continuous functions (simply) converges, then its limit is a continuous function. In particular, we want to show that there exists $k \in \mathbb{R}_{\geq 0}$ such that, for all $i \in \mathbb{N}$, V_i is k-Lipschitz-continuous.

Definition 2.6. Let $\Lambda \in \mathbb{R}_{\geq 0}$. A function $f \colon \mathbb{R}_{\geq 0} \to \mathbb{R}_\infty$ is *continuous (resp. Λ-Lipschitz-continuous) on regions* if for all regions r, the restriction of f over each region is continuous (resp. Λ-Lipschitz-continuous, i.e. for all regions r and all valuations $\nu, \nu' \in r$, $|f(\nu) - f(\nu')| \leq \Lambda|\nu - \nu'|$).

A function $X \colon L \times \mathbb{R}_{\geq 0} \to \mathbb{R}_\infty$ is continuous (resp. Λ-Lipschitz-continuous) on regions if, for all locations $\ell \in L$, the restriction of X on ℓ is continuous (resp. Λ-Lipschitz-continuous) on regions.

Since V_0 is constant on regions, we deduce that V_0 is Λ-Lipschitz-continuous on regions for all constants Λ. Moreover, in all WTGs and for all $i \in \mathbb{N}$, it is shown (by [Bus19, Lemma 10.10]) that V_i is Λ_i-Lipschitz-continuous on regions for a constant Λ_i that depends on i. We now refine the proof for our one-clock setting to show that the same constant can be chosen for all i.

Lemma 2.7. *For all $i \in \mathbb{N}$, V_i is W_{loc}-Lipschitz-continuous on regions.*

Proof. We rely on the knowledge that for all $i \in \mathbb{N}$, V_i is continuous and piecewise affine on each region, with finitely many pieces.

We reason by induction on $i \in \mathbb{N}$ showing that the derivative of V_i is bounded by W_{loc} in absolute values. The base case $i = 0$ is trivially satisfied as seen above. Let $i \in \mathbb{N}$ be such that V_i is continuous on regions and piecewise affine with finitely many pieces that have a derivative bounded by W_{loc}. We consider $\ell \in L \setminus L_T$ (otherwise, we conclude as for $i = 0$). By massaging the definition of $\mathcal{F}$, we have that

$$V_{i+1}(\ell, \nu) = \begin{cases} \min_\delta \inf_{(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')} \big(\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + V_i(\ell', \nu')\big) & \text{if } \ell \in L_{\mathsf{Min}}\,; \\ \max_\delta \sup_{(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')} \big(\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + V_i(\ell', \nu')\big) & \text{if } \ell \in L_{\mathsf{Max}}\,. \end{cases}$$

For a fixed valuation ν, and once chosen the transition δ in the minimum or maximum, there are finitely many delays t to consider in the infimum or supremum since V_i is piecewise affine: they are either delay 0 or all delays t such that $\nu + t$ are cutpoints[3] ν^c of $V_i(\ell', \cdot)$. In particular, since there is only a finite number of

[3] A cutpoint is the value of the clock in-between two affine pieces.

such cutpoints, the function $\mathcal{F}(V_i)(\ell, \cdot)$ can be written as a finite nesting of min and max operations over affine terms, each corresponding to a choice of delay and a transition to take. There are several cases to define those terms, depending on the chosen transition δ and cutpoint ν^c. First, we suppose that the transition δ resets x:

- if a delay 0 is chosen, then the affine term is $V_i(\ell', 0) + \mathsf{wt}(\delta)$ that has derivative 0;
- otherwise, the affine term that it generates is of the form

$$(\nu^c - \nu)\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + V_i(\ell', 0)$$

  whose derivative is bounded by W_{loc} in absolute value.

Finally, we suppose that the transition δ does not reset x:

- if a delay 0 is chosen, then the affine term is $\mathsf{wt}(\delta) + V_i(\ell', \nu)$, whose derivative is the same as in $V_i(\ell', \cdot)$ and thus bounded by W_{loc} in absolute value;
- otherwise, the affine term that it generates is of the form

$$(\nu^c - \nu)\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + V_i(\ell', \nu^c)\,.$$

  whose derivative is bounded by W_{loc} in absolute value. □

To conclude the first part of the proof, it remains to prove that iterates of $\mathcal{F}$ converge to the greatest fixpoint of $\mathcal{F}$.

Lemma 2.8. $\lim_i V_i$ *is the greatest fixpoint of* $\mathcal{F}$.

Proof. We start by proving that $\lim_i V_i = \inf_i V_i$ is a fixpoint of $\mathcal{F}$, i.e. $\inf_i V_i = \mathcal{F}(\inf_i V_i)$. By Lemma 2.7, we know that for all $i \in \mathbb{N}$, V_i is W_{loc}-Lipschitz-continuous over regions. Thus, we deduce that the sequence $(V_i)_i$ converges to a continuous function over regions, i.e. $\inf_i V_i(\ell)$ is continuous over regions for all locations ℓ. Now, by Dini's theorem, we deduce that $(V_i)_i$ uniformly converges over regions to $\inf_i V_i$. Finally, we apply Lemma 2.3-3 to conclude that $\inf_i V_i = \inf_i \mathcal{F}(V_i) = \mathcal{F}(\inf_i V_i)$, and thus that $\inf_i V_i$ is a fixpoint of $\mathcal{F}$.

Finally, we prove that $\inf_i V_i$ is the greatest fixpoint V of $\mathcal{F}$. As V is the greatest fixpoint, we have $\inf_i V_i \leq V$. Conversely, we prove by induction on $i \in \mathbb{N}$ that $V \leq V_i$. First, we suppose that $i = 0$ and if $\ell \notin L_T$, then $V_0(\ell, \nu) = +\infty$ thus $V(\ell, \nu) \leq V_0(\ell, \nu)$; otherwise, $\ell \in L_T$ and $V_0(\ell, \nu) = 0$, while $V(\ell, \nu) = 0$ (by definition of $\mathcal{F}$ and since V is a fixpoint). Then, we suppose that $i \in \mathbb{N}$ is such that $V \leq V_i$, as $\mathcal{F}$ is monotonous, we have $\mathcal{F}(V) \leq \mathcal{F}(V_i)$. Thus, we conclude the proof, since V is a fixpoint of $\mathcal{F}$: $V = \mathcal{F}(V) \leq \mathcal{F}(V_i) = V_{i+1}$. □

2.3.2. The greatest fixpoint of $\mathcal{F}$ is equal to the value function

To conclude the proof of Proposition 2.5, it remains to prove that $\lim_i V_i = \mathsf{dVal}$. To do it, we adapt the proof given in [Bou16] in our context (turn-based games with negative and positive weights).

The main idea of this proof is the definition of a link between V_i and the value obtained when we consider only plays with at most i steps. We fix W_i be the set of configurations from where Min can guarantee that a target location is reached within i steps from an attractor computation (see Section 1.2.3). Intuitively, for a configuration that is not in W_i, V_i is equal to $+\infty$ since Max can avoid the target locations in the i first steps. Formally, we define the sequence of $(W_i)_i$ by induction on $i \in \mathbb{N}$: $(\ell, \nu) \in W_0$ if $\ell \in L_T$, and for all $i \in \mathbb{N}$, $(\ell, \nu) \in W_{i+1}$ if $(\ell, \nu) \in W_i$, or

1. $\ell \in L_{\mathsf{Min}}$ and there exists an edge $(\ell, \nu) \xrightarrow{\delta,t} (\ell', \nu')$ such that $(\ell', \nu') \in W_i$;
2. $\ell \in L_{\mathsf{Max}}$, and for all edges $(\ell, \nu) \xrightarrow{\delta,t} (\ell', \nu')$, we have $(\ell', \nu') \in W_i$.

The following lemma recalls that V_i is more precise that W_i:

Lemma 2.9. *Let $i \in \mathbb{N}$ and (ℓ, ν) be a configuration. Then, $(\ell, \nu) \in W_i$ if and only if $V_i(\ell, \nu) < +\infty$.*

Proof. We prove the equivalence by induction on $i \in \mathbb{N}$. If $i = 0$, we conclude by definitions of W_0 and V_0. Now, we fix $i \in \mathbb{N}$ such that for all configurations (ℓ', ν'), we have $(\ell', \nu') \in W_i$ if and only if $V_i(\ell', \nu') < +\infty$. Let (ℓ, ν) be a configuration. First, we suppose that $\ell \in L_T$. In this case, $(\ell, \nu) \in W_{i+1}$ and $V_{i+1}(\ell, \nu) = 0 < +\infty$.

Now, we suppose that $\ell \in L_{\mathsf{Min}}$ and we have

$$V_{i+1}(\ell, \nu) = \mathcal{F}(V_i)(\ell, \nu) = \inf_{(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')} \left[(\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + V_i(\ell', \nu') \right].$$

In particular, $V_{i+1}(\ell, \nu) < +\infty$ if and only if there exists an edge $(\ell, \nu) \xrightarrow{\delta,t} (\ell', \nu')$ such that $V_i(\ell', \nu') < +\infty$. We deduce that $V_{i+1}(\ell, \nu) < +\infty$ if and only if there exists an edge $(\ell, \nu) \xrightarrow{\delta,t} (\ell', \nu')$ such that $(\ell', \nu') \in W_i$ (by applying the hypothesis of induction on (ℓ', ν')). We conclude that $V_{i+1}(\ell, \nu) < +\infty$ if and only if $(\ell, \nu) \in W_{i+1}$, by item 1 of the definition of W_{i+1}.

Finally, we suppose that $\ell \in L_{\mathsf{Max}}$ and we have

$$V_{i+1}(\ell, \nu) = \mathcal{F}(V_i)(\ell, \nu) = \sup_{(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')} \left[\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + V_i(\ell', \nu') \right].$$

In particular, by hypothesis of induction, $V_{i+1}(\ell,\nu) < +\infty$ if and only if for all edges $(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')$, we have $(\ell',\nu') \in W_i$. Thus, by item 2 of the definition of W_{i+1}, we conclude that $V_{i+1}(\ell,\nu) < +\infty$ if and only if $(\ell,\nu) \in W_{i+1}$. □

Remark. When we consider (general) WTGs with only non-negative weights, Lemma 2.9 can be naturally reinforced with a lower bound on the sequence of $(V_i)_i$. In [Bou+04a], the authors prove that for all configurations (ℓ,ν), $(\ell,\nu) \in W_i$ if and only if $0 \leq V_i(\ell,\nu) < +\infty$. However, with negative weights, $0 \leq V_i(\ell,\nu)$ obviously can be not satisfied.

To prove that value iteration converges to the value function, we relate configurations in W_i with some particular strategies of Min. Given a configuration (ℓ,ν), we fix $\mathsf{dStrat}_i(\ell,\nu)$ to be the set of strategies of Min such that all plays from (ℓ,ν) conforming to it reach the target *in at most i steps*. More precisely, we require that for all plays starting from (ℓ,ν) and conforming to a strategy of $\mathsf{dStrat}_i(\ell,\nu)$, the j-th configuration of the play belongs to W_{i-j}: in particular, the first configuration, (ℓ,ν) must be in W_i, and the last one in W_0 (i.e. with a location being a target).

For all $\varepsilon > 0$, we inductively define a sequence of strategies $(\sigma_i^\varepsilon)_i$ whose i-th strategy will be shown to belong to $\mathsf{dStrat}_i(\ell,\nu)$ if $(\ell,\nu) \in W_i$. We let σ_0^ε be any fixed strategy of Min. For $i \in \mathbb{N}$, relying on $\sigma_i^{\varepsilon/2}$, we define σ_{i+1}^ε for all finite plays ending in a location of Min according to its length. If the play contains only one configuration, we fix $\sigma_{i+1}^\varepsilon(\ell,\nu)$ be any feasible decision (δ,t) such that $(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')$ and $\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + V_i(\ell',\nu') \leq V_{i+1}(\ell,\nu) + \varepsilon/2$ (that exists by definition of $V_{i+1}(\ell,\nu)$ as an infimum). Otherwise, the play can be decomposed as $(\ell,\nu) \xrightarrow{\delta,t} \rho$ with (ℓ',ν') the first configuration of ρ, and we let

$$\sigma_{i+1}^\varepsilon((\ell,\nu) \xrightarrow{\delta,t} \rho) = \sigma_i^{\varepsilon/2}(\rho)\,.$$

Lemma 2.10. *For all $i \in \mathbb{N}$, $\varepsilon > 0$ and $(\ell,\nu) \in W_i$,*

$$\sigma_i^\varepsilon \in \mathsf{dStrat}_i(\ell,\nu) \quad \text{and} \quad V_i(\ell,\nu) + \varepsilon \geq \mathsf{dVal}^{\sigma_i^\varepsilon}(\ell,\nu)\,.$$

Proof. We reason by induction on $i \in \mathbb{N}$. If $i = 0$, since $(\ell,\nu) \in W_0$, we have $\ell \in L_T$ and thus any strategy (and thus the fixed strategy σ_0^ε) is in $\mathsf{dStrat}_0(\ell,\nu)$, and $V_0(\ell,\nu) = 0 = \mathsf{dVal}^{\sigma_0^\varepsilon}(\ell,\nu)$.

Now, consider $i \in \mathbb{N}$ such that for all configurations $(\ell,\nu) \in W_i$, we have $\sigma_i^{\varepsilon/2} \in \mathsf{dStrat}_i(\ell,\nu)$, and $V_i(\ell,\nu) + \frac{\varepsilon}{2} \geq \mathsf{dVal}^{\sigma_i^{\varepsilon/2}}(\ell,\nu)$. We show that σ_{i+1}^ε satisfies the properties for a given configuration $(\ell,\nu) \in W_{i+1}$. If $\ell \in L_T$, we conclude as in the case $i = 0$ (since $\ell \in W_0$). Otherwise, we show $\sigma_{i+1}^\varepsilon \in \mathsf{dStrat}_{i+1}(\ell,\nu)$ by contradiction. We thus suppose that there exists a finite play ρ' of length $i+1$

conforming to $\sigma^{\varepsilon}_{i+1}$ that does not reach L_T. It can be decomposed as $(\ell,\nu) \xrightarrow{\delta,t} \rho$ where $\sigma^{\varepsilon}_{i+1}(\ell,\nu) = (\delta,t)$ and ρ is conforming to $\sigma_i^{\varepsilon/2}$. We show that $(\ell',\nu') \in W_i$ where (ℓ',ν') is the first configuration of ρ.

- If $\ell \in L_{\mathsf{Max}}$, then we conclude that $(\ell',\nu') \in W_i$ by item 2 of definition of W_{i+1}: all edges from (ℓ,ν) reach a configuration in W_i.
- If $\ell \in L_{\mathsf{Min}}$, then, by item 1 of definition of W_{i+1}, there exists an edge $(\ell,\nu) \xrightarrow{\delta',t'} (\ell'',\nu'')$ such that $(\ell'',\nu'') \in W_i$, i.e. such that $V_i(\ell'',\nu'') < +\infty$ (by Lemma 2.9). The choice of $\sigma^{\varepsilon}_{i+1}(\ell,\nu)$ is taken along all possible edges from (ℓ,ν), at most ε away from the infimum. Thus, it chooses an edge $(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')$ such that $V_i(\ell',\nu') < +\infty$, i.e. $(\ell',\nu') \in W_i$ (by Lemma 2.9).

By the induction hypothesis applied to $(\ell',\nu') \in W_i$, $\sigma_i^{\varepsilon/2} \in \mathsf{dStrat}_i(\ell',\nu')$, and thus ρ reaches L_T within i steps which contradicts the hypothesis.

We then prove that $V_{i+1}(\ell,\nu) + \varepsilon \geq \mathsf{dVal}^{\sigma^{\varepsilon}_{i+1}}(\ell,\nu)$. By definition of $\sigma^{\varepsilon}_{i+1}$ with $\sigma_i^{\varepsilon/2}$, we remark that, for all finite plays $(\ell,\nu) \xrightarrow{\delta,t} \rho$ of length at least one, we have $\sigma^{\varepsilon}_{i+1}((\ell,\nu) \xrightarrow{\delta,t} \rho) = \sigma_i^{\varepsilon/2}(\rho)$. In particular, the weight of all plays from $(\ell,\nu) \xrightarrow{\delta,t} \rho$ and conforming to $\sigma^{\varepsilon}_{i+1}$ is equal to the weight of the play from ρ and conforming to $\sigma_i^{\varepsilon/2}$ under the same strategy of Max, i.e. for all strategies τ of Max, we have

$$\mathbf{P}\big(\mathsf{Play}((\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu'), \sigma^{\varepsilon}_{i+1}, \tau)\big) = \mathbf{P}\big(\mathsf{Play}((\ell',\nu'), \sigma_i^{\varepsilon/2}, \tau)\big)\,.$$

Thus, by applying the supremum over strategies of Max, we deduce that

$$\mathsf{dVal}^{\sigma^{\varepsilon}_{i+1}}((\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')) = \mathsf{dVal}^{\sigma_i^{\varepsilon/2}}(\ell',\nu')\,. \tag{2.1}$$

- If $\ell \in L_{\mathsf{Max}}$, then we have

$$V_{i+1}(\ell,\nu) = \mathcal{F}(V_i)(\ell,\nu) = \sup_{(\ell,\nu)\xrightarrow{\delta,t}(\ell',\nu')} \big(\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + V_i(\ell',\nu')\big)\,.$$

  Moreover, since $(\ell,\nu) \in W_{i+1}$, then $(\ell',\nu') \in W_i$ (by item 2) and, by induction hypothesis on $V_i(\ell',\nu')$: $V_i(\ell',\nu') + \varepsilon/2 \geq \mathsf{dVal}^{\sigma_i^{\varepsilon/2}}(\ell',\nu')$. Thus, for all decisions (δ,t)

$$\begin{aligned} V_{i+1}(\ell,\nu) &\geq \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + V_i(\ell',\nu') \\ &\geq \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \mathsf{dVal}^{\sigma_i^{\varepsilon/2}}(\ell',\nu') - \frac{\varepsilon}{2} \\ &\geq \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \mathsf{dVal}^{\sigma^{\varepsilon}_{i+1}}((\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')) - \frac{\varepsilon}{2} \qquad \text{(by (2.1))}. \end{aligned}$$

Finally, since this inequality holds for all edges from (ℓ, ν), we conclude that

$$\begin{aligned} V_{i+1}(\ell,\nu) &\geq \sup_{(\ell,\nu)\xrightarrow{\delta,t}(\ell',\nu')} \left[\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \mathsf{dVal}^{\sigma^{\varepsilon}_{i+1}}\big((\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')\big)\right] - \frac{\varepsilon}{2} \\ &\geq \mathsf{dVal}^{\sigma^{\varepsilon}_{i+1}}(\ell,\nu) - \varepsilon \qquad \text{(by Lemma 2.4).} \end{aligned}$$

- If $\ell \in L_{\mathsf{Min}}$, then, by definition of $\sigma^{\varepsilon}_{i+1}$, and letting $\sigma^{\varepsilon}_{i+1}(\ell,\nu) = (\delta, t)$:

$$V_{i+1}(\ell,\nu) \geq \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + V_i(\ell',\nu') - \frac{\varepsilon}{2}$$

Now, since $(\ell',\nu') \in W_i$ (as explained in the proof of $\sigma^{\varepsilon}_{i+1} \in \mathsf{dStrat}_{i+1}(\ell,\nu)$), by induction hypothesis, $V_i(\ell',\nu') + \varepsilon/2 \geq \mathsf{dVal}^{\sigma_i^{\varepsilon/2}}(\ell',\nu')$. Thus

$$\begin{aligned} V_{i+1}(\ell,\nu) &\geq \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \mathsf{dVal}^{\sigma_i^{\varepsilon/2}}(\ell',\nu') - \varepsilon \\ &\geq \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \mathsf{dVal}^{\sigma^{\varepsilon}_{i+1}}((\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')) - \varepsilon \qquad \text{(by (2.1))} \\ &\geq \mathsf{dVal}^{\sigma^{\varepsilon}_{i+1}}(\ell,\nu) - \varepsilon \qquad \text{(by Lemma 2.4).} \end{aligned}$$

□

As a corollary, we obtain:

Lemma 2.11. *For all $i \in \mathbb{N}$, and $(\ell,\nu) \in W_i$, $V_i(\ell,\nu) = \inf_{\sigma\in\mathsf{dStrat}_i(\ell,\nu)} \mathsf{dVal}^{\sigma}(\ell,\nu)$.*

Proof. We reason by induction on $i \in \mathbb{N}$. If $i = 0$, since $\ell \in L_T$ for all strategies $\sigma \in \mathsf{dStrat}_0(\ell,\nu)$, $V_0(\ell,\nu) = 0 = \mathsf{dVal}^{\sigma}(\ell,\nu)$.

For $i \in \mathbb{N}$ such that the property holds, let $(\ell,\nu) \in W_{i+1}$. If $\ell \in L_T$, we have $(\ell,\nu) \in W_0$ and we conclude as in the case $i = 0$. Otherwise, Lemma 2.10 directly implies that

$$V_{i+1}(\ell,\nu) + \varepsilon \geq \mathsf{dVal}^{\sigma^{\varepsilon}_{i+1}}(\ell,\nu) \geq \inf_{\sigma\in\mathsf{dStrat}_{i+1}(\ell,\nu)} \mathsf{dVal}^{\sigma}(\ell,\nu)\,.$$

We conclude that $V_{i+1}(\ell,\nu) \geq \inf_{\sigma\in\mathsf{dStrat}_{i+1}(\ell,\nu)} \mathsf{dVal}^{\sigma}(\ell,\nu)$ since the inequality holds for all $\varepsilon > 0$.

Conversely, we show that $V_{i+1}(\ell,\nu) \leq \inf_{\sigma\in\mathsf{dStrat}_{i+1}(\ell,\nu)} \mathsf{dVal}^{\sigma}(\ell,\nu)$ by proving that for all $\sigma \in \mathsf{dStrat}_{i+1}(\ell,\nu)$, we have $V_{i+1}(\ell,\nu) \leq \mathsf{dVal}^{\sigma}(\ell,\nu)$. Let $\sigma \in \mathsf{dStrat}_{i+1}(\ell,\nu)$.

- If $\ell \in L_{\mathsf{Min}}$, then we let $(\delta, t) = \sigma(\ell,\nu)$ with $(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')$, so that $(\ell',\nu') \in W_i$. By induction hypothesis, we have

$$V_i(\ell',\nu') = \inf_{\sigma'\in\mathsf{dStrat}_i(\ell',\nu')} \mathsf{dVal}^{\sigma'}(\ell',\nu')\,.$$

Now, we consider the strategy σ' obtained from σ by adding as a first move the edge $(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')$. Formally, it is defined by

$$\sigma'_{\ell',\nu'}(\rho) = \begin{cases} \sigma\big((\ell,\nu) \xrightarrow{\delta,t} \rho\big) & \text{if } \rho \text{ starts in } (\ell',\nu'); \\ \sigma(\rho) & \text{otherwise.} \end{cases}$$

Given a play ρ' conforming to σ' starting from (ℓ',ν'), we remark that the play $(\ell,\nu) \xrightarrow{\delta,t} \rho'$ is conforming to σ. In particular, we obtain that

$$\mathsf{Play}((\ell,\nu),\sigma',\tau) = \mathsf{Play}((\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu'),\sigma,\tau) \tag{2.2}$$

and we deduce that $\sigma' \in \mathsf{dStrat}_i(\ell',\nu')$. Thus, by hypothesis of induction,

$$V_i(\ell',\nu') \leq \mathsf{dVal}^{\sigma'}(\ell',\nu')\,.$$

In particular, we deduce that

$$\begin{aligned} V_{i+1}(\ell,\nu) &= \mathcal{F}(V_i)(\ell,\nu) \\ &\leq \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + V_i(\ell',\nu') \\ &\leq \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \mathsf{dVal}^{\sigma'}(\ell',\nu')\,. \end{aligned}$$

Moreover, by (2.2), we also obtain that, for all strategies τ of Max,

$$\mathbf{P}\big(\mathsf{Play}((\ell,\nu),\sigma',\tau)\big) = \mathbf{P}\big(\mathsf{Play}((\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu'),\sigma,\tau)\big)\,.$$

In particular, we deduce that $\mathsf{dVal}^{\sigma'}(\ell',\nu') = \mathsf{dVal}^{\sigma}((\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu'))$, and we can rewrite, by Lemma 2.4, the previous inequality as

$$V_{i+1}(\ell,\nu) \leq \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \mathsf{dVal}^{\sigma}((\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')) \leq \mathsf{dVal}^{\sigma}(\ell,\nu)\,.$$

- If $\ell \in L_{\mathsf{Max}}$, then, by Lemma 2.4, we have

$$\mathsf{dVal}^{\sigma}(\ell,\nu) = \sup_{(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')} \big(\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + \mathsf{dVal}^{\sigma}((\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu'))\big)$$

Letting $(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')$ be an edge from (ℓ,ν), since $\sigma \in \mathsf{dStrat}_{i+1}(\ell,\nu)$, we have $(\ell',\nu') \in W_i$, and thus by induction hypothesis,

$$V_i(\ell',\nu') \leq \inf_{\sigma' \in \mathsf{dStrat}_i(\ell',\nu')} \mathsf{dVal}^{\sigma'}(\ell',\nu')\,.$$

By considering the same strategy σ' as the one defined in the case of Min, we obtain that

$$V_i(\ell', \nu') \leq \mathsf{dVal}^{\sigma'}(\ell', \nu') = \mathsf{dVal}^{\sigma}((\ell, \nu) \xrightarrow{\delta,t} (\ell', \nu'))\,.$$

Thus, we deduce that

$$\mathsf{dVal}^{\sigma}(\ell, \nu) \geq \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + V_i(\ell', \nu')$$

Since this holds for all edges $(\ell, \nu) \xrightarrow{\delta,t} (\ell', \nu')$, we conclude that

$$\mathsf{dVal}^{\sigma}(\ell, \nu) \geq \sup_{(\ell,\nu)\xrightarrow{\delta,t}(\ell',\nu')} \big(\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell) + V_i(\ell', \nu')\big) = V_{i+1}(\ell, \nu)\,. \qquad \square$$

Finally, we prove that the limit $V = \inf_i V_i$ of the sequence $(V_i)_i$ equals the value function of $\mathcal{G}$. In particular, we fix W to be the set of configurations from where Min can ensure to reach L_T (without restriction on the number of steps), that is the limit of $(W_i)_i$: $W = \bigcup_i W_i$. Now, by the attractor computation in RTGs (see Section 1.2.3), we know that there exists a finite $N \in \mathbb{N}$ such that $W = \bigcup_{i=0}^{N} W_i$. Now, we can finally prove that $V = \mathsf{dVal}$.

We reason by double inequalities, and we start by proving that $V \geq \mathsf{dVal} = \inf_\sigma \mathsf{dVal}^\sigma$. If $(\ell, \nu) \notin W$, we have for all $i \in \mathbb{N}$, $V_i(\ell, \nu) = +\infty$ (by Lemma 2.9), and thus $V(\ell, \nu) = +\infty$. Otherwise, $(\ell, \nu) \in W_N$. By letting $\varepsilon > 0$, there exists $k \geq N$ such that $V_k(\ell, \nu) \leq V(\ell, \nu) + \varepsilon$. By using Lemma 2.11, $\inf_{\sigma \in \mathsf{dStrat}_k(\ell,\nu)} \mathsf{dVal}^\sigma(\ell, \nu) \leq V(\ell, \nu) + \varepsilon$. By considering the infimum over all strategies, and since this holds for all $\varepsilon > 0$, we get $\mathsf{dVal} = \inf_\sigma \mathsf{dVal}^\sigma \leq V$.

Conversely, we prove that $V \leq \mathsf{dVal} = \inf_\sigma \mathsf{dVal}^\sigma$. By contradiction, we suppose that there exists a strategy σ of Min and an initial configuration (ℓ, ν) such that $V(\ell, \nu) > \mathsf{dVal}^\sigma(\ell, \nu)$. Since then $\mathsf{dVal}^\sigma(\ell, \nu) < +\infty$, all plays conforming to σ reach a target location. We (inductively) build a play ρ from (ℓ, ν) conforming to σ such that at each step we guarantee that $\mathsf{last}(\rho) = (\ell', \nu')$ satisfies $\ell' \notin L_T$, and $V(\ell', \nu') > \mathsf{dVal}^\sigma(\rho)$. In particular, this would imply that ρ is an infinite play that never reaches a target, contradicting the hypothesis.

To initiate the inductive construction of ρ, we let $\rho = (\ell, \nu)$. Since $V(\ell, \nu) > \mathsf{dVal}^\sigma(\ell, \nu)$, we deduce that $\ell \notin L_T$ (otherwise $V(\ell, \nu) = 0 = \mathsf{dVal}^\sigma(\ell, \nu)$ by Lemma 2.4).

Otherwise, we suppose that ρ is a play from (ℓ, ν) conforming to σ such that $V(\ell', \nu') > \mathsf{dVal}^\sigma(\rho)$ where $\mathsf{last}(\rho) = (\ell', \nu')$ and $\ell' \notin L_T$. We define a new step for ρ as follows.

- If $\ell' \in L_{\mathsf{Min}}$, then we extend ρ by $\rho' = \rho \xrightarrow{\delta,t} (\ell'', \nu'')$, by letting $\sigma(\rho) = (\delta, t)$.

Since ρ is conforming to σ, then ρ' is also conforming to σ. By induction hypothesis and Lemma 2.4,

$$V(\ell', \nu') > \mathsf{dVal}^\sigma(\rho) = \mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell') + \mathsf{dVal}^\sigma(\rho')$$

Since V is a fixpoint of $\mathcal{F}$ by Lemma 2.8, with $V(\ell', \nu')$ being thus equal to an infimum over all possible edges, we obtain

$$V(\ell'', \nu'') \geq V(\ell', \nu') - \mathsf{wt}(\delta) - t\,\mathsf{wt}(\ell') > \mathsf{dVal}^\sigma(\rho')\,.$$

- If $\ell' \in L_{\mathsf{Max}}$, then we prove that there exists an edge $(\ell', \nu') \xrightarrow{\delta,t} (\ell'', \nu'')$ such that $V(\ell'', \nu'') > \mathsf{dVal}^\sigma(\rho[\delta, t])$, and we define the new step of ρ with this edge (the resulting play is conforming to σ). To do that, we reason by contradiction, and we suppose that for all edges $(\ell', \nu') \xrightarrow{\delta,t} (\ell'', \nu'')$, we have $V(\ell'', \nu'') \leq \mathsf{dVal}^\sigma(\rho[\delta, t])$. In particular, by induction hypothesis, Lemma 2.4, and monotonicity of $\mathcal{F}$, we have

$$\begin{aligned} V(\ell', \nu') > \mathsf{dVal}^\sigma(\rho) &= \sup_{(\ell',\nu') \xrightarrow{\delta,t} (\ell'',\nu'')} \left[\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell') + \mathsf{dVal}^\sigma(\rho[\delta, t])\right] \\ &\geq \sup_{(\ell',\nu') \xrightarrow{\delta,t} (\ell'',\nu'')} \left[\mathsf{wt}(\delta) + t\,\mathsf{wt}(\ell') + V(\ell'', \nu'')\right] = V(\ell', \nu') \end{aligned}$$

  as V is a fixpoint of $\mathcal{F}$ (by Lemma 2.8). Thus, we obtain a contradiction.

This concludes the proof that $\lim_i V_i = \mathsf{dVal}$, and thus of Proposition 2.5.

2.4. The case of divergent weighted timed games

In this section, we consider a divergent WTG, and we rely on results from [BMR17] to conclude that the value function of all divergent WTGs is the greatest fixpoint of the operator $\mathcal{F}$.

Proposition 2.12. *Let $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt}\rangle$ be a divergent WTG. Then, $\mathcal{F}(\mathsf{dVal}_\mathcal{G}) = \mathsf{dVal}_\mathcal{G}$. Moreover, for all $V\colon L \times \mathbb{R}^\mathcal{X}_{\geq 0} \to \mathbb{R}_\infty$ fixpoints of $\mathcal{F}$, $V \leq \mathsf{dVal}_\mathcal{G}$.*

Proof. Let (ℓ, ν) be a configuration of $\mathcal{G}$. Since $\mathcal{G}$ is divergent, we recall that locations of the region game $\mathcal{R}(\mathcal{G})$ may be divide into two subsets [BMR17, Proposition 5]: region states in a negative SCC (i.e. an SCC where all plays following a cyclic path have a weight less than -1) and region states in a positive SCC (i.e. an SCC where all plays following a cyclic path have a weight greater than 1).

The computation of $\mathsf{dVal}_{\mathcal{G}}(\ell, \nu)$ differs according to the SCC in where the region state (ℓ, r) is when $\nu \in r$. Indeed, the authors of [BMR17, Proposition 6] propose a value iteration algorithm based on $\mathcal{F}$ from the function V_0 defined by the constant function equal to $+\infty$ if (ℓ, r) is in a positive SCC, and the constant function equal to $-\infty$ otherwise. More precisely, the authors prove that this algorithm stabilises (for both cases) after $H \in \mathbb{N}$ steps (where H is the number of regions states of the WTG), i.e. $\mathsf{dVal}(\ell, \nu) = V_H(\ell, \nu) = V_{H+1}(\ell, \nu)$ where V_n is the function obtained after n applications of $\mathcal{F}$ in V^0. Thus, we deduce that

$$\mathsf{dVal}(\ell, \nu) = V_H(\ell, \nu) = \mathcal{F}(V_H)(\ell, \nu) = \mathcal{F}(\mathsf{dVal})(\ell, \nu)\,.$$

To conclude the proof, we prove that dVal is the greatest fixpoint of $\mathcal{F}$ by adapting the proof of Lemma 2.8. We suppose that V is the greatest fixpoint of $\mathcal{F}$. By letting $(V_n^+)_n$ the sequence of functions defined by applications of $\mathcal{F}$ from the constant function equal to $+\infty$ except in target locations where the constant function is equal to 0, we know that the sequence $\inf_n V_n^+ = \mathsf{dVal}$ by [BMR23, Theorem 3.9]. Now, as V is the greatest fixpoint, we have $\mathsf{dVal} = \inf_i V_i^+ \leq V$ since dVal is a fixpoint of $\mathcal{F}$. Conversely, we prove by induction on $i \in \mathbb{N}$ that $V \leq V_i^+$. First, we suppose that $i = 0$ and if $\ell \notin L_T$, then $V_0^+(\ell, \nu) = +\infty$ thus $V(\ell, \nu) \leq V_0^+(\ell, \nu)$; otherwise, $\ell \in L_T$ and $V_0^+(\ell, \nu) = 0 = V(\ell, \nu)$ (by definition of $\mathcal{F}$ and since V is a fixpoint of $\mathcal{F}$). Then, we suppose that $i \in \mathbb{N}$ is such that $V \leq V_i^+$, as $\mathcal{F}$ is monotonous, we have $\mathcal{F}(V) \leq \mathcal{F}(V_i^+)$. Thus, we conclude the proof, since V is a fixpoint of $\mathcal{F}$: $V = \mathcal{F}(V) \leq \mathcal{F}(V_i^+) = V_{i+1}^+ \leq \mathsf{dVal}$. □

3. Switching Strategies

Min needs finite memory to (almost-)optimally play in WTGs. However, (almost-)optimal strategies with a particular shape may be defined: we call them switching strategies. Such strategies consist of a combination with a counter of two memoryless strategies. The existence of an (almost-)optimal switching strategy for Min is shown in SPGs [Bri+17] and simple one-clock WTGs [Bri+15]. Our contribution is to prove the existence of an almost-optimal switching strategy for Min in all divergent WTGs.

Table of contents

3.1. A memory combination of two well-chosen memoryless strategies . . 70
3.1.1. How Min controls negative cycles in shortest path games . . 70
3.1.2. Fake-optimal strategies in weighted timed games 71
3.1.3. The case of reset-acyclic one-clock weighted timed games . . 73
3.2. Almost-optimal strategies in divergent weighted timed games 75
3.2.1. Switching strategies for Min 75
3.2.2. Memoryless strategies for Max 79

A switching strategy σ is described by two deterministic memoryless strategies σ^1 and σ^2, as well as a switching threshold κ. Intuitively, σ^1 aims at reaching a cyclic play with negative weight, while σ^1 is an attractor to the target. Formally, we fix

Definition 3.1. A *switching strategy* $\sigma = \langle \sigma^1, \sigma^2, \kappa \rangle$ with two deterministic memoryless strategies σ^1, and σ^2, and a switching threshold $\kappa \in \mathbb{N}$ describes a strategy that plays strategy σ^1 until either we reach a target location or the finite play has a length of at least κ, in which case we switch to strategy σ^2.

To establish our next contribution, we are inspired by the proof of the existence of an optimal switching strategy in SPGs given in [Bri+17].

Theorem 3.2. *Let $\mathcal{G}$ be a divergent WTG and $\varepsilon > 0$. Then,*

1. Min *has an ε-optimal switching strategy;*

2. Max *has an ε-optimal memoryless strategy.*

Before presenting the proof of this claim, we start by recalling the main tools used in SPGs to prove the existence of optimal switching strategy for Min and their generalisation into reset-acyclic one-clock WTGs given in [Bri+22].

3.1. A memory combination of two well-chosen memoryless strategies

To define an (almost)-optimal switching strategy in a given WTG, we need to guarantee some properties on both memoryless strategies σ^1 and σ^2. Indeed, we ask that σ^2 ensures us that all plays always reach a target location: it is thus defined by an attractor (that exists in all WTGs, see Section 1.2.3). About σ^1, we ask that every play only reaches a "cycle" (not only cyclic play but cyclic paths) with a negative weight.

3.1.1. How Min controls negative cycles in shortest path games

For now, we consider only SPGs. Remember that Min does not always have optimal memoryless strategies (see Example 1.30). In particular, memory in deterministic optimal strategies appears since Min wants to stay in a cyclic path during only a bounded time (otherwise, a target location may not be reached). To define switching strategies, authors of [Bri+17] define a class of strategies such that Min controls cycles along plays conforming to it.

This class of strategies is called the *fake-optimal strategies* which are memoryless strategies that allow Min either: to reach the target using a play whose payoff is lower than the value of the game; or to arbitrarily decrease the cumulative weight along the play (in other words, these cumulative weights tend to $-\infty$ as the play goes on). So fake-optimal strategies are not optimal in general: they do not guarantee that plays conforming to it reach a target location. However, when a target location is reached, fake-optimal strategies guarantee that Min will play consistently with their objective by decreasing the cumulative weight of the play prefixes.

Definition 3.3. Given an initial vertex ℓ_{i}, a *fake-optimal strategy* is a memoryless (deterministic) strategy σ for Min such that, for all plays ρ starting in ℓ_{i} and conforming to σ either: ρ reaches a target location and $\mathbf{P}(\rho) \leq \mathsf{dVal}(\ell_{\mathsf{i}})$; or for all cyclic plays $\rho' = \ell_0 \xrightarrow{e_0} \cdots \xrightarrow{e_k} \ell_0$ followed by ρ, $\mathsf{wt}_\Sigma(\rho') < 0$, i.e. $\sum_{i=0}^{k} \mathsf{wt}(e_i) < 0$.

Example 3.4. We consider the SPG depicted in Figure 1.8 and the memoryless strategy σ^1 such that $\sigma^1(\ell_{\mathsf{Min}}) = e_1$. We note that σ^1 is fake-optimal: given a play ρ conforming to it, either it reaches the target location by a decision of Max (since σ^1 always chooses e_1 from ℓ_{Min}) via e_3, and its payoff is at most -10 (the value); or it creates one of the two possible cyclic plays: $\ell_{\mathsf{Min}} \xrightarrow{e_1} \ell_{\mathsf{Max}} \xrightarrow{e_2} \ell_{\mathsf{Min}}$ and $\ell_{\mathsf{Max}} \xrightarrow{e_2} \ell_{\mathsf{Min}} \xrightarrow{e_1} \ell_{\mathsf{Max}}$, both having a cumulative weight of -1. However, we note that $\mathsf{dVal}^{\sigma^1}(\ell_{\mathsf{Min}}) = +\infty$, since there exists a play (that stays in the cyclic path forever) conforming to σ^1. ⌟

Since we do not ask for the optimality of fake-optimal strategies, to define an optimal strategy for Min, we need to combine it with another strategy that guarantees reaching a target location, obtaining a switching strategy.

Example 3.5. From the fake-optimal strategy σ^1 defined in Example 3.4, we can define an optimal switching strategy for the SPG that is depicted in Figure 1.8. To do it, we fix $\sigma = \langle \sigma^1, \sigma^2, \kappa \rangle$ where $\sigma^2(\ell_{\mathsf{Min}}) = e_4$ and $\kappa = 10$. ⌟

Authors of [Bri+17] generalise this process to all SPGs by defining a fake-optimal strategy σ^1 from the fixpoint characterisation of the value with the untimed version of operator $\mathcal{F}$ (introduced in Chapter 2). Thus, they obtain the following theorem.

Theorem 3.6 [Bri+17, Proposition 14]**.** *Let $\mathcal{G} = \langle V_{\mathsf{Max}}, V_{\mathsf{Min}}, V_T, E, \mathsf{wt} \rangle$ be a SPG. Then, there exists an optimal switching strategy $\sigma = \langle \sigma^1, \sigma^2, \kappa \rangle$ for Min such that κ is polynomial in $|V|$ and W.*

An additional property given by switching strategies is the capability to limit by $\kappa + |V|$ the length of plays conforming to them. Indeed, when we consider a switching strategy, every play conforming to it follows the first strategy during at most κ steps, and the attractor that defines the second one, during at most $|V|$ steps (see Definition 1.22).

3.1.2. Fake-optimal strategies in weighted timed games

The definition of fake-optimal strategies in SPGs carries two problems when we want to adapt it in WTGs. For the same reason that optimal strategies may not exist in WTGs, the first condition must be relaxed to guarantee the existence of strategies that satisfy it. Indeed, even if "fake" means that the considered strategy is not obliged to guarantee a target location is reached, we want to control the payoff obtained when it is done. In particular, since we can not ask that the payoff is at most the value, we must guarantee a payoff close to this value, i.e. the tolerated error factor compared to the value depending linearly on the size of the play.

Next, two notions of cycles coexist in WTGs: cyclic paths and cyclic plays. Thus, the second condition needs to look after the definition of a cycle that we

consider: we want to guarantee that when the length of the play increases, its cumulative weight decreases. However, only asking that every cyclic play has a negative weight does not suffice since cyclic plays may never exist even if the size of the play increases (and follows a cyclic region path). Now, when we consider the elementary region path, its weight may be non-negative but belongs to a larger cycle with a negative weight. In particular, unlike in SPGs, we do not ask that every cyclic region path conforming to the strategy is followed by a play with a negative weight, but only for long enough cyclic paths, the length depending on the considered WTG.

Definition 3.7. Let $\mathcal{G}$ be a WTG and $\varepsilon > 0$. A strategy σ is *fake-ε-optimal* if

- each finite play ρ conforming to σ starting from the configuration (ℓ, ν) and reaching the target has a payoff at most $\mathsf{dVal}(\ell, \nu) + |\rho|\,\varepsilon$. Moreover, we suppose that if $\mathsf{dVal}(\ell, \nu) = -\infty$, no such plays must exist;
- there exists $H \in \mathbb{N}$ such that each finite play conforming to σ following a cyclic region path of length at least H, in the region game $\mathcal{R}(\mathcal{G})$, has a cumulated weight at most -1.

From this class of strategies that we can combine with an attractor strategy, we obtain switching strategies in WTGs that are good candidates to build ε-optimal switching strategies :

Lemma 3.8. *Let $\mathcal{G}$ be a WTG such that for all $\varepsilon' > 0$, such that there exists a fake-ε'-optimal strategy σ^1 for Min. Let $\varepsilon > 0$ and $N \in \mathbb{N}$, then there exists a switching strategy $\sigma = \langle \sigma^1, \sigma^2, \kappa \rangle$ such that for all configurations (ℓ, ν) of $\mathcal{G}$,*

$$\mathsf{dVal}^{\sigma}(\ell, \nu) \leq \max(-N, \mathsf{dVal}(\ell, \nu)) + \varepsilon\,.$$

Proof. We fix the switching strategy σ defined such that σ^1 is a fake-ε/κ-optimal strategy, σ^2 is an attractor strategy and $\kappa = (W_{\mathsf{e}}\,|Q|\,(H+1)+N)(|Q|\,H+1)$ where Q is the set of region states in $\mathcal{R}(\mathcal{G})$ and H is given by σ^1. We show that this switching strategy satisfies, for all configurations (ℓ, ν) of $\mathcal{G}$, the desired inequality. Let ρ be a play conforming to σ from (ℓ, ν) that reaches the target (ℓ_t, ν_t). We distinguish two cases according to the length of ρ. First, we suppose that $|\rho| \leq \kappa$, then ρ is only conforming to σ^1 which is a fake-ε/κ-optimal strategy. In particular, $\mathsf{wt}_{\Sigma}(\rho) \leq \mathsf{dVal}(\ell, \nu) + |\rho|\frac{\varepsilon}{\kappa}$, and we conclude since $|\rho| \leq \kappa$, i.e.

$$\mathsf{wt}_{\Sigma}(\rho) \leq \mathsf{dVal}(\ell, \nu) + \varepsilon \leq \max(-N, \mathsf{dVal}(\ell, \nu)) + \varepsilon.$$

Otherwise, we suppose that $|\rho| > \kappa$, then we can decompose ρ as ρ_1 conforming to σ^1, followed by ρ_2 conforming to σ^2 with $|\rho_1| = \kappa$. As ρ_2 is conforming to σ^2,

an attractor strategy on the region game $\mathcal{R}(\mathcal{G})$, it is acyclic in $\mathcal{R}(\mathcal{G})$, so $|\rho_2| \leq |Q|$. In particular,

$$\mathsf{wt}_\Sigma(\rho_2) \leq W_\mathsf{e} |Q|.$$

As $|\rho_1| = \kappa$, by definition of $\kappa = (W_\mathsf{e} |Q| (H+1) + N)(|Q| H + 1)$, ρ_1 goes at least $(W_\mathsf{e} |Q| (H+1) + N) H$ times through the same pair of location and region, and thus contains at least $W_\mathsf{e} |Q| (H+1) + N$ region cycles of length at least H. Moreover, all of them are conforming to σ^1 which is a fake-ε/κ-optimal strategy. In particular, each cycle has a weight at most -1, and the weight of all cycles of ρ_1 is therefore at most $-W_\mathsf{e} |Q| (H+1) - N$. Moreover, ρ_1 has at most $|Q| H$ remaining edges (otherwise it would contain a region cycle of length at least H), once the region cycles are removed. The weight of one of these edges is at most W_e, so the weight of this set of edges is at most $W_\mathsf{e} |Q| H$. We can deduce that

$$\mathsf{wt}_\Sigma(\rho_1) \leq -W_\mathsf{e} |Q| (H+1) - N + W_\mathsf{e} |Q| H = -W_\mathsf{e} |Q| - N.$$

So, we can deduce that

$$\mathsf{wt}_\Sigma(\rho) = \mathsf{wt}_\Sigma(\rho_1) + \mathsf{wt}_\Sigma(\rho_2) \leq -W_\mathsf{e} |Q| - N + W_\mathsf{e} |Q| = -N.$$

In particular, we obtain that $\mathsf{wt}_\Sigma(\rho) \leq \max(-N, \mathsf{dVal}(\ell, \nu)) + \varepsilon$. To conclude the proof, we remark that ρ is a play reaching the target, so

$$\mathbf{P}(\rho) = \mathsf{wt}_\Sigma(\rho) \leq \max(-N, \mathsf{dVal}(\ell, \nu)) + \varepsilon\,. \qquad \square$$

Finally, in this context, as well as for SPGs, we also note that (by definition), every play conforming to a switching strategy has a length at most $\kappa + |Q|$ where $Q = L \times \mathsf{Reg}(\mathcal{X}, \mathsf{M})$ is the set of region states.

3.1.3. The case of reset-acyclic one-clock weighted timed games

Switching strategies have been considered in simple[1] one-clock WTGs in [Bri+15]. Using the particular structure of simple one-clock WTGs, the authors of [Bri+15] define a notion of fake-ε-optimal strategy distinct from ours (a priori). In particular, they remark that the time inside such WTGs only passes from 0 to 1 one time: so the maximal weight that we can obtain from the time is always W_loc. Knowing this fact, fake-ε-optimal strategies can focus only on the weight of the transitions used: the second condition deals with cyclic paths where the sum of the weights of transitions is negative (this hypothesis does not imply that the cyclic path has

[1] A simple one-clock WTG is a one-clock WTG without any reset and where all guards are $0 \leq x \leq 1$.

a negative weight). Next, these authors have extended their definition to reset-acyclic[2] one-clock WTGs and obtain the existence of a switching almost-optimal strategy in this case.

Remark. The fake-ε-optimal strategy defined in [Bri+15] fits with our definition when we consider cyclic paths that are longer than $W_{\mathsf{loc}} \times |Q|$ (we note that the converse is not true) where Q is the set of region states in $\mathcal{R}(\mathcal{G})$. ◁

Knowing the existence of an almost-optimal switching strategy in reset-acyclic one-clock WTGs does not suffice in our following contributions. We want to obtain an upper bound on the length of plays conforming to such strategies, i.e. we want an upper bound over the switching threshold κ. Indeed, by estimating the maximal number of steps needed by σ^1 to reach the target and combining this with κ, we can deduce an explicit pseudo-polynomial bound κ' that upper bounds the number of steps that the switching strategy σ needs to reach the target. From [Bri+22, Lemma 3.9], we know that

$$\kappa = O(|Q| \times (W_{\mathsf{loc}} + |\sigma^1| \times W_{\mathsf{tr}}|Q|) + |\sigma^1|)$$

where $|\sigma^1|$ is the size of this strategy, i.e. the number of cutpoints in $\mathsf{dVal}_{\mathcal{R}(\mathcal{G})}$ (by [Bri+22, Theorem 5.9]). Moreover, by [Bri+22, Theorem 5.13], we have a bound over the number of cutpoints in $\mathsf{dVal}_{\mathcal{R}(\mathcal{G})}$, i.e. $|\sigma^1| = O(W_{\mathsf{tr}}^4|Q|^9)$. Thus, we deduce that the switching threshold κ is approximated by

$$\kappa = O\left(|Q| \times \left[W_{\mathsf{loc}} + W_{\mathsf{tr}}^4|Q|^9 \times W_{\mathsf{tr}}|Q|\right] + W_{\mathsf{tr}}^4|Q|^9\right) = O\left(|Q|^{11}(W_{\mathsf{loc}} + W_{\mathsf{tr}}^5)\right) .$$

Then, we fix κ'' to be the number of turns taken by σ^2 to reach the target location, which is polynomial in the number of locations of the region automaton underlying the game, thus polynomial in the number of locations of the WTG (since it is only one-clock). Overall, this defines κ' as

$$\kappa' = \kappa + \kappa'' = O\left(|Q|^{12}(W_{\mathsf{loc}} + W_{\mathsf{tr}}^5)\right)$$

that is polynomial in $|L|$ (since $|Q|$ is polynomial in $|L|$) and in W. Thus, we obtain the following result.

Theorem 3.9 [Bri+22]. *Let $\mathcal{G}$ be a reset-acyclic WTG and $\varepsilon > 0$.* Min *has an ε-optimal switching strategy σ such that all plays conforming to σ reach the target within a number of steps polynomial in $|Q|$ and W.*

[2] A reset-acyclic one-clock WTG is a one-clock WTG where no cyclic region path contains a reset.

3.2. Almost-optimal strategies for both players in divergent weighted timed games

We conclude this chapter by proving Theorem 3.2. In particular, we explain how we define almost-optimal switching strategies for Min and, more precisely, the fake-ε-optimal strategies for all divergent WTGs. Moreover, we also remark that the definition of such strategies can be adapted to build memoryless almost-optimal strategies for Max.

3.2.1. Switching strategies for Min

We begin by looking at the switching strategies of Min. From Lemma 3.8, we know that showing that the existence of fake-ε-optimal strategies in all divergent WTGs for Min suffices to prove Theorem 3.2-1. The definition of such strategies relies on the computation of the value (by Theorem 1.34) and its fixpoint characterisation (by Theorem 2.1-2).

To begin with, let us recall how the value can be computed in Theorem 1.34. First, configurations (ℓ, ν) of value $\mathsf{dVal}(\ell, \nu) = -\infty$ are found by using a decomposition of the region game $\mathcal{R}(\mathcal{G})$ into SCC. Indeed, in divergent WTGs, configurations of value $-\infty$ are all the ones from which Min has a strategy to visit infinitely many times configurations of a single location (ℓ, r) of $\mathcal{R}(\mathcal{G})$ contained in a negative SCC. This is, thus, a Büchi objective on the region game that can easily be solved with some attractor computations. Notice that if a configuration (ℓ, ν) has value $-\infty$, this implies that all configurations (ℓ, ν') with ν' in the same region as ν have value $-\infty$. As we explained at the end of Section 1.3.2 for the value $+\infty$, we can remove configurations of value $-\infty$ by strengthening the guards on transitions while letting unchanged other finite values.

Then, the (finite) value function is obtained as an iterate V^H of the operator $\mathcal{F}$ (defined in Chapter 2), with H polynomial in the size of the region game and the maximal weights of $\mathcal{G}$. This means that playing for only a bounded number of steps is equivalent to the original WTG. As a side effect, this allows one to decompose the clock space $\mathbb{R}^{\mathcal{X}}_{\geq 0}$ into a finite number α of *cells* (a refinement of the classical regions[3]) such that the value function is affine on each cell.

In particular, we prove the following property:

Proposition 3.10. *In all divergent WTGs $\mathcal{G}$, for all $\varepsilon > 0$ and $N \in \mathbb{N}$, there exists a switching strategy $\sigma = \langle \sigma^1, \sigma^2, \kappa \rangle$ for* Min *such that, for all configurations (ℓ, ν) of $\mathcal{G}$, $\mathsf{dVal}^\sigma(\ell, \nu) \leq \max(-N, \mathsf{dVal}(\ell, \nu)) + \varepsilon$.*

[3] A more precise definition of these objects introduced in [ABM04; Bus19] is given in Chapter 10.

From this proposition, we deduce Theorem 3.2-1 since if all configurations have a finite value, there exists an almost-optimal switching strategy. In the presence of a configuration with a value $-\infty$, we build from Proposition 3.10 a family of switching strategies (indexed by the parameter N) whose value tends to $-\infty$.

In the rest of this section, we prove Proposition 3.10 by building strategy σ^1 as a fake-ε-optimal strategy. In the absence of values $-\infty$, the first property allows one to obtain a $\kappa\,\varepsilon$-optimal strategy even in the case where the switch does not occur (because we reach the target prematurely). The construction of a fake-ε/κ-optimal strategy σ (the linear dependency on the length of the play in the first property of fake-optimality is thus taken care of by a division by κ here) relies on the fact that $\mathcal{F}(\mathsf{dVal}) = \mathsf{dVal}$ (by Theorem 2.1-2) to play almost-optimally at horizon 1. More formally:

- For all configurations of value $-\infty$, σ^1 is built as a winning strategy for the Büchi objective "visit infinitely often configurations of a location (ℓ, r) of $\mathcal{R}(\mathcal{G})$ contained in a negative SCC". By definition, all cyclic paths following σ^1 will be inside a negative SCC, and thus be of cumulated weight at most -1, by divergence of the WTG. Moreover, no plays conforming to σ^1 from such a configuration of value $-\infty$ will reach a target location, since the chosen negative SCC is a trap controlled by Min.

- For the remaining configurations of finite value, we rely upon operator $\mathcal{F}$, letting σ^1 choose a decision that minimises the value at horizon 1. However, because of the guards on clocks, infimum/supremum operators in $\mathcal{F}$ are not necessarily minima/maxima, and we thus need to allow for a small error at each step of the strategy: this is the main difference with the untimed setting, which by the way explains why the definition of switching strategy from SPGs needed to be adapted. We will use the $\operatorname{arginf}^{\varepsilon}$ operator defined for all mappings $f\colon A \to \mathbb{R}$ and $B \subseteq A$ by $\operatorname{arginf}^{\varepsilon}_B f = \{a \in B \mid f(a) \leq \inf_B f + \varepsilon\}$. Then, for all configurations (ℓ, ν), we choose $\sigma(\ell, \nu)$ as a decision (δ, t) such that
$$\sigma^1(\ell,\nu) \in \operatorname{arginf}^{\varepsilon/\kappa}_{(\ell,\nu)\xrightarrow{\delta,t}(\ell',\nu')} \big[t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + \mathsf{dVal}(\ell',\nu')\big] \tag{3.1}$$
We note that this set is non-empty since the value function is a fixpoint of operator $\mathcal{F}$.

The strategy σ^1 thus built makes a small error wrt the optimal at each step. But once again strongly relying on the divergence of the WTG, we can nevertheless show that σ^1 is fake-ε/κ-optimal.

Lemma 3.11. *The strategy σ^1 of Min is fake-ε/κ-optimal, i.e.*

1. *each finite play ρ conforming to σ^1 from (ℓ,ν) and reaching the target has a payoff at most $\mathsf{dVal}(\ell,\nu)+|\rho|\frac{\varepsilon}{\kappa}$: in particular, if $\mathsf{dVal}(\ell,\nu)=-\infty$, no such play exists;*

2. *each finite play conforming to σ^1 following a cycle in the region game $\mathcal{R}(\mathcal{G})$ of length at least $|L|\alpha+1$ has a cumulated weight at most -1 where α is the number of cells.*

Proof. We show independently the two properties.

We begin by showing item 1. If $\mathsf{dVal}(\ell,\nu)=-\infty$, we have already seen before that no such play exists (since σ^1 is defined by a Büchi objective that does not contain the set of target locations). We thus restrict ourselves to considering configurations of value different from $-\infty$ in the following. We then reason by induction on the length of the plays ρ. If $|\rho|=0$, $\ell\in L_T$ and

$$\mathbf{P}(\rho)=0=\mathsf{dVal}(\ell,\nu)\leq\mathsf{dVal}(\ell,\nu)+0\times\frac{\varepsilon}{\kappa}\,.$$

If $|\rho|=n>0$, we write $\rho=(\ell,\nu)\xrightarrow{\delta,t}\rho'$, and let (ℓ',ν') be the first configuration of ρ'. By definition, we have $\mathbf{P}(\rho)=t\,\mathsf{wt}(\ell)+\mathsf{wt}(\delta)+\mathsf{wt}(\rho')$. By induction hypothesis, we obtain

$$\mathbf{P}(\rho)\leq t\,\mathsf{wt}(\ell)+\mathsf{wt}(\delta)+\mathsf{dVal}(\ell',\nu')+(n-1)\frac{\varepsilon}{K} \tag{3.2}$$

- If $\ell\in L_{\mathsf{Max}}$, then we can rewrite (3.2) (by applying the supremum) as

$$\begin{aligned}\mathbf{P}(\rho)&\leq\sup_{(\ell,\nu)\xrightarrow{\delta,t}(\ell',\nu')}\big(t\,\mathsf{wt}(\ell)+\mathsf{wt}(\delta)+\mathsf{dVal}(\ell',\nu')\big)+(n-1)\frac{\varepsilon}{\kappa}\\&\leq\mathcal{F}(\mathsf{dVal})(\ell,\nu)+n\frac{\varepsilon}{\kappa}\,.\end{aligned}$$

- If $\ell\in L_{\mathsf{Min}}$, then we can rewrite (3.2) as

$$\begin{aligned}\mathbf{P}(\rho)&\leq\inf_{(\ell,\nu)\xrightarrow{\delta,t}(\ell',\nu')}\big(t\,\mathsf{wt}(\ell)+\mathsf{wt}(\delta)+\mathsf{dVal}(\ell',\nu')\big)+\frac{\varepsilon}{\kappa}+(n-1)\frac{\varepsilon}{\kappa}\\&\leq\mathcal{F}(\mathsf{dVal})(\ell,\nu)+n\frac{\varepsilon}{\kappa}\end{aligned}$$

  by using the definition of σ^1 given in (3.1).

Finally, since dVal is a fixpoint of $\mathcal{F}$ (by Theorem 2.1-2), we conclude that $\mathbf{P}(\rho)\leq\mathsf{dVal}(\ell,\nu)+n\frac{\varepsilon}{\kappa}$.

Now, we prove item 2. The second property is trivial (and already discussed before) when the strategy σ^1 is chosen according to the case of values $-\infty$. We

therefore once again suppose that there are no remaining configurations of value $-\infty$ in this proof.

We first show that, for all edges $(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')$ conforming to σ^1, we have

$$\mathsf{dVal}(\ell,\nu) \geq t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + \mathsf{dVal}(\ell',\nu') - \frac{\varepsilon}{\kappa} \tag{3.3}$$

- If $\ell \in L_{\mathsf{Min}}$, since dVal is a fixpoint of $\mathcal{F}$ (by Theorem 2.1-2), we have

$$\begin{aligned}\mathsf{dVal}(\ell,\nu) &= \inf_{(\ell,\nu)\xrightarrow{\delta_1,t_1}(\ell_1,\nu_1)} \left[t_1\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta_1) + \mathsf{dVal}(\ell_1,\nu_1)\right] \\ &\geq t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + \mathsf{dVal}(\ell',\nu') - \frac{\varepsilon}{\kappa}\end{aligned}$$

  when $\sigma^1(\ell,\nu) = (\delta,t)$.

- If $\ell \in L_{\mathsf{Max}}$, since dVal is a fixpoint of $\mathcal{F}$ (by Theorem 2.1-2), we have

$$\begin{aligned}\mathsf{dVal}(\ell,\nu) &= \sup_{(\ell,\nu)\xrightarrow{\delta_1,t_1}(\ell_1,\nu_1)} \left[t_1\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta_1) + \mathsf{dVal}(\ell_1,\nu_1)\right] \\ &\geq t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + \mathsf{dVal}(\ell',\nu') \\ &\geq t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + \mathsf{dVal}(\ell',\nu') - \frac{\varepsilon}{\kappa}\end{aligned}$$

  for all feasible decisions (δ,t) from (ℓ,ν).

Then, let π be a cyclic path in $\mathcal{R}(\mathcal{G})$ of length at least $|L|\alpha+1$, and ρ be a finite play following π and conforming to σ^1. It goes through only states of a single SCC of $\mathcal{R}(\mathcal{G})$, that is either positive or negative, by the divergence of the WTG $\mathcal{G}$. We show that this SCC is necessarily negative, therefore proving $\mathsf{wt}_\Sigma(\rho) \leq -1$. Suppose, on the contrary, that the SCC is positive.

First, since π has a length of at least $|L|\alpha+1$, it must go through the same pair of locations and cells of dVal (remember that dVal is piecewise affine, and we have called cell each piece where it is affine) at least twice. Consider the cycle π' (of length at most $|L|\alpha$) obtained by considering the part of π between these two occurrences, and ρ' a corresponding finite play starting in a configuration (ℓ,ν), ending in (ℓ,ν') (with ν and ν' in the same cell). Since the SCC is positive, ρ' has a weight of at least 1. By the definition of σ^1 taking a uniform decision on each cell, we can mimic this loop once again from (ℓ,ν'), for $2\lceil\beta\rceil$ times, where β is the maximum over the cells c (where dVal is not $-\infty$) of the differences $\sup_c \mathsf{dVal} - \inf_c \mathsf{dVal}$, that is finite since the value function is affine in each cell that is bounded. The obtained play ρ'' has a weight of at least $2\lceil\beta\rceil$ (since it is decomposed as $2\lceil\beta\rceil$ cycles in a positive SCC), and a length at most $2\lceil\beta\rceil\,|L|\alpha$.

By summing up all the inequalities given by (3.3) along the play ρ'' (from (ℓ, ν) to (ℓ, ν'')), and by simplifying the result, we obtain that

$$\mathsf{wt}_\Sigma(\rho'') \leq \mathsf{dVal}(\ell,\nu) - \mathsf{dVal}(\ell,\nu'') + \frac{\varepsilon}{\kappa}\,|\rho''| \leq \mathsf{dVal}(\ell,\nu) - \mathsf{dVal}(\ell,\nu'') + \frac{\varepsilon}{\kappa}\,2\,\lceil\beta\rceil\,|L|\,\alpha.$$

However, by definition of β (the maximum difference of values in a cell), we obtain

$$\mathsf{wt}_\Sigma(\rho'') \leq \lceil\beta\rceil\left(1 + \frac{\varepsilon}{\kappa}\,2\,|L|\,\alpha\right) \leq \lceil\beta\rceil\,(1+2\varepsilon) < 2\,\lceil\beta\rceil$$

since $\kappa \geq 2\,|L|\,\alpha$ and $\varepsilon < 1/2$. This contradicts the fact that ρ'' has a weight of at least $2\,\lceil\beta\rceil$ and concludes the proof of the second item. □

Remark. From Lemma 3.11, we deduce that the switching threshold κ can be set to $(W_{\mathsf{e}}\,|Q|(|L|\alpha+2)+N)(|Q|\,(|L|\alpha+1)+1)$. ◁

3.2.2. Memoryless strategies for Max

In a divergent WTG, with the same technique as for switching strategies for Min, we can prove that Max has an almost-optimal memoryless strategy. To make this proof, we take the point of view of Max; looking for good strategies for this player. This is possible since all WTGs are determined [Bri+22, Theorem 2.2]. We may thus associate a value with any strategy τ of Max:

$$\mathsf{dVal}^\tau(\ell,\nu) = \inf_{\sigma\in\mathsf{dStrat}_{\mathsf{Min}}} \mathbf{P}\big(\mathsf{Play}((\ell,\nu),\sigma,\tau)\big)\,.$$

Then, the strategy τ is ε-optimal if $\mathsf{dVal}^\tau(\ell,\nu) \geq \mathsf{dVal}(\ell,\nu) - \varepsilon$ for all configurations (ℓ,ν).

As Max does not wish to go to the target, we show that no switch is necessary to play almost-optimally: memoryless strategies are sufficient to guarantee a value as close as wanted to the value of the WTG. For a configuration with a value equal to $-\infty$, all the strategies for Max are equivalent where they are all equally bad. Without loss of generality, we can therefore suppose that there are no configurations in $\mathcal{G}$ with a value equal to $-\infty$. Then, it is shown in [BMR17] that the remaining values are bounded in absolute value by $W_{\mathsf{e}}|Q|$ where Q is the set of region states in $\mathcal{R}(\mathcal{G})$, since optimal plays have no cycles.

In the rest of this section, we prove Theorem 3.2-2, by using that fact to build a memoryless strategy τ analogous to strategy σ^1 before. Once again, we use that the value function is a fixpoint of the operator $\mathcal{F}$ (once we have removed configurations of value $-\infty$). We must still accommodate small errors at each step, because of the guards on clocks. Similarly to the switching parameter κ for the switching strategy (with $N = 0$), we let $\kappa = W_{\mathsf{e}}\,|Q|\,(|L|\,\alpha+2)\,(|Q|(|L|\,\alpha+1)+1)$ where

Q is the set of region states in $\mathcal{R}(\mathcal{G})$. We use the argsup$^\varepsilon$ operator, analogously to arginf$^\varepsilon$, defined for all mappings $f\colon A \to \mathbb{R}$ and $B \subseteq A$ by $\operatorname{argsup}^\varepsilon_B f = \{a \in B \mid f(a) \geq \sup_B f - \varepsilon\}$. Then, for all configurations (ℓ, ν), we choose $\tau(\ell, \nu)$ as a decision (δ, t) such that

$$\tau(\ell,\nu) \in \operatorname{argsup}^{\varepsilon/\kappa}_{(\ell,\nu)\xrightarrow{\delta,t}(\ell',\nu')} (t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + \mathsf{dVal}(\ell',\nu'))\,. \tag{3.4}$$

In a completely symmetrical way as for Lemma 3.11, we show that τ satisfies the following useful properties:

Lemma 3.12. *The strategy τ satisfies the following properties:*

1. *each finite play ρ conforming to τ from (ℓ, ν) and reaching the target has a cumulated weight at least $\mathsf{dVal}(\ell, \nu) - |\rho|\frac{\varepsilon}{\kappa}$;*
2. *each finite play conforming to τ, that follows a cycle in the region game $\mathcal{R}(\mathcal{G})$ of length at least $|L|\alpha + 1$, has a cumulated weight at least 1 where α is the number of cells.*

Now, we have tools to prove that τ is an ε-optimal strategy for Max from (ℓ, ν). In other words, we show that $\mathsf{dVal}^\tau(\ell, \nu) \geq \mathsf{dVal}(\ell, \nu) - \varepsilon$.

Let ρ be a play conforming to τ from (ℓ, ν). We distinguish two cases according to the length of ρ. First, we suppose that $|\rho| \leq \kappa$, then, by Lemma 3.12-1, we have

$$\mathbf{P}(\rho) \geq \mathsf{dVal}(\ell,\nu) - \frac{\varepsilon}{\kappa}|\rho| \geq \mathsf{dVal}(\ell,\nu) - \varepsilon\,.$$

Otherwise, we suppose that $|\rho| > \kappa$, then by definition of $\kappa = W_{\mathsf{e}}\,|Q|\,(|L|\,\alpha + 2)\,(|Q|\,(|L|\,\alpha + 1) + 1)$, ρ contains at least $W_{\mathsf{e}}\,|Q|\,(|L|\,\alpha + 2)$ cycles of length at least $|L|\,\alpha + 1$ in $\mathcal{R}(\mathcal{G})$. By Lemma 3.12-2, all of these cycles have a cumulated weight of at least 1. So, the cumulated weight of ρ induced by the cycles in $\mathcal{R}(\mathcal{G})$ is at least $W_{\mathsf{e}}\,|Q|\,(|L|\,\alpha + 2)$. Moreover, ρ has at most $|Q|\,(|L|\,\alpha + 1)$ remaining edges once the region cycles are removed. The weight of each of these edges is at least $-W_{\mathsf{e}}$, so the weight of this set of edges is at least $-W_{\mathsf{e}}\,|Q|\,(|L|\,\alpha + 1)$. Now, since $W_{\mathsf{e}}\,|Q| \geq \mathsf{dVal}(\ell, \nu)$ (by [BMR17]), we deduce that

$$\mathbf{P}(\rho) \geq W_{\mathsf{e}}\,|Q|\,(|L|\,\alpha + 2) - W_{\mathsf{e}}\,|Q|\,(|L|\,\alpha + 1) = W_{\mathsf{e}}\,|Q| \geq \mathsf{dVal}(\ell,\nu)\,.$$

In particular, we obtain that $\mathbf{P}(\rho) \geq \mathsf{dVal}(\ell, \nu) - \varepsilon$.

4. Deterministic Value Problem in One-clock Weighted Timed Games

Despite several recent works, the decidability of the deterministic value problem in one-clock WTGs, i.e. deciding if the deterministic value of a given configuration is less than a fixed threshold, remained open. Our main contribution concerning the value is to push the decidability frontier by proving that the value function is computable for all one-clock WTGs. Moreover, we present a first implementation of a value iteration-based algorithm to solve all one-clock WTGs.

Table of contents

4.1. Closure of one-clock weighted timed games 84
4.2. How Max can control negative cycles in the closure 90
4.3. A partial unfolding of the closure 95
4.3.1. How Max controls negative "cycles" in the unfolding 99
4.3.2. Value of the unfolding 103
4.4. Decidability of the value problem 112
4.5. Implementation 112

In this chapter, we will prove that the value function of all one-clock WTGs is computable in exponential time. Even if the effective computation of the value function is based on techniques of [Bri+22], to obtain the result, we use a technique more similar to the one in [Bou+06] used to obtain the decidability in one-clock WTGs with only non-negative weights.

Theorem 4.1. *For all one-clock WTGs $\mathcal{G}$ and all initial locations ℓ_{i}, the mapping $\nu \mapsto \mathsf{dVal}_{\mathcal{G}}(\ell_{\mathsf{i}}, \nu)$ is computable in time exponential in $|L|$ and W.*

Remark. The complexities of Theorem 4.1 would be more traditionally considered as doubly-exponential if weights of the one-clock WTG were encoded in binary as

usual. In this work, we thus count the complexities as if all weights were encoded in unary and, thus, consider W to be the bound of interest. ◁

To provide the proof of Theorem 4.1, we let $\mathcal{G}$ be a one-clock WTG and ℓ_{i} be an initial location. The first step of the proof, described in Section 4.1, is to define a special region game $\overline{\mathcal{G}}$ [Bri+22], called *closure*, in which we close all the guards and we keep in mind the region that we originally wanted to reach by its closure: for example a guard $0 \leq x \leq 1$ is used to reach the region $(0, 1)$. Since this transformation preserves the value function [Bri+22], we deduce the result by computing the value functions $\nu \mapsto \mathsf{dVal}_{\overline{\mathcal{G}}}((\ell_{\mathsf{i}}, r), \nu)$ for all regions r. Regions r over which $\nu \mapsto \mathsf{dVal}_{\overline{\mathcal{G}}}((\ell_{\mathsf{i}}, r), \nu)$ is constantly equal to $+\infty$ or $-\infty$ are computable in polynomial time, as explained in [Bri+22]. We, therefore, remove them from $\overline{\mathcal{G}}$ from now on. We now fix an initial region r_{i} and let $q_{\mathsf{i}} = (\ell_{\mathsf{i}}, r_{\mathsf{i}})$, and explain how to compute $\nu \mapsto \mathsf{dVal}_{\overline{\mathcal{G}}}((\ell_{\mathsf{i}}, r_{\mathsf{i}}), \nu)$ on the interval r_{i}.

As in the non-negative case [Bou+06], the objective is to limit the number of transitions with a reset taken into the plays while not modifying the value of the WTG. When all weights are non-negative, this is fairly easy to achieve since, intuitively speaking, Min has no interest in using any cycles containing such a transition (since it has non-negative weight and is thus non-beneficial for Min). The WTG can thus be transformed such that each transition with a reset is taken at most once. To obtain a smaller WTG, it is even possible to simply count the number of transitions with a reset taken so far in the play and stop the WTG (with a final weight $+\infty$) in case the counter goes above the number of such transitions in the WTG. The transformed WTG has a polynomial number of locations with respect to the original WTG and is reset-acyclic, which allows one to solve it by using Theorem 1.32, with a time complexity polynomial in $|L|$ and W (instead of the exponential time complexity originally achieved in [Bou+06; Rut11] with respect to $|L|$).

The situation is much more intricate in the presence of negative weights since negative cycles containing a transition with a reset can be beneficial for Min, as we have seen in Example 1.26. Notice that this is still true in the closure of the WTG, as can be checked in Figure 1.7. Moreover, some cyclic paths may have plays following it with a positive weight and plays following it with a negative weight, making it difficult to determine whether it is beneficial to Min (or not). To overcome this situation, we will consider the point of view of Max, benefiting from the determinacy of the WTG. We will show that, in the closure, Max can play optimally with memoryless strategies while avoiding negative cyclic plays. This will simplify our further study since, by following this strategy, Max ensures that only non-negative cyclic plays will be encountered, which is not beneficial to Min. Therefore, as in [Bou+06], we will limit the firing of transitions with a reset to at most once. However, we are not able to do it without blowing up

exponentially the number of locations of the WTG. Indeed, along the unfolding of the WTG, we need to record enough information to know, in case a cyclic path ending with a reset is closed, whether this cyclic path has a potential negative weight (in which case Max will indeed not follow it) or non-negative weight (in which case it is not beneficial for Min to close the cycle). Determining in which case we are will be made possible by introducing the notion of the value of a cyclic path in Section 4.2. Then, Max has even an optimal strategy to avoid closing cyclic paths with a negative value (which is stronger than only avoiding creating negative cyclic plays). The unfolding, denoted $\mathcal{U}$, will be defined in Section 4.3. Section 4.3.1 shows that Max keeps its ability to play without falling into a negative "cycle" in the unfolding. This allows us to show in Section 4.3.2 that the unfolding WTG has a value equal to the one of the closure WTG. This allows us to wrap up the proof of Theorem 4.1 in Section 4.4.

Final weight functions For technical reasons in the computation of the value function, we need to equip our one-clock WTGs with a *final weight function*: when a target configuration is reached, we add the final weight induced by the function of the final location applied to the valuation to the cumulative weight. This definition has been previously used used to decompose into SCC the solution of divergent WTGs [Bus19]. Here, we use this technique to close our unfolding and, thus, obtain a finite unfolding. In the rest of this chapter, we therefore consider $\mathcal{G}$ to be a one-clock WTG defined by the following tuple $\langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \{x\}, \Delta, \mathsf{wt}, \mathsf{wt}_T\rangle$ such that $\langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \{x\}, \Delta, \mathsf{wt}\rangle$ is a one-clock WTG and $\mathsf{wt}_T : L_T \times \mathbb{R}_{\geq 0} \to \mathbb{R}_\infty$ is a function mapping each target configuration to a final weight.

We also assume that the final weight functions satisfy a sufficient property ensuring that they can be encoded in finite space: we require final weight functions to be piecewise affine with a finite number of pieces and continuous on each region. More precisely, we assume that cutpoints (the value of the clock in-between two affine pieces) and coefficients are rational and given in binary. In particular, we let W_{fin} be the maximum absolute value of weights of final functions, i.e.

$$W_{\mathsf{fin}} = \sup_{\substack{q \in Q_t \\ \mathsf{wt}_T(q,\cdot) \notin \{+\infty, -\infty\}}} \sup_{\nu} |\mathsf{wt}_T(q,\nu)|$$

We also let W be the maximum of W_{loc}, W_{tr}, and W_{fin}.

Finally, introducing final weight functions modifies the payoff of plays. Formally, given a finite play ρ reaching a target configuration (ℓ, ν) in $\mathcal{G}$, we extend the payoff of ρ with the final weight function, i.e. we fix $\mathbf{P}(\rho) = \mathsf{wt}_\Sigma(\rho) + \mathsf{wt}_T(\ell, \nu)$ in this case.

4.1. Closure of one-clock weighted timed games

As a reminder, a WTG can be populated with the region information without loss of generality in its region game (see Section 1.2.2 for details). However, to solve one-clock WTGs, this transformation does not suffice. Indeed, to solve reset-acyclic one-clock WTGs in [Bri+22], authors do not use the usual region game: their method is based on a construction that consists in not only enhancing the locations with regions (as the region game) but also closing all guards while preserving the value of the original game.

Definition 4.2. The *closure* of a one-clock WTG $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \{x\}, \Delta, \mathsf{wt}, \mathsf{wt}_T \rangle$ is the one-clock WTG $\overline{\mathcal{G}} = \langle Q_{\mathsf{Min}}, Q_{\mathsf{Max}}, Q_t, Q_u, \{x\}, \overline{\Delta}, \overline{\mathsf{wt}}, \overline{\mathsf{wt}_T} \rangle$ where:

- $Q = Q_{\mathsf{Min}} \uplus Q_{\mathsf{Max}} \uplus Q_t$ is the set of region[1] states from $\mathcal{G}$;
- for all $(\ell, r) \in Q$, the transition $\big((\ell, r), \overline{g \cap r''}, Y, (\ell', r')\big) \in \overline{\Delta}$ if and only if there exist a transition $(\ell, g, Y, \ell') \in \Delta$, and a region r'' such that the interval[2] $g \cap r''$ is not empty, the lower bound of r'' is a time successor of r, and r' is equal to r'' if $Y = \emptyset$ and to $\{0\}$ otherwise: $\overline{g \cap r''}$ stands for the topological closure of the non-empty interval $g \cap r''$. Moreover, in this case, we let $\overline{\mathsf{wt}}\big((\ell, r), \overline{g \cap r''}, Y, (\ell', r')\big) = \mathsf{wt}(\ell, g, Y, \ell')$;
- for all region states (ℓ, r), we have $\overline{\mathsf{wt}}(\ell, r) = \mathsf{wt}(\ell)$;
- for all $(\ell, r) \in Q_t$, for $\nu \in r$, $\overline{\mathsf{wt}_T}((\ell, r), \nu) = \mathsf{wt}_T(\ell, \nu)$ and extend $\nu \mapsto \overline{\mathsf{wt}_T}((\ell, r), \nu)$ by continuity on $\bar{r}$, the closure of the interval r. We may also let $\overline{\mathsf{wt}_T}((\ell, r), \nu) = +\infty$ for all $\nu \notin \bar{r}$, even though we will never use this in the following.

An example of closure is given in Figure 4.1, which depicts the closure of the one-clock WTG depicted in Figure 1.7, restricted to locations ℓ_0, ℓ_1, ℓ_2, and ☺ (we have seen that ℓ_3 is anyway useless).

The semantics of the closure is obtained by concentrating on the following set of configurations which is an invariant of the closure (i.e. starting from a configuration fulfilling the invariant, we can only reach configurations fulfilling the invariant):

- configurations $((\ell, \{M_k\}), M_k)$;
- and configurations $((\ell, (M_k, M_{k+1})), \nu)$ with $\nu \in [M_k, M_{k+1}]$ (and not only in (M_k, M_{k+1}) as one might expect in the region game);

[1] As explained in Section 1.3.3.2, we use the definition induced by [LMS04] for regions.

[2] We recall that since we consider only one-clock WTG, we represent guards by an interval of real value.

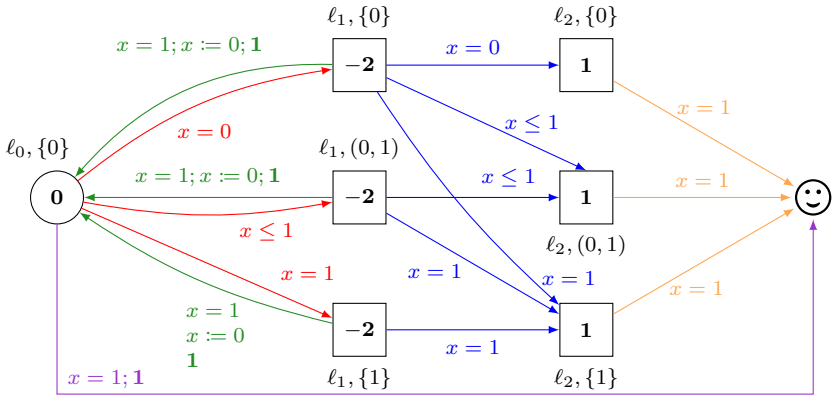


Figure 4.1 – The restriction of the closure of the one-clock WTG depicted in Figure 1.7 to locations ℓ_0, ℓ_1, ℓ_2 and ☺.

where $(M_i)_i$ is the set of constants that appear in guards of the WTG and that define regions (like in Section 1.3.3.2).

The closure of the guards allows players to mimic a decision in $\mathcal{G}$ "arbitrarily close" to M_k (or M_{k+1}) in (M_k, M_{k+1}) to be simulated by jumping on M_k (or M_{k+1}) while staying in the region (M_k, M_{k+1}). In particular, it is shown in [Bri+22] that we can transform an ε-optimal strategy of $\overline{\mathcal{G}}$ into an ε'-optimal strategy of $\mathcal{G}$ with $\varepsilon' < 2\varepsilon$ and vice-versa. Thus, the closure of a one-clock WTG preserves its value.

Lemma 4.3 [Bri+22, Lemma 6.1]**.** *For all one-clock WTGs $\mathcal{G}$, $(\ell, r) \in Q$ and $\nu \in r$,*

$$\mathsf{dVal}_{\mathcal{G}}(\ell, \nu) = \mathsf{dVal}_{\overline{\mathcal{G}}}((\ell, r), \nu)\,.$$

Moreover, the closure construction also makes the value function more manageable for our purpose. Indeed, as shown in [Bri+22], the mapping $\nu \mapsto \mathsf{dVal}_{\mathcal{G}}(q, \nu)$ is continuous over all regions, but there might be discontinuities at the borders of the regions. The closure construction clears this issue by softening the borders of each region independently: we show the continuity of the value function on each closed region (and not only on the regions) in the closure game by following a very similar sketch as the one of [Bri+22, Theorem 3.2].

Lemma 4.4. *For all one-clock WTGs $\mathcal{G}$ and initial location $(\ell, r) \in Q$, the mapping $\nu \mapsto \mathsf{dVal}_{\overline{\mathcal{G}}}((\ell, r), \nu)$ is continuous over $\overline{r}$.*

Proof. The main ingredient of our proof is, given a strategy σ of Min in $\overline{\mathcal{G}}$, a location $q = (\ell, r)$ of $\overline{\mathcal{G}}$, and valuations $\nu, \nu' \in \overline{r}$ (and not only $\nu, \nu' \in r$ as in the proof of [Bri+22, Theorem 3.2]), to show how to build a strategy σ' in $\overline{\mathcal{G}}$ and a length-preserving function f that maps plays of $\overline{\mathcal{G}}$ starting in (q, ν') and conforming to σ' to plays of $\overline{\mathcal{G}}$ starting in (q, ν) conforming to σ with similar behaviour and

weight. More precisely, we define σ' and f by induction on the length k of the finite play that is given as an argument and relies on the following set of induction hypotheses:

Induction hypothesis: There exist a strategy σ', only defined on plays of length at most $k-1$ starting in (q,ν'), and a function f mapping plays of length k starting in (q,ν') conforming to σ' to plays of length k starting in (q,ν) conforming to σ such that for all plays $\rho' = (q_0 = q, \nu'_0 = \nu') \xrightarrow{\delta'_0,t'_0} \cdots \xrightarrow{\delta'_{k-1},t'_{k-1}} (q_k,\nu'_k)$ conforming to σ', letting $(q_0,\nu_0=\nu) \xrightarrow{\delta_0,t_0} \cdots \xrightarrow{\delta_{k-1},t_{k-1}} (q_k,\nu_k)$ the play $f(\rho')$, we have:

1. $|\nu_k - \nu'_k| \leq |\nu - \nu'|$;
2. $\mathsf{wt}_\Sigma(\rho') \leq \mathsf{wt}_\Sigma(f(\rho')) + W_{\mathsf{loc}}(|\nu-\nu'| - |\nu_k - \nu'_k|)$.

We note that no property is required on the strategy σ' for finite plays that do not start in (q,ν'). Moreover, by the invariants of $\overline{\mathcal{G}}$, we have that for every $i \in \{0,\ldots,k\}$, ν_i and ν'_i belong to the interval $\overline{r_i}$ such that $q_i = (\ell_i, r_i)$.

Let us explain how this result would imply the desired result before going through the induction itself, i.e. why the mapping $\nu \mapsto \mathsf{dVal}_{\overline{\mathcal{G}}}((\ell,r),\nu)$ is continuous over $\overline{r}$. We remark first that the result directly implies that if the value of $\overline{\mathcal{G}}$ is finite for some valuation ν in $\overline{r}$, then it is finite for all other valuations ν' in $\overline{r}$. Indeed, a finite value of the game in (q,ν) implies that there exists a strategy σ such that every play starting in (q,ν) and conforming to it reaches a target location in a final valuation such that the final weight function appired in this last configuration is finite. Moreover, denoting σ' the strategy obtained from σ thanks to the above result, any play ρ' starting in (q,ν') and conforming to σ' reaches a target location (since $f(\rho')$ does as a play conforming to σ). Moreover, we recall that final weight function is either always finite or always infinite within a region (by hypothesis). In particular, the final weight function of $\mathsf{last}(\rho')$ is finite as the final valuations of ρ' and $f(\rho')$ sit in the same region.

Now, assuming the value of $\mathcal{G}$ is finite over $\overline{r}$, we show that the value function is continuous over $\overline{r}$. To do it, we show that, for all $\nu \in \overline{r}$, for all $\varepsilon > 0$, there exists $\delta > 0$ such that for all $\nu' \in \overline{r}$ with $|\nu - \nu'| \leq \delta$, we have $|\mathsf{dVal}_{\overline{\mathcal{G}}}(q,\nu) - \mathsf{dVal}_{\overline{\mathcal{G}}}(q,\nu')| \leq \varepsilon$. To this end, we can show that:

$$|\mathsf{dVal}_{\overline{\mathcal{G}}}(q,\nu) - \mathsf{dVal}_{\overline{\mathcal{G}}}(q,\nu')| \leq (W_{\mathsf{loc}} + K)|\nu - \nu'| \tag{4.1}$$

where K is the greatest absolute value of the slopes appearing in the piecewise affine functions within wt_T. Indeed, assume that this inequality holds, and consider $\nu \in \overline{I}$ and a positive real number ε. Then, we let $\delta = \frac{\varepsilon}{W_{\mathsf{loc}}+K}$, and we consider a

valuation ν' such that $|\nu - \nu'| \leq \delta$. In this case, (4.1) becomes

$$|\mathsf{dVal}_{\overline{\mathcal{G}}}(q,\nu) - \mathsf{dVal}_{\overline{\mathcal{G}}}(q,\nu')| \leq (W_{\mathsf{loc}} + K)|\nu - \nu'| \leq (W_{\mathsf{loc}} + K)\frac{\varepsilon}{W_{\mathsf{loc}} + K} \leq \varepsilon\,.$$

Thus, proving (4.1) is sufficient to establish continuity.

On the other hand, (4.1) is equivalent to:

$$\begin{aligned}\mathsf{dVal}_{\overline{\mathcal{G}}}(q,\nu) &\leq \mathsf{dVal}_{\overline{\mathcal{G}}}(q,\nu') + (W_{\mathsf{loc}} + K)|\nu - \nu'| \quad \text{and}\\ \mathsf{dVal}_{\overline{\mathcal{G}}}(q,\nu') &\leq \mathsf{dVal}_{\overline{\mathcal{G}}}(q,\nu) + (W_{\mathsf{loc}} + K)|\nu - \nu'|\,.\end{aligned}$$

As those last two equations are symmetric with respect to ν and ν', we only have to show either of them. We thus focus on the latter, which, by using the upper value, can be reformulated as: for all strategies σ of Min, there exists a strategy σ' such that

$$\mathsf{dVal}^{\sigma'}_{\overline{\mathcal{G}}}(q,\nu') \leq \mathsf{dVal}^{\sigma}_{\overline{\mathcal{G}}}(q,\nu) + (W_{\mathsf{loc}} + K)|\nu - \nu'|\,.$$

We note that this last equation is equivalent to saying that there exists a function f mapping plays ρ' from (q,ν') conforming to σ' to plays from (q,ν) conforming to σ such that, for all such ρ' the final valuations of ρ' and $f(\rho')$ differ by at most $|\nu - \nu'|$ and

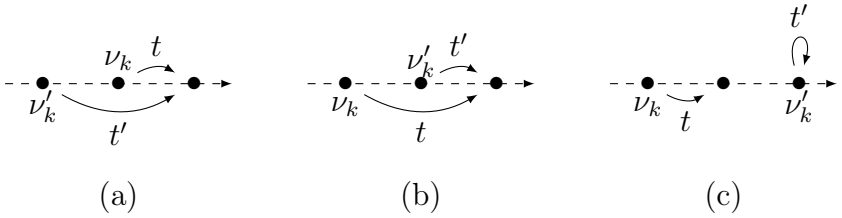
$$\mathsf{wt}_\Sigma(\rho') \leq \mathsf{wt}_\Sigma(f(\rho')) + W_{\mathsf{loc}}|\nu - \nu'|$$

which is exactly what we claimed induction achieves since $-|\nu_k - \nu'_k| \leq 0$. Thus, to conclude this proof, let us now define σ' and f, by induction on the length k of ρ'.

Base case $k = 0$: In this case, σ' does not have to be defined since there are no plays of length -1. Moreover, in that case, $\rho' = (q,\nu')$ and $f(\rho') = (q,\nu)$, in which case both properties are trivial.

Inductive case: Let us suppose now that the construction is done for a given $k \geq 0$ and perform it for $k+1$. We start with the construction of σ'. To that extent, we consider a play $\rho' = (q_0 = q, \nu'_0 = \nu') \xrightarrow{\delta'_0,t'_0} \cdots \xrightarrow{\delta'_{k-1},t'_{k-1}} (q_k,\nu'_k)$ conforming to σ' (provided by induction hypothesis) such that $q_k \in Q_{\mathsf{Min}}$. Let (δ,t) be the decision made by σ on $f(\rho')$, i.e. $\sigma(f(\rho')) = (\delta,t)$. Then, we define $\sigma'(\rho') = (\delta',t')$ where $t' = \max(0,\nu_k + t - \nu'_k)$. The delay t' respects the guard of transition δ, as can be seen from Figure 4.2. Indeed, either $\nu_k + t = \nu'_k + t'$ (cases (a) and (b) in Figure 4.2) or $\nu_k \leq \nu_k + t \leq \nu'_k$ (case (c) in Figure 4.2 where $t' = 0$), in which case ν'_k is in the same closure of region as $\nu_k + t$ since ν_k and ν'_k are in the same closure of region by induction hypothesis: we conclude by noticing that the guard of δ is closed.

Let us now build the mapping f. Let $\rho' = (q_0 = q, \nu'_0 = \nu') \xrightarrow{\delta'_0,t'_0} \cdots \xrightarrow{\delta'_k,t'_k}$



(a) (b) (c)

Figure 4.2 – The definition of t' when (a) $\nu'_k \leq \nu_k$; (b) $\nu_k < \nu'_k < \nu_k + t$; or (c) $\nu_k < \nu_k + t < \nu'_k$.

(q_{k+1}, ν'_{k+1}) be a play conforming to σ' and let $\tilde{\rho}' = (q_0, \nu'_0) \xrightarrow{\delta'_0, t'_0} \cdots \xrightarrow{\delta'_{k-1}, t'_{k-1}} (q_k, \nu'_k)$ its prefix of length k. Using the construction of f over plays of length k by induction, the play $f(\tilde{\rho}') = (q_0, \nu_0 = \nu) \xrightarrow{\delta_0, t_0} \cdots \xrightarrow{\delta_{k-1}, t_{k-1}} (q_k, \nu_k)$ satisfies properties 1 and 2. Then:

- if $q_k \in Q_{\mathsf{Min}}$ and $\sigma(f(\tilde{\rho}')) = (\delta, t)$, then $f(\rho') = f(\tilde{\rho}') \xrightarrow{\delta, t} (q_{k+1}, \nu_{k+1})$ is obtained by applying those choices on $f(\tilde{\rho}')$. By the construction of σ', we moreover have $\delta'_k = \delta$;
- if $q_k \in Q_{\mathsf{Max}}$, the last valuation ν_{k+1} of $f(\rho')$ is rather obtained by choosing action (δ'_k, t) verifying $t = \max(0, \nu'_k + t'_k - \nu_k)$. We note that transition δ'_k is allowed since both $\nu_k + t$ and $\nu'_k + t'_k$ are in the same closure of region (for similar reasons as above).

Moreover, by induction hypothesis $\tilde{\rho}'$ and $f(\tilde{\rho}')$ have the same length.

Now, to prove property 1, we notice that we always have either

$$\nu_k + t = \nu'_k + t'_k \quad \text{or} \quad \nu_k \leq \nu_k + t \leq \nu'_k = \nu'_k + t'_k \quad \text{or} \quad \nu'_k \leq \nu'_k + t_k \leq \nu_k = \nu_k + t_k \,.$$

In all of these possibilities, we have $|(\nu_k + t) - (\nu'_k + t'_k)| \leq |\nu_k - \nu'_k|$.

We finally check property 2. Either q_k belongs to Min or to Max, using the induction hypothesis, we have

$$\begin{aligned}
\mathsf{wt}_\Sigma(\rho') &= \mathsf{wt}_\Sigma(\tilde{\rho}') + \overline{\mathsf{wt}}(\delta'_k) + t'_k\, \overline{\mathsf{wt}}(q_k) \\
&\leq \mathsf{wt}_\Sigma(f(\tilde{\rho}')) + W_{\mathsf{loc}}(|\nu - \nu'| - |\nu_k - \nu'_k|) + \overline{\mathsf{wt}}(\delta'_k) + t'_k\, \overline{\mathsf{wt}}(q_k) \\
&= \mathsf{wt}_\Sigma(f(\rho')) + (t'_k - t)\, \overline{\mathsf{wt}}(q_k) + W_{\mathsf{loc}}(|\nu - \nu'| - |\nu_k - \nu'_k|) \,.
\end{aligned}$$

To conclude, let us claim that

$$|t'_k - t| \leq |\nu_k - \nu'_k| - |\nu'_{k+1} - \nu_{k+1}| \tag{4.2}$$

Thus, since $|\mathsf{wt}(q_k)| \leq W_{\mathsf{loc}}$, we conclude that:

$$\mathsf{wt}_\Sigma(\rho') \leq \mathsf{wt}_\Sigma(f(\rho')) + W_{\mathsf{loc}}(|\nu - \nu'| - |\nu_{k+1} - \nu'_{k+1}|)$$

which concludes the induction.

To conclude the proof, we prove (4.2). First, we suppose that δ'_k does not contain a reset. In particular, we have $t'_k = \nu'_{k+1} - \nu'_k$ and $t = \nu_{k+1} - \nu_k$, thus $|t'_k - t| = |\nu'_{k+1} - \nu'_k - (\nu_{k+1} - \nu_k)|$. Then, two cases are possible: either $t'_k = \max(0, \nu_k + t - \nu'_k)$ or $t = \max(0, \nu'_k + t'_k - \nu_k)$. So we have three different possibilities:

- if $t'_k + \nu'_k = t + \nu_k$, then $\nu'_{k+1} = \nu_{k+1}$, thus

$$|t'_k - t| = |\nu_k - \nu'_k| = |\nu_k - \nu'_k| - |\nu'_{k+1} - \nu_{k+1}|\,;$$

- if $t = 0$, then $\nu_k = \nu_{k+1} \geq \nu'_{k+1} \geq \nu'_k$, thus

$$|t'_k - t| = \nu'_{k+1} - \nu'_k = (\nu_k - \nu'_k) - (\nu_{k+1} - \nu'_{k+1}) = |\nu_k - \nu'_k| - |\nu'_{k+1} - \nu_{k+1}|\,;$$

- if $t'_k = 0$, then $\nu'_k = \nu'_{k+1} \geq \nu_{k+1} \geq \nu_k$, thus

$$|t'_k - t| = \nu_{k+1} - \nu_k = (\nu'_k - \nu_k) - (\nu'_{k+1} - \nu_{k+1}) = |\nu_k - \nu'_k| - |\nu'_{k+1} - \nu_{k+1}|\,.$$

Otherwise, δ'_k contains a reset, then $\nu'_{k+1} = \nu_{k+1} = 0$. If $t'_k = \nu_k + t - \nu'_k$, we have that $|t'_k - t| = |\nu_k - \nu'_k|$. Otherwise, $t'_k = 0$ and $t \leq \nu'_k - \nu_k$. In all cases, we have proved (4.2). □

In [Bri+22], it is also shown that the mapping $\nu \mapsto \mathsf{dVal}_{\mathcal{G}}(q, \nu)$ is piecewise affine on each region where it is not infinite, that the total number of pieces (and thus of cutpoints in-between two such affine pieces) is pseudo-polynomial (i.e. polynomial in the number of locations and the greatest weight W), and that all cutpoints and the value associated to such a cutpoint are rational numbers. We will only use this result on reset-acyclic WTGs, i.e. that do not contain cyclic paths with a transition with a reset, which we formally cite in Theorem 1.32.

Example 4.5. Notice that the one-clock WTG depicted in Figure 1.7 does not fulfil this hypothesis: indeed, the play $(\ell_0, 0) \xrightarrow{\delta_1, 1/2-\varepsilon} (\ell_1, 1/2 - \varepsilon) \xrightarrow{\delta_2, 1/2+\varepsilon} (\ell_0, 0)$ is a cyclic play that contains a transition with a reset and of weight -2ε negative and arbitrarily close to 0, also doesn't satisfy the property mentioned immediately after Theorem 1.32. ⌟

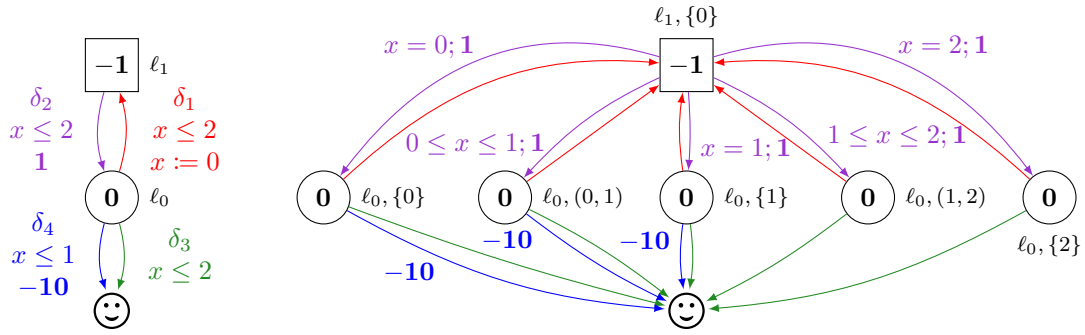


Figure 4.3 – On the left is a one-clock WTG where Max needs memory to play almost optimally. On the right is its closure, where we merged several transitions by removing unnecessary guards.

4.2. How Max can control negative cycles in the closure

One of the main arguments of our proof is that, in the closure of a one-clock WTG, Max can *optimally* play with *memoryless* strategies while *avoiding negative cyclic plays*. As already noticed in [Bri+22], this is not always true in all (one-clock) WTGs: Max may need memory to almost optimally play without the possibility of avoiding some negative cyclic plays.

Example 4.6. In the one-clock WTG depicted on the left in Figure 4.3, we can see that $\mathsf{dVal}(\ell_1, 0) = 0$, but Max, who does not have an optimal strategy, needs memory to play almost optimally and cannot avoid negative cyclic plays. Indeed, an (almost-)optimal strategy for Max always chooses a delay greater than 1: if at some point, a strategy of Max chooses a delay less than or equal to 1, then Min can always choose δ_4, and the value of this strategy is -10. However, Max must choose a delay closer and closer to 1. Otherwise, we suppose that there exists $\beta > 0$ such that all delays chosen by a strategy of Max are greater than $1 + \beta$. Thus, Min has a family of strategies that stay longer and longer in the cycle with a weight at most $-\beta$, and the value of this strategy will tend to $-\infty$. In particular, Max does not have an optimal strategy, and, for all $\varepsilon > 0$, the ε-optimal strategy requires infinite memory to play with delays closer and closer to 1: for instance, after the n-th round in the cycle, Max delays $\frac{\varepsilon}{2^n}$ time units, to sum up, all weights to a value at most $-\varepsilon$. ⌟

Such convergence phenomena needed by Max do not exist in $\overline{\mathcal{G}}$ since all guards are closed (this is not sufficient alone though) and by the regularity of dVal given by Lemma 4.4.

Example 4.7. We consider the closure of the one-clock WTG depicted in Figure 4.3. The almost-optimal strategy (with infinite memory) of Max in $\mathcal{G}$ translates into an optimal memoryless strategy in $\overline{\mathcal{G}}$: in $(\ell_1, \{0\})$, Max can delay 1 time unit and jump into the location $(\ell_0, (1, 2))$. Then, cyclic plays that Min can create have a zero weight and are thus not profitable for either player. ⌟

To generalise this explanation, we start by defining the value of cyclic paths ending with a reset in a given one-clock WTG. Intuitively, the value of this cyclic path is the weight that Min (or Max) can guarantee regardless of the delays chosen by Max (or Min) during this one.

Definition 4.8. Let $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \{x\}, \Delta, \mathsf{wt}, \mathsf{wt}_T \rangle$ be a one-clock WTG. By induction, we define the *(deterministic) value* of a finite path π in $\mathcal{G}$ from an initial valuation ν of the clock and we denote it by $\mathsf{dVal}^{\nu}_{\mathcal{G}}(\pi)$. If π has length 0 (i.e. if $\pi \in L$), we let

$$\mathsf{dVal}^{\nu}_{\mathcal{G}}(\pi) = 0\,.$$

Otherwise, π can be written $\ell_0 \xrightarrow{\delta_0} \pi'$ (with π' starting in location ℓ_1), and we let

$$\mathsf{dVal}^{\nu}_{\mathcal{G}}(\pi) = \begin{cases} \inf_{(\ell_0,\nu) \xrightarrow{\delta_0,t_0} (\ell_1,\nu')} \left[t_0\, \mathsf{wt}(\ell_0) + \mathsf{wt}(\delta_0) + \mathsf{dVal}^{\nu'}_{\mathcal{G}}(\pi') \right] & \text{if } \ell_0 \in L_{\mathsf{Min}} \\ \sup_{(\ell_0,\nu) \xrightarrow{\delta_0,t_0} (\ell_1,\nu')} \left[t_0\, \mathsf{wt}(\ell_0) + \mathsf{wt}(\delta_0) + \mathsf{dVal}^{\nu'}_{\mathcal{G}}(\pi') \right] & \text{if } \ell_0 \in L_{\mathsf{Max}} \end{cases}$$

Then, for a cyclic path π of $\mathcal{G}$ ending by a transition with a reset, we let $\mathsf{dVal}_{\mathcal{G}}(\pi) = \mathsf{dVal}^{0}_{\mathcal{G}}(\pi)$.

Example 4.9. Let $\pi = \ell_0 \xrightarrow{\delta_1} \ell_1 \xrightarrow{\delta_2} \ell_0$ be the cyclic path of the one-clock WTG $\mathcal{G}$ depicted in Figure 1.7. To evaluate the value of π, Min only needs to choose a delay $t_1 \in [0, 1]$ when firing δ_1, while Max has no choice but to play a delay $1 - t_1$ when firing δ_2, generating a finite play ρ of weight $\mathsf{wt}_{\Sigma}(\rho) = 2\,t_1 - 1$. We deduce that $\mathsf{dVal}_{\mathcal{G}}(\pi) = \inf_{t_1 \in [0,1]}(2\,t_1 - 1) = -1$ (when Min chooses $t_1 = 0$). ⌟

A cyclic path with a negative value ensures that Min can always guarantee that a cyclic play is obtained and followed it with a negative weight, even when there are other cyclic plays (that follow it) with a non-negative weight. It is exactly those cycles that are problematic for Max since Min can benefit from them. We now show our key lemma: in the closure of a one-clock WTG, Max can play optimally and avoid cyclic paths of negative value.

Lemma 4.10. *In a closure one-clock WTG $\overline{\mathcal{G}}$,* Max *has a memoryless optimal strategy τ, such that*

1. *all cyclic plays conforming to τ have a non-negative weight;*

2. all cyclic paths ending by a reset conforming to τ have a non-negative value.

Proof. We use[3] the fixpoint characterisation of the value, given in Theorem 2.1-1, to define the memoryless optimal strategy τ for Max. Indeed, the identity $\mathsf{dVal}_{\overline{\mathcal{G}}} = \mathcal{F}(\mathsf{dVal}_{\overline{\mathcal{G}}})$, applied over configurations belonging to Max, suggests a choice of transition and delay to play almost optimally. As $\mathcal{F}$ computes a supremum on the set of possible (transitions and) delays, this does not directly lead to a specific decision: in general, this would give rise to almost-optimal strategies and not an optimal one. This is where we rely on the continuity of $\mathsf{dVal}_{\overline{\mathcal{G}}}$ (via Lemma 4.4) on each closure of region to deduce that this supremum is indeed a maximum. More precisely, for a location $q \in Q_{\mathsf{Max}}$, we can write $\mathcal{F}(\mathsf{Val}_{\overline{\mathcal{G}}})(q,\nu)$ as

$$\max_{\delta\in\Delta} \sup_{t \text{ s.t. } (q,\nu)\xrightarrow{\delta,t}(q',\nu')} \left[\overline{\mathsf{wt}}(\delta) + t\,\overline{\mathsf{wt}}(q) + \mathsf{dVal}_{\overline{\mathcal{G}}}(q',\nu')\right].$$

The guard of transition δ is the closure $\overline{r}$ of a region $r \in \mathsf{Reg}(\{x\},\mathsf{M})$, therefore, t is in a closed interval J of values such that $\nu+t$ falls in $\overline{r}$. Notice that ν' is either 0 if δ contains a reset or is $\nu+t$: in both cases, this is a continuous function of t. Relying on the continuity of $\mathsf{dVal}_{\overline{\mathcal{G}}}$, the mapping $t \in J \mapsto \overline{\mathsf{wt}}(\delta) + t\,\overline{\mathsf{wt}}(q) + \mathsf{dVal}_{\overline{\mathcal{G}}}(q',\nu')$ is thus continuous over a compact set so that its supremum is indeed a maximum.

We thus let the memoryless strategy τ be such that, for all configurations (q,ν), $\tau(q,\nu)$ is chosen arbitrarily in

$$\operatorname*{argmax}_{\delta\in\Delta} \operatorname*{argmax}_{t \text{ s.t. } (q,\nu)\xrightarrow{\delta,t}(q',\nu')} \left[\overline{\mathsf{wt}}(\delta) + t\,\overline{\mathsf{wt}}(q) + \mathsf{dVal}_{\overline{\mathcal{G}}}(q',\nu')\right] \tag{4.3}$$

This mapping τ is then extended into a memoryless strategy, defining it over finite plays by only considering the last configuration of the play. To conclude the proof, we show that τ is an optimal strategy that satisfies the two properties of the lemma.

We first show that τ is an optimal strategy, by proving that $\mathsf{dVal}^{\tau}_{\overline{\mathcal{G}}}(q,\nu) \geq \mathsf{dVal}_{\overline{\mathcal{G}}}(q,\nu)$ for all configurations (q,ν). In particular, we show that for all plays ρ from (q,ν) conforming to τ, we have $\mathbf{P}(\rho) \geq \mathsf{dVal}_{\overline{\mathcal{G}}}(q,\nu)$. We remark that if ρ does not reach Q_t, then $\mathbf{P}(\rho) = +\infty$, and the inequality is satisfied. Now, we suppose that ρ reaches Q_t, and we reason by induction on the length of ρ

[3] Unlike in Chapter 2, we have equipped one-clock WTGs with final weight functions (continuous over each region). Nevertheless, we note that the proof of Theorem 2.1-1 can be extended when we suppose that final weight functions are always finite in region states (details of this version of this proof are given in [MPR23a, Section 8]). Moreover, we also remark that, without loss of generality, we may also suppose that final weights $\mathsf{wt}_T(q,\nu)$ are different from $+\infty$, for all configurations (q,ν) with $q \in Q_t$. To do so, it suffices to forbid the jump into a region r where the final weight function is constant equal to $+\infty$, by modifying the guard on the ingoing transitions.

to show that $\mathbf{P}(\rho) \geq \mathsf{dVal}_{\overline{\mathcal{G}}}(q,\nu)$ for all plays ρ starting in a configuration (q,ν) reaching Q_t. If ρ has length 0, it starts directly in $q \in Q_t$, and $\mathbf{P}(\rho) = \overline{\mathsf{wt}_T}(q,\nu) = \mathsf{dVal}_{\overline{\mathcal{G}}}(q,\nu)$. Otherwise, $\rho = (q,\nu) \xrightarrow{\delta,t} \rho'$, with ρ' starting in a configuration (q',ν'). In particular, by the hypothesis of induction, we have

$$\mathbf{P}(\rho) = \overline{\mathsf{wt}}(\delta) + t\,\overline{\mathsf{wt}}(q) + \mathbf{P}(\rho') \geq \overline{\mathsf{wt}}(\delta) + t\,\overline{\mathsf{wt}}(q) + \mathsf{dVal}_{\overline{\mathcal{G}}}(q',\nu')\,.$$

Now, if $q \in Q_{\mathsf{Min}}$, then we conclude by using that $\mathsf{dVal}_{\overline{\mathcal{G}}}$ is a fixed point of $\mathcal{F}$, i.e.

$$\mathbf{P}(\rho) \geq \inf_{(q,\nu)\xrightarrow{\delta,t}(q',\nu')} \left[\overline{\mathsf{wt}}(\delta) + t\,\overline{\mathsf{wt}}(q) + \mathsf{dVal}_{\overline{\mathcal{G}}}(q',\nu')\right] = \mathsf{dVal}_{\overline{\mathcal{G}}}(q,\nu)\,.$$

Otherwise, we suppose that $q \in Q_{\mathsf{Max}}$ and (δ,t) is defined by τ. Thus, by (4.3) and using again that $\mathsf{dVal}_{\overline{\mathcal{G}}}$ is a fixed point of $\mathcal{F}$, we obtain that

$$\mathbf{P}(\rho) \geq \sup_{(q,\nu)\xrightarrow{\delta,t}(q',\nu')} \left[\overline{\mathsf{wt}}(\delta) + t\,\overline{\mathsf{wt}}(q,r) + \mathsf{dVal}_{\overline{\mathcal{G}}}(q',\nu')\right] = \mathsf{dVal}_{\overline{\mathcal{G}}}(q,\nu)\,.$$

Finally, we conclude the proof by showing that τ satisfies the two properties of the lemma.

We begin by proving the property 1. Let $\rho = (q_1,\nu_1) \xrightarrow{\delta_1,t_1} \cdots (q_k,\nu_k) \xrightarrow{\delta_k,t_k} (q_{k+1},\nu_{k+1}) = (q_1,\nu_1)$ be a cyclic play conforming to τ. We show that $\mathsf{wt}_\Sigma(\rho) \geq 0$ by claiming that for all $i \in \{1,\dots,k\}$,

$$\mathsf{dVal}_{\overline{\mathcal{G}}}(q_i,\nu_i) \leq \overline{\mathsf{wt}}(\delta_i) + t_i\,\overline{\mathsf{wt}}(q_i) + \mathsf{dVal}_{\overline{\mathcal{G}}}(q_{i+1},\nu_{i+1}) \tag{4.4}$$

Indeed by summing this inequality along ρ, we obtain

$$\sum_{i=1}^{k} \mathsf{dVal}_{\overline{\mathcal{G}}}(q_i,\nu_i) \leq \sum_{i=1}^{k} \left(\overline{\mathsf{wt}}(\delta_i) + t_i\,\overline{\mathsf{wt}}(q_i) + \mathsf{dVal}_{\overline{\mathcal{G}}}(q_{i+1},\nu_{i+1})\right)$$

i.e., since ρ is a cyclic play,

$$\mathsf{wt}_\Sigma(\rho) = \sum_{i=1}^{k} \left(\overline{\mathsf{wt}}(\delta_i) + t_i\,\overline{\mathsf{wt}}(q_i)\right) \geq 0\,.$$

To conclude this point, we show (4.4). For $i \in \{1,\dots,k\}$, we distinguish two cases.

- If $q_i \in Q_{\mathsf{Min}}$, then we conclude as $\mathsf{dVal}_{\overline{\mathcal{G}}}$ is a fixed point of $\mathcal{F}$:

$$\begin{aligned}\mathsf{dVal}_{\overline{\mathcal{G}}}(q_i,\nu_i) &= \inf_{(q_i,\nu_i)\xrightarrow{\delta,t}(q',\nu')} \left[\overline{\mathsf{wt}}(\delta) + t\,\overline{\mathsf{wt}}(q_i) + \mathsf{dVal}_{\overline{\mathcal{G}}}(q',\nu')\right] \\ &\leq \overline{\mathsf{wt}}(\delta_i) + t_i\,\overline{\mathsf{wt}}(q_i) + \mathsf{dVal}_{\overline{\mathcal{G}}}(q_{i+1},\nu_{i+1})\,.\end{aligned}$$

- If $q_i \in Q_{\mathsf{Max}}$ then, as $\mathsf{dVal}_{\overline{\mathcal{G}}}$ is a fixed point of $\mathcal{F}$ and by using (4.3), we have

$$\begin{aligned}\mathsf{dVal}_{\overline{\mathcal{G}}}(q_i,\nu_i) &= \sup_{(q_i,\nu_i)\xrightarrow{\delta,t}(q',\nu')} \left[\overline{\mathsf{wt}}(\delta) + t\,\overline{\mathsf{wt}}(q_i) + \mathsf{dVal}_{\overline{\mathcal{G}}}(q',\nu')\right] \\ &\leq \overline{\mathsf{wt}}(\delta_i) + t_i\,\overline{\mathsf{wt}}(q_i) + \mathsf{dVal}_{\overline{\mathcal{G}}}(q_{i+1},\nu_{i+1})\,.\end{aligned}$$

To conclude the proof, we prove property 2. Let $\pi = q_0 \xrightarrow{\delta_0} q_1 \cdots q_k \xrightarrow{\delta_k} q_0$ be a cyclic path conforming to τ such that δ_k contains a reset. By grouping all infimum/supremum together[4] in the previous definition, we can see that $\mathsf{dVal}_{\overline{\mathcal{G}}}(\pi)$ can be rewritten as

$$\inf_{\left(f_i\colon (t_0,\ldots,t_{i-1})\mapsto t_i\right)_{\substack{0\leq i\leq k\\ q_i\in Q_{\mathsf{Min}}}}} \ \sup_{\left(f_i\colon (t_0,\ldots,t_{i-1})\mapsto t_i\right)_{\substack{0\leq i\leq k\\ q_i\in Q_{\mathsf{Max}}}}} \mathsf{wt}_\Sigma(\rho)$$

where ρ is the finite play $(q_0,0) \xrightarrow{\delta_0,t_0=f_0} (q_1,\nu_1) \xrightarrow{\delta_1,t_1=f_1(t_0)} \cdots \xrightarrow{\delta_k,t_k=f_k(t_0,\ldots,t_{k-1})} (q_0,0)$. Notice that the mapping f_i, chosen by the player owning location q_i, describes the delay before taking the transition δ_i as a function of the previously chosen delays. In particular, for all $\varepsilon > 0$, there exists $\left(f_i\colon (t_0,\ldots,t_{i-1})\mapsto t_i\right)_{\substack{0\leq i\leq k\\ q_i\in Q_{\mathsf{Min}}}}$ such that for all $\left(f_i\colon (t_0,\ldots,t_{i-1})\mapsto t_i\right)_{\substack{0\leq i\leq k\\ q_i\in Q_{\mathsf{Max}}}}$,

$$\mathsf{wt}_\Sigma(\rho) \leq \mathsf{dVal}_{\overline{\mathcal{G}}}(\pi) + \varepsilon$$

with ρ the finite play described above. Since π is conforming to τ, a particular choice of delays $\left(f_i\colon (t_0,\ldots,t_{i-1})\mapsto t_i\right)_{\substack{0\leq i\leq k\\ q_i\in Q_{\mathsf{Max}}}}$ is given by τ itself. In this case, the latter finite play ρ is conforming to τ. By the previous property, we know that $\mathsf{wt}_\Sigma(\rho) \geq 0$, therefore

$$\mathsf{dVal}_{\overline{\mathcal{G}}}(\pi) \geq -\varepsilon\,.$$

Since this holds for all $\varepsilon > 0$, we deduce that $\mathsf{dVal}_{\overline{\mathcal{G}}}(\pi) \geq 0$ as expected. □

As a side note, it is tempting to strengthen Lemma 4.10-2 to ensure that all plays

[4] The value of a path is always determined. Indeed, by grouping choices to define strategies, the order between the infimum and the supremum is not significant.

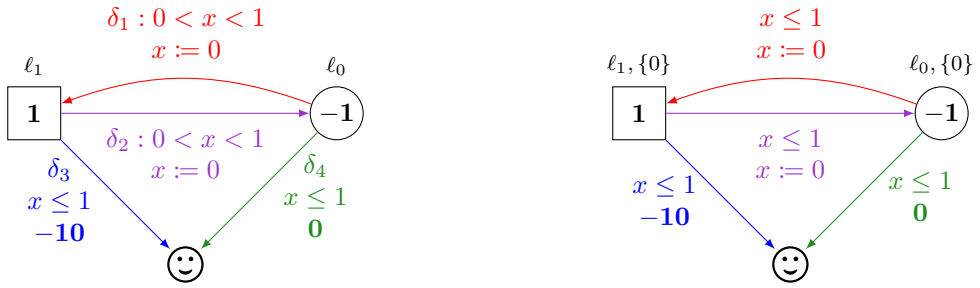


Figure 4.4 – On the left, a WTG such that its closure on the right contains a cyclic path of value 0 and some cyclic paths of negative weight. Moreover, Max uses this cyclic path to play optimally.

following a cyclic path ending with a reset conforming to τ have a non-negative weight. Unfortunately, this does not hold, as shown in the following example.

Example 4.11. We consider the closure of the one-clock WTG depicted in Figure 4.4. Let $\pi = (\ell_0, \{0\}) \xrightarrow{\delta_1} (\ell_1, \{0\}) \xrightarrow{\delta_2} (\ell_0, \{0\})$. It is a cyclic path such that all plays following it have a weight in $[-1, 1]$. To evaluate the value of π in $\overline{\mathcal{G}}$, Min and Max need to choose delays $t_1, t_2 \in [0, 1]$ when firing δ_1 and δ_2. We obtain a set of finite plays ρ parametrised by t_1 and t_2 of weight $\mathsf{wt}_\Sigma(\rho) = -t_1 + t_2$. We deduce that $\mathsf{dVal}_{\overline{\mathcal{G}}}(\pi) = \inf_{t_1} \sup_{t_2} [t_2 - t_1] = 0$ (when Min and Max choose $t_1 = t_2 = 1$). In particular, from the configuration $((\ell_0, \{0\}), 0)$, the cyclic path π is not interesting for Min since he only can guarantee the weight 0. Thus, he must play transition δ_4 after a delay of 1 unit of time to lead to a value of -1. To play optimally, Max must avoid the transition δ_3, i.e. all optimal strategies of Max play in the previous cyclic path π that has a non-negative value but such that certain plays following it have a negative weight. ⌟

Finally, we note that Lemma 4.10 does not allow us to conclude on the decidability of the value problem since we use the unknown value $\mathsf{dVal}_{\overline{\mathcal{G}}}$ to define the optimal strategy.

4.3. A partial unfolding of the closure

To compute $\mathsf{dVal}_{\overline{\mathcal{G}}}$, we now define the partial *unfolding* of the one-clock WTG $\overline{\mathcal{G}}$ by allowing only one occurrence of each cyclic path (from $\overline{\mathcal{G}}$) ending by a reset. In particular, when a transition with a reset is taken for the first time, we go into a new copy of the WTG, from which, if this transition happens to be chosen one more time, we stop the play by jumping into a new target location. The final weight of this target location is determined by the value of the cyclic path (ending

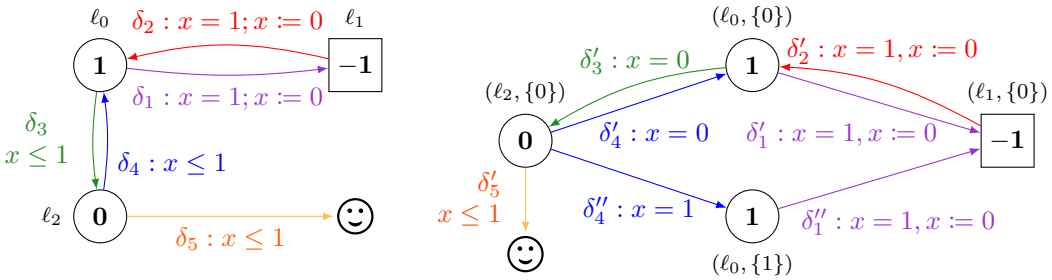


Figure 4.5 – A one-clock WTG (left) and a portion of its closure (right) where δ'_2 belongs to a cyclic path of non-negative value and another cyclic path of negative value.

with a reset) that would have just been closed. If the cyclic path has a negative value, then we go in a leaf $\mathsf{t}_{<0}$ of final weight $-\infty$ since this is a desirable cycle for Min. Otherwise, we go in a leaf $\mathsf{t}_{\geq 0}$ of final weight big enough to be an undesirable behaviour for Min, i.e. $|Q|\, W_{\mathsf{e}} + W_{\mathsf{fin}}$ (for technical reasons that will become clear later, we can not simply put a final weight $+\infty$).

A single transition with a reset can be part of two distinct cyclic paths, one of negative value and the other of non-negative value, as demonstrated in Example 4.12. Thus, knowing the last transition of the cycle is not enough to compute the value of the cyclic path. Instead, we need to record the whole cyclic path: copying the game (as done in the non-negative setting [Bou+06]) is not enough. Our unfolding needs to remember the path followed so far: its locations are thus finite paths of $\overline{\mathcal{G}}$.

Example 4.12. In Figure 4.5, we have depicted a one-clock WTG (left) and a portion of its closure (right), where δ'_2 is contained in a cyclic path of negative value

$$(\ell_0, \{0\}) \xrightarrow{\delta'_3} (\ell_2, \{0\}) \xrightarrow{\delta''_4} (\ell_0, \{1\}) \xrightarrow{\delta''_1} (\ell_1, \{0\}) \xrightarrow{\delta'_2} (\ell_0, \{0\})$$

and another cyclic path of non-negative (zero) value

$$(\ell_0, \{0\}) \xrightarrow{\delta'_1} (\ell_1, \{0\}) \xrightarrow{\delta'_2} (\ell_0, \{0\})\,. \qquad \lrcorner$$

In order to obtain a finite acyclic unfolding, we also need to stop cyclic paths without resets. To do so, we will rely on Theorem 3.9 stating that in reset-acyclic one-clock WTGs, there exists an almost-optimal switching strategy σ, for Min such that all plays conforming to σ reach the target within a known number κ of steps.

Now, between two transitions with a reset, we obtain a reset-acyclic one-clock WTG. As a consequence, we can bound the number of steps between two transitions with a reset by κ. This property allows us to avoid incorporating cycles in

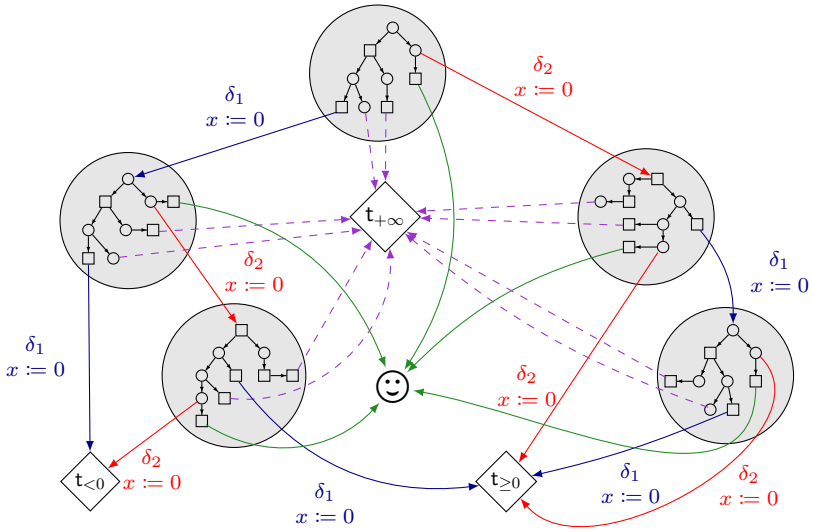


Figure 4.6 – Scheme of the unfolding of a closure of a one-clock WTG.

the unfolding: we cut the unfolding when the play becomes longer than κ since the last seen transition with a reset. In this case, we will jump into a new target location, $\mathsf{t}_{+\infty}$, whose final weight is equal to $+\infty$ since it is an undesirable behaviour for Min.

The scheme of the unfolding is depicted in Figure 4.6 when the closure of a WTG contains two transitions with a reset, δ_1 and δ_2, each belonging to several cycles of different values (negative and non-negative). Inside each grey component, only transitions with no reset are unfolded for at most κ steps by only keeping the path followed so far in the current location. Transitions with a reset induce a change of components: these are in between the components. The second time they are visited, the value of the cycle it closes is computed, and we jump in $\mathsf{t}_{<0}$ or $\mathsf{t}_{\geq 0}$ depending on the sign of the value.

Definition 4.13. The *unfolding* of $\overline{\mathcal{G}}$ from the initial location q_{i} is the (a priori infinite) one-clock WTG $\mathcal{U} = \langle L'_{\mathsf{Min}}, L'_{\mathsf{Max}}, L'_T, L'_u, \{x\}, \Delta', \mathsf{wt}', \mathsf{wt}'_T \rangle$ with $L'_{\mathsf{Min}} \subseteq \mathsf{FPaths}_{\mathsf{Min}}$, $L'_{\mathsf{Max}} \subseteq \mathsf{FPaths}_{\mathsf{Max}}$, $L'_T \subseteq Q_t \cup \{\mathsf{t}_{\geq 0}, \mathsf{t}_{<0}, \mathsf{t}_{+\infty}\}$ such that

- $L' = L'_{\mathsf{Min}} \uplus L'_{\mathsf{Max}} \uplus L'_T$ and Δ' are the smallest sets such that $q_{\mathsf{i}} \in L'$ and for all $\pi \in L'_{\mathsf{Min}} \uplus L'_{\mathsf{Max}}$ and $\delta \in \overline{\Delta}$, if $\textsc{Next}(\pi, \delta) = (\pi', \delta')$ then $\pi' \in L'$ and $\delta' \in \Delta'$ (where $\textsc{Next}$ is defined in Algorithm 1);
- $L'_u = \{\pi \in L' \mid \mathsf{last}(\pi) \in Q_u\}$;
- for all $\pi \notin L'_T$, $\mathsf{wt}'(\pi) = \overline{\mathsf{wt}}(\mathsf{last}(\pi))$, and all $\delta' \in \Delta'$, $\mathsf{wt}'(\delta') = \overline{\mathsf{wt}}(\Delta\mathsf{proj}(\delta'))$;

Algorithm 1 Function NEXT that maps pairs $(\pi, \delta) \in \mathsf{FPaths}_{\overline{\mathcal{G}}} \times \Delta$ to pairs (π', δ') composed of a finite path π' of $\overline{\mathcal{G}}$ (or $\mathsf{t}_{\geq 0}$, or $\mathsf{t}_{<0}$, or $\mathsf{t}_{+\infty}$) and a new transition δ' of the unfolding $\mathcal{U}$.

function NEXT$(\pi, \delta = (q_1, r, Y, q_2))$: ▷ $\mathsf{last}(\pi) = q_1$
  if $q_2 \in Q_t$ **then** $\pi' \coloneqq q_2$
  else if $Y = \{x\}$ **then**
    if $|\pi|_\delta = 0$ **then** $\pi' \coloneqq \pi \xrightarrow{\delta} q_2$
    else $\{$ $\pi \coloneqq \pi_1 \xrightarrow{\delta} \pi_2$
      if $\mathsf{dVal}_{\overline{\mathcal{G}}}(\pi_2 \xrightarrow{\delta} q_2) \geq 0$ **then** $\pi' \coloneqq \mathsf{t}_{\geq 0}$ **else** $\pi' \coloneqq \mathsf{t}_{<0}$ $\}$
  else $\{$ $\pi \coloneqq \pi_1 \cdot \pi_2$ where π_2 contains no reset and $|\pi_2|$ is maximal
    if $|\pi_2| = \kappa$ **then** $\pi' \coloneqq \mathsf{t}_{+\infty}$ **else** $\pi' \coloneqq \pi \xrightarrow{\delta} q_2$ $\}$
  $\delta' \coloneqq (\pi, r, Y, \pi')$ ▷ $\Delta\mathsf{proj}(\delta') \coloneqq \delta$
  return (π', δ')

- for all $\pi \in L'_T$, for all ν,

$$\mathsf{wt}'_T(\pi, \nu) = \overline{\mathsf{wt}_T}(\pi, \nu) \quad \text{if } \pi \in Q_t \qquad \mathsf{wt}'_T(\mathsf{t}_{\geq 0}, \nu) = |Q|\, W_{\mathsf{e}} + W_{\mathsf{fin}}$$
$$\mathsf{wt}'_T(\mathsf{t}_{<0}, \nu) = -\infty \qquad \mathsf{wt}'_T(\mathsf{t}_{+\infty}, \nu) = +\infty\,.$$

As expected, the definition of NEXT guarantees that a "new" target location is reached when the length between two resets is too long or when a transition with a reset appears twice. Moreover, the length of the path in a location that is not a target, given by the application of NEXT, strictly increases. This allows us to show that $\mathcal{U}$ is a finite and acyclic WTG.

Lemma 4.14. *The one-clock WTG $\mathcal{U}$ is acyclic and has a finite set of locations of cardinality at most exponential in $|L|$ and W.*

Proof. We start by proving that $\mathcal{U}$ is an acyclic WTG. The function NEXT never removes a transition from a path $\pi \in L'_{\mathsf{Min}} \cup L'_{\mathsf{Max}}$ that is given as input. In particular, the function NEXT produces a transition from π to π' that is an extension of π (i.e. $|\pi'| > |\pi|$) or a target location. Thus, all paths in $\mathcal{U}$ are acyclic, i.e. $\mathcal{U}$ is acyclic.

Now, we prove that $\mathcal{U}$ is a finite WTG by proving that for all locations $\pi \in L'_{\mathsf{Min}} \cup L'_{\mathsf{Max}}$, the length of the path is upper-bounded by $(|\Delta_R| + 1)\kappa$, where we let Δ_R be the subset of transitions of $\overline{\mathcal{G}}$ with a reset. Locations of $\mathcal{U}$ are built by successive applications of NEXT. We show by induction (on the number of such applications) that every location $\pi \in L'_{\mathsf{Min}} \cup L'_{\mathsf{Max}}$ can be decomposed as follows: $\pi = \pi'_0 \xrightarrow{\delta_0} \pi'_1 \xrightarrow{\delta_1} \cdots \xrightarrow{\delta_k} \pi'_k$ where transitions δ_i belong to Δ_R and are pairwise

distinct, and where π'_i belong to $\mathsf{FPaths}_{\overline{\mathcal{G}}}$ and have length at most κ. As a direct consequence of this property, we have $k \leq |\Delta_R|$, and we easily deduce the expected bound on $|\pi|$, and thus the desired bound on the number of locations of $\mathcal{U}$ by the bound on κ of Theorem 3.9.

We now proceed to the induction on the number of applications of Next. As a base case, we have $\pi = q_{\mathsf{i}}$, and the property trivially holds. We assume now that the property holds for π, with a decomposition $\pi = \pi'_0 \xrightarrow{\delta_0} \pi'_1 \xrightarrow{\delta_1} \cdots \xrightarrow{\delta_k} \pi'_k$. We fix some $\delta = (q_1, r, Y, q_2) \in \overline{\Delta}$ with $\mathsf{last}(\pi) = q_1$, and we consider $\textsc{Next}(\pi, \delta) = (\pi', \delta')$ with $\pi' \in L'_{\mathsf{Min}} \cup L'_{\mathsf{Max}}$. We distinguish cases according to the definition of Next. We observe that $\pi' \in L'_{\mathsf{Min}} \cup L'_{\mathsf{Max}}$ excludes the case when Next sets π' in lines 2, 6, or 8 (when $|\pi'_k| = \kappa$). The following cases may occur:

- if Next sets π' in line 4, then we have $\delta \in \overline{\Delta}_R$, $|\pi|_\delta = 0$, and $\pi' = \pi \xrightarrow{\delta} q_2$. Hence, a correct decomposition of π' is obtained by adding δ and an empty path π'_{k+1} to the ones of π;
- if Next sets π' in line 8, while $\pi' \neq \mathsf{t}_{+\infty}$ (by hypothesis), then $|\pi'_k| < \kappa$. A correct decomposition of π' is then obtained from the one of π by replacing π'_k with $\pi'_k \xrightarrow{\delta} q_2$. □

4.3.1. How Max controls negative "cycles" in the unfolding

In this section, we show another good property of $\mathcal{U}$ by mimicking the one of Lemma 4.10: Max has an optimal memoryless strategy in $\overline{\mathcal{G}}$ avoiding cyclic plays with a negative weight. In the unfolding $\mathcal{U}$, there are no cyclic plays, but we will be able to obtain a similar result: Max can play optimally with a memoryless strategy while making sure that in-between two occurrences of the same transition with a reset (which would result in a cyclic play in the original game $\overline{\mathcal{G}}$), the play has a non-negative weight. More formally, we want to obtain the following lemma, which we show in the rest of the section:

Lemma 4.15. *In the WTG $\mathcal{U}$,* Max *has a memoryless optimal strategy τ such that if a finite play $\rho = \rho_1 \xrightarrow{\delta'_1, t_1} \rho_2 \xrightarrow{\delta'_2, t_2} (\mathsf{t}_{\geq 0}, 0)$ is conforming to τ with $\Delta\mathsf{proj}(\delta'_1) = \Delta\mathsf{proj}(\delta'_2) \in \overline{\Delta}_R$ (i.e. the same transition with a reset in the original WTG $\overline{\mathcal{G}}$), then $\mathsf{wt}_\Sigma\big(\rho_2 \xrightarrow{\delta'_2, t_2} (\mathsf{t}_{\geq 0}, 0)\big) \geq 0$.*

Remark. The fact that $\mathcal{U}$ is acyclic is crucial in this result: we can not guarantee that the value of the path ending in $\mathsf{t}_{\geq 0}$ is non-negative if we would have defined $\mathcal{U}$ with grey components (in Figure 4.6) containing cyclic paths without a reset. Indeed, the values of cyclic paths are not preserved by concatenation. For instance, in the one-clock WTG $\mathcal{G}$ depicted on the left of Figure 4.5, we can see

that $\mathsf{dVal}_{\mathcal{G}}(\ell_0 \xrightarrow{\delta_1} \ell_1 \xrightarrow{\delta_2} \ell_0) = 0$ (Min and Max must delay 1 in each location), and $\mathsf{dVal}_{\mathcal{G}}(\ell_0 \xrightarrow{\delta_3} \ell_2 \xrightarrow{\delta_4} \ell_0) = 0$. However, when we concatenate these two cyclic paths, we obtain the cycle $\ell_0 \xrightarrow{\delta_3} \ell_2 \xrightarrow{\delta_4} \ell_0 \xrightarrow{\delta_1} \ell_1 \xrightarrow{\delta_2} \ell_0$ of value -1. ◁

The proof of Lemma 4.15 essentially consists in applying Lemma 4.10 in $\mathcal{U}$ to define a memoryless optimal strategy with the desired property. However, Lemma 4.10 holds only in the closure of WTGs (Example 4.6 gives a counter-example when the WTG is not a closure), i.e. a priori, the result holds only on the closure $\overline{\mathcal{U}}$ of $\mathcal{U}$. Nevertheless, we notice that the guards of all transitions of $\mathcal{U}$ come from $\overline{\mathcal{G}}$ and the regions of $\mathcal{U}$ are thus the ones of $\overline{\mathcal{G}}$. Therefore, apart from target locations, only locations (π, r) in $\overline{\mathcal{U}}$ with π ending in a location of the form (q, r) are reachable. Thus $\mathcal{U}$ and $\overline{\mathcal{U}}$ are the same WTG, and the result of Lemma 4.10 transfers to $\mathcal{U}$ as well.

The second argument of the proof is checking that $\mathcal{U}$ preserves the value of paths, i.e. the value in $\mathcal{U}$ of a finite path $\pi_{\mathcal{U}}$, $\mathsf{dVal}^{\nu}_{\mathcal{U}}(\pi_{\mathcal{U}})$, is equal to the value in $\overline{\mathcal{G}}$ of its projection given by $\Delta\mathsf{proj}$. In particular, we define a new projection function $\Pi\mathsf{proj}$ as an extension over finite paths of $\Delta\mathsf{proj}$ such that for all finite paths in $\mathcal{U}$ with at least one transition, $\pi_{\mathcal{U}} = \pi_1 \xrightarrow{\delta'} \pi'_{\mathcal{U}}$ with $\pi_1 \in L' \setminus L'_T$, we let $\Pi\mathsf{proj}(\pi_{\mathcal{U}})$ be equal to

$$\begin{cases} \mathsf{last}(\pi_1) \xrightarrow{\Delta\mathsf{proj}(\delta')} q_2 & \text{if } \pi'_{\mathcal{U}} \in L' \text{ and } \Delta\mathsf{proj}(\delta') = (\mathsf{last}(\pi_1), r, Y, q_2); \\ \mathsf{last}(\pi_1) \xrightarrow{\Delta\mathsf{proj}(\delta')} \Pi\mathsf{proj}(\pi'_{\mathcal{U}}) & \text{otherwise .} \end{cases}$$

We note that, $\Pi\mathsf{proj}(\pi_{\mathcal{U}})$ is always a finite path in $\overline{\mathcal{G}}$ with the same length of $\pi_{\mathcal{U}}$ and it satisfies the following properties:

Lemma 4.16. *Let $\pi_{\mathcal{U}} \in \mathsf{FPaths}_{\mathcal{U}}$ be a path with at least one transition, then*

1. *for all valuations ν, $\mathsf{dVal}^{\nu}_{\mathcal{U}}(\pi_{\mathcal{U}}) = \mathsf{dVal}^{\nu}_{\overline{\mathcal{G}}}(\Pi\mathsf{proj}(\pi_{\mathcal{U}}))$;*
2. *if $\mathsf{last}(\pi_{\mathcal{U}}) \notin L'_T$, then $\Pi\mathsf{proj}(\pi_{\mathcal{U}})$ is a suffix of $\mathsf{last}(\pi_{\mathcal{U}})$.*

Proof. We independently prove the two properties given by the lemma.

1. We prove this property when the first location of $\pi_{\mathcal{U}}$ belongs to Min. The case where it belongs to Max is analogous when we replace the infimum with a supremum. We reason by induction on the length of $\pi_{\mathcal{U}}$. First, we suppose that $\pi_{\mathcal{U}}$ contains exactly one transition, i.e. $\pi_{\mathcal{U}} = \pi_1 \xrightarrow{\delta'} \pi_2$ and

$$\mathsf{dVal}^{\nu}_{\mathcal{U}}(\pi_{\mathcal{U}}) = \inf_t \left[t\, \mathsf{wt}'(\pi_1) + \mathsf{wt}'(\delta') + \mathsf{dVal}^{\nu'}_{\mathcal{U}}(\pi_2) \right]$$

where $\nu' = \nu + t$ if δ' does not contain a reset, or $\nu' = 0$ otherwise. Since the value of an empty path is null, we have $\mathsf{dVal}^{\nu'}_{\mathcal{U}}(\pi_2) = 0 = \mathsf{dVal}^{\nu'}_{\overline{\mathcal{G}}}(q_2)$ where q_2 is given by $\Delta\mathsf{proj}(\delta')$. Moreover, as $\mathcal{U}$ preserves the weight of transitions and locations, we obtain that

$$\mathsf{dVal}^{\nu}_{\mathcal{U}}(\pi_{\mathcal{U}}) = \inf_t \left[t\,\mathsf{wt}(\mathsf{last}(\pi_1)) + \mathsf{wt}(\Delta\mathsf{proj}(\delta')) + \mathsf{dVal}^{\nu'}_{\overline{\mathcal{G}}}(q_2) \right].$$

Now, by definition of $\Pi\mathsf{proj}$, we remark that $\Pi\mathsf{proj}(\pi_{\mathcal{U}}) = \mathsf{last}(\pi_1) \xrightarrow{\Delta\mathsf{proj}(\delta')} q_2$. Thus, since $\mathcal{U}$ preserves transitions with a reset, we deduce that $\mathsf{dVal}^{\nu}_{\mathcal{U}}(\pi_{\mathcal{U}}) = \mathsf{dVal}^{\nu}_{\overline{\mathcal{G}}}(\Pi\mathsf{proj}(\pi_{\mathcal{U}}))$.

Now, we suppose that $\pi_{\mathcal{U}} = \pi_1 \xrightarrow{\delta'} \pi'_{\mathcal{U}}$ with $\pi'_{\mathcal{U}}$ a path in $\mathcal{U}$. Since $\mathcal{U}$ preserves the weight of transitions and locations, we have

$$\begin{aligned}\mathsf{dVal}^{\nu}_{\mathcal{U}}(\pi_{\mathcal{U}}) &= \inf_t \left[t\,\mathsf{wt}'(\pi_1) + \mathsf{wt}'(\delta') + \mathsf{dVal}^{\nu'}_{\mathcal{U}}(\pi'_{\mathcal{U}}) \right] \\ &= \inf_t \left[t\,\mathsf{wt}(\mathsf{last}(\pi_1)) + \mathsf{wt}(\Delta\mathsf{proj}(\delta')) + \mathsf{dVal}^{\nu'}_{\mathcal{U}}(\pi'_{\mathcal{U}}) \right]\end{aligned}$$

where $\nu' = \nu + t$ if δ' does not contain a reset, or $\nu' = 0$ otherwise. Now, the induction hypothesis applied to $\pi'_{\mathcal{U}}$ implies that

$$\mathsf{dVal}^{\nu}_{\mathcal{U}}(\pi_{\mathcal{U}}) = \inf_t \left[t\,\mathsf{wt}(\mathsf{last}(\pi_1)) + \mathsf{wt}(\Delta\mathsf{proj}(\delta')) + \mathsf{dVal}^{\nu'}_{\overline{\mathcal{G}}}(\Pi\mathsf{proj}(\pi'_{\mathcal{U}})) \right].$$

We conclude that $\mathsf{dVal}^{\nu}_{\mathcal{U}}(\pi_{\mathcal{U}}) = \mathsf{dVal}^{\nu}_{\overline{\mathcal{G}}}(\Pi\mathsf{proj}(\pi_{\mathcal{U}}))$, since $\mathcal{U}$ preserves transitions with a reset and $\Pi\mathsf{proj}(\pi_{\mathcal{U}}) = \mathsf{last}(\pi_1) \xrightarrow{\Delta\mathsf{proj}(\delta')} \Pi\mathsf{proj}(\pi'_{\mathcal{U}})$.

2. We reason by induction on the length of $\pi_{\mathcal{U}}$. First, we suppose that $\pi_{\mathcal{U}}$ contains only one transition, i.e. $\pi_{\mathcal{U}} = \pi_1 \xrightarrow{\delta'} \pi_2$ with $\pi_2 \in L'_{\mathsf{Min}} \cup L'_{\mathsf{Max}}$, and $\Pi\mathsf{proj}(\pi_{\mathcal{U}}) = \mathsf{last}(\pi_1) \xrightarrow{\Delta\mathsf{proj}(\delta')} q_2$ where q_2 is given by $\Delta\mathsf{proj}(\delta')$. By definition of Next, since $\pi_2 \notin L'_T$, we note that $\pi_2 = \pi_1 \xrightarrow{\Delta\mathsf{proj}(\delta')} q_2$. We conclude that $\Pi\mathsf{proj}(\pi_{\mathcal{U}})$ is a suffix of $\mathsf{last}(\pi_{\mathcal{U}}) = \pi_2$, since $\mathsf{last}(\pi_1)$ is a suffix of π_1.

   Otherwise, we suppose that $\pi_{\mathcal{U}} = \pi_1 \xrightarrow{\delta'} \pi'_{\mathcal{U}}$ with $\mathsf{last}(\pi'_{\mathcal{U}}) \notin L'_T$, and

$$\Pi\mathsf{proj}(\pi_{\mathcal{U}}) = \mathsf{last}(\pi_1) \xrightarrow{\Delta\mathsf{proj}(\delta')} \Pi\mathsf{proj}(\pi'_{\mathcal{U}}).$$

   By induction hypothesis, $\Pi\mathsf{proj}(\pi'_{\mathcal{U}})$ is a suffix of $\mathsf{last}(\pi'_{\mathcal{U}})$, i.e. there exists a finite path π of $\overline{\mathcal{G}}$ such that $\mathsf{last}(\pi'_{\mathcal{U}}) = \pi \cdot \Pi\mathsf{proj}(\pi'_{\mathcal{U}})$. Now, we remark that $\pi = \pi_1$, since each application of Next (that does not reach a target location) exactly adds one transition in the path of the next location: δ' is a

transition between π_1 and π_2 where $\pi_2 = \pi_1 \xrightarrow{\Delta\mathsf{proj}(\delta')} q_2$ is the first location of $\pi'_{\mathcal{U}}$. We conclude the proof since $\mathsf{last}(\pi_1)$ is a suffix of π_1. □

Finally, we have the tools to finish the proof of Lemma 4.15. As explained before, we apply the result of Lemma 4.10 in $\mathcal{U}$ (since the closure of $\mathcal{U}$ describes the same WTG as $\mathcal{U}$) to obtain a memoryless optimal strategy τ for Max. It remains to show that if $\rho = \rho_1 \xrightarrow{\delta'_1,t} \rho_2 \xrightarrow{\delta'_2,t_2} (\mathsf{t}_{\geq 0}, 0)$ is conforming to τ with $\Delta\mathsf{proj}(\delta'_1) = \Delta\mathsf{proj}(\delta'_2)$ containing a reset, then $\mathsf{wt}_\Sigma(\rho_2 \xrightarrow{\delta'_2,t_2} (\mathsf{t}_{\geq 0}, 0)) \geq 0$. Let $\pi_{\mathcal{U}}$ be the path of $\mathcal{U}$ followed by $\rho_2 \xrightarrow{\delta'_2,t_2} (\mathsf{t}_{\geq 0}, 0)$. We start by claiming that

$$\mathbf{P}(\rho_2 \xrightarrow{\delta'_2,t_2} (\mathsf{t}_{\geq 0}, 0)) \geq \mathsf{dVal}_{\mathcal{U}}(\pi_{\mathcal{U}}) + \mathsf{wt}'_T(\mathsf{t}_{\geq 0}, 0) \tag{4.5}$$

where this is the place where the fact that $\mathsf{wt}'_T(\mathsf{t}_{\geq 0}, 0)$ is not equal to $+\infty$ is crucial.

Equation (4.5) and Lemma 4.16-1 (with $\nu = 0$) allow us to conclude as follows. First, we know that

$$\mathsf{wt}_\Sigma(\rho_2 \xrightarrow{\delta'_2,t_2} (\mathsf{t}_{\geq 0}, 0)) = \mathbf{P}(\rho_2 \xrightarrow{\delta'_2,t_2} (\mathsf{t}_{\geq 0}, 0)) - \mathsf{wt}'_T(\mathsf{t}_{\geq 0}, 0) \geq \mathsf{dVal}_{\mathcal{U}}(\pi_{\mathcal{U}}) \geq \mathsf{dVal}_{\overline{\mathcal{G}}}(\pi)$$

where $\pi = \Pi\mathsf{proj}(\pi_{\mathcal{U}})$. Then, since $\pi_{\mathcal{U}} = \pi'_{\mathcal{U}} \xrightarrow{\delta'_2} \mathsf{t}_{\geq 0}$ where ρ_2 follows $\pi'_{\mathcal{U}}$, we deduce that π' is a suffix of $\pi'_{\mathcal{U}}$ where $\pi = \pi' \xrightarrow{\delta} q$ (by Lemma 4.16-2 applied to $\pi'_{\mathcal{U}}$). In particular, the definition of NEXT on π' and δ (as $\mathsf{t}_{\geq 0}$ is reached) guarantees that $\mathsf{dVal}_{\overline{\mathcal{G}}}(\pi) \geq 0$.

To conclude the proof, we need to show (4.5). We reason by induction on suffixes ρ' of $\rho_2 \xrightarrow{\delta'_2,t_2} (\mathsf{t}_{\geq 0}, 0)$ showing that

$$\mathbf{P}(\rho') \geq \mathsf{dVal}^{\nu'}_{\mathcal{U}}(\pi'_{\mathcal{U}}) + \mathsf{wt}'_T(\mathsf{t}_{\geq 0}, 0)$$

where $\pi'_{\mathcal{U}}$ is the path followed by ρ', and ν' is the first valuation of ρ'. For the suffix $\rho' = (\mathsf{t}_{\geq 0}, 0)$, then

$$\mathbf{P}(\rho') = \mathsf{wt}'_T(\mathsf{t}_{\geq 0}, 0) = \mathsf{dVal}^0_{\mathcal{U}}(\mathsf{t}_{\geq 0}) + \mathsf{wt}'_T(\mathsf{t}_{\geq 0}, 0)\,.$$

Otherwise, we suppose that $\rho' = (\pi, \nu') \xrightarrow{\delta',t} \rho''$. In particular, we fix $\nu'' = \nu' + t$ the first valuation of ρ'' ($\nu'' \neq 0$ since ρ_2 does not contain a transition with a reset) and $\pi'_{\mathcal{U}} = \pi \xrightarrow{\delta'} \pi''_{\mathcal{U}}$ with ρ'' follows $\pi''_{\mathcal{U}}$. Moreover, we deduce that,

$$\begin{aligned}\mathbf{P}(\rho') &= t\,\mathsf{wt}'(\pi) + \mathsf{wt}'(\delta') + \mathbf{P}(\rho'') \\ &\geq t\,\mathsf{wt}'(\pi) + \mathsf{wt}'(\delta') + \mathsf{dVal}^{\nu'+t}_{\mathcal{U}}(\pi''_{\mathcal{U}}) + \mathsf{wt}'_T(\mathsf{t}_{\geq 0}, 0)\end{aligned}$$

by induction hypothesis. To conclude the induction case, we distinguish two cases.

- If $\pi \in L'_{\mathsf{Min}}$, then

$$\begin{aligned}\mathbf{P}(\rho') &\geq \inf_{t \text{ s.t. } (\pi,\nu') \xrightarrow{\delta',t} (\pi',\nu'+t)} \left[t\,\mathsf{wt}'(\pi) + \mathsf{wt}'(\delta') + \mathsf{dVal}_{\mathcal{U}}^{\nu'+t}(\pi''_{\mathcal{U}}) + \mathsf{wt}'_T(\mathsf{t}_{\geq 0}, 0) \right] \\ &= \inf_{t \text{ s.t. } (\pi,\nu') \xrightarrow{\delta',t} (\pi',\nu'+t)} \left[t\,\mathsf{wt}'(\pi) + \mathsf{wt}'(\delta') + \mathsf{dVal}_{\mathcal{U}}^{\nu'+t}(\pi''_{\mathcal{U}}) \right] + \mathsf{wt}'_T(\mathsf{t}_{\geq 0}, 0) \\ &= \mathsf{dVal}_{\mathcal{U}}^{\nu'}(\pi'_{\mathcal{U}}) + \mathsf{wt}'_T(\mathsf{t}_{\geq 0}, 0)\,.\end{aligned}$$

- If $\pi \in L'_{\mathsf{Max}}$, since τ chooses δ', we can deduce that

$$\begin{aligned}\mathbf{P}(\rho') &\geq \sup_{t \text{ s.t. } (\pi,\nu') \xrightarrow{\delta',t} (\pi',\nu'+t)} \left[t\,\mathsf{wt}'(\pi) + \mathsf{wt}'(\delta') + \mathsf{dVal}_{\mathcal{U}}^{\nu'+t}(\pi''_{\mathcal{U}}) + \mathsf{wt}'_T(\mathsf{t}_{\geq 0}, 0) \right] \\ &= \sup_{t \text{ s.t. } (\pi,\nu') \xrightarrow{\delta',t} (\pi',\nu'+t)} \left[t\,\mathsf{wt}'(\pi) + \mathsf{wt}'(\delta') + \mathsf{dVal}_{\mathcal{U}}^{\nu'+t}(\pi''_{\mathcal{U}}) \right] + \mathsf{wt}'_T(\mathsf{t}_{\geq 0}, 0) \\ &= \mathsf{dVal}_{\mathcal{U}}^{\nu'}(\pi'_{\mathcal{U}}) + \mathsf{wt}'_T(\mathsf{t}_{\geq 0}, 0)\,.\end{aligned}$$

Since it is obtained after transition δ'_1 that resets the clock, the first valuation of $\rho_2 \xrightarrow{\delta'_2,t_2} (\mathsf{t}_{\geq 0}, 0)$ is 0. Thus, by induction, we obtain (4.5) as expected.

4.3.2. Value of the unfolding

The most difficult part of the proof of Theorem 4.1 is to show that the unfolding preserves the value from $\overline{\mathcal{G}}$. Remember that we have fixed an initial location $q_{\mathsf{i}} = (\ell_{\mathsf{i}}, r_{\mathsf{i}})$ to build $\mathcal{U}$.

Proposition 4.17. *For all valuations $\nu \in r_{\mathsf{i}}$, $\mathsf{dVal}_{\overline{\mathcal{G}}}(q_{\mathsf{i}}, \nu) = \mathsf{dVal}_{\mathcal{U}}(q_{\mathsf{i}}, \nu)$.*

We prove Proposition 4.17 in this section, splitting the proof into two inequalities.

First inequality We prove first that $\mathsf{dVal}_{\overline{\mathcal{G}}}(q_{\mathsf{i}}, \nu) \leq \mathsf{dVal}_{\mathcal{U}}(q_{\mathsf{i}}, \nu)$, which can be rewritten as

$$\mathsf{dVal}_{\overline{\mathcal{G}}}(q_{\mathsf{i}}, \nu) \leq \sup_{\tau_{\mathcal{U}}} \mathsf{dVal}_{\mathcal{U}}^{\tau_{\mathcal{U}}}(q_{\mathsf{i}}, \nu)\,.$$

We must thus show that Max can guarantee to always do at least as good in $\mathcal{U}$ as in $\overline{\mathcal{G}}$. We thus fix an optimal strategy $\tau_{\overline{\mathcal{G}}}$ in $\overline{\mathcal{G}}$ obtained by Lemma 4.10: in particular, $\mathsf{dVal}_{\overline{\mathcal{G}}}(q_{\mathsf{i}}, \nu) = \mathsf{dVal}_{\overline{\mathcal{G}}}^{\tau_{\overline{\mathcal{G}}}}(q_{\mathsf{i}}, \nu)$. We show the existence of a strategy $\tau_{\mathcal{U}}$ in $\mathcal{U}$ such that $\mathsf{dVal}_{\overline{\mathcal{G}}}^{\tau_{\overline{\mathcal{G}}}}(q_{\mathsf{i}}, \nu) \leq \mathsf{dVal}_{\mathcal{U}}^{\tau_{\mathcal{U}}}(q_{\mathsf{i}}, \nu)$, i.e. for all plays ρ conforming to $\tau_{\mathcal{U}}$, there

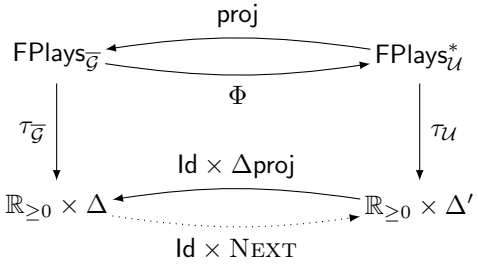


Figure 4.7 – Scheme showing the links between the different objects defined for the proof of Proposition 4.17 where $\mathsf{FPlays}^*_{\mathcal{U}}$ is the set of finite plays of $\mathcal{U}$ avoiding target locations $\mathsf{t}_{\geq 0}$ and $\mathsf{t}_{<0}$.

exists a play conforming to $\tau_{\overline{\mathcal{G}}}$ with a weight at most the weight of ρ. As it is depicted in Figure 4.7, the strategy $\tau_{\mathcal{U}}$ is defined via a *projection* of plays of $\mathcal{U}$ in $\overline{\mathcal{G}}$: we use the mapping NEXT to send back transitions of Δ to Δ'.

More formally, the projection operator proj projects finite plays of $\mathcal{U}$ starting in q_{i} (since these are the only ones we need to take care of) to finite plays of $\overline{\mathcal{G}}$. For this reason, from now on, $\mathsf{FPlays}_{\mathcal{U}}$ and $\mathsf{FPlays}_{\overline{\mathcal{G}}}$ denote the subsets of plays that start in location q_{i}. Moreover, we limit ourselves to projecting plays of $\mathcal{U}$ that do not reach the targets $\mathsf{t}_{<0}$ and $\mathsf{t}_{\geq 0}$ since, otherwise, there is no canonical projection in $\overline{\mathcal{G}}$. We thus let $\mathsf{FPlays}^*_{\mathcal{U}}$ be all such finite plays of $\mathsf{FPlays}_{\mathcal{U}}$ that do not end in $\mathsf{t}_{<0}$ or $\mathsf{t}_{\geq 0}$. The projection function $\mathsf{proj}\colon \mathsf{FPlays}^*_{\mathcal{U}} \to \mathsf{FPlays}_{\overline{\mathcal{G}}}$ is defined inductively on finite plays $\rho \in \mathsf{FPlays}^*_{\mathcal{U}}$ by letting $\mathsf{proj}(\rho)$ be

$$\begin{cases} (q_{\mathsf{i}}, \nu) & \text{if } \rho = (q_{\mathsf{i}}, \nu) \in L'; \\ \mathsf{proj}(\rho') \xrightarrow{\Delta\mathsf{proj}(\delta'),t} (\mathsf{last}(\pi), \nu) & \text{if } \rho = \rho' \xrightarrow{\delta',t} (\pi, \nu); \\ \mathsf{proj}(\rho') \xrightarrow{\Delta\mathsf{proj}(\delta'),t} (q', \nu) & \text{if } \rho = \rho' \xrightarrow{\delta',t} (\mathsf{t}_{+\infty}, \nu) \text{ and } \Delta\mathsf{proj}(\delta') = (q, r, Y, q'), \end{cases}$$

It fulfils the following properties:

Lemma 4.18. *For all plays $\rho \in \mathsf{FPlays}^*_{\mathcal{U}}$,*

1. *if $\mathsf{last}(\rho) = (\pi, \nu)$ with $\pi \neq \mathsf{t}_{+\infty}$, then $\mathsf{last}(\mathsf{proj}(\rho)) = (\mathsf{last}(\pi), \nu)$;*
2. *$\mathsf{wt}_\Sigma(\rho) = \mathsf{wt}_\Sigma(\mathsf{proj}(\rho))$;*
3. *if $\mathsf{last}(\rho) = (\pi, \nu)$ with $\pi \notin Q_t$, then $\mathsf{proj}(\rho)$ follows π.*

Proof. 1. Since $\pi \neq \mathsf{t}_{+\infty}$, this is direct from a case analysis on the definition of proj.

2. We reason by induction on the length of $\rho \in \mathsf{FPlays}^*_{\mathcal{U}}$. First, we suppose that $\rho = (q_{\mathsf{i}}, \nu)$, then we have $\mathsf{proj}(\rho) = \rho$ and $\mathsf{wt}_\Sigma(\rho) = 0 = \mathsf{wt}_\Sigma(\mathsf{proj}(\rho))$. Now,

we suppose that $\rho = \rho' \xrightarrow{\delta',t} (\pi,\nu)$, with $\rho' \in \mathsf{FPlays}^*_{\mathcal{U}}$ ending in location π' such that $\pi' \notin L'_T$. Then,

$$\begin{aligned}\mathsf{wt}_\Sigma(\rho) &= \mathsf{wt}_\Sigma(\rho') + t\,\mathsf{wt}'(\pi') + \mathsf{wt}'(\delta') \\ &= \mathsf{wt}_\Sigma(\rho') + t\,\overline{\mathsf{wt}}(\mathsf{last}(\pi')) + \overline{\mathsf{wt}}(\Delta\mathsf{proj}(\delta'))\end{aligned}$$

since $\mathcal{U}$ preserves the weights of $\overline{\mathcal{G}}$, i.e. $\mathsf{wt}'(\pi') = \overline{\mathsf{wt}}(\mathsf{last}(\pi'))$, and $\mathsf{wt}'(\delta') = \overline{\mathsf{wt}}(\Delta\mathsf{proj}(\delta'))$. Moreover, the induction hypothesis applied to ρ' implies that

$$\begin{aligned}\mathsf{wt}_\Sigma(\rho) &= \mathsf{wt}_\Sigma(\mathsf{proj}(\rho')) + t\,\overline{\mathsf{wt}}(\mathsf{last}(\pi')) + \overline{\mathsf{wt}}(\Delta\mathsf{proj}(\delta')) \\ &= \mathsf{wt}_\Sigma(\mathsf{proj}(\rho')) + t\,\overline{\mathsf{wt}}(\mathsf{last}(\mathsf{proj}(\rho'))) + \overline{\mathsf{wt}}(\Delta\mathsf{proj}(\delta'))\end{aligned}$$

since $\mathsf{last}(\pi') = \mathsf{last}(\mathsf{proj}(\rho'))$ by the first property (as $\pi' \neq \mathsf{t}_{+\infty}$). Finally, by the definition of $\mathsf{proj}(\rho)$, we conclude that $\mathsf{wt}_\Sigma(\rho) = \mathsf{wt}_\Sigma(\mathsf{proj}(\rho))$.

3. We reason by induction on the length of $\rho \in \mathsf{FPlays}^*_{\mathcal{U}}$ that does not reach a target location. If $\rho = (q_{\mathrm{i}},\nu)$, the property is trivial. Now, we suppose that $\rho = \rho' \xrightarrow{\delta',t} (\pi,\nu)$, with $\rho' \in \mathsf{FPlays}^*_{\mathcal{U}}$ ending in a configuration (π',ν') such that $\pi' \notin L'_T$. In particular, we have $\mathsf{proj}(\rho) = \mathsf{proj}(\rho') \xrightarrow{\delta,t} (\mathsf{last}(\pi),\nu)$ with $\delta = \Delta\mathsf{proj}(\delta')$, and, by the induction hypothesis, $\mathsf{proj}(\rho')$ follows π'. Moreover, we have $\textsc{Next}(\pi',\delta) = (\pi,\delta')$ such that π must be obtained from π' on lines 4 or 8 of Algorithm 1, i.e. $\pi = \pi' \xrightarrow{\delta} q_2$ where q_2 is given by δ. Thus, we deduce that $\mathsf{proj}(\rho)$ follows π. □

Now, for all plays $\rho \in \mathsf{FPlays}^*_{\mathcal{U}}$ such that $\mathsf{last}(\rho) = (\pi,\nu)$ and $\pi \in L'_{\mathsf{Max}}$ (for plays not starting in q_{i}, the decision over ρ is irrelevant), we define a strategy $\tau_{\mathcal{U}}$ for Max in $\mathcal{U}$ by

$$\tau_{\mathcal{U}}(\rho) = (\delta',t) \quad \text{if } \tau_{\overline{\mathcal{G}}}(\mathsf{proj}(\rho)) = (\delta,t) \text{ and } \textsc{Next}(\pi,\delta) = (\pi',\delta')$$

We note that this is a valid decision for Max: we apply the same delay (since delays chosen in $\tau_{\overline{\mathcal{G}}}$ and $\tau_{\mathcal{U}}$ are identical) from the same configuration (as $\mathsf{last}(\mathsf{proj}(\rho)) = (\mathsf{last}(\pi),\nu)$, by Lemma 4.18-1), through the same guard (since guards of δ and δ' are identical). Thus, whether or not the location π is urgent (i.e. $\mathsf{last}(\pi)$ is urgent), the decision (δ',t) gives rise to an edge in $[\![\mathcal{U}]\!]$. Moreover, since the definition of $\tau_{\mathcal{U}}$ relies on the projection, it is of no surprise that:

Lemma 4.19. *Let $\rho \in \mathsf{FPlays}^*_{\mathcal{U}}$ be a play conforming to $\tau_{\mathcal{U}}$. Then, $\mathsf{proj}(\rho)$ is conforming to $\tau_{\overline{\mathcal{G}}}$.*

Proof. We reason by induction on the length of ρ. If $\rho = (q_{\mathrm{i}},\nu)$, then $\mathsf{proj}(\rho) = (q_{\mathrm{i}},\nu)$, and the property is trivial. Otherwise, we suppose that $\rho = \rho' \xrightarrow{\delta',t} (\pi,\nu)$

and $\mathsf{proj}(\rho) = \mathsf{proj}(\rho') \xrightarrow{\Delta\mathsf{proj}(\delta'),t} (\mathsf{last}(\pi), \nu)$. By the induction hypothesis, $\mathsf{proj}(\rho')$ is conforming to $\tau_{\overline{\mathcal{G}}}$. Letting $\mathsf{last}(\mathsf{proj}(\rho')) = (q', \nu')$, we conclude by distinguishing two cases. First, if $q' \in Q_{\mathsf{Min}}$, we directly conclude that $\mathsf{proj}(\rho)$ is conforming to $\tau_{\overline{\mathcal{G}}}$ too. Otherwise, we suppose that $q' \in Q_{\mathsf{Max}}$. Since ρ is conforming to $\tau_{\mathcal{U}}$ and ρ' also belongs to Max (by Lemma 4.18-1), we have $\tau_{\mathcal{U}}(\rho') = (\delta', t)$. In particular, by definition of $\tau_{\mathcal{U}}$, $\tau_{\overline{\mathcal{G}}}(\mathsf{proj}(\rho')) = (\Delta\mathsf{proj}(\delta'), t)$. Thus, $\rho_{\overline{\mathcal{G}}}$ is conforming to $\tau_{\overline{\mathcal{G}}}$. □

Finally, we prove $\mathsf{dVal}^{\tau_{\overline{\mathcal{G}}}}_{\mathcal{R}(\mathcal{G})}(q_{\mathsf{i}}, \nu) \leq \mathsf{dVal}^{\tau_{\mathcal{U}}}_{\mathcal{U}}(q_{\mathsf{i}}, \nu)$ by showing that for all plays $\rho_{\mathcal{U}}$ from (q_{i}, ν) conforming to $\tau_{\mathcal{U}}$, there exists a play $\rho_{\overline{\mathcal{G}}}$ from (q_{i}, ν) conforming to $\tau_{\overline{\mathcal{G}}}$ such that $\mathbf{P}(\rho_{\overline{\mathcal{G}}}) \leq \mathbf{P}(\rho_{\mathcal{U}})$. If $\rho_{\mathcal{U}}$ does not reach a target location of $\mathcal{U}$ or reaches target $\mathsf{t}_{+\infty}$, then $\mathbf{P}(\rho_{\mathcal{U}}) = +\infty$, and for all plays $\rho_{\overline{\mathcal{G}}}$ conforming to $\tau_{\overline{\mathcal{G}}}$, we have $\mathbf{P}(\rho_{\overline{\mathcal{G}}}) \leq +\infty = \mathbf{P}(\rho_{\mathcal{U}})$. Now, we suppose that $\rho_{\mathcal{U}}$ reaches a target location different from $\mathsf{t}_{+\infty}$.

- If the target location reached by $\rho_{\mathcal{U}}$ is not in $\{\mathsf{t}_{\geq 0}, \mathsf{t}_{<0}\}$, then $\rho_{\mathcal{U}} \in \mathsf{FPlays}^*_{\mathcal{U}}$, and we can use the projector operator to let $\rho_{\overline{\mathcal{G}}} = \mathsf{proj}(\rho_{\mathcal{U}})$. It is conforming to $\tau_{\overline{\mathcal{G}}}$ (by Lemma 4.19). Moreover, by letting $\mathsf{last}(\rho_{\mathcal{U}}) = (\pi, \nu)$ (with $\pi \neq \mathsf{t}_{+\infty}$ by hypothesis), we have $\mathsf{wt}'_T(\pi, \nu) = \overline{\mathsf{wt}_T}(\mathsf{last}(\pi), \nu)$ since $\mathsf{last}(\rho_{\overline{\mathcal{G}}}) = (\mathsf{last}(\pi), \nu)$, by Lemma 4.18-1. We conclude that $\mathbf{P}(\rho_{\overline{\mathcal{G}}}) = \mathbf{P}(\rho_{\mathcal{U}})$ since proj preserves the weight (by Lemma 4.18-2).

- If the target location reached by $\rho_{\mathcal{U}}$ is $\mathsf{t}_{\geq 0}$, then we decompose $\rho_{\mathcal{U}}$ as $\rho_{\mathcal{U}} = \rho^1_{\mathcal{U}} \xrightarrow{\delta',t} (\mathsf{t}_{\geq 0}, \nu)$ with $\rho^1_{\mathcal{U}} \in \mathsf{FPlays}^*_{\mathcal{U}}$ and $(\pi', \nu') = \mathsf{last}(\rho^1_{\mathcal{U}})$. Since the value in $\overline{\mathcal{G}}$ is supposed to be finite (we removed configurations of value $+\infty$ or $-\infty$), Min can always guarantee that the target is reached, i.e. there exists an (attractor) memoryless strategy $\sigma_{\overline{\mathcal{G}}}$ that guarantees that Q_t is reached. Now, let $\rho_{\overline{\mathcal{G}}} = \rho^1_{\overline{\mathcal{G}}}\rho^2_{\overline{\mathcal{G}}}$ be such that $\rho^1_{\overline{\mathcal{G}}} = \mathsf{proj}(\rho^1_{\mathcal{U}}) \xrightarrow{\delta,t} (q, \nu)$ with $\delta = \Delta\mathsf{proj}(\delta')$ and $\rho^2_{\overline{\mathcal{G}}}$ be the play from (q, ν) conforming to $\tau_{\overline{\mathcal{G}}}$ and $\sigma_{\overline{\mathcal{G}}}$. To conclude this case, we prove that $\rho_{\overline{\mathcal{G}}}$ is conforming to $\tau_{\overline{\mathcal{G}}}$ and $\mathbf{P}(\rho_{\overline{\mathcal{G}}}) \leq \mathbf{P}(\rho_{\mathcal{U}})$.

  First, since $\mathsf{proj}(\rho^1_{\mathcal{U}})$ is conforming to $\tau_{\overline{\mathcal{G}}}$ (by Lemma 4.19), then $\rho^1_{\overline{\mathcal{G}}}$ is conforming to $\tau_{\overline{\mathcal{G}}}$ if and only if its last decision is. If $\pi' \in L'_{\mathsf{Min}}$, then $\mathsf{proj}(\rho^1_{\mathcal{U}})$ belongs to Min (by Lemma 4.18-1) and $\rho^1_{\overline{\mathcal{G}}}$ is conforming to $\tau_{\overline{\mathcal{G}}}$. Otherwise, we suppose that $\pi' \in L'_{\mathsf{Max}}$, then $\tau_{\mathcal{U}}(\rho^1_{\mathcal{U}}) = (\delta', t)$ and $\textsc{Next}(\pi', \delta) = (\mathsf{t}_{<0}, \delta')$. Thus, since $\mathsf{proj}(\rho^1_{\mathcal{U}})$ belongs to Max (by Lemma 4.18-1) and by the construction of $\tau_{\mathcal{U}}$, we deduce that $\tau_{\overline{\mathcal{G}}}(\mathsf{proj}(\rho^1_{\mathcal{U}})) = (\delta, t)$, i.e. $\rho^1_{\overline{\mathcal{G}}}$ is conforming to $\tau_{\overline{\mathcal{G}}}$. Finally, we conclude that $\rho_{\overline{\mathcal{G}}}$ is conforming to $\tau_{\overline{\mathcal{G}}}$ by the choice of $\rho^2_{\overline{\mathcal{G}}}$.

  Now, we prove that $\mathbf{P}(\rho_{\overline{\mathcal{G}}}) \leq \mathbf{P}(\rho_{\mathcal{U}})$. First, we remark that

$$\mathbf{P}(\rho_{\overline{\mathcal{G}}}) = \mathsf{wt}_\Sigma(\rho^1_{\overline{\mathcal{G}}}) + \mathbf{P}(\rho^2_{\overline{\mathcal{G}}}) = \mathsf{wt}_\Sigma(\mathsf{proj}(\rho^1_{\mathcal{U}})) + t\,\overline{\mathsf{wt}}(\mathsf{last}(\pi')) + \overline{\mathsf{wt}}(\delta) + \mathbf{P}(\rho^2_{\overline{\mathcal{G}}})\,.$$

In particular, since $\overline{\mathsf{wt}}(\mathsf{last}(\pi')) = \mathsf{wt}'(\pi')$ (by definition of $\mathcal{U}$) and also by using Lemma 4.18-2, we obtain

$$\mathbf{P}(\rho_{\overline{\mathcal{G}}}) = \mathsf{wt}_\Sigma(\rho^1_{\mathcal{U}}) + t\,\mathsf{wt}'(\pi') + \mathsf{wt}'(\delta') + \mathbf{P}(\rho^2_{\overline{\mathcal{G}}}) = \mathsf{wt}_\Sigma(\rho_{\mathcal{U}}) + \mathbf{P}(\rho^2_{\overline{\mathcal{G}}})\,.$$

Moreover, the length of $\rho^2_{\overline{\mathcal{G}}}$ is bounded by $|Q|$ (since it is conforming to an attractor, and since regions are already encoded in $\overline{\mathcal{G}}$) and each of its edges has a weight bounded in absolute values by W_{e}. By adding its final weight, we obtain

$$\mathbf{P}(\rho_{\overline{\mathcal{G}}}) \leq \mathsf{wt}_\Sigma(\rho_{\mathcal{U}}) + |Q|\,W_{\mathsf{e}} + W_{\mathsf{fin}}\,.$$

To conclude, we remark that $\rho_{\mathcal{U}}$ reaches $\mathsf{t}_{\geq 0}$, and its weight is thus

$$\mathbf{P}(\rho_{\mathcal{U}}) = \mathsf{wt}_\Sigma(\rho_{\mathcal{U}}) + |Q|\,W_{\mathsf{e}} + W_{\mathsf{fin}}\,.$$

Therefore, $\mathbf{P}(\rho_{\overline{\mathcal{G}}}) \leq \mathbf{P}(\rho_{\mathcal{U}})$.

- Finally, we prove that the case where the target location reached by $\rho_{\mathcal{U}}$ is $\mathsf{t}_{<0}$ is not possible. As before we decompose $\rho_{\mathcal{U}}$ as $\rho_{\mathcal{U}} = \rho^1_{\mathcal{U}} \xrightarrow{\delta',t} (\mathsf{t}_{\geq 0}, \nu)$ with $\rho^1_{\mathcal{U}} \in \mathsf{FPlays}^*_{\mathcal{U}}$ and $(\pi', \nu') = \mathsf{last}(\rho^1_{\mathcal{U}})$. We consider the play $\rho^1_{\overline{\mathcal{G}}} = \mathsf{proj}(\rho^1_{\mathcal{U}}) \xrightarrow{\delta,t} (q, \nu)$ with $\delta = \Delta\mathsf{proj}(\delta')$ that is conforming to $\tau_{\overline{\mathcal{G}}}$ (by the same reasoning as the previous case) and we prove that $\rho^1_{\overline{\mathcal{G}}}$ finishes with a play that follows the cyclic path with a negative value that contradicts Lemma 4.10-2. By definition of $\mathcal{U}$, we have $\textsc{Next}(\pi', \delta) = (\mathsf{t}_{<0}, \delta')$ with $|\pi'|_\delta > 0$, so by letting $\pi' = \pi_1 \xrightarrow{\delta} \pi_2$ with $|\pi_2|_\delta = 0$, we have $\mathsf{dVal}_{\overline{\mathcal{G}}}(\pi_2 \xrightarrow{\delta} q_2) < 0$ where q_2 is given by δ. Moreover, since $\mathsf{proj}(\rho^1_{\mathcal{U}})$ follows π (by Lemma 4.18-3), $\rho^1_{\overline{\mathcal{G}}}$ follows $\pi \xrightarrow{\delta} q_2$ that contains a cyclic path $\pi_2 \xrightarrow{\delta} q_2$ with a negative value.

To conclude the proof, we have shown that for all plays $\rho_{\mathcal{U}}$ from (q_{i}, ν) conforming to $\tau_{\mathcal{U}}$, we can build a play $\rho_{\overline{\mathcal{G}}}$ from (q_{i}, ν) conforming to $\tau_{\overline{\mathcal{G}}}$ such that $\mathbf{P}(\rho_{\overline{\mathcal{G}}}) \leq \mathbf{P}(\rho_{\mathcal{U}})$. In particular,

$$\begin{aligned}
\mathsf{dVal}^{\tau_{\overline{\mathcal{G}}}}_{\overline{\mathcal{G}}}(q_{\mathsf{i}}, \nu) &= \inf_{\tau_{\overline{\mathcal{G}}} \in \mathsf{dStrat}_{\mathsf{Min}}(\overline{\mathcal{G}})} \mathbf{P}(\mathsf{Play}((q_{\mathsf{i}}, \nu), \sigma_{\overline{\mathcal{G}}}, \tau_{\overline{\mathcal{G}}})) \\
&\leq \inf_{\tau_{\mathcal{U}} \in \mathsf{dStrat}_{\mathsf{Min}}(\mathcal{U})} \mathbf{P}(\mathsf{Play}((q_{\mathsf{i}}, \nu), \sigma_{\mathcal{U}}, \tau_{\mathcal{U}})) \\
&\leq \mathsf{dVal}^{\tau_{\mathcal{U}}}_{\mathcal{U}}(q_{\mathsf{i}}, \nu)\,.
\end{aligned}$$

Second inequality We prove the reciprocal inequality $\mathsf{dVal}_{\overline{\mathcal{G}}}(q_{\mathrm{i}}, \nu) \geq \mathsf{dVal}_{\mathcal{U}}(q_{\mathrm{i}}, \nu)$ that can be rewritten as

$$\mathsf{dVal}_{\overline{\mathcal{G}}}(q_{\mathrm{i}}, \nu) \geq \sup_{\tau_{\mathcal{U}}} \mathsf{dVal}_{\mathcal{U}}^{\tau_{\mathcal{U}}}(q_{\mathrm{i}}, \nu)\,.$$

It thus amounts to showing that Max can guarantee to always do at least as good in $\overline{\mathcal{G}}$ as in $\mathcal{U}$. We, thus, fix the optimal strategy $\tau_{\mathcal{U}}$ in $\mathcal{U}$ given by Lemma 4.15, and show that $\mathsf{dVal}_{\overline{\mathcal{G}}}(q_{\mathrm{i}}, \nu) \geq \mathsf{dVal}_{\mathcal{U}}^{\tau_{\mathcal{U}}}(q_{\mathrm{i}}, \nu)$.

To do so, we show that there exists a strategy $\tau_{\overline{\mathcal{G}}}$ in $\overline{\mathcal{G}}$ such that for a particular play ρ conforming to $\tau_{\overline{\mathcal{G}}}$, there exists a play conforming to $\tau_{\mathcal{U}}$ with a weight at most the weight of ρ. As depicted in Figure 4.7, the strategy $\tau_{\overline{\mathcal{G}}}$ is defined via a function Φ that maps plays of $\overline{\mathcal{G}}$ into plays of $\mathcal{U}$. Intuitively, this function removes all cyclic plays ending with a reset from plays in $\overline{\mathcal{G}}$. Formally, it is defined by induction on the length of the plays by letting $\Phi(q_{\mathrm{i}}, \nu) = (q_{\mathrm{i}}, \nu)$, and for all finite plays $\rho \in \mathsf{FPlays}_{\overline{\mathcal{G}}}$, letting $\rho' = \rho \xrightarrow{\delta, t} (q, \nu)$,

1. if $\Phi(\rho)$ ends in $\mathsf{t}_{+\infty}$, we fix $\Phi(\rho') = \Phi(\rho)$;
2. else, if δ contains a reset and $\Phi(\rho) = \rho_1 \xrightarrow{\delta', t'} \rho_2$ with $\Delta\mathsf{proj}(\delta') = \delta$, letting π the first location of ρ_2, we fix $\Phi(\rho') = \rho_1 \xrightarrow{\delta', t'} (\pi, 0)$;
3. otherwise, letting $\textsc{Next}(\pi, \delta) = (\pi', \delta')$ with π the last location of $\Phi(\rho)$, we fix $\Phi(\rho') = \Phi(\rho) \xrightarrow{\delta', t} (\pi', \nu)$.

This function satisfies the following properties:

Lemma 4.20. *For all plays $\rho \in \mathsf{FPlays}_{\overline{\mathcal{G}}}$, if $\mathsf{last}(\Phi(\rho)) = (\pi, \nu)$ with $\pi \neq \mathsf{t}_{+\infty}$, then we have $\pi \notin \{\mathsf{t}_{<0}, \mathsf{t}_{\geq 0}\}$ and*

$$\mathsf{last}(\rho) = \begin{cases} (\mathsf{last}(\pi), \nu) & \text{if } \pi \notin Q_t\,; \\ (\pi, \nu) & \text{otherwise}\,. \end{cases}$$

Proof. We show the property by induction on the length of ρ. If $\rho = (q_{\mathrm{i}}, \nu)$, then $\Phi(\rho) = \rho$, and the property holds. Otherwise, we let $\rho' = \rho \xrightarrow{t, \delta} (q, \nu)$ such that the property holds for ρ (since it does not end in L'_T), and we follow the definition of Φ.

1. If $\Phi(\rho)$ ends in $\mathsf{t}_{+\infty}$, we have $\Phi(\rho') = \Phi(\rho)$, and this case is thus not possible (since $\Phi(\rho')$ is supposed to not end in $\mathsf{t}_{+\infty}$).
2. Else, if δ contains a reset and $\Phi(\rho) = \rho_1 \xrightarrow{\delta', t'} \rho_2$ with $\mathsf{last}(\rho_1) = (\pi_1, \nu)$ and $\Delta\mathsf{proj}(\delta') = \delta = (q, r, Y, q')$, we have $\Phi(\rho') = \rho_1 \xrightarrow{\delta', t'} (\pi', 0)$, by letting π'

be the first location of ρ_2. Moreover, we have $\textsc{Next}(\pi_1, \delta) = (\pi', \delta')$. Now, by definition of $\textsc{Next}$, if $q' \in Q_t$, then $\pi' = q' \in Q_t$. Thus, we conclude that $\mathsf{last}(\rho') = (q', 0) = (\mathsf{last}(\Phi(\rho')), 0)$ as expected. Otherwise, $q' \notin Q_t$ and we have $\mathsf{last}(\Phi(\rho)) = (\pi, 0)$. We note that $\pi \notin \{\mathsf{t}_{<0}, \mathsf{t}_{\geq 0}\}$ since ρ_1 does not contain a transition δ_1' such that $\Delta\mathsf{proj}(\delta_1') = \delta$ (otherwise, in $\Phi(\rho)$, we would have already fired twice the transition δ with a reset, before trying to fire it a third time). Thus $\pi = \pi' \xrightarrow{\delta} q'$, and we conclude.

3. Otherwise, $\Phi(\rho') = \Phi(\rho) \xrightarrow{\delta', t} (\pi', \nu)$ if $\textsc{Next}(\pi, \delta) = (\pi', \delta')$ with π the last location of $\Phi(\rho)$. Once again, we are in a case where $\pi' = \pi \xrightarrow{\delta} q'$, by letting $\delta = (q, r, Y, q')$. Thus, we conclude as before. □

Now, we define $\tau_{\overline{\mathcal{G}}}$ such that its behaviour is the same as the one given by $\tau_{\mathcal{U}}$ after the application of Φ on the finite play, i.e. after the removal of all cyclic paths between the same transition with a reset. Formally, for all plays $\rho \in \mathsf{FPlays}_{\overline{\mathcal{G}}}$, we let $\tau_{\overline{\mathcal{G}}}(\rho)$ be defined as any valid decision (δ, t) if $\Phi(\rho)$ ends in $\mathsf{t}_{+\infty}$, and otherwise,

$$\tau_{\overline{\mathcal{G}}}(\rho) = (\Delta\mathsf{proj}(\delta'), t) \qquad \text{if } \tau_{\mathcal{U}}(\Phi(\rho)) = (\delta', t)$$

This is a valid decision for Max. First, by Lemma 4.20, $\mathsf{last}(\rho) = (\mathsf{last}(\pi), \nu)$ when $\mathsf{last}(\Phi(\rho)) = (\pi, \nu)$. Moreover, delays chosen in $\tau_{\overline{\mathcal{G}}}$ and $\tau_{\mathcal{U}}$ are the same, and the guards of δ' and $\Delta\mathsf{proj}(\delta')$ are identical. Thus, whether or not the location π is urgent, the decision $(\Delta\mathsf{proj}(\delta'), t)$ gives rise to an edge in $[\![\overline{\mathcal{G}}]\!]$. Since the definition of $\tau_{\overline{\mathcal{G}}}$ relies on the operation Φ, it is again not surprising that:

Lemma 4.21. *Let $\rho \in \mathsf{FPlays}_{\overline{\mathcal{G}}}$ be a play conforming to $\tau_{\overline{\mathcal{G}}}$. Then $\Phi(\rho)$ is conforming to $\tau_{\mathcal{U}}$.*

Proof. We reason by induction on the length of ρ. If $\rho = (q_{\mathrm{i}}, \nu)$, then $\Phi(\rho) = (q_{\mathrm{i}}, \nu)$ and the property is trivial. Otherwise, we suppose that $\rho' = \rho \xrightarrow{\delta, t} (q, \nu)$. By the induction hypothesis, $\Phi(\rho)$ conforms to $\tau_{\mathcal{U}}$.

1. If $\Phi(\rho)$ ends in $\mathsf{t}_{+\infty}$, we have $\Phi(\rho') = \Phi(\rho)$ that is conforming to $\tau_{\mathcal{U}}$.

2. If δ contains a reset and $\Phi(\rho) = \rho_1 \xrightarrow{\delta', t'} \rho_2$ with $\Delta\mathsf{proj}(\delta') = \delta$, letting π be the first location of ρ_2, we have $\Phi(\rho') = \rho_1 \xrightarrow{\delta', t'} (\pi, 0)$. This is a prefix of $\Phi(\rho)$ that is conforming to $\tau_{\mathcal{U}}$. Thus, $\Phi(\rho')$ is conforming to $\tau_{\mathcal{U}}$ too.

3. Otherwise, $\Phi(\rho') = \Phi(\rho) \xrightarrow{\delta', t} (\pi', \nu)$ if $\textsc{Next}(\pi, \delta) = (\pi', \delta')$ with π the last location of $\Phi(\rho)$. If $\Phi(\rho)$ ends in a location of Min, since it is conforming to $\tau_{\mathcal{U}}$, so does $\Phi(\rho')$. Otherwise, $\tau_{\overline{\mathcal{G}}}(\rho) = (\delta, t)$ which implies that $\tau_{\mathcal{U}}(\Phi(\rho)) = (\delta'', t)$ with $\Delta\mathsf{proj}(\delta'') = \delta$, meaning that $\textsc{Next}(\pi, \delta) = (\pi', \delta'')$, i.e. $\delta'' = \delta'$: in this case too, $\Phi(\rho')$ is conforming to $\tau_{\mathcal{U}}$. □

Finally, we prove that $\mathsf{dVal}_{\overline{\mathcal{G}}}(q_\mathsf{i},\nu) \geq \mathsf{dVal}^{\tau_\mathcal{U}}_{\mathcal{U}}(q_\mathsf{i},\nu)$. We note that we do not aim at comparing $\mathsf{dVal}^{\tau_\mathcal{U}}_{\mathcal{U}}(q_\mathsf{i},\nu)$ with $\mathsf{dVal}^{\tau_{\overline{\mathcal{G}}}}_{\overline{\mathcal{G}}}(q_\mathsf{i},\nu)$ but instead directly with $\mathsf{dVal}_{\overline{\mathcal{G}}}(q_\mathsf{i},\nu)$. This is helpful here since we do not need to start with any play ρ conforming to $\tau_{\overline{\mathcal{G}}}$. Instead, we pick a special play, choosing well the strategy followed by Min. Indeed, we suppose that Min follows an almost-optimal switching strategy σ in $\overline{\mathcal{G}}$, as given in Theorem 3.9, that ensures that in all plays $\rho_{\overline{\mathcal{G}}}$ conforming to σ, the target is reached fast enough (with a number of transitions bounded by κ). We can easily enrich this result to take into account resets. Indeed, as performed in [Bri+22, Theorem 6.6] to show that all one-clock WTGs have an (a priori non-computable) value function that is piecewise affine with a finite number of cutpoints, we can replace each transition with a reset by a new transition jumping in a fresh target location of final weight given by the value function we aim at computing. From a strategy perspective, this means that in each component of our unfolding (in-between two transitions with a reset), Min follows a switching strategy. We remark that such strategies are a priori not known to be computable (since we cannot perform the transformation described above using the value function), but we use only its existence in this proof.

Given $\varepsilon > 0$, we thus consider an ε-optimal strategy σ for Min in $\overline{\mathcal{G}}$ such that in all plays $\rho_{\overline{\mathcal{G}}}$ conforming to σ, in-between two transitions with a reset and after the last such transition, the number of transitions is bounded by κ. We now fix the special play ρ from (q_i,ν) conforming to σ and $\tau_{\overline{\mathcal{G}}}$. It reaches a target location since σ is ε-optimal and $\mathsf{dVal}_{\overline{\mathcal{G}}}(q_\mathsf{i},\nu) \neq +\infty$. We show that

$$\exists \rho_\mathcal{U} \in \mathsf{FPlays}_\mathcal{U} \text{ conforming to } \tau_\mathcal{U} \quad \mathbf{P}(\rho_\mathcal{U}) \leq \mathbf{P}(\rho) \tag{$\star$}$$

As a consequence, we obtain

$$\begin{aligned}\mathsf{dVal}^{\tau_\mathcal{U}}_{\mathcal{U}}(q_\mathsf{i},\nu) &= \inf_{\sigma_\mathcal{U} \in \mathsf{dStrat}_{\mathsf{Min}}(\mathcal{U})} \mathbf{P}(\mathsf{Play}((q_\mathsf{i},\nu),\sigma_\mathcal{U},\tau_\mathcal{U})) \\ &\leq \mathbf{P}(\rho_\mathcal{U}) \leq \mathbf{P}(\rho) \leq \mathsf{dVal}_{\overline{\mathcal{G}}}(q_\mathsf{i},\nu) + \varepsilon\,.\end{aligned}$$

Since this holds for all $\varepsilon > 0$, we have $\mathsf{dVal}^{\tau_\mathcal{U}}_{\mathcal{U}}(q_\mathsf{i},\nu) \leq \mathsf{dVal}_{\overline{\mathcal{G}}}(q_\mathsf{i},\nu)$ as expected.

To show ($\star$), we proceed by induction on the prefixes ρ' of ρ, proving that ($\star$) holds or that $\Phi(\rho')$ does not end in $\mathsf{t}_{+\infty}$ and $\mathsf{wt}_\Sigma(\Phi(\rho')) \leq \mathsf{wt}_\Sigma(\rho')$. Indeed, at the end of the induction, we, therefore, obtain ($\star$) or that $\Phi(\rho)$ does not end in $\mathsf{t}_{+\infty}$ and $\mathsf{wt}_\Sigma(\Phi(\rho)) \leq \mathsf{wt}_\Sigma(\rho)$. In the case where ($\star$) does not hold, we fix $\rho_\mathcal{U} = \Phi(\rho)$ and $\mathsf{last}(\rho_\mathcal{U}) = (\pi,\nu)$. In particular, we have $\pi \in Q_t$, and $\mathsf{last}(\rho) = (\pi,\nu)$: by Lemma 4.20, if $\pi \notin Q_t$, then $\mathsf{last}(\rho) = (\mathsf{last}(\pi),\nu)$, with $\mathsf{last}(\pi) \notin Q_t$ that contradicts the fact that ρ reaches the target. Therefore,

$$\mathbf{P}(\rho_\mathcal{U}) = \mathbf{P}(\Phi(\rho)) = \mathsf{wt}_\Sigma(\Phi(\rho)) + \mathsf{wt}'_T(\pi,\nu) \leq \mathsf{wt}_\Sigma(\rho) + \mathsf{wt}'_T(\pi,\nu) = \mathbf{P}(\rho)\,.$$

Since $\rho_{\mathcal{U}}$ is conforming to $\tau_{\mathcal{U}}$ (by Lemma 4.21), we obtain ($\star$) here too.

Finally, we proceed to the proof by induction. First, we suppose that $\rho' = (q_{\mathrm{i}}, \nu)$ and $\mathsf{wt}_\Sigma(\Phi(\rho')) = 0 = \mathsf{wt}_\Sigma(\rho')$. Otherwise, we suppose that $\rho' = \rho'' \xrightarrow{\delta,t} (q, \nu)$. By induction on ρ'', if ($\star$) does not (already) hold, we know that $\Phi(\rho'')$ does not end in $\mathsf{t}_{+\infty}$ and $\mathsf{wt}_\Sigma(\Phi(\rho'')) \leq \mathsf{wt}_\Sigma(\rho'')$. We follow the three cases of the definition of $\Phi(\rho')$.

1. We cannot have $\Phi(\rho'')$ ending in $\mathsf{t}_{+\infty}$ by hypothesis.

2. Suppose now that δ contains a reset and $\Phi(\rho'') = \rho_1 \xrightarrow{\delta',t'} \rho_2$ with $\Delta\mathsf{proj}(\delta') = \delta$. Letting π be the first location of ρ_2, we have $\Phi(\rho') = \rho_1 \xrightarrow{\delta',t'} (\pi, 0)$. Thus

$$\mathsf{wt}_\Sigma(\Phi(\rho')) = \mathsf{wt}_\Sigma(\Phi(\rho'')) - \mathsf{wt}_\Sigma(\rho_2) \leq \mathsf{wt}_\Sigma(\rho'') - \mathsf{wt}_\Sigma(\rho_2) \tag{4.6}$$

   Let $(\pi', \nu') = \mathsf{last}(\rho_2)$, and $\rho_{\mathcal{U}} = \Phi(\rho'') \xrightarrow{\delta',t} (\pi'', 0)$, with $\textsc{Next}(\pi', \delta) = (\pi'', \delta'')$. Notice that $\rho_{\mathcal{U}}$ is conforming to $\tau_{\mathcal{U}}$, since $\Phi(\rho'')$ does and if π' belongs to Max, this follows directly from the definition of $\tau_{\overline{\mathcal{G}}}$ from $\tau_{\mathcal{U}}$ (since $\tau_{\overline{\mathcal{G}}}(\rho''_{\overline{\mathcal{G}}}) = (\delta, t)$, and $\Phi(\rho'') \notin \mathsf{t}_{+\infty}$). Moreover, it contains twice a transition with a reset coming from the same transition δ of $\overline{\mathcal{G}}$, therefore $\pi'' \in \{\mathsf{t}_{<0}, \mathsf{t}_{\geq 0}\}$. If $\pi'' = \mathsf{t}_{<0}$, $\mathbf{P}(\rho_{\mathcal{U}}) = -\infty$ and ($\star$) holds. Otherwise, if $\pi'' = \mathsf{t}_{\geq 0}$, by Lemma 4.15 applied on $\rho_{\mathcal{U}}$, $\mathsf{wt}_\Sigma(\rho_2 \xrightarrow{\delta',t} (\mathsf{t}_{\geq 0}, 0)) \geq 0$, i.e. $\mathsf{wt}_\Sigma((\pi', \nu') \xrightarrow{\delta',t} (\mathsf{t}_{\geq 0}, 0)) \geq -\mathsf{wt}_\Sigma(\rho_2)$. Combined with (4.6), we obtain that

$$\begin{aligned}
\mathsf{wt}_\Sigma(\Phi(\rho')) &\leq \mathsf{wt}_\Sigma(\rho'') + \mathsf{wt}_\Sigma\big((\pi', \nu') \xrightarrow{\delta',t} (\mathsf{t}_{\geq 0}, 0)\big) \\
&= \mathsf{wt}_\Sigma(\rho'') + t\,\mathsf{wt}'(\pi') + \mathsf{wt}'(\delta'') \\
&= \mathsf{wt}_\Sigma(\rho'') + t\,\overline{\mathsf{wt}}(q') + \overline{\mathsf{wt}}(\delta) = \mathsf{wt}_\Sigma(\rho')
\end{aligned}$$

   where we have set q' the last location of ρ'', that is also the last location of π'.

3. Otherwise, $\Phi(\rho') = \Phi(\rho'') \xrightarrow{\delta',t} (\pi', \nu)$ if $\textsc{Next}(\pi, \delta) = (\pi', \delta')$ with π the last location of $\Phi(\rho'')$. In this case,

$$\begin{aligned}
\mathsf{wt}_\Sigma(\Phi(\rho')) &= \mathsf{wt}_\Sigma(\Phi(\rho'')) + t\,\mathsf{wt}'(\pi) + \mathsf{wt}'(\delta') \\
&\leq \mathsf{wt}_\Sigma(\rho'') + t\,\overline{\mathsf{wt}}(q') + \overline{\mathsf{wt}}(\delta) = \mathsf{wt}_\Sigma(\rho')
\end{aligned}$$

   where we have written q' the last location of ρ''.

This ends the proof by induction.

4.4. Decidability of the value problem

By using the unfolding, we are now able to conclude the proof of Theorem 4.1, i.e.

Theorem 4.1. *For all one-clock WTGs $\mathcal{G}$ and all initial locations ℓ_{i}, the mapping $\nu \mapsto \mathsf{dVal}_{\mathcal{G}}(\ell_{\mathsf{i}}, \nu)$ is computable in time exponential in $|L|$ and W.*

We remember (by Lemma 4.3) that we only need to explain how to compute $\nu \mapsto \mathsf{dVal}_{\overline{\mathcal{G}}}((\ell_{\mathsf{i}}, r_{\mathsf{i}}), \nu)$ over r_{i}. By Proposition 4.17, this is equivalent to computing the mapping $\nu \mapsto \mathsf{dVal}_{\mathcal{U}}((\ell_{\mathsf{i}}, r_{\mathsf{i}}), \nu)$ over r_{i}. We now explain why this is internal.

First, the definition of $\mathcal{U}$ is effective: we can compute it entirely, making use of Lemma 4.14 showing that it is a finite one-clock WTG. The only non-trivial part is the test of the sign of $\mathsf{dVal}_{\overline{\mathcal{G}}}(\pi_2 \xrightarrow{\delta} q_2)$ in line 4 of Algorithm 1 to determine in which target location we jump. Since $\pi_2 \xrightarrow{\delta} q_2$ is a finite path, we can apply Theorem 1.32 to compute the value of the corresponding game, which is exactly the value $\mathsf{dVal}_{\overline{\mathcal{G}}}(\pi_2 \xrightarrow{\delta} q_2)$. The complexity of computing the value of a path is polynomial in the length of this path (that is exponential in $|L|$ and W, by Lemma 4.14) and polynomial in $|L|$ and W (notice that weights of $\overline{\mathcal{G}}$ are the same as the ones in $\mathcal{G}$): this is thus of complexity exponential in $|L|$ and W. Since $\mathcal{U}$ has an exponential number of locations with respect to $|L|$ and W, the total time required to compute $\mathcal{U}$ is also exponential with respect to $|L|$ and W.

Lemma 4.14 ensures that $\mathcal{U}$ is acyclic, so we can apply Theorem 1.32 to compute the value function $\nu \mapsto \mathsf{dVal}_{\mathcal{U}}((\ell_{\mathsf{i}}, r_{\mathsf{i}}), \nu)$ as a piecewise affine and continuous function. It requires a complexity polynomial in the number of locations of $\mathcal{U}$, and in W (since weights of $\mathcal{U}$ all come from $\mathcal{G}$). Knowing the previous bound on the number of locations of $\mathcal{U}$, this complexity translates into an exponential time complexity with respect to $|L|$ and W, as announced.

4.5. Implementation

We conclude this chapter by turning our attention to an implementation of our algorithm to solve one-clock WTGs. This implementation, written in C++, under GPL Licence, is available on the following GitHub page:

`https://github.com/jparr2/1clock-WTG-solver/tree/main`

For convenience reasons, we use the value iteration-based algorithm instead of the algorithm designed in [Bri+22] to compute the value function. In particular, given a one-clock WTG described by a text file (as illustrated on the left of Figure 4.8), our implementation symbolically computes the value function with two versions of the value iteration algorithm: the standard version that may not converge (as illustrated in Example I.1) and a restricted version that computes only a

```
WTG
States
l0 Min 0
l1 Max -1
l2 Max 1
lf Target 0
Transitions
l0 0 [0,1] false l1
l0 1 [0,1] false lf
l1 0 [1,1] true l0
l1 0 [0,1] false l2
l2 0 [1,1] false lf
```

```
Value Iteration did not converge after 3 steps
Values obtains after 3 steps
Value in l0: Function:
cutpoints: [(0, 1/2); (1/2, 1/2); (1, 1)]
lines: {((+ 1/2), 0, 1/2), ((x + 0), 1/2, 1),}
Value in l1: Function:
cutpoints: [(0, 1); (3/4, 1/4); (1, 1/2)]
lines: {((-1 x +1), 0, 3/4), ((x -1/2), 3/4, 1),}
Value in l2: Function:
cutpoints: [(0, 1); (1, 0)]
lines: {((-1 x + 1), 0, 1),}
Value in lf: Function:
cutpoints: [(0, 0); (1, 0)]
lines: {((+ 0), 0, 1),}
```

Figure 4.8 – On the left, the input file representing the one-clock WTG depicted in Figure I.1, i.e. a one-clock WTG for which value iteration does not converge in finite time. On the right, the output obtained when we execute the command C++ to compute the value iteration algorithm three times on this one-clock WTG.

given number of steps of the algorithm chosen by the user. Since value iteration is known to converge on acyclic WTGs (in [AD94]), to guarantee the convergence of the value-iteration algorithm, the user can ask to apply the first version in a finite *unfolding* automatically computed with Algorithm 1 from the original one-clock WTG. Finally, we connect this implementation to a Python environment to draw the value function obtained from the C++ implementation.

Usage of our implementation Our implementation presents two interfaces: a C++ interface based on the C++ solver with only textual output (as on the right in Figure 4.8), or a Python interface only used to draw the value functions computed by the C++ solver (as in Figure 4.9). Moreover, the C++ solver can be managed to adapt the algorithm used to solve the chosen one-clock WTG. For instance, we can cite the following options:

- -slow: use the value iteration algorithm with only the value function computed in the previous iteration to compute the current one, otherwise we use a version of value iteration that uses the most recent value;
- -nb n: restrict to n the number of iterations on each state;
- -reg: compute the closure of the game before applying the solver;
- -unfolding *label*: compute value iteration on the unfolding from the state labelled by *label* from the region $\{0\}$.

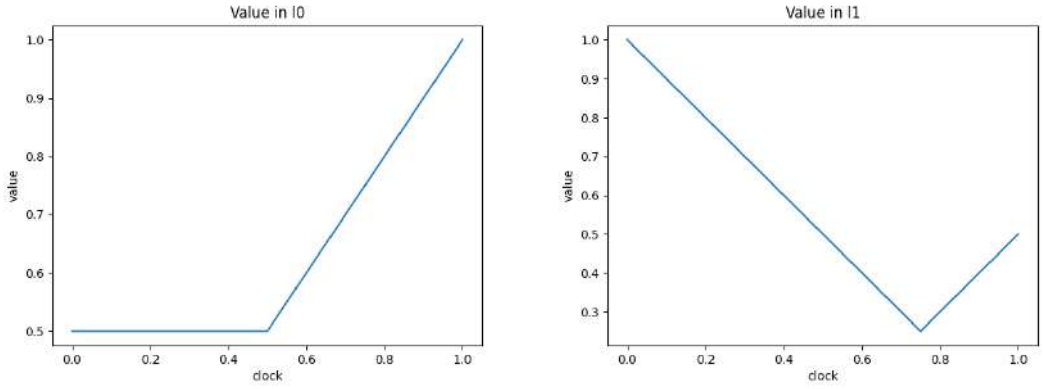


Figure 4.9 – The value functions (for ℓ_0 on the left and ℓ_1 on the right) obtained after three iterations of the value iteration algorithm on the WTG depicted in Figure I.1. As expected, our implementation reports that value iteration does not converge (in three steps).

Experimental work This implementation is immature and naive. In particular, we can not design experimentation over it: we will quickly obtain time out and memory limitation. In particular, we can guarantee a result from our implementation only for small WTGs that is not sufficient to present good experimentation.

The execution time of our implementation is triply exponential in the size of the WTG instead of only exponential as claimed in Theorem 4.1. This is because the implementation of value iteration does not rely on the structure of the one-clock WTG: in particular, no special version of this algorithm is implemented to only work on acyclic WTGs with a backward version. The main limitation to implementing such a version is the data structure used for our WTGs. For now, we represent it with a list of states and a list of transitions where states do not store ingoing or outgoing transitions, whereas a backward version must know the ingoing transitions of each step (to avoid browsing the whole set of transitions at each step).

Concerning the memory limit, when we apply the unfolding version, we need to compute the complete unfolding (its size being exponential in the size of the original WTG). Moreover, each location defined along the unfolding stores a finite path from the original WTG needed to compute the value if it induces a cyclic path ending with a reset. With our Algorithm 1, we can do no better since our unfolding is defined from the top (the initial location) to the bottom (the target locations) where the value iteration (on acyclic WTGs) computes the value from the bottom (the target locations) to the top (the initial location).

Discussion

In this part, we have solved the value problem in one-clock WTGs with arbitrary weights, an open problem for several years. We strongly rely on the determinacy of the WTGs taking the point of view of Max, instead of the one of Min as was done in previous work with only non-negative weights. We also use technical ingredients such as the closure of a one-clock WTG, switching strategies for Min, the fixpoint characterisation of the value function, and acyclic unfolding.

Solving weighted timed games

In this work, we have introduced new notions like the value of (cyclic) paths. In particular, a natural question involves extending these notions to identify other classes of WTGs for which the value problem is decidable, or at least the value function is approximable. Moreover, a complexity gap remains in solving the one-clock WTGs. Our algorithm runs in exponential time (with weights encoded in unary), which does not match the known PSPACE lower bound with weights in binary [FIS20]. Observe that this lower bound only uses non-negative weights. This complexity gap deserves further study.

A research direction concerns the unfolding. In this work, we unfold the one-clock WTG into a finite tree. It would be interesting to develop a symbolic approach that shares computation between subtrees to obtain a more efficient algorithm. A way to do it consists in folding some cyclic paths along the unfolding: in the best way, the grey bubbles in Figure 4.6 will be only a copy of the game (as in the non-negative case). This solution is not yet realistic, but a tradeoff may be achievable if we can obtain some guarantee on the value of the cyclic path when we unfold many times the same elementary cyclic path.

Extend decidability result to other classes of WTGs The analysis of the cyclic paths and, more precisely, their values in the setting of one-clock WTGs can be an inspiration to identify new decidable (or approximable) classes of WTGs with arbitrarily many clocks. In particular, the value of the cyclic path may be used to relax the definition of divergent WTGs without losing the decidability of the value problem.

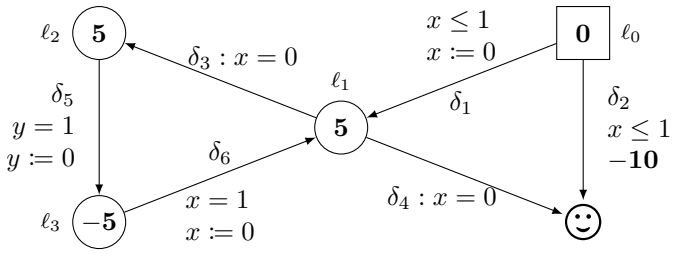


Figure I.2 – A general WTG where the function of value in ℓ_0 regarding to the clock y, i.e. the function $y \mapsto \mathsf{dVal}(\ell_0, (0, y))$, is bounded but not continuous on the region $(0, 1)$.

Another research direction concerns the classes of WTGs for which clocks are not supposed to be bounded. In particular, we aim to look after classes of WTGs in which deciding when the value function is equal to $-\infty$ is decidable, like one-clock WTGs or divergent WTGs. From the juxtaposition of locations with a value equal to $-\infty$ or $+\infty$, the extension of the method of [Beh+01] with weights compensation for the unbounded delays in the case of WTGs is not applicable.

About the deterministic value function

To obtain our result about the value problem in one-clock WTGs, we have exploited some properties of the value function as its continuity or fixpoint characterisation. To have a better understanding of the value function, we aim to extend these properties to other classes of WTGs.

About the continuity of the value function The continuity of the value function guaranteed in one-clock WTGs as well as in divergent WTGs does not extend to all WTGs. In particular, authors of [Bri+22, Remark 3.3] provide a two-clock non-divergent WTG in which the value function is not continuous.

Example I.2. We consider the general WTG depicted in Figure I.2 such that its value function in ℓ_0 admits a point of discontinuity on $y = 0.5$. To compute this value function, we focus on the value function of ℓ_1 that is given by cyclic plays that follow the path $\pi = \ell_1 \xrightarrow{\delta_3} \ell_2 \xrightarrow{\delta_5} \ell_3 \xrightarrow{\delta_6} \ell_1$. From the configuration $(\ell_1, 0, 0.5)$ (i.e. from the location ℓ_1 where $x = 0$ and $y = 0.5$), the unique cyclic play that follows π is $\rho = (\ell_1, 0, 0.5) \xrightarrow{\delta_3, 0} (\ell_2, 0, 0.5) \xrightarrow{\delta_5, 0.5} (\ell_3, 0.5, 0) \xrightarrow{\delta_6, 0.5} (\ell_1, 0, 0.5)$ and $\mathsf{wt}_\Sigma(\rho) = 0$. Thus, in ℓ_1, Min directly goes to the target, and $\mathsf{dVal}(\ell_1, 0, 0.5) = 0$.

Now, from the configuration $(\ell_1, 0, 0.6)$, the unique cyclic play that follows π is $(\ell_1, 0, 0.6) \xrightarrow{\delta_3, 0} (\ell_2, 0, 0.6) \xrightarrow{\delta_5, 0.4} (\ell_3, 0.4, 0) \xrightarrow{\delta_6, 0.6} (\ell_1, 0, 0.5)$ and its weight is -1.

Thus, for this configuration, $\mathsf{dVal}(\ell_1, 0, 0.6) = -\infty$.

In particular, we can deduce that for all configurations of ℓ_0, Max goes to the cycle when $y \leq 0.5$ and goes to the target otherwise. Thus, we have

$$\mathsf{dVal}(\ell_0, (0, y)) = \begin{cases} 0 & \text{if } y \in [0, 0.5]; \\ -10 & \text{if } y \in (0.5, 1]; \end{cases}$$

even if all iterations of $\mathcal{F}$ are continuous over regions. ⌟

Towards a fixpoint characterisation for all WTGs We have shown that the value is a fixpoint of the operator $\mathcal{F}$ defined in Chapter 2 in one-clock and divergent WTGs. A natural question is the extension of this result to all WTGs (as originally claimed in [Bou+04a]). In particular, obtaining this result would allows us to define a pseudo-algorithm that may not terminate.

However, we remark that our proof can not be extended to general WTGs. Indeed, since the value function may be discontinuous over regions (see Example I.2), in this case, iterates of $\mathcal{F}$ cannot uniformly converge to the value. Thus, either

1. the sequence uniformly converges, and the limit will be continuous (in [Bus19, Lemma 10.10], it was shown that each iterate of $\mathcal{F}$ is continuous over each region) that is not the value function that may thus not be the greatest fixpoint of $\mathcal{F}$; or

2. the sequence of iterates of $\mathcal{F}$ only simply converges to its limit, which could be the value function or not.

In particular, this implies that our proof can not a priori be adapted to all WTGs with negative weights.

Finally, in the context of almost-divergent WTGs (i.e. a divergent WTG in which cyclic plays with weight exactly zero are permitted), we note, from [Bus19, Theorem 9.3] that iterates of $\mathcal{F}$ converge to the value function. However, since $\mathcal{F}$ is not Scott-continuous (Example 2.2) and iterates of $\mathcal{F}$ do not converge in finite time (i.e. the proof for divergent WTGs cannot be extend), this convergence does not suffice to conclude the fixpoint characterisation for this class of WTGs.

Existence of switching strategies for all WTGs We prove that almost-optimal switching strategies exist only for SPGs, divergent WTGs and reset-acyclic one-clock WTGs. The question is: can we obtain such results for all one-clock WTGs or almost-divergent WTGs?

About one-clock WTGs, we know that we need to relax the definition of a switching strategy and more precisely the definition of fake-optimal strategy. In-

deed, as illustrated in Example 1.26, the negative value that Min can guarantee in the cyclic path may depend on the precision of the strategy, i.e. of ε.

Regarding almost-divergent WTGs, an extension of our proof for divergent WTGs does not hold. First, we do not have a fixpoint characterisation, even if we know that the value iteration converges to the value function (by [BMR18]). Second, even if we obtain this characterisation, the way to define the almost-optimal strategies with a small error about the optimal choice at horizon 1 is not sufficient. Indeed, we obtain that the weight of plays following a cyclic path are at most ε, so in almost-divergent WTGs, it is not sufficient to conclude that it is negative weight (since null weight cyclic paths are permitted).

Implementation

Our first implementation can be improved in terms of temporal and spatial complexity. As explained in Section 4.5, some ways may be explored: another encoding of WTGs to improve the complexity of value iteration over acyclic WTGs, and a reverse definition for the unfolding (modulo some extended storage) to avoid its whole storage. Moreover, the unfolding algorithm may be improved with a better algorithm to compute the value of the path. Indeed, with the backward algorithm, if we can store less than all the finite paths of the current stages, we decrease the memory needed to compute the value.

How to more efficiently compute the value of a cyclic path Even if we backward define the unfolding, for now, the weight of the cyclic path is necessarily computed when the cyclic path is closed and thus fully known. However, from the next lemma that ensures that delays inside the cyclic path are the minimum or the supremum in the interval of possible delays given a transition along the cyclic paths without reset transition, we can store partial computation of this value when we define the unfolding (from the target locations).

Given a closure $\overline{\mathcal{G}}$ of the one-clock WTG, we consider some cyclic path π without resets, i.e. π starts and ends in the same location of $\mathcal{R}(\mathcal{G})$, and contains no transition with reset. In particular, as an immediate consequence of transitions of $\mathcal{R}(\mathcal{G})$, this entails that every location along π has the same region, that we denote by r. Observe that if r is a singleton interval, then no delay may occur along any run following π. As a consequence, we will focus on the case where π is not a singleton, and write $r = (\alpha, \beta)$.

For technical reasons, let us write $\pi = \delta_1 \cdots \delta_n$, and write $\delta_i = (\ell_i, r) \to (\ell_{i+1}, r)$. For every $i \in \{1, \ldots, n\}$, we denote by a_i the weight of location (ℓ_i, r) and by b_i the weight of transition δ_i. We also let $b = \sum_{i=1}^{n} b_i$. Using these notations, if we denote by t_i the time spent in location (ℓ_i, r), then the weight of the unique play

that follows path π with delays $t_1, \ldots, t_n$ is equal to

$$\sum_{i=1}^{n} a_i \times t_i + b_i = b + \sum_{i=1}^{n} a_i \times t_i =: g(t_1, \ldots, t_n)\,.$$

Depending on which player owns each location, this value is defined as a succession of the infima and suprema over delays, compatible with region r. Formally, we let Opt_i denote the operator inf if location (ℓ_i, r) belongs to Player Min, and the operator sup otherwise.

We denote by ν the initial valuation, we define:

$$\mathsf{dVal}^{\nu}_{\pi} = \underset{t_1 \mid \nu + t_1 \in r}{\mathsf{Opt}_1} \quad \underset{t_2 \mid \nu + t_1 + t_2 \in r}{\mathsf{Opt}_2} \quad \cdots \quad \underset{t_n \mid \nu + \sum_{i=1}^{n} t_i \in r}{\mathsf{Opt}_n} \quad g(t_1, \ldots, t_n)\,.$$

Lemma I.3. *Let $\pi = \delta_1 \cdots \delta_n$ be a cyclic path without reset in $\overline{\mathcal{G}}$, whose associated interval is denoted $r = (\alpha, \beta)$. We claim that the optimal value $\mathsf{dVal}^{\nu}_{\pi}$ is achieved for some extremal values of delays t_i's,* i.e. *that :*

- *either we have $t_i = 0$, for every i,*
- *or there exists a unique i_0 such that $t_{i_0} = \beta - \nu$, and we have $t_i = 0$, for every $i \neq i_0$.*

In addition, we claim that which case holds does not depend on the initial valuation ν.

Proof. We will prove the property by induction on the length of π. To that end, we need a stronger induction property. We first introduce some additional notations.

Given ν, and an index k, we define the following set:

$$K^{\nu,k} = \{(t_1, \ldots, t_k) \in \mathbb{R}^k_{\geq 0} \mid \nu + t_1 \in r \wedge \nu + t_1 + t_2 \in r \wedge \ldots \wedge \nu + \sum_{j \leq k} t_j \in r\}\,.$$

For every index $k \in \{0, 1, \ldots, n\}$, we denote by $\mathsf{Opt}_{>k}$ the function mapping $(t_1, \ldots, t_k) \in K^{\nu,k}$ to

$$\underset{t_{k+1} \mid \nu + \sum_{i=1}^{k+1} t_i \in r}{\mathsf{Opt}_{k+1}} \quad \cdots \quad \underset{t_n \mid \nu + \sum_{i=1}^{n} t_i \in r}{\mathsf{Opt}_n} \quad g(t_1, \ldots, t_n)\,.$$

Observe that for $k = 0$, we obtain the constant mapping (no argument) $\mathsf{dVal}^{\nu}_{\pi}$ and for $k = n$, we obtain the mapping $(t_1, \ldots, t_n) \mapsto g(t_1, \ldots, t_n)$.

We will prove by decreasing induction on k, that for every k, for every $t_1, \ldots, t_k$, there exist Boolean values $b_{k+1}, \ldots, b_n \in \{\mathtt{T}, \mathtt{F}\}$ such that for any value of ν, the optimal value $\mathsf{Opt}_{>k}(t_1, \ldots, t_k)$ is obtained for the input delays $t_{k+1}, \ldots, t_n$ defined,

for every i, by $t_i = 0$ if $b_i = \mathtt{F}$, and $t_i = \beta - \nu - \sum_{j<i} t_j$ if $b_i = \mathtt{T}$. This mapping is affine w.r.t. its arguments, so that if it is optimized over some variables, then there exist some deterministic values of them, such that this optimal value is reached for these values, and these values are "on the border".

Assume delays $t_1, \ldots, t_{k-1}$ have been chosen in $K^{\nu,k-1}$, and we consider some Boolean $b_k \in \{\mathtt{F}, \mathtt{T}\}$, then we define t_k such that $(t_1, \ldots, t_k) \in K^{\nu,k}$ as follows:

- if $b_k = \mathtt{F}$, then we choose for t_k the minimal delay, i.e. we let $t_k = 0$;
- if $b_k = \mathtt{T}$, then we choose for t_k the maximal delay, i.e. we let $t_k = \beta - \nu - \sum_{j<k} t_j$.

We prove the following property by (descending) induction on $0 \leq k \leq n$:

- the mapping $\mathsf{Opt}_{>k}$ is affine w.r.t. its arguments $t_1, \ldots, t_k$;
- there exist Boolean values $b_i \in \{\mathtt{F}, \mathtt{T}\}$, for every $i \in \{k+1, \ldots, n\}$, such that, defining t_i from b_i by minimal delay if $b_i = \mathtt{F}$, and maximal delay otherwise, we have $\mathsf{Opt}_{>k}(t_1, \ldots, t_k) = g(t_1, \ldots, t_n)$.

Base case: $k = n$. In that case, as previously observed, $\mathsf{Opt}_{>k}$ is the mapping g. The property trivially holds as there is no variable to be optimized.
Induction case: Pick some $n \geq k > 0$, assume that the property holds for k, and let us show it also holds for $k-1$.

We fix $t_1, \ldots, t_{k-1}$ in $K^{\nu,k-1}$. We can decompose $\mathsf{Opt}_{>k-1}(t_1, \ldots, t_{k-1})$ as follows:

$$\mathsf{Opt}_{>k-1}(t_1, \ldots, t_{k-1}) = \underset{t_k \mid \nu + \sum_{j \leq k} t_j \in I}{\mathsf{Opt}_k} \mathsf{Opt}_{>k}(t_1, \ldots, t_k)\,.$$

By induction property, the mapping $\mathsf{Opt}_{>k}$ is affine. Thus, the optimisation Opt_k will be easily identified, by distinguishing cases: whether Opt_k is an infimum or a supremum and whether the coefficient of t_k in $\mathsf{Opt}_{>k}$ is positive or not. This gives rise to four cases, but for each of them, the optimal value is reached either by choosing for t_k the minimal delay (0), or the maximal one ($\beta - \nu - \sum_{j<k} t_j$). In particular, we can define a Boolean value b_k accordingly: we set it to $\mathtt{F}$ if there is no delay, and to $\mathtt{T}$ otherwise. We can easily verify that the expected induction property follows.

Last, observe that the exact value of ν has never been considered in the previous reasoning, hence the values $b_1, \ldots, b_n$ do not depend on it. □

Part II.

Stochastic Value

Introduction

Even if probabilities do not exist in the definition of WTGs, it is interesting to introduce them as a feature for strategies, i.e. we allow both players to use stochastic choices in their strategies. Indeed, probabilities can strengthen the modelisation by introducing incertitude for the opponent. For instance, stochastic strategies for Max (the environment) obscure his choices since he can play a set of edges (possibly infinite) at each turn. On Min's side, he can expect to obtain a better value by introducing some incertitude in the response for Max. In this part, we will study the values obtained when both players can play with stochastic strategies.

Probabilities to extend power of deterministic strategies From a game theory point of view, using stochastic strategies is crucial since deterministic strategies are not always powerful enough. For example, a Nash equilibrium in finite strategic games (like in rock-paper-scissors games) may exist only when we allow both players to use stochastic strategies [Nas50].

From a controller synthesis point of view, we claim that using stochastic choices is natural too to compensate for the lack of power of deterministic strategies. Indeed, for some classes of games, winning/optimal deterministic strategies may not exist. For instance, in the case of concurrent (multi-player) stochastic games under probabilistic alternating-time temporal logic with rewards (rPATL for short), stochastic strategies are needed to win [Kwi+18; Kwi+21].

However, in Chapter 6, we show that considering probabilities in WTGs contributes nothing to deterministic strategies, i.e. deterministic strategies are not less powerful than probabilistic ones. To do it, we prove that Max has always a deterministic best response strategy against a stochastic strategy of Min.

Trading memory for probabilities From a controller synthesis point of view, we claim that using stochastic choices is also natural for reducing the use of memory in the strategies of the controller. For some classes of games (where memory is required to win or play optimally), the *tradeoff between memory and randomisation* has already been investigated.

This tradeoff has been studied, first, in turn-based stochastic games like Street or Müller games. In particular, authors of [CAH04] show that randomised memoryless strategies are sufficient to win in one-player games (with or without probabilities

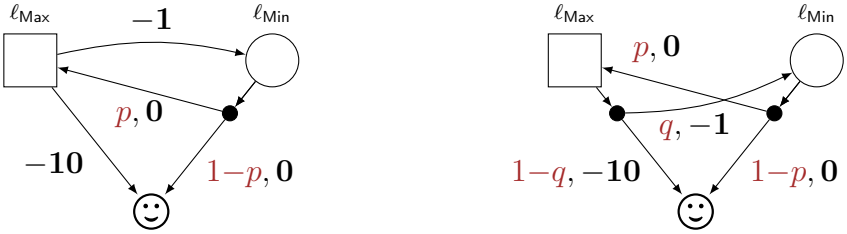


Figure II.1 – On the left, the Markov Decision Process obtained from the SPG depicted in Figure 1.8 where Min requires memory to play optimally, when letting Min play at random, with a parametric probability $p \in (0, 1)$. On the right is the Markov Chain obtained when Max plays along a memoryless randomised strategy, with a parametric probability $q \in [0, 1]$.

in the arena). However, for two-player games, finite memory is always needed (even if players use probabilistic strategies). A precise bound on the memory used (in probabilistic strategies) by both players to win is given in [Hor09].

Then, this tradeoff has been extended to the context of non-stochastic games. For example, in timed games, the goal in [CHP08b] is to reduce the information about the real-time values of clocks used in strategies. For the reachability objective, the authors prove that stochastic strategies have a simpler structure than deterministic ones: they are easier to implement.

Finally, the tradeoff has been extended for multi-dimensional quantitative objectives. For instance, in mean-payoff parity games with a threshold, if there exists a deterministic finite-memory winning strategy, then there exists a randomised memoryless almost-sure winning strategy [CRR13]. Moreover, this article [CRR13, Theorem 3] gives a characterisation of the tradeoff between memory and probabilities for energy games, multi-dimensional mean-payoff games and mean-payoff parity games.

Trading memory in WTGs We will study the tradeoff between the use of memory and probabilities in WTGs (and in SPGs) since Min may require exponential memory to play optimally (see Chapter 3). We aim at proving that the memory required by Min could be traded for stochastic choices instead (and vice versa). An example of SPG where Min needs memory to play optimally (for the deterministic value) is given by Example 1.30. We recall that the deterministic value of both locations (except the target location) is -10 since Min guarantees it from ℓ_{Min} by jumping 10 times into ℓ_{Max} before going to the target location ☺. Now, by playing a switching strategy (for instance, described in Example 3.5), i.e. turning in the negative cycle long enough so that no matter how he reaches the target finally. Thus, Min optimally plays by using pseudo-polynomial memory with respect to

Model	PPTA [BJK06]	PTGA [For+16]	TAMG [BS12]	CTMG [RS11]	STA [Bai+08]	STG [Ber+14]
Transitions	✓	✓		✓	✓	✓
Delays			Rate	Rate	✓	✓

Table II.1 – Extension of timed models with probability distributions on transitions or delays.

the weights of the arena.

In this example, such a switching strategy can be mimicked using randomisation only (and no memory): Min deciding to go to ℓ_{Max} with high probability $p < 1$ and to go to the target vertex with the remaining low probability $1 - p > 0$ (we enforce this probability to be positive, to reach the target with probability 1, no matter how the opponent is playing). The resulting *Markov Decision Process* (MDP) is depicted on the left of Figure II.1. The shortest path problem in such MDPs has been thoroughly studied in [BT91], where it is proved that Max does not require memory to play optimally. Denoting by q the probability that Max jumps in ℓ_{Min} in its memoryless strategy, we obtain the *Markov chain* (MC) on the right of Figure II.1. We can compute (see[1] Example 7.12) the expected value in this MC, as well as the best strategy for both players: the overall, optimal (stochastic) value remains -10, even if Min no longer has an optimal strategy. He rather has an ε-optimal strategy, consisting in choosing $p = 1 - \frac{\varepsilon}{10}$ that ensures an expected value at most $-10 + \varepsilon$.

In Chapter 7, we generalise the previous example for all SPGs and divergent WTGs. In particular, we introduce a new value, the *memoryless value*, by considering only memoryless strategies (for both players). Thus, we study the tradeoff between memory and probabilities by proving the equality of the memoryless value and the deterministic one. Moreover, we remark that this tradeoff does not hold for all WTGs.

How to introduce probabilities in strategies of WTGs? Before generalising this example to all WTGs, we must first introduce stochastic strategies in this context, which has never been formally explored until now, as far as we are aware. In the literature, many models extend timed models with some distributions of probability and, sometimes, with a reward extension. A summary of these known models is given in Table II.1. However, few of them may be used to define the obtained systems when we have fixed a stochastic strategy (as MDP or MC in SPGs settings).

[1] To simplify computation of the value of the MC, we need some tools, such as Bellman equations, introduced in Section 7.2.

The simplest way to introduce probabilistic distributions on timed systems consists of the definition of a probabilistic distribution over transitions from each location of a timed automaton. In particular, a *probabilistic timed game arena* (PTGA for short) [For+16] is a concurrent two-player game over a timed automaton such that at each turn the player that proposes the smaller delay chooses a distribution over transitions under a *reachability-time objective*[2]. We remark that the reachability-time objective in a PTGA is not determined (i.e. knowing the strategy of the opponent may influence the time to reach a target location). However, when the controller fixes its strategy first, the reachability-time problem over PTGAs is known as decidable [For+16, Theorem 18]. Moreover, a PTGA is a game extension (without weights) of a *priced probabilistic timed automaton* (PPTA for short) [BJK06; Kwi+06], that that is a probabilistic extension of priced timed automata by the same principle as an MDP: the player chooses a timed action, but a probabilistic distribution (induced by the chosen action) defines the next location (without modifying the delay). In PPTA, the shortest path problem is known as undecidable for at least two clocks [BCJ09]. In the case where we consider PPTA with one clock, the decidability of the problem is unknown even if a pseudo-algorithm[3] exists [BJK06]. However, the problem is open for PPTA with one-clock, simple or divergent PPTA with at least two clocks.

A second way to introduce probabilities is to define a continuous distribution for delays. For instance, a timed automaton Markov game (TAMG for short) [BS12] is a turn-based game based on a timed automaton where the delay spent in each location is chosen by the rate, i.e. an exponential distribution whose parameter is encoded in the game. In particular, players only choose a transition, they do not affect the time. In such games, the reachability is determined and decidable [BS12, Theorem 2].

Now, a way to combine both points of view is given in the definition of *continuous time Markov games* (CTMG for short), introduced in [RS11] and in which the reachability problem is known as decidable [RS11, Theorem 7]. Indeed, a CTMG encodes probabilities (chosen by the current player) into two types of locations: discrete locations to choose a transition, and continuous locations to choose a delay according to an exponential distribution. However, we note that the time is encoded with only one clock (since continuous locations only choose a delay) and that time is spent only in continuous locations (discrete location is done in urgent time).

Finally, in our setting, the better way to combine both points of view is given in *stochastic timed games* (STG for short) introduced in [BF09; Aks+16]. An STG

[2]Under a reachability-time objective, Min wants to minimise the time used, i.e. the sum of delays, to reach a target location whereas Max wants to maximise this time.

[3]A pseudo-algorithm is a correct algorithm for which the termination is not guaranteed.

is a turn-based $2-1/2$-player timed game where the $1/2$ player chooses a delay and a transition according to a measure of probability over the possible choices. he qualitative reachability (i.e. testing if the probability of reaching the target location is equal to 1) is known as decidable for one-clock $1-1/2$ STGs [BF09, Theorem 4] and conjectured decidable in one-clock STGs [Aks+16, Table 1] (for more clocks the problem remains open). The authors of these works also study quantitative reachability (i.e. testing if the probability of reaching a target location is less than $q \in (0,1)$): they prove that the problem is known as undecidable in SPGs for at least three clocks [BF09, Theorem 9] and decidable for one-clock SPGs [Aks+16, Corollary 9]. However, we want to compute an expectation over the set of plays that reach a target location, so the quantitative reachability problem is not helpful.

Remember, our objective is to introduce stochastic strategies in WTGs. Intuitively, when a stochastic strategy is fixed in WTGs, the resulting model is close[4] to a $1-1/2$-player timed game. In particular, our interest is in *stochastic timed automata* (STA for short) introduced in [Ber+14; Bou+16; Ber+18; Bou+20] which is a reduction to $1/2$-player of STG (i.e. like when we fix two stochastic strategies in WTGs). The reachability problem is decidable for one-clock STA [Bai+08, Corollary 18], or fair STA [Ber+14, Theorem 8.4] (the problem remains open for all STA).

Even if significant formal work is done to define a probabilistic measure on the set of plays in STA, by multiple integrals, the authors of [Ber+14] fix insufficient hypotheses (see Example 5.2) on the considered distribution of probabilities to guarantee their existence. Indeed, the existence of this probabilistic measure defined by a multiple integral needs to be checked. In Chapter 5, based on works on STA, we introduce *stochastic strategies* in WTGs and the needed hypotheses to define the probabilistic measure on the set of plays conforming to them. In particular, we extend the results on STA in the context of two-player games (instead of model-checking) and with weights, which indeed represents the main challenge to give meaning to the expected payoff and the *stochastic value*.

[4] According to the shape of the strategy, sometimes, the resulting models can not be encoded in a timed automaton. We note that am interesting problem is to restric to strategies that can be represented somehow as a TA or as a subclass of TA. However, in this work, we do not study this class of strategies.

5. Playing Stochastically in Weighted Timed Games

Introducing stochastic strategies in WTGs requires to take some precautions. In this chapter, we present the notion of stochastic strategy and ground mathematically the associated definitions of expected weight and stochastic value. We also adapt the notion of deterministic value to restrict deterministic strategies to be smooth (in order for them to be considered as particular stochastic strategies).

Table of contents

5.1. Probability measure on plays 128
5.1.1. A restriction over possible stochastic strategies 129
5.1.2. Construction of the probability measure 133
5.2. Expected payoff of plays 138
5.2.1. Values defined with proper (stochastic) strategies 138
5.2.2. A sufficient condition for proper strategies 143
5.3. Discrete strategies 148

In this chapter, we fix $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt}\rangle$ be a (general) WTG. Naturally, deterministic strategies for Min in $\mathcal{G}$ are extended to more general stochastic strategies. We let $\mathsf{Dist}(S)$ be the set of all probability distributions over a set S (equipped with an underlying σ-algebra). Stochastic strategies are then mappings $\eta\colon \mathsf{FPlays}_{\mathsf{Min}} \to \mathsf{Dist}(\Delta \times \mathbb{R}_{\geq 0})$ where each finite play is associated to a probability distribution over the set of pairs of transition and delay. Since Δ is a finite set, we choose to decouple the distribution on pairs of $\Delta \times \mathbb{R}_{\geq 0}$ by first selecting a transition and then a delay. Authors of [Ber+14] consider independent choices, the one on transitions being described by some weights on transitions (depending on the current region). Thus, the strategy η can be decoupled as first choosing a transition via $\eta_\Delta\colon \mathsf{FPlays}_{\mathsf{Min}} \to \mathsf{Dist}(\Delta)$, and then, knowing the chosen transition, choosing a delay via $\eta_{\mathbb{R}^+}\colon \mathsf{FPlays}_{\mathsf{Min}} \times \Delta \to \mathsf{Dist}(\mathbb{R}_{\geq 0})$, the support of the distribution $\eta_{\mathbb{R}^+}(\rho, \delta)$, denoted by $\mathrm{supp}(\eta_{\mathbb{R}^+}(\rho, \delta))$, being included in the interval $I(\rho, \delta)$ of

valid delays. Similar definitions hold for Max whose stochastic strategies will be denoted by θ.

We remark that deterministic strategies are a special case of stochastic strategies, where the distributions are chosen to be Dirac distributions. Finally, we let $\mathsf{FPlays}_{\rho}^{\eta,\theta}$ be the set of plays reaching L_T from ρ (without containing ρ) conforming to η and θ.

5.1. Probability measure on plays

We fix a strategy η and θ for each player, and an initial configuration (ℓ_0, ν_0). Our goal is to define a probability measure on plays. To do so, and following the contribution of [Ber+14] for stochastic timed automata, the set of plays of a WTG $\mathcal{G}$ starting from (ℓ_0, ν_0) and conforming to η and θ that we denote by $\mathsf{Plays}_{\ell_0,\nu_0}^{\eta,\theta}$ can be equipped with a structure of σ-algebra generated by *cylinders*, that are all subsets of plays that start with a finite prefix following the same finite path π (remember that paths are sequences of transitions, with no information on the delayed time) with some Lebesgue-measurable constraints on the delays taken along π. The idea is thus to define a probability measure $\mathbb{P}_{\ell_0,\nu_0}^{\eta,\theta}$ on the algebra generated by such cylinders (i.e. the closure of cylinders by finite union and complement) which extends uniquely as a probability measure over the whole σ-algebra (i.e. the closure of cylinders by countable union and complement), by Carathéodory's extension theorem. First, we formally define the notion of cylinders, generalising the single configuration (ℓ_0, ν_0) by a finite play ρ_0 to later take into account the possible use of memory in the strategies of both players.

Definition 5.1. Let ρ_0 be a finite play, π be a finite path, and $\mathcal{C}$ be a Lebesgue-measurable subset of $\mathbb{R}_{\geq 0}^{|\pi|}$, the *cylinder* $\mathsf{Cyl}_{\rho_0}(\pi, \mathcal{C})$ (denoted by $\mathsf{Cyl}_{\rho_0}(\pi)$ when $\mathcal{C} = \mathbb{R}_{\geq 0}^{|\pi|}$) is the set of maximal plays ρ that start in the last configuration of ρ_0 and such that the maximal play $\rho_0\rho$ *satisfies* $\pi, \mathcal{C}$ (denoted by $\rho_0\rho \models \pi, \mathcal{C}$), i.e. the prefix of length $|\pi|$ of the maximal play $\rho_0\rho$ follows π and its sequence of delays belongs to $\mathcal{C}$.

We define the probability of the cylinder $\mathsf{Cyl}_{\rho_0}(\pi, \mathcal{C})$ under the strategies η of Min and θ of Max, and denote it by $\mathbb{P}_{\rho_0}^{\eta,\theta}(\pi, \mathcal{C})$ (instead of the longer $\mathbb{P}^{\eta,\theta}(\mathsf{Cyl}_{\rho_0}(\pi, \mathcal{C}))$). If the cylinder is empty (because the finite play ρ_0 does not follow a prefix of π or contains delays not consistent with the constraint $\mathcal{C}$), we let its probability be 0. Otherwise, the probability of the non-empty cylinder is defined by induction on the length of ρ_0. If $|\rho_0| = |\pi|$, we let $\mathbb{P}_{\rho_0}^{\eta,\theta}(\pi, \mathcal{C}) = 1$. Otherwise, letting δ be the $(|\rho_0|+1)$-th transition of π, and ξ be the strategy η or θ according if $\rho_0 \in \mathsf{FPlays}_{\mathsf{Min}}$

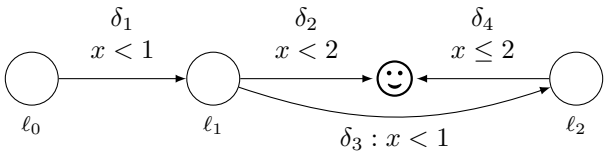


Figure 5.1 – A (one-player) WTG with a single clock x and all weights equal to 0.

or $\rho_0 \in \mathsf{FPlays}_{\mathsf{Max}}$, we let

$$\mathbb{P}^{\eta,\theta}_{\rho_0}(\pi, \mathcal{C}) = \int_{I(\rho_0,\delta)} \xi_\Delta(\rho_0)(\delta) \times \mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi, \mathcal{C}) \, \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0, \delta)(t) \tag{5.1}$$

As in the previous definition, when $\mathcal{C} = \mathbb{R}^{|\pi|}_{\geq 0}$, we denote by $\mathbb{P}^{\eta,\theta}_{\rho_0}(\pi)$ the probability of $\mathsf{Cyl}_{\rho_0}(\pi)$.

5.1.1. A restriction over possible stochastic strategies

For modelling purposes, the authors of [Ber+14] enforce that probability distributions on delays in $\eta_{\mathbb{R}^+}$ do not forbid any delays of the interval $I(\rho_0, \delta)$ of possible delays, thus ruling out singular distributions like Dirac ones that would consider taking a single possible delay (like deterministic strategies do). More formally, they require $\eta_{\mathbb{R}^+}(\rho_0, \delta)$ to be absolutely continuous (i.e. equivalent to the Lebesgue measure) on the interval $I(\rho_0, \delta)$. We claim that even with this assumption, the previous definition of the probability of a cylinder may not be well-founded, as demonstrated in the following example.

Example 5.2. We consider the WTG of Figure 5.1, where only Min plays and all weights equal 0 (so we can see it as a stochastic timed automaton), and the memoryless strategy η that is (partially) defined as follows. We let A be a non-Lebesgue-measurable subset of $[0, 1)$. We denote $\overline{A}$ as the complement set $[0, 1) \setminus A$ and χ_A the characteristic function of the set A.

We start by defining the delays to match as closely as possible the setting of [Ber+14] here. For delays in ℓ_0 and ℓ_2, we consider uniform probability distributions on the appropriate intervals. In ℓ_1, for all $t_1 \in [0, 1)$, we let $\eta_{\mathbb{R}^+}((\ell_1, t_1), \delta_2)$ be the uniform distribution on $[0, 2 - t_1]$ if $t_1 \in A$, and the truncated exponential distribution on $[0, 2 - t_1]$ with parameter $\lambda = 1$ otherwise. For the choice of transitions from ℓ_1 (the only place where there is a choice), for all $t_1 \in [0, 1)$, we let

$$\eta_\Delta(\ell_1, t_1)(\delta_2) = \begin{cases} (3 - t_1)/(4 - 2t_1) = f(t_1) & \text{if } t_1 \in A; \\ (1 + e^{-(1-t_1)} - 2e^{-(2-t_1)})/(2 - 2e^{-(2-t_1)}) = g(t_1) & \text{if } t_1 \in \overline{A}. \end{cases}$$

In the setting of [Ber+14], the strategy in ℓ_1 can be obtained by first choosing a delay similar to ours, and then choosing transition δ_2 with either probability $\frac{1}{2}$ or 1 depending on whether transition δ_3 is fireable after letting the chosen delay elapse: authors of [Ber+14] would describe this by putting weight 1 on both transitions δ_2 and δ_3 in the stochastic timed automaton. The intricate formulas above for the transitions are thus simply a way to mimic their setting in ours.

Let us try to compute the probability of the cylinder $\mathsf{Cyl}_{\ell_0,0}(\delta_1\delta_2)$: its probability with respect to $\mathbb{P}^{\eta}_{\ell_0,0}$ is

$$\mathbb{P}^{\eta}_{\ell_0,0}(\delta_1\delta_2) = \int_{I((\ell_0,0),\delta_1)} \eta_{\Delta}(\ell_0,0)(\delta_1) \times \mathbb{P}^{\eta}_{(\ell_0,0)[\delta_1,t_1]}(\delta_2)\, \mathrm{d}\eta_{\mathbb{R}^+}((\ell_0,0),\delta_1)(t_1)\,.$$

Moreover, since δ_1 is the only transition from ℓ_0, then $\eta_{\Delta}(\ell_0,0)(\delta_1) = 1$ and

$$\mathbb{P}^{\eta}_{\ell_0,0}(\delta_1\delta_2) = \int_0^1 \mathbb{P}^{\eta}_{(\ell_0,0)[\delta_1,t_1]}(\delta_2)\, \mathrm{d}t_1$$

that requires $h\colon t_1 \mapsto \mathbb{P}^{\eta}_{(\ell_0,0)[\delta_1,t_1]}(\delta_2)$ to be a measurable function on $[0,1]$ to be well-defined. For $t_1 \in [0,1) \cap A$,

$$\begin{aligned} h(t_1) &= \int_{I((\ell_1,t_1),\delta_1)} \eta_{\Delta}(\ell_1,t_1)(\delta_2) \times 1\, \mathrm{d}\eta_{\mathbb{R}^+}((\ell_1,t_1),\delta_2)(t_2) \\ &= \int_0^{2-t_1} f(t_1)\frac{\mathrm{d}t_2}{2-t_1} = f(t_1)\,. \end{aligned}$$

Similarly, if $t_1 \in [0,1) \cap \overline{A}$, then $h(t_1) = g(t_1)$.

Functions f and g are measurable and never match over $[0,1)$. Thus, would h be measurable, so would be $(h-g)/(f-g)$ that is equal to the characteristic function of A: this contradicts the non-measurability of A. And thus, it is not possible to define the probability $\mathbb{P}^{\eta}_{\ell_0,0}(\delta_1\delta_2)$. ⌟

From this example, we see the importance to moreover enforce that the distributions $\eta_{\Delta}(\rho_0)$ and $\eta_{\mathbb{R}^+}(\rho_0,\delta)$ are "measurable w.r.t. the sequence of delays along the play ρ_0". This is easy to define for the transition part. For delays, we know that absolutely continuous distribution, as [Ber+14] requires, does not suffice to guarantee measurability. Moreover, since we want deterministic strategies to be a subset of stochastic strategies, we must be able to choose delays by using Dirac distributions, and, by extension discrete, distributions.

We restrict ourselves to discrete distributions whose *cumulative distribution functions* (CDF) are a finite linear combination of translated Heaviside functions. Precisely, we let the Heaviside function H denote the mapping from $\mathbb{R}$ to $[0,1]$ such that $H(t) = 0$ if $t < 0$ and $H(t) = 1$ otherwise. Recall that it is the CDF of

the Dirac distribution choosing $t = 0$. This results in the following hypothesis:

Hypothesis 1. *A (stochastic) strategy ξ satisfies this hypothesis if for all finite plays $\rho_0 = (\ell_0, \nu_0)[\delta_0, t_0] \cdots [\delta_{k-1}, t_{k-1}]$ and transitions δ,*

1. *the mapping $(t_0, \ldots, t_{k-1}) \mapsto \xi_\Delta(\rho_0)(\delta)$ is Lebesgue-measurable; and*
2. *the probability distribution $\xi_{\mathbb{R}^+}(\rho_0, \delta)$ (of the random variable t) is described by a CDF of the form*

$$t \mapsto G(\rho_0, \delta)(t) + \sum_{i=0}^{n} \alpha_i(\rho_0, \delta) H(t - a_i(\rho_0, \delta))$$

where $G(\rho_0, \delta)$ is an absolutely continuous function, $\alpha_i(\rho_0, \delta) \in [0, 1]$ and $a_i(\rho_0, \delta) \in \mathbb{R}_{\geq 0}$, for all $i \in \{0, \ldots, n\}$.

Moreover, the mappings $(t_0, \ldots, t_{k-1}) \mapsto \alpha_i(\rho_0, \delta)$, $(t_0, \ldots, t_{k-1}) \mapsto a_i(\rho_0, \delta)$, and $(t_0, \ldots, t_{k-1}, t) \mapsto G(\rho_0, \delta)(t)$ must be Lebesgue-measurable.

Under this hypothesis, we prove that the integral in (5.1) is always well-defined:

Lemma 5.3. *For all finite plays $\rho_0 = (\ell_0, \nu_0)[\delta_0, t_0] \cdots [\delta_{k-1}, t_{k-1}]$, finite paths π starting in ℓ_0 (such that $|\pi| \geq k$), and Lebesgue-measurable sets $\mathcal{C}$ of $\mathbb{R}_{\geq 0}^{|\pi|}$,*

1. *the mapping $(t_0, \ldots, t_{k-1}) \mapsto \mathbb{P}_{\rho_0}^{\eta,\theta}(\pi, \mathcal{C})$ is Lebesgue-measurable;*
2. *$0 \leq \mathbb{P}_{\rho_0}^{\eta,\theta}(\pi, \mathcal{C}) \leq 1$.*

Proof. We reason by induction on the length of ρ_0. If $|\rho_0| = |\pi|$, then the mapping $(t_0, \ldots, t_{k-1}) \mapsto \mathbb{P}_{\rho_0}^{\eta,\theta}(\pi, \mathcal{C})$ is equal to 1 if $\pi = \delta_0 \delta_1 \cdots \delta_{k-1}$ and $(t_0, \ldots, t_{k-1}) \in \mathcal{C}$ (otherwise it is equal to 0). In particular, in case $\pi = \delta_0 \delta_1 \cdots \delta_{k-1}$, this mapping is equal to the characteristic function of $\mathcal{C}$ (regarding delays) that is Lebesgue measurable by definition of a cylinder. Thus, $(t_0, \ldots, t_{k-1}) \mapsto \mathbb{P}_{\rho_0}^{\eta,\theta}(\pi, \mathcal{C})$ is Lebesgue-measurable. Otherwise, we suppose that the property holds for all finite plays of length $k + 1 \leq |\pi|$ and we consider a finite play ρ_0 of length k. Letting ξ be the strategy η or θ according as $\rho_0 \in \mathsf{FPlays}_{\mathsf{Min}}$, or $\rho_0 \in \mathsf{FPlays}_{\mathsf{Max}}$, then by using the notations of (5.1) and Hypothesis 1-2, we can decompose $\mathbb{P}_{\rho_0}^{\eta,\theta}(\pi, \mathcal{C})$ as $\xi_\Delta(\rho_0)(\delta)\,(A + B)$ where

$$A = \int_{I(\rho_0,\delta) \cap \mathcal{C}^k} \mathbb{P}_{\rho_0[\delta,t]}^{\eta,\theta}(\pi, \mathcal{C}) \times G(\rho_0, \delta)(t)\, \mathrm{d}t \tag{5.2}$$

and

$$B = \sum_{i=0}^{\infty} \alpha_i(\rho_0, \delta) \times \mathbb{P}_{\rho_0[\delta, a_i(\rho_0,\delta)]}^{\eta,\theta}(\pi, \mathcal{C})\,. \tag{5.3}$$

We start by studying (5.2). In particular, by induction hypothesis applied to the finite play $\rho_0[\delta, t]$, we know that the mapping $(t_0, \dots, t_{k-1}, t) \mapsto \mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi, \mathcal{C})$ is Lebesgue-measurable. Therefore, the mapping

$$(t_0, \dots, t_{k-1}, t) \mapsto \mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi, \mathcal{C}) \times G(\rho_0, \delta)(t)$$

is Lebesgue-measurable, non-negative and upper-bounded by 1. Since the interval $I(\rho_0, \delta)$ is bounded, the integral exists. Moreover, by Fubini Theorem[1], the resulting integral is Lebesgue-measurable w.r.t. the delays in ρ, i.e. (5.2) is Lebesgue-measurable w.r.t. $(t_0, \dots, t_{k-1})$.

Now, we focus on (5.3). By composition of the Lebesgue-measurable mappings $(t_0, \dots, t_{k-1}, t) \mapsto \mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi, \mathcal{C})$ (by induction hypothesis applied on $\rho_0[\delta, t]$) and $(t_0, \dots, t_{k-1}) \mapsto a_i(\rho_0, \delta)$ (by Hypothesis 1-2), the mapping $(t_0, \dots, t_{k-1}) \mapsto \mathbb{P}^{\eta,\theta}_{\rho_0[\delta,a_i(\rho,\delta)]}(\pi, \mathcal{C})$ is Lebesgue-measurable. Then, thanks to Hypothesis 1-2, the countable sum of (5.3) absolutely converges. Thus, by countable sum of Lebesgue-measurable functions, (5.3) is Lebesgue-measurable w.r.t. $(t_0, \dots, t_{k-1})$.

Finally, the mapping $(t_0, \dots, t_{k-1}) \mapsto \xi_\Delta(\rho_0)(\delta)$ is Lebesgue-measurable, by using Hypothesis 1-1. Thus, $(t_0, \dots, t_{k-1}) \mapsto \mathbb{P}^{\eta,\theta}_{\rho_0}(\pi, \mathcal{C})$ is Lebesgue-measurable as a linear combination of Lebesgue-measurable functions. Moreover, by using the fact that $\xi_\Delta(\rho_0)$ is a probability distribution and the induction hypothesis bounding the probabilities between 0 and 1, we obtain, from the definition,

$$0 \leq \mathbb{P}^{\eta,\theta}_{\rho_0}(\pi, \mathcal{C}) \leq \int_{I(\rho,\delta)} \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0, \delta)(t) = 1$$

since $\xi_{\mathbb{R}^+}(\rho_0, \delta)$ is a probability measure. □

In the following, we let $\mathsf{Strat}_{\mathsf{Min}}$ and $\mathsf{Strat}_{\mathsf{Max}}$ be the sets of (stochastic) strategies satisfying Hypothesis 1, for both players. We let $\mathsf{mStrat}_{\mathsf{Min}}$ and $\mathsf{mStrat}_{\mathsf{Max}}$ be the respective subsets of memoryless strategies. We let $\mathsf{sdStrat}_{\mathsf{Min}}$ and $\mathsf{sdStrat}_{\mathsf{Max}}$ be the respective subsets of deterministic strategies of both players that moreover satisfy Hypothesis 1: we later call them the *smooth* deterministic strategies.

Remark. By modifying Example 5.2 (by using two different constants instead functions f and g), it can be shown that imposing the smoothness of deterministic strategies does not come without a loss of generality. However, in divergent WTGs, we will show in Section 6.3 that there exists, for both players, almost-optimal smooth deterministic strategies with respect to the deterministic value. ◁

[1]The dependency on ρ_0 in the interval $I(\rho_0, \delta) \cap \mathcal{C}^k$ of integration can be replaced by the full space $[0, M]$ since clocks are bounded by M, and the characteristic function of $t \in I(\rho_0, \delta) \cap \mathcal{C}^k$ which is measurable w.r.t. $(t_0, \dots, t_{k-1}, t)$.

5.1.2. Construction of the probability measure

Under Hypothesis 1, we prove that $\mathbb{P}^{\eta,\theta}_{\rho_0}$ can be extended as a probability distribution over the set $\mathsf{Plays}^{\eta,\theta}_{\rho_0}$ of maximal plays starting in the last configuration of ρ_0 and conforming to η and θ, equipped with the σ-algebra Σ_{ρ_0} generated by the cylinders $\mathsf{Cyl}_{\rho_0}(\pi, \mathcal{C})$, following a similar proof as [Ber+14, Proposition 3.2].

Proposition 5.4. *If η and θ are strategies satisfying Hypothesis 1, then, for all finite plays ρ_0, there exists a probability measure $\mathbb{P}^{\eta,\theta}_{\rho_0}$ over $(\mathsf{Plays}^{\eta,\theta}_{\rho_0}, \Sigma_{\rho_0})$.*

The rest of this section consists of the proof of this proposition. To do it, we follow a similar proof schema as in [Ber+14, Appendix A]. We start by proving that $\mathbb{P}^{\eta,\theta}_{\rho_0}$ is a probability measure over the algebra $(\mathsf{Plays}^{\eta,\theta}_{\rho_0}, \mathcal{A}_{\rho_0})$ generated by cylinders defined from paths (then closed by a finite union and complement). Then, we use Carathéodory's extension theorem (see Theorem 5.7) to conclude that $\mathbb{P}^{\eta,\theta}_{\rho_0}$ is a probability measure over the σ-algebra generated by all maximal paths.

To define the probability $\mathbb{P}^{\eta,\theta}_{\rho_0}$ over elements of the algebra $\mathcal{A}_{\rho_0}$, we consider an inductive definition of $\mathcal{A}_{\rho_0}$. We remark that $\mathcal{A}_{\rho_0} = \bigcup_n \mathcal{A}^n_{\rho_0}$ where $\mathcal{A}^n_{\rho_0}$ is the algebra generated by all cylinders defined with a path of length at most[2] n from ρ_0. More precisely, we let $\mathsf{FPaths}^n_{\rho_0}$ be the set of paths from ρ_0 that are either non-maximal and of length n or maximal and of length at most n. Then, $\mathcal{A}^n_{\rho_0}$ is generated by all cylinders of the form $\mathsf{Cyl}_{\rho_0}(\pi, \mathcal{C})$ with $\pi \in \mathsf{FPaths}^n_{\rho_0}$ and $\mathcal{C}$ a Lebesgue-measurable set of $\mathbb{R}^{|\pi|}_{\geq 0}$. Elements of $\mathcal{A}^n_{\rho_0}$ are finite unions of such disjoint cylinders and we let their probability be the sum of the probability of each cylinder.

Lemma 5.5. *Let $n \in \mathbb{N}$ and ρ_0 be a finite play, then $\mathbb{P}^{\eta,\theta}_{\rho_0}$ is a measure[3] over the algebra $(\mathsf{Plays}^{\eta,\theta}_{\rho_0}, \mathcal{A}^n_{\rho_0})$.*

Proof. By definition and Lemma 5.3-2, $\mathbb{P}^{\eta,\theta}_{\rho_0}$ is additive, non-negative and finite over $(\mathsf{Plays}^{\eta,\theta}_{\rho_0}, \mathcal{A}^n_{\rho_0})$. In particular, by [KSK76], $\mathbb{P}^{\eta,\theta}_{\rho_0}$ is σ-additive, if and only if for all sequences $(A_i)_i$ of elements of $\mathcal{A}^n_{\rho_0}$ such that $A_0 \subseteq A_1 \subseteq \cdots$ and $A = \bigcup_i A_i \in \mathcal{A}^n_{\rho_0}$, we have $\mathbb{P}^{\eta,\theta}_{\rho_0}(A) = \lim_i \mathbb{P}^{\eta,\theta}_{\rho_0}(A_i)$. So, let $(A_i)_i$ be such a sequence. Without loss of generality, we can suppose that each A_i is generated by the same path $\pi = \delta_1 \cdots \delta_k$ with $k \leq n$ (since π may be a maximal path of a length less than n), otherwise it suffices to intersect each A_i with $\mathsf{Cyl}_{\rho_0}(\pi)$. In particular, letting $(\mathcal{C}^j_i)_j$ be the sequence of disjoint sets of constraints that appear in cylinders of A_i, we

[2] In [Ber+14], authors defined this set by cylinders with a length equal to n. Here, we need to relax this hypothesis since maximal paths may be finite. Indeed if we do not keep maximal paths with a length less than n, we will lose some mass of probability.

[3] More precisely, we can prove that $\mathbb{P}^{\eta,\theta}_{\rho_0}$ is a probability measure over $(\mathsf{Plays}^{\eta,\theta}_{\rho_0}, \mathcal{A}^n_{\rho_0})$. However, since this property is not useful in the following, we do not prove that $\mathbb{P}^{\eta,\theta}_{\rho}(\mathsf{Plays}^{\eta,\theta}_{\rho_0}) = 1$ at this step of the proof.

have $A_i = \bigsqcup_{j=0}^{m_i} \mathsf{Cyl}_{\rho_0}(\pi, \mathcal{C}_i^j)$ where m_i is the number of cylinders in A_i. Now, by definition of probabilities (and its additivity on cylinders), we have

$$\mathbb{P}^{\eta,\theta}_{\rho_0}(A_i) = \sum_{j=0}^{m_i} \mathbb{P}^{\eta,\theta}_{\rho_0}(\pi, \mathcal{C}_i^j)\,.$$

In particular, $\mathbb{P}^{\eta,\theta}_{\rho_0}(A_i)$ is equal to

$$\sum_{j=0}^{m_i} \int_{I(\rho_0,\delta_1)} \cdots \int_{I(\rho_0[\delta_1,t_1]\cdots[\delta_{k-1},t_{k-1}],\delta_k)} \xi_\Delta(\rho_0)(\delta_1)\cdots\xi_\Delta(\rho_0[\delta_1,t_1]\cdots[\delta_{k-1},t_{k-1}])(\delta_k) \times \mathbb{P}^{\eta,\theta}_{\rho_0[\delta_1,t_1]\cdots[\delta_k,t_k]}(\pi,\mathcal{C}_i^j) \\ \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0[\delta_1,t_1]\cdots[\delta_{k-1},t_{k-1}],\delta_k)(t_k)\cdots\ \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta_1)(t_1)\,.$$

Moreover, by letting χ_Y be the characteristic function of the set Y, we remark that

$$\mathbb{P}^{\eta,\theta}_{\rho_0[\delta_1,t_1]\cdots[\delta_k,t_k]}(\pi,\mathcal{C}_i^j) = \chi_{\mathsf{Cyl}(\pi,\mathcal{C}_i^j)}(\rho_0[\delta_1,t_1]\cdots[\delta_k,t_k])$$

(by definition of the probability) and $\mathbb{P}^{\eta,\theta}_{\rho_0}(A_i)$ is equal to

$$\sum_{j=0}^{m_i} \int_{I(\rho_0,\delta_1)} \cdots \int_{I(\rho_0[\delta_1,t_1]\cdots[\delta_{k-1},t_{k-1}],\delta_k)} \xi_\Delta(\rho_0)(\delta_1)\cdots\xi_\Delta(\rho_0[\delta_1,t_1]\cdots[\delta_{k-1},t_{k-1}])(\delta_k) \times \chi_{\mathsf{Cyl}(\pi,\mathcal{C}_i^j)}(\rho_0[\delta_1,t_1]\cdots[\delta_k,t_k]) \\ \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0[\delta_1,t_1]\cdots[\delta_{k-1},t_{k-1}],\delta_k)(t_k)\cdots\ \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta_1)(t_1)\,.$$

In particular by linearity of the integral, we obtain that $\mathbb{P}^{\eta,\theta}_{\rho_0}(A_i)$ is equal to

$$\int_{I(\rho_0,\delta_1)} \cdots \int_{I(\rho_0[\delta_1,t_1]\cdots[\delta_{k-1},t_{k-1}],\delta_k)} \xi_\Delta(\rho_0)(\delta_1)\cdots\xi_\Delta(\rho_0[\delta_1,t_1]\cdots[\delta_{k-1},t_{k-1}])(\delta_k) \times \chi_{A_i}(\rho_0[\delta_1,t_1]\cdots[\delta_k,t_k]) \\ \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0[\delta_1,t_1]\cdots[\delta_{k-1},t_{k-1}],\delta_k)(t_k)\cdots\ \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta_1)(t_1)$$

since $\sum_{j=0}^{m_i} \chi_{\mathsf{Cyl}(\pi,\mathcal{C}_i^j)} = \chi_{A_i}$ (as $\mathcal{C}_i^j$ are disjoint sets). Moreover, since, for all i, χ_{A_i} is

bounded by 1, by dominated convergence, we deduce that $\lim_i \mathbb{P}^{\eta,\theta}_{\rho_0}(A_i)$ is equal to

$$\int_{I(\rho_0,\delta_1)} \cdots \int_{I(\rho_0[\delta_1,t_1]\cdots[\delta_{k-1},t_{k-1}],\delta_k)} \xi_\Delta(\rho_0)(\delta_1)\cdots\xi_\Delta(\rho_0[\delta_1,t_1]\cdots[\delta_{k-1},t_{k-1}])(\delta_k) \times \left(\lim_i \chi_{A_i}(\rho_0[\delta_1,t_1]\cdots[\delta_k,t_k])\right) \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0[\delta_1,t_1]\cdots[\delta_{k-1},t_{k-1}],\delta_k)(t_k)\cdots \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta_1)(t_1)\,.$$

Moreover, since $A = \bigcup_i A_i$, then $\lim_i \chi_{A_i} = \chi_A$. Thus, we deduce that $\lim_i \mathbb{P}^{\eta,\theta}_{\rho_0}(A_i)$ is equal to

$$\int_{I(\rho_0,\delta_1)} \cdots \int_{I(\rho_0[\delta_1,t_1]\cdots[\delta_{k-1},t_{k-1}],\delta_k)} \xi_\Delta(\rho_0)(\delta_1)\cdots\xi_\Delta(\rho_0[\delta_1,t_1]\cdots[\delta_{k-1},t_{k-1}])(\delta_k) \times \chi_A(\rho_0[\delta_1,t_1]\cdots[\delta_k,t_k]) \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0[\delta_1,t_1]\cdots[\delta_{k-1},t_{k-1}],\delta_k)(t_k)\cdots \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta_1)(t_1)\,.$$

Thus, we conclude that $\lim_i \mathbb{P}^{\eta,\theta}_{\rho_0}(A_i) = \mathbb{P}^{\eta,\theta}_{\rho_0}(A)$ by the same way as for $\mathbb{P}^{\eta,\theta}_{\rho_0}(A_i)$, since A is a finite union of cylinders (as $A \in \mathcal{A}^n_{\rho_0}$). □

To prove that $\mathbb{P}^{\eta,\theta}_{\rho_0}$ is a probability measure on the algebra $(\mathsf{Plays}^{\eta,\theta}_{\rho_0}, \mathcal{A}_{\rho_0})$, we use another property over $\mathcal{A}^n_{\rho_0}$: for all $n \in \mathbb{N}$, $\mathcal{A}^n_{\rho_0} \subseteq \mathcal{A}^{n+1}_{\rho_0}$. Let A be an element of $\mathcal{A}^n_{\rho_0}$: either A is a union of cylinders generated by maximal paths (of length at most n) and $A \in \mathcal{A}^{n+1}_{\rho_0}$, or there exists a cylinder of A generated by a non-maximal path π of length n. Since π is non-maximal, it can be extended by a transition δ. In particular, for all constraints $\mathcal{C} \subseteq \mathbb{R}^n_{\geq 0}$, $\mathsf{Cyl}_{\rho_0}(\pi,\mathcal{C}) = \bigcup_\delta \mathsf{Cyl}_{\rho_0}(\pi\delta, \mathcal{C}\times\mathbb{R}_{\geq 0})$. Thus, we conclude that $A \in \mathcal{A}^{n+1}_{\rho_0}$ by definition of $\mathcal{A}^{n+1}_{\rho_0}$.

Lemma 5.6. *Let ρ_0 be a finite play, then $\mathbb{P}^{\eta,\theta}_{\rho_0}$ is a probability measure over $(\mathsf{Plays}^{\eta,\theta}_{\rho_0}, \mathcal{A}_{\rho_0})$.*

Proof. We start by proving that $\mathbb{P}^{\eta,\theta}_{\rho_0}$ is a measure on $\mathcal{A}_{\rho_0}$. Since $\mathbb{P}^{\eta,\theta}_{\rho_0}$ is additive (as $\mathcal{A}^n_{\rho_0} \subseteq \mathcal{A}^{n+1}_{\rho_0}$), non-negative and finite over $\mathcal{A}^n_{\rho_0}$ (by Lemma 5.3-2), the σ-additivity can be obtained by showing that $\lim_n \mathbb{P}^{\eta,\theta}_{\rho_0}(B_n) = 0$ for all sequences $(B_n)_n$ of elements of $\mathcal{A}_{\rho_0}$ such that $B_0 \supseteq B_1 \supseteq \cdots$ and $\bigcap_n B_n = \emptyset$ (by [KSK76]). Let $(B_n)_n$ be such a sequence. Without loss of generality, we suppose that for every n, $B_n \in \mathcal{A}^n_{\rho_0}$ (otherwise, all B_n are in a fixed $\mathcal{A}^{n_0}_{\rho_0}$ and Lemma 5.5 allows us to conclude).

To easily write the limit of probabilities of B_n, we begin by proving that all B_n are generated by a unique cylinder where paths follow the same prefix. Indeed, by [KSK76, Lemma 3.3], we know that if $\lim_n \mathbb{P}^{\eta,\theta}_{\rho_0}(B_n) > 0$, then there exists a sequence of cylinders $(\mathsf{Cyl}_n)_n$ such that for all n, $\mathsf{Cyl}_n \subseteq B_n$ and $\mathsf{Cyl}_{n+1} \subseteq \mathsf{Cyl}_n$. In other words, we can suppose, without loss of generality, that there exists a maximal

path π such that $B_n = \bigsqcup_{j=0}^{m_n} \mathsf{Cyl}_{\rho_0}(\pi_n, \mathcal{C}_n^j)$ where $\pi_n = \delta_1 \cdots \delta_n$ is the prefix of π of length n, $(\mathcal{C}_n^j)_j$ is a sequence of constraints for π_n and m_n the finite number of cylinders of B_n (since $B_n \in \mathcal{A}_{\rho_0}^n$). Otherwise, when $\mathbb{P}_{\rho_0}^{\eta,\theta}$ is not a measure on $\mathcal{A}_{\rho_0}$, such a sequence suffices to prove it. In particular, $B_n = \mathsf{Cyl}_{\rho_0}(\pi_n, \mathcal{C}_n)$ where $\mathcal{C}_n = \bigsqcup_{j=0}^{m_n} \mathcal{C}_n^j$ (which is a Lebesgue-measurable set of $\mathbb{R}^n$ since each $\mathcal{C}_n^j$ is). Moreover, with the same reasoning as in the previous lemma and by letting χ_Y the characteristic function of the set Y, $\mathbb{P}_{\rho_0}^{\eta,\theta}(B_n)$ is equal to

$$\int_{I(\rho_0,\delta_1)} \xi_\Delta(\rho_0)(\delta_1) \times \int_{I(\rho_0[\delta_1,t_1],\delta_2)} \xi_\Delta(\rho_0[\delta_1,t_1])(\delta_2) \cdots$$
$$\int_{I(\rho_0[\delta_1,t_1]\cdots[\delta_{n-1},t_{n-1}],\delta_n)} \xi_\Delta(\rho_0[\delta_1,t_1]\cdots[\delta_{n-1},t_{n-1}])(\delta_n) \times \chi_{B_n}(\rho_0[\delta_1,t_1]\cdots[\delta_n,t_n])$$
$$\mathrm{d}\xi_{\mathbb{R}^+}(\rho_0[\delta_1,t_1]\cdots[\delta_{n-1},t_{n-1}],\delta_n)(t_n)\cdots \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0[\delta_1,t_1],\delta_2)(t_2)\, \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta_1)(t_1).$$

Now, to prove that the limit of probabilities is equal to 0, we will prove that there exists $i \in \mathbb{N}$ such that $B_i = \emptyset$ (more precisely, $\mathcal{C}_i = \emptyset$) from the hypothesis of $\bigcap_n B_n = \emptyset$. For all $i > 0$, we let $p_i \colon \mathbb{R}_{\geq 0}^{i+1} \mapsto \mathbb{R}_{\geq 0}^i$ be the continuous (for product topologies) projection over the i first components of a constraint in $\mathbb{R}_{\geq 0}^{i+1}$. For all $i < n$, we can define the constraint for the i first steps by induction along $\mathcal{C}_n$ such that $\mathcal{C}_n^i = p_i(\mathcal{C}_n^{i+1})$. Moreover, since $B_{n+1} \subseteq B_n$ and all sets follow the same prefix, we have for all $i \leq n$ that $\mathcal{C}_{n+1}^i \subseteq \mathcal{C}_n^i$. In particular, for some fixed i, the sequence $(\mathcal{C}_n^i)_n$ decreases, thus it converges to $\mathcal{C}^i \subseteq \mathbb{R}_{\geq 0}^i$. Now, we want to prove that there exists $i > 0$ such that $\mathcal{C}^i = \emptyset$. To do so, we reason by contradiction, and we suppose that for all i, $\mathcal{C}^i \neq \emptyset$, and we want to define a play where its delays are in $\bigcap_i \mathcal{C}^i$. To define this play, we define, by induction on i, a sequence of delays that satisfies $\mathcal{C}^i$: for all i if we have a sequence of delays $(t_i)_i$ that satisfies $\mathcal{C}^i$ (for all i), then since $\mathcal{C}^i = p_i(\mathcal{C}^{i+1})$ (by continuity of p_i), there exists t_{i+1} such that $(t_1, \ldots, t_i, t_{i+1})$ satisfying $\mathcal{C}^{i+1}$ (with $(t_1, \ldots, t_i)$ satisfying $\mathcal{C}^i$). This sequence of delays defines a play $\rho \in \bigcap_i \mathsf{Cyl}_{\rho_0}(\pi_i, \mathcal{C}^i) = \bigcap_i B_i$. However, by hypothesis on $(B_n)_n$, $\bigcap_i B_i = \emptyset$, thus we obtain a contradiction and there exists i such that $\mathcal{C}^i = \emptyset$, i.e. $B_i = \emptyset$.

To conclude about the σ-additivity, we prove that $\lim_n \mathbb{P}_{\rho_0}^{\eta,\theta}(B_n) = 0$. We let B_i be the subset such that $B_i = \emptyset$, and for all $n \geq i$, by letting χ_Y the characteristic function of the set Y, $\mathbb{P}_{\rho_0}^{\eta,\theta}(B_n)$ is equal to

$$\int_{I(\rho_0,\delta_1)} \xi_\Delta(\rho_0)(\delta_1) \times \int_{I(\rho_0[\delta_1,t_1],\delta_2)} \xi_\Delta(\rho_0[\delta_1,t_1])(\delta_2) \cdots$$
$$\int_{I(\rho_0[\delta_1,t_1]\cdots[\delta_{n-1},t_{n-1}],\delta_n)} \xi_\Delta(\rho_0[\delta_1,t_1]\cdots[\delta_{n-1},t_{n-1}])(\delta_n) \times \chi_{B_n}(\rho_0[\delta_1,t_1]\cdots[\delta_n,t_n])$$
$$\mathrm{d}\xi_{\mathbb{R}^+}(\rho_0[\delta_1,t_1]\cdots[\delta_{n-1},t_{n-1}],\delta_n)(t_n)\cdots \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0[\delta_1,t_1],\delta_2)(t_2)\, \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta_1)(t_1).$$

In particular, since $i \leq n$, by restricting the probability to the i first components, $\mathbb{P}^{\eta,\theta}_{\rho_0}(B_n)$ is bounded by

$$\int_{I(\rho_0,\delta_1)} \xi_\Delta(\rho_0)(\delta_1) \times \int_{I(\rho_0[\delta_1,t_1],\delta_2)} \xi_\Delta(\rho_0[\delta_1,t_1])(\delta_2) \cdots$$
$$\int_{I(\rho_0[\delta_1,t_1]\cdots[\delta_{i-1},t_{i-1}],\delta_i)} \xi_\Delta(\rho_0[\delta_1,t_1]\cdots[\delta_{i-1},t_{i-1}])(\delta_i) \times \chi_{B_i}(\rho_0[\delta_1,t_1]\cdots[\delta_i,t_i])$$
$$\mathrm{d}\xi_{\mathbb{R}^+}(\rho_0[\delta_1,t_1]\cdots[\delta_{i-1},t_{i-1}],\delta_i)(t_i)\cdots\ \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0[\delta_1,t_1],\delta_2)(t_2)\ \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta_1)(t_1).$$

Since $\chi_{B_i} \leq 1$, we obtain that $\lim_n \mathbb{P}^{\eta,\theta}_{\rho_0}(B_n)$ is bounded by

$$\int_{I(\rho_0,\delta_1)} \xi_\Delta(\rho_0)(\delta_1) \times \int_{I(\rho_0[\delta_1,t_1],\delta_2)} \xi_\Delta(\rho_0[\delta_1,t_1])(\delta_2) \cdots$$
$$\int_{I(\rho_0[\delta_1,t_1]\cdots[\delta_{i-1},t_{i-1}],\delta_n)} \xi_\Delta(\rho_0[\delta_1,t_1]\cdots[\delta_{i-1},t_{i-1}])(\delta_i) \times \lim_n \chi_{B_n}(\rho_0[\delta_1,t_1]\cdots[\delta_i,t_i])$$
$$\mathrm{d}\xi_{\mathbb{R}^+}(\rho_0[\delta_1,t_1]\cdots[\delta_{i-1},t_{i-1}],\delta_i)(t_i)\cdots\ \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0[\delta_1,t_1],\delta_2)(t_2)\ \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta_1)(t_1)$$

by dominated convergence. Now, since the sequence $(B_n)_n$ decreases, we know that for all $n \geq i$, $B_n = \emptyset$. Thus, $\lim_n \chi_{B_n}(\rho_0[\delta_1,t_1]\cdots[\delta_i,t_i]) = 0$, and we deduce that $\lim_n \mathbb{P}^{\eta,\theta}_{\rho_0}(B_n) = 0$.

Finally, we conclude the proof by proving that $\mathbb{P}^{\eta,\theta}_{\rho_0}(\mathsf{Plays}^{\eta,\theta}_{\rho_0}) = 1$. To do that, we remark that $\mathcal{A}^0_{\rho_0} = (\mathsf{Plays}^{\eta,\theta}_{\rho_0}, \mathsf{Cyl}_{\rho_0}(\pi_0))$, i.e. $\mathsf{Plays}^{\eta,\theta}_{\rho_0} = \mathsf{Cyl}_{\rho_0}(\pi_0)$ where ρ_0 follows π_0. Thus, $\mathsf{Cyl}_{\rho_0}(\pi_0) \neq \emptyset$, and by definitions, we obtain that

$$\mathbb{P}^{\eta,\theta}_{\rho_0}(\mathsf{Plays}^{\eta,\theta}_{\rho_0}) = \mathbb{P}^{\eta,\theta}_{\rho_0}(\pi_0) = 1. \qquad \square$$

Finally, we conclude the proof of Proposition 5.4 by using the following theorem.

Theorem 5.7 (Carathéodory's extension theorem)**.** *Let S be a set, and μ be a σ-finite measure defined on an algebra $A \subseteq 2^S$. Then, μ can be extended in a unique manner to the σ-algebra generated by A.*

Indeed, to conclude this proof, we apply Theorem 5.7 to the set $S = \mathsf{Plays}^{\eta,\theta}_{\rho_0}$, the algebra $A = \mathcal{A}_{\rho_0}$, and the σ-finite measure $\mu = \mathbb{P}^{\eta,\theta}_{\rho_0}$ on $\mathcal{A}_{\rho_0}$ (by Lemma 5.6). Hence, there is a unique extension of $\mathbb{P}^{\eta,\theta}_{\rho_0}$ on the σ-algebra generated by the cylinders which is a probability measure over maximal plays.

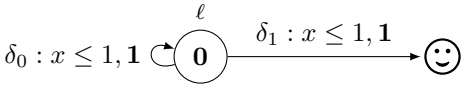


Figure 5.2 – A WTG where Min has a strategy reaching the target with probability 1 but with an expected weight equal to $+\infty$.

5.2. Expected payoff of plays

We now define the expected weight over the maximal plays starting from the last configuration of a finite play ρ_0. We denote it by $\mathbb{E}^{\eta,\theta}_{\rho_0}$ instead of $\mathbb{E}^{\eta,\theta}_{\rho_0}(\mathsf{wt})$ since we only consider the expectation of the weight wt:

$$\mathbb{E}^{\eta,\theta}_{\rho_0} = \int_{\rho} \mathbf{P}(\rho)\, \mathrm{d}\mathbb{P}^{\eta,\theta}_{\rho_0}(\rho) \tag{5.4}$$

This definition of the expectation makes sense by the following result:

Lemma 5.8. *The mapping $\rho \mapsto \mathbf{P}(\rho)$ is a $\mathbb{P}^{\eta,\theta}_{\rho_0}$-measurable function.*

Proof. By using the results of [Bou+07, Section 3.2], we know that given a finite path π ending in a target location of L_T and an open interval of possible weights I_{wt} (that is a basic Lebesgue-measurable set of $\mathbb{R}$), the set of maximal plays following π and with a weight in I_{wt} is defined by constraints over the delays taken along the play. More precisely, this set of plays can be defined by a set of delays $(t_i)_{0\leq i\leq |\pi|-1}$ that satisfy a linear program where the constraints are over partial sums $\sum_{i=j}^{k} t_i$. Thus, the set of maximal plays with a weight in I_{wt} is a (countable) union of cylinders, defined by finite maximal paths reaching the target and constraints on delays described by a linear program (and thus a Lebesgue-measurable set of $\mathbb{R}^{|\pi|}_{\geq 0}$). □

5.2.1. Values defined with proper (stochastic) strategies

Even if the integral defining $\mathbb{E}^{\eta,\theta}_{\rho_0}$ always exists, it may be infinite. In particular, it is finite only if the probability to reach a target location is equal to 1, since otherwise there is a non-zero probability of not reaching the target location, the weight of all such plays being $+\infty$, leading to an infinite expectation. We thus now require that $\mathbb{P}^{\eta,\theta}_{\rho_0}(\mathsf{TPlays}_{\rho_0}) = 1$ (i.e. the probability to follow an infinite path is 0) that is possible since TPlays_{ρ_0} is a $\mathbb{P}^{\eta,\theta}_{\rho_0}$-measurable set as a countable union of cylinders generated by all finite maximal paths ending in L_T. However, this is not a sufficient condition to ensure that the expected weight is finite.

Example 5.9. We consider the (one-player) WTG of Figure 5.2, where only Min plays, and the memoryless strategy η defined as follows. For all valuations $\nu \leq 1$, we fix

$$\eta_\Delta(\ell,\nu)(\delta_0) = \nu \qquad \text{and} \qquad \eta_\Delta(\ell,\nu)(\delta_1) = 1-\nu\,.$$

Moreover, for all $i \geq 2$ and $\delta \in \{\delta_0,\delta_1\}$, $\eta_{\mathbb{R}^+}\left(\left(\ell, 1-\frac{1}{i}\right),\delta\right)$ is the Dirac distribution selecting the delay $t = \frac{1}{i} - \frac{1}{i+1}$. Notice that we can extend the delay distribution (continuously for instance) so that this strategy satisfies Hypothesis 1. For all $i \geq 1$, there is a unique play conforming to η of length i that reaches the target from configuration $(\ell, \frac{1}{2})$: it has a weight i and a probability $\frac{1}{i+1}\prod_{j=2}^{i}\left(1-\frac{1}{j}\right) = \frac{1}{i(i+1)}$. In particular,

$$\mathbb{P}^{\eta}_{\ell,\frac{1}{2}}(\mathsf{TPlays}_{\ell,\frac{1}{2}}) = \sum_{i\geq 1}\frac{1}{i(i+1)} = 1\,.$$

Moreover, by previous computation, we can deduce that

$$\mathbb{E}^{\eta,\theta}_{\ell,\frac{1}{2}} = \int_\rho \mathbf{P}(\rho)\,\mathrm{d}\mathbb{P}^{\eta,\theta}_{\ell,\frac{1}{2}}(\rho) = \sum_i \int_\rho \mathbf{P}(\rho)\,\chi_{\mathsf{TPlays}^i_{\ell,\frac{1}{2}}}(\rho)\,\mathrm{d}\mathbb{P}^{\eta,\theta}_{\ell,\frac{1}{2}}(\rho)$$

where $\mathsf{TPlays}^i_{\ell,\frac{1}{2}}$ is a subset of $\mathsf{TPlays}_{\ell,\frac{1}{2}}$ that contains all plays of length i. The expected weight would thus be

$$\mathbb{E}^{\eta,\theta}_{\ell,\frac{1}{2}} = \sum_{i\geq 1} i \times \frac{1}{i(i+1)} = \sum_{i\geq 1}\frac{1}{i+1}\,,$$

which is a diverging series. This example can easily be adapted to only consider continuous distributions on the delays (instead of Dirac ones). ⌟

We thus need a stronger hypothesis to ensure that the expectation is finite. We adopt here an asymmetrical point of view, relying only on a hypothesis on the strategy η of Min. Our choice is grounded in our controller synthesis view, Min being the controller desiring to reach a target location with minimum expected weight, while Max is an uncontrollable environment. First, we focus on smooth deterministic strategies.

Definition 5.10. A smooth deterministic strategy $\sigma \in \mathsf{sdStrat}_{\mathsf{Min}}$ of Min is said *proper* if for all finite plays ρ_0 and (stochastic) strategies $\theta \in \mathsf{Strat}_{\mathsf{Max}}$, we have $\mathbb{P}^{\sigma,\theta}_{\rho_0}(\mathsf{TPlays}_{\rho_0}) = 1$ and $\mathbb{E}^{\sigma,\theta}_{\rho_0}$ is finite.

We remark first that not all smooth deterministic strategies that guarantee reaching a target location are proper.

Example 5.11. We consider the WTG depicted in Figure 5.3, $(\ell_0, 0)$ the initial configuration. We define σ to be a deterministic strategy of Min such that σ always

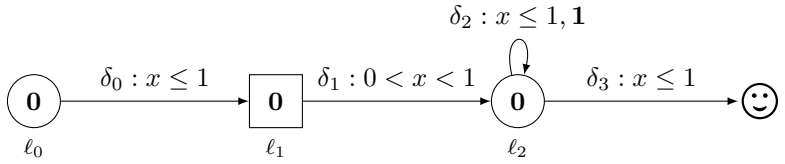


Figure 5.3 – A WTG where Min has a non-proper deterministic strategy that reaches the target with probability 1 against all (stochastic) strategies for Max.

chooses $(\delta_0, 0)$ from ℓ_0, and chooses i times $(\delta_2, 0)$ before $(\delta_3, 0)$ when ℓ_2 is reached with a valuation $\nu \in [1 - \frac{1}{i}, 1 - \frac{1}{i+1})$. First, this strategy guarantees that all plays conforming to it reach a target location, even if the number of steps to take depends on the first configuration reached in ℓ_2.

Now, we consider $\theta \in \mathsf{Strat}_{\mathsf{Max}}$ defined such that, for all configurations (ℓ_1, ν),

$$\theta_{\Delta}(\ell_1, \nu) = \mathsf{Dirac}_{\delta_1} \qquad \text{and} \qquad \theta_{\mathbb{R}^+}((\ell, \nu), \delta_1) = \mathcal{U}(\nu, 1)$$

where $\mathcal{U}(\nu, 1)$ is the uniform distribution over the open interval $(\nu, 1)$, and we compute $\mathbb{E}^{\sigma,\theta}_{\ell_0,0}$. By Lemma 5.18 (and the cylinder decomposition), we have

$$\begin{aligned}
\mathbb{E}^{\sigma,\theta}_{\ell_0,0} &= \sum_{\pi \in \mathsf{TPaths}_{\ell_0,0}} \mathbb{E}^{\sigma,\theta}_{\ell_0,0}(\pi) = \sum_n \sum_{\pi \in \mathsf{TPaths}^{n+2}_{\ell_0,0}} \mathbb{E}^{\sigma,\theta}_{\ell_0,0}(\pi) \\
&= \sum_n \sum_{\pi \in \mathsf{TPaths}^{n+2}_{\ell_0,0}} \mathbb{E}^{\sigma,\theta}_{(\ell_0,0)[\delta_0,0]}(\pi) \qquad \text{(by definition of } \sigma) \\
&= \sum_n \sum_{\pi \in \mathsf{TPaths}^{n+2}_{\ell_0,0}} \int_0^1 \mathbb{E}^{\sigma,\theta}_{(\ell_0,0)[\delta_0,0][\delta_1,t]}(\pi) \mathrm{d}\eta_{\mathbb{R}^+}((\ell_1, 0), \delta_1)(t) \quad \text{(by definition of } \theta).
\end{aligned}$$

Now, since σ is a deterministic strategy, for all n, there exists only one path $\pi_n \in \mathsf{TPaths}^{n+2}_{\ell_0,0}$. In particular, we obtain that

$$\mathbb{E}^{\sigma,\theta}_{\ell_0,0} = \sum_n \int_0^1 \mathbb{E}^{\sigma,\theta}_{(\ell_0,0)[\delta_0,0][\delta_1,t]}(\pi_n) \mathrm{d}\eta_{\mathbb{R}^+}((\ell_1, 0), \delta_1)(t).$$

By definition of σ, we know that π_n is conforming to σ only when $t \in [1 - \frac{1}{n}, 1 - \frac{1}{n+1})$, i.e.

$$\mathbb{E}^{\sigma,\theta}_{(\ell_0,0)[\delta_0,0][\delta_1,t]}(\pi_n) = n \times \chi_{[1-\frac{1}{n}, 1-\frac{1}{n+1})}(t).$$

Since $\eta_{\mathbb{R}^+}((\ell_1, 0), \delta_1)$ follows a uniform distribution over $(0, 1)$, we deduce that

$$\mathbb{E}^{\sigma,\theta}_{\ell_0,0} = \sum_n n \int_{\frac{n-1}{n}}^{\frac{n}{n+1}} \mathrm{d}t = \sum_n \frac{n}{n(n+1)}$$

that diverges. Thus, σ is not proper. ⌟

For stochastic strategies, we have also seen above that reaching the target set of locations with probability 1 is a necessary but not sufficient condition to be proper. Not only Min must reach the target almost surely but must do it *quickly enough* so that the expectation converges. We now give a sufficient condition for a strategy to be proper.

Hypothesis 2. *A strategy $\eta \in \mathsf{Strat}_{\mathsf{Min}}$ of* Min *satisfies this hypothesis if there exist $m \in \mathbb{N}$ and $\alpha \in (0, 1)$ such that for all finite plays ρ_0 and strategies $\theta \in \mathsf{Strat}_{\mathsf{Max}}$,*

$$\mathbb{P}^{\eta,\theta}_{\rho_0}\Big(\bigcup_{n \leq m} \mathsf{TPlays}^n_{\rho_0}\Big) \geq \alpha\,.$$

From this Hypothesis, we define the notion of proper stochastic strategies.

Definition 5.12. A strategy $\eta \in \mathsf{Strat}_{\mathsf{Min}}$ of Min is said *proper* if η satisfies Hypothesis 2 or is a proper smooth deterministic strategy.

As expected, (non-proper) strategies considered in Examples 5.9 and 5.11 do not satisfy this hypothesis.

Example 5.13. We consider the memoryless strategy η defined in Example 5.9 for the WTG depicted in Figure 5.2. Let $m \in \mathbb{N}$ and $\alpha \in (0, 1)$, and we fix $k = \left\lceil \frac{m+1}{\alpha} - 1 \right\rceil$. Now, for all $n \leq m$, since there exists only one play conforming to η and length n, we deduce that

$$\mathbb{P}^{\eta}_{\ell,\frac{1}{k}}(\mathsf{TPlays}^n_{\rho_0}) = \frac{1}{k+n} \prod_{j=k}^{k+n} \left(1 - \frac{1}{j}\right) \leq \frac{1}{k+n}\,.$$

In particular, since $k + 1 \leq k + n$, we obtain that

$$\mathbb{P}^{\eta}_{\ell,\frac{1}{k}}\Big(\bigcup_{n \leq m} \mathsf{TPlays}^n_{\rho_0}\Big) = \frac{1}{k+1} + \sum_{n=1}^{m} \frac{1}{k+n} \leq \frac{m+1}{k+1} < \alpha\,.$$

Thus, η does not satisfy Hypothesis 2. We note that a strategy satisfies Hypothesis 2 independently of weights in the considered WTG. ⌟

We let $\mathsf{sdStrat}^{\mathsf{p}}_{\mathsf{Min}}(\mathcal{G})$ be the set of proper smooth deterministic strategies[4] of Min. We also let $\mathsf{Strat}^{\mathsf{p}}_{\mathsf{Min}}(\mathcal{G})$ be the set of proper strategies of Min given by the subset of $\mathsf{Strat}_{\mathsf{Min}}$. Finally, we let $\mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}(\mathcal{G})$ be the subset of $\mathsf{Strat}^{\mathsf{p}}_{\mathsf{Min}}$ that defines the set of memoryless proper strategies. To lighten the notations, we do not mention $\mathcal{G}$ when the context is clear.

Now that we have associated an expected payoff with each convenient pair of strategies, we are able to mimic the classical definitions of *values* to stochastic strategies. Let ℓ be a location and ν be a valuation. For all $\eta \in \mathsf{Strat}^{\mathsf{p}}_{\mathsf{Min}}(\mathcal{G})$ and $\theta \in \mathsf{Strat}_{\mathsf{Max}}(\mathcal{G})$, we let

$$\mathsf{Val}^{\eta}_{\mathcal{G}}(\ell,\nu) = \sup_{\theta \in \mathsf{Strat}_{\mathsf{Max}}(\mathcal{G})} \mathbb{E}^{\eta,\theta}_{\ell,\nu} \qquad \text{and} \qquad \mathsf{Val}^{\theta}_{\mathcal{G}}(\ell,\nu) = \inf_{\eta \in \mathsf{Strat}^{\mathsf{p}}_{\mathsf{Min}}(\mathcal{G})} \mathbb{E}^{\eta,\theta}_{\ell,\nu} .$$

This definition can be generalised by replacing configurations by finite plays ρ_0: we let

$$\mathsf{Val}^{\eta}_{\mathcal{G}}(\rho_0) = \sup_{\theta \in \mathsf{Strat}_{\mathsf{Max}}(\mathcal{G})} \mathbb{E}^{\eta,\theta}_{\rho_0} \qquad \text{and} \qquad \mathsf{Val}^{\theta}_{\mathcal{G}}(\rho_0) = \sup_{\eta \in \mathsf{Strat}^{\mathsf{p}}_{\mathsf{Min}}(\mathcal{G})} \mathbb{E}^{\eta,\theta}_{\rho_0}$$

be the generalised versions. Then, we let $\overline{\mathsf{Val}}_{\mathcal{G}}(\ell,\nu)$ and $\underline{\mathsf{Val}}_{\mathcal{G}}(\ell,\nu)$ be the upper- and lower-values of $\mathcal{G}$ in (ℓ,ν), defined as the best-expected payoff Min and Max can hope for, respectively:

$$\overline{\mathsf{Val}}_{\mathcal{G}}(\ell,\nu) = \inf_{\eta \in \mathsf{Strat}^{\mathsf{p}}_{\mathsf{Min}}(\mathcal{G})} \mathsf{Val}^{\eta}_{\mathcal{G}}(\ell,\nu) \qquad \text{and} \qquad \underline{\mathsf{Val}}_{\mathcal{G}}(\ell,\nu) = \sup_{\theta \in \mathsf{Strat}_{\mathsf{Max}}(\mathcal{G})} \mathsf{Val}^{\theta}_{\mathcal{G}}(\ell,\nu) .$$

We also define the *memoryless values* $\mathsf{mVal}^{\eta}_{\mathcal{G}}(\ell,\nu)$, $\mathsf{mVal}^{\theta}_{\mathcal{G}}(\ell,\nu)$, $\overline{\mathsf{mVal}}_{\mathcal{G}}(\ell,\nu)$, and $\underline{\mathsf{mVal}}_{\mathcal{G}}(\ell,\nu)$, where all strategies are taken memoryless. Finally, where all strategies are taken smooth deterministic, we define the *smooth deterministic values* $\mathsf{sdVal}^{\eta}_{\mathcal{G}}(\ell,\nu)$, $\mathsf{sdVal}^{\theta}_{\mathcal{G}}(\ell,\nu)$, $\overline{\mathsf{sdVal}}_{\mathcal{G}}(\ell,\nu)$, and $\underline{\mathsf{sdVal}}_{\mathcal{G}}(\ell,\nu)$,. As usual, for all configurations (ℓ,ν), we always have

$$\underline{\mathsf{Val}}_{\mathcal{G}}(\ell,\nu) \leq \overline{\mathsf{Val}}_{\mathcal{G}}(\ell,\nu) , \ \underline{\mathsf{mVal}}_{\mathcal{G}}(\ell,\nu) \leq \overline{\mathsf{mVal}}_{\mathcal{G}}(\ell,\nu) \text{ and } \underline{\mathsf{sdVal}}_{\mathcal{G}}(\ell,\nu) \leq \overline{\mathsf{sdVal}}_{\mathcal{G}}(\ell,\nu).$$

Again, to lighten the notations, we do not mention $\mathcal{G}$ when the context is clear.

We have introduced the (stochastic) values to generalise the deterministic one. However, to ensure the existence of mathematical objects like the measure of probabilities or the expectation (against all (stochastic) strategies for Max), we need to constrain deterministic strategies and define new deterministic values. In particular, we do not have tools to prove that smooth deterministic values are determined

[4] The classical attractor strategy of Min (see Section 1.2.3) that guarantees that the target is reached can be chosen continuous over regions and is thus smooth and proper.

(unlike in the case of the deterministic value), and are equal to the deterministic value. An idea to prove it will be proving the existence of an almost-optimal proper smooth deterministic strategy for the deterministic value: intuitively it consists of proving some smooth properties on the deterministic value function as well as the capability of Min to ensure the deterministic value by going quickly enough to a target location.

5.2.2. A sufficient condition for proper strategies

To conclude the definition of values with stochastic strategies, it remains to prove that Hypothesis 2 is indeed a sufficient condition for guaranteeing a strategy is proper. The rest of the section consists of proving the following proposition:

Proposition 5.14. *All strategies of* Min *satisfying Hypothesis 2 are proper.*

The main idea of the proof is to decompose the expectation $\mathbb{E}^{\eta,\theta}_{\rho_0}$ according to the size of maximal plays such that we can bound their weight by an affine function (that is always the case) and their probabilities by the exponential function since Hypothesis 2 formalises the idea that Min guarantees that the target is quickly reached enough whatever the starting point of the play. In particular, we can exponentially bound the probability of each cylinder according to its length as illustrated in the following lemma.

Lemma 5.15. *Let $\eta \in \mathsf{Strat}_{\mathsf{Min}}$ be a strategy of* Min *satisfying Hypothesis 2 for the bound α and m. Let $\theta \in \mathsf{Strat}_{\mathsf{Max}}$ be a strategy of* Max *and $\rho_0 \in \mathsf{FPlays}$ be a finite play following a finite path π_0. For all n, we have*

$$\sum_{\pi \in \mathsf{TPaths}^n_{\rho_0}} \mathbb{P}^{\eta,\theta}_{\rho_0}(\pi_0\pi) \leq (1-\alpha)^{\lfloor n/m \rfloor}.$$

Proof. We proceed by induction, showing first that for all $n \in \mathbb{N}$, and all finite plays ρ_0 following π_0,

$$\sum_{\pi \in \mathsf{TPaths}^{nm}_{\rho_0}} \mathbb{P}^{\eta,\theta}_{\rho_0}(\pi_0\pi) \leq (1-\alpha)^n .$$

For $n = 0$, the property holds since $(1-\alpha)^0 = 1$. Otherwise, suppose that the property holds for n, and we prove it for $n+1$ by decomposing the sum as

$$\sum_{\pi \in \mathsf{TPaths}^{(n+1)m}_{\rho_0}} \mathbb{P}^{\eta,\theta}_{\rho_0}(\pi_0\pi) = \sum_{\delta_0}\sum_{\delta_1}\cdots\sum_{\delta_{m-1}}\sum_{\pi \in \mathsf{TPaths}^{nm}_{\rho_0}} \mathbb{P}^{\eta,\theta}_{\rho_0}(\pi_0{\cdot}\delta_0{\cdot}\delta_1\cdots\delta_{m-1}{\cdot}\pi)\,. \quad (5.5)$$

For a fixed $\delta_0 \cdots \delta_{m-1} \cdot \pi \in \mathsf{TPaths}_{\rho_0}^{(n+1)m}$, letting $\pi' = \pi_0 \cdot \delta_0 \cdot \delta_1 \cdots \delta_{m-1} \cdot \pi$, we can unravel the m first steps of the definition of $\mathbb{P}_{\rho_0}^{\eta,\theta}(\pi')$. Letting ξ^i denote the strategy η or θ depending if the source location of the transition δ_i belongs to Min or Max, we can write $\mathbb{P}_{\rho_0}^{\eta,\theta}(\pi')$ under the form:

$$\int_{I(\rho_0,\delta_0)} \cdots \int_{I(\rho_0[\delta_0,t_0]\cdots[\delta_{m-1},t_{m-1}])} \xi_\Delta^0(\rho_0)(\delta_0)\cdots\xi_\Delta^{m-2}\big(\rho_0[\delta_0,t_0]\cdots[\delta_{m-1},t_{m-2}]\big)(\delta_{m-1})\mathbb{P}^{\eta,\theta}_{\rho_0[\delta_0,t_0]\cdots[\delta_{m-1},t_{m-1}]}(\pi') \, \mathrm{d}\xi_{\mathbb{R}^+}^{m-1}\big(\rho_0[\delta_0,t_0]\cdots[\delta_{m-2},t_{m-2}],\delta_{m-1}\big)(t_{m-1})\cdots \mathrm{d}\xi_{\mathbb{R}^+}^0(\rho_0,\delta_0)(t_0)\,.$$

Thus, (5.5) can be rewritten as

$$\sum_{\delta_0}\sum_{\delta_1}\cdots\sum_{\delta_{m-1}}\int_{I(\rho_0,\delta_0)}\int_{I(\rho_0[\delta_0,t_0])}\cdots\int_{I(\rho_0[\delta_0,t_0]\cdots[\delta_{m-1},t_{m-1}])} \xi_\Delta^0(\rho_0)(\delta_0)\cdots\xi_\Delta^{m-1}\big(\rho_0[\delta_0,t_0]\cdots[\delta_{m-2},t_{m-2}]\big)(\delta_{m-1}) \sum_{\pi\in\mathsf{TPaths}_{\rho_0}^{nm}} \mathbb{P}^{\eta,\theta}_{\rho_0[\delta_0,t_0]\cdots[\delta_{m-1},t_{m-1}]}(\pi_0\delta_0\delta_1\cdots\delta_{m-1}\pi) \mathrm{d}\xi_{\mathbb{R}^+}^{m-1}\big(\rho_0[\delta_0,t_0]\cdots[\delta_{m-2},t_{m-2}],\delta_{m-1}\big)(t_{m-1})\cdots \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta_0)(t_0).$$

By induction hypothesis,

$$\sum_{\pi\in\mathsf{TPaths}_{\rho_0}^{nm}} \mathbb{P}^{\eta,\theta}_{\rho_0[\delta_0,t_0]\cdots[\delta_{m-1},t_{m-1}]}(\pi_0\delta_0\delta_1\cdots\delta_{m-1}\pi) \le (1-\alpha)^n$$

so that $\sum_{\pi\in\mathsf{TPaths}_{\rho_0}^{(n+1)m}} \mathbb{P}_{\rho_0}^{\eta,\theta}(\pi_0\delta_0\delta_1\cdots\delta_{m-1}\pi)$ can be bounded by

$$(1-\alpha)^n\sum_{\delta_0}\sum_{\delta_1}\cdots\sum_{\delta_{m-1}}\int_{I(\rho_0,\delta_0)}\cdots\int_{I(\rho_0[\delta_0,t_0]\cdots[\delta_{m-1},t_{m-1}])} \xi_\Delta^0(\rho_0)(\delta_0)\cdots\xi_\Delta^{m-1}\big(\rho_0[\delta_0,t_0]\cdots[\delta_{m-2},t_{m-2}]\big)(\delta_{m-1}) \mathrm{d}\xi_{\mathbb{R}^+}^{m-1}\big(\rho_0[\delta_0,t_0]\cdots[\delta_{m-2},t_{m-2}],\delta_{m-1}\big)(t_{m-1})\cdots \mathrm{d}\xi_{\mathbb{R}^+}^0(\rho_0,\delta_0)(t_0)$$

that is $(1-\alpha)^n$ multiplied by the probability of all paths that can continue after ρ for at least m steps without having reached the target set L_T, i.e. by Hypothesis 2

$$(1-\alpha)^n\,\mathbb{P}_{\rho_0}^{\eta,\theta}\big(\bigcup_{n\ge m}\mathsf{TPlays}_{\rho_0}^n\big) \le (1-\alpha)^n(1-\alpha)\,.$$

This concludes the induction proof. To conclude, we simply need to consider paths

of length that is not a multiple of m: we thus unravel similarly the first $k \in (0, m)$ transitions, bounding all their probabilities by 1. □

A natural choice to decompose the expectation is using cylinders (as for probability). Thus, we need to define the expected weight of a cylinder. We will not need to consider specific constraints on the delays and thus only consider cylinders of the form $\mathsf{Cyl}_{\rho_0}(\pi)$ (with the set of constraints $\mathbb{R}_{\geq 0}^{|\pi|}$).

Definition 5.16. Let ρ_0 be a finite play and π be a finite path. If the cylinder $\mathsf{Cyl}_{\rho_0}(\pi)$ is non-empty, the *expected weight* $\mathbb{E}^{\eta,\theta}_{\rho_0}(\pi)$ of plays that are in $\mathsf{Cyl}_{\rho_0}(\pi)$ is defined by induction on the length of ρ_0. If $|\rho_0| = |\pi|$, we let $\mathbb{E}^{\eta,\theta}_{\rho_0}(\pi) = 0$, meaning that there are no more steps to take into account in the expected weight. Otherwise, by letting $\delta = (\ell, g, Y, \ell')$ be the $(|\rho_0|+1)$-th transition of π, and ξ be the strategy η or θ depending if $\rho_0 \in \mathsf{FPlays}_{\mathsf{Min}}$ or $\rho_0 \in \mathsf{FPlays}_{\mathsf{Max}}$, we let

$$\mathbb{E}^{\eta,\theta}_{\rho_0}(\pi) = \int_{I(\rho_0,\delta)} \xi_{\Delta}(\rho_0)(\delta)\Big[\big(t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta)\big)\mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi) + \mathbb{E}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi)\Big]\,\mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta)(t)\,.$$

For the sake of completeness, if the cylinder $\mathsf{Cyl}_{\rho_0}(\pi)$ is empty, we let $\mathbb{E}^{\eta,\theta}_{\rho_0}(\pi) = 0$.

By adapting the proof of Lemma 5.3, Hypothesis 1 is sufficient to show that all expectations $\mathbb{E}^{\eta,\theta}_{\rho_0}(\pi)$ are finite:

Lemma 5.17. *For all finite plays $\rho_0 = (\ell_0, \nu_0)[\delta_0, t_0]\cdots[\delta_{k-1}, t_{k-1}]$, we let $\pi_0 = \delta_0 \cdots \delta_{k-1}$ be the finite path following by ρ_0. For all finite paths π starting in the last location of π_0, $\eta \in \mathsf{Strat}_{\mathsf{Min}}$ and $\theta \in \mathsf{Strat}_{\mathsf{Max}}$:*

1. *$\mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi)$ exists;*
2. *$|\mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi)| \leq \mathbb{P}^{\eta,\theta}_{\rho_0}(\pi_0\pi)\,|\pi| W_{\mathsf{e}}$ (so $\mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi)$ is finite);*
3. *the mapping $(t_0, \ldots, t_{k-1}) \mapsto \mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi)$ is Lebesgue-measurable.*

Proof. We start by proving properties 1 and 2 by induction on the length k of ρ_0. Since ρ_0 follows π_0, the result is immediate when $k = |\pi_0\pi|$ (i.e. $|\pi| = 0$). Otherwise, we suppose that the property holds for all finite plays of length $k+1$ and we consider a finite play ρ_0 of length k. We let ξ be the strategy η or θ according if $\rho_0 \in \mathsf{FPlays}_{\mathsf{Min}}$ or $\rho_0 \in \mathsf{FPlays}_{\mathsf{Max}}$, $\delta = (\ell, g, Y, \ell')$ be a transition with ℓ the last location of ρ_0 and π' be a finite path starting in ℓ'. We also let $\pi = \delta\pi'$. By definition of the expectation, $\mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi)$ is equal to

$$\int_{I(\rho_0,\delta)} \xi_{\Delta}(\rho_0)(\delta)\Big[(t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta))\mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\pi) + \mathbb{E}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\pi)\Big]\,\mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta)(t)\,.$$

By induction hypothesis for all finite plays $\rho_0[\delta, t]$, we know that $\mathbb{E}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\pi)$ exists and is a measurable function of t. Therefore, the mapping

$$t \mapsto (t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta))\mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\pi) + \mathbb{E}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\pi)$$

is measurable (by Lemma 5.3). Thus, since $(t_0, \dots, t_{k-1}) \mapsto \xi_\Delta(\rho)(\delta)$ is Lebesgue-measurable (by Hypothesis 1-1), $\mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi)$ exists, i.e. property 1 holds. Moreover, by triangular inequality, we bound $|\mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi)|$ by

$$\int_{I(\rho_0,\delta)} \xi_\Delta(\rho_0)(\delta)\Big[|t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta)|\mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\pi) + |\mathbb{E}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\pi)|\Big]\,\mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta)(t)\,.$$

By definition of W_e and induction hypothesis applied to $\rho_0[\delta, t]$,

$$|t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta)| \le W_\mathsf{e} \qquad \text{and} \qquad |\mathbb{E}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\pi)| \le \mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\pi)\,|\pi'|\,W_\mathsf{e}\,.$$

By properties of the integral, we deduce that

$$|\mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi)| \le \underbrace{(|\pi'|+1)}_{=|\pi|} W_\mathsf{e} \underbrace{\int_{I(\rho_0,\delta)} \xi_\Delta(\rho_0)(\delta)\mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\pi)\,\mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta)(t)}_{=\mathbb{P}^{\eta,\theta}_{\rho_0}(\pi_0\pi)}$$

and property 2 holds.

To conclude the proof, we remark that by property 2 and Fubini theorem, $\mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi)$ is Lebesgue-measurable w.r.t. the delays in ρ_0, i.e. $(t_0, \dots, t_{k-1}) \mapsto \mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi)$ is Lebesgue-measurable. In particular, property 3 holds. □

Now, we give the link between the expected weight and the expected weight restricted to a cylinder. In particular, we prove that the expected weight can be decomposed over cylinders.

Lemma 5.18. *Let ρ_0 be a finite play following the path π_0, then*

$$\mathbb{E}^{\eta,\theta}_{\rho_0} = \sum_{\pi \in \mathsf{TPaths}_{\rho_0}} \mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi)\,.$$

Proof. By property of the integral, we can split the integral into cylinders and obtain that[5]

$$\mathbb{E}^{\eta,\theta}_{\rho_0} = \sum_{\pi \in \mathsf{TPaths}_{\rho_0}} \underbrace{\int_\rho \mathbf{P}(\rho)\,\chi_{\mathsf{Cyl}_{\rho_0}(\pi_0\pi)}(\rho)\,\mathrm{d}\mathbb{P}^{\eta,\theta}_{\rho_0}(\rho)}_{=A}\,.$$

[5]The definition of A could correspond to that of $\mathbb{E}^{\eta,\theta}_{\rho_0}(\pi)$. However, in the following, we need the inductive formula given by Definition 5.16.

To conclude the proof, we prove by induction on the length of ρ_0 that $A = \mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi)$. If $|\rho_0| = |\pi_0\pi|$, since $\pi \in \mathsf{TPaths}_{\rho_0}$, then $\mathbf{P}(\rho) = 0$ for all $\rho \in \mathsf{Cyl}_{\rho_0}(\pi_0\pi)$. Thus, the property holds in this case. Otherwise, we suppose that the property holds for all finite plays of length $k+1$ and we consider a finite play ρ_0 of length k, $\delta = (\ell, g, Y, \ell') \in \Delta$ the first transition of π, and we let ξ be a strategy η or θ according if ρ_0 ends in a configuration of Min or Max. In particular, we obtain that A is equal to

$$\int_{I(\rho_0,\delta)} \xi_\Delta(\rho_0)(\delta)\Big(\int_\rho (t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + \mathbf{P}(\rho))\,\chi_{\mathsf{Cyl}_{\rho_0[\delta,t]}(\pi_0\pi)}(\rho)\;\mathrm{d}\mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\rho)\Big) \\ \mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta)(t)\,.$$

In particular, by linearity of the integral, A can be rewritten as

$$\int_{I(\rho_0,\delta)} \xi_\Delta(\rho_0)(\delta)\Big[\big(t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta)\big)\underbrace{\int_\rho \chi_{\mathsf{Cyl}_{\rho_0[\delta,t]}(\pi_0\pi)}(\rho)\mathrm{d}\mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\rho)}_{=\mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\pi)} + \\ \int_\rho \mathbf{P}(\rho)\,\chi_{\mathsf{Cyl}_{\rho_0[\delta,t]}(\pi_0\pi)}(\rho)\;\mathrm{d}\mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\rho)\Big]\mathrm{d}\eta_{\mathbb{R}^+}(\rho_0,\delta)(t)\,.$$

By hypothesis of induction applied on $\rho_0[\delta,t]$, we have

$$\int_\rho \mathbf{P}(\rho)\,\chi_{\mathsf{Cyl}_{\rho_0[\delta,t]}(\pi_0\pi)}(\rho)\;\mathrm{d}\mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\rho) = \mathbb{E}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\pi)$$

so that A can be rewritten as

$$A = \int_{I(\rho_0,\delta)} \xi_\Delta(\rho_0)(\delta)\Big[\big(t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta)\big)\mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\pi) + \mathbb{E}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\pi)\Big]\mathrm{d}\eta_{\mathbb{R}^+}(\rho_0,\delta)(t)$$

that is the definition of $\mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi)$. □

Finally, we can conclude the proof of Proposition 5.14. Let $\eta \in \mathsf{Strat}_{\mathsf{Min}}$ be a strategy that satisfies Hypothesis 2. We let α and m be the bounds of Hypothesis 2. Let $\theta \in \mathsf{Strat}_{\mathsf{Max}}$, and $\rho_0 \in \mathsf{FPlays}$ that follows a path π_0.

We first show that $\mathbb{P}^{\eta,\theta}_{\rho_0}(\mathsf{TPlays}_{\rho_0}) = 1$. The probability of not reaching the target can be obtained as the limit

$$\lim_{n\to\infty} \sum_{\pi\in\mathsf{TPaths}^n_{\rho_0}} \mathbb{P}^{\eta,\theta}_{\rho_0}(\pi_0\pi)$$

By Lemma 5.15, this limit is 0, so the probability to reach the target is indeed 1.

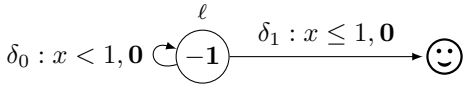


Figure 5.4 – A WTG where Min has a proper strategy that does not satisfy Hypothesis 2.

We then show that the infinite sum $\mathbb{E}^{\eta,\theta}_{\rho_0} = \sum_{\pi \in \mathsf{TPaths}_{\rho_0}} \mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi)$ (given by Lemma 5.18) converges. Notice that this sum can be decomposed as

$$\mathbb{E}^{\eta,\theta}_{\rho_0} = \sum_{n=0}^{\infty} \sum_{\pi \in \mathsf{TPaths}^n_{\rho_0}} \mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi) \tag{5.6}$$

Since $\mathsf{TPaths}^n_{\rho_0}$ is a finite set, only the first sum must be shown to be converging. We prove that it is absolutely converging by bounding $\sum_{\pi \in \mathsf{TPaths}^n_{\rho_0}} |\mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi)|$ with respect to n. By Lemmas 5.17-2 and 5.15, it is bounded by $n\,(1-\alpha)^{\lfloor \frac{n}{m} \rfloor} W_{\mathsf{e}}$, which is the general term of a convergent series since $0 < \alpha < 1$. That concludes the proof.

Remark. Hypothesis 2 is not a necessary condition for guaranteeing that a strategy is proper: some strategies are proper even if this hypothesis is not satisfied (see the following example). We note that to obtain a necessary condition, we must take account of weights in the considered WTG. ◁

Example 5.19. We consider the WTG depicted in Figure 5.4. We consider the memoryless strategy η defined in Example 5.9. In particular, η does not satisfy Hypothesis 2, as shown in Example 5.13.

Finally, we prove that the expectation of this strategy is finite, i.e. that η is proper. As explained in Example 5.9, for all $i \geq 1$, there is a unique play conforming to η of length i that reaches the target from configuration $(\ell, \frac{1}{2})$: it has a weight $\sum_{n=1}^{i} -(\frac{1}{n} - \frac{1}{n+1}) = -\frac{i}{i+1}$ and a probability $\frac{1}{i(i+1)}$. In particular, by the previous computation, we can deduce that

$$\mathbb{E}^{\eta,\theta}_{\ell,\frac{1}{2}} = \sum_{i \geq 1} -\frac{i}{i+1} \times \frac{1}{i(i+1)} = \sum_{i \geq 1} -\frac{1}{(i+1)^2}\,,$$

which is a converging series to $1 - \frac{\pi^2}{6} \simeq -0.6449$. Thus, this strategy is proper. ⌟

5.3. Discrete strategies

Between deterministic and stochastic strategies, we define a new class of strategies that we call *discrete strategies*. Intuitively, discrete strategies generalise deter-

ministic strategies by only allowing discrete probabilities for both transitions *and* delays. We will later use such particular strategies and thus study them here.

Definition 5.20. A strategy ξ satisfying Hypothesis 1 is a *discrete strategy* if for all $\rho \in \mathsf{FPlays}$ and $\delta \in \Delta$, $\xi_{\mathbb{R}^+}(\rho, \delta)$ is a piecewise-constant function, i.e. there exist $\alpha_0, \dots, \alpha_n \in (0, 1]$, and $a_0, \dots, a_n \in \mathbb{R}_{\geq 0}$ such that, for all $t \in I(\rho, \delta)$,

$$\xi_{\mathbb{R}^+}(\rho, \delta)(t) = \sum_{i=0}^{n} \alpha_i(\rho, \delta) H(t - a_i(\rho, \delta)) \,.$$

Under a discrete strategy, the current player chooses a delay in a finite set of possible choices. However, even if a discrete strategy for Min only uses Dirac distributions for delays, and verifies that the probability to reach the target (against all strategies of Max) is 1, the expectation may not exist. Thus, to ensure the existence of the expectation, the strategies of Min must moreover satisfy Hypothesis 2.

We consider $\eta \in \mathsf{Strat}^{\mathsf{p}}_{\mathsf{Min}}$ and $\theta \in \mathsf{Strat}_{\mathsf{Max}}$ two discrete strategies (with η satisfying Hypothesis 2), and we will rewrite the expectation they induce. Intuitively, since η and θ only use discrete distributions for transitions and delays, a single play reaching a target and conforming to both strategies forms a $\mathbb{P}^{\eta,\theta}_{\rho_0}$-measurable set. Moreover, when we consider a finite path conforming to both discrete strategies, the number of finite plays following it is countable. Thus, we prove that $\mathbb{P}^{\eta,\theta}_{\rho_0}$ is a discrete measure, and we obtain a simpler formula for the expectation.

Lemma 5.21. *Let $\eta \in \mathsf{Strat}^{\mathsf{p}}_{\mathsf{Min}}$ and $\theta \in \mathsf{Strat}_{\mathsf{Max}}$ be two discrete strategies. For all finite plays ρ_0, we have*

$$\mathbb{E}^{\eta,\theta}_{\rho_0} = \sum_{\rho \in \mathsf{TPlays}^{\eta,\theta}_{\rho_0}} \mathbf{P}(\rho) \times \mathbb{P}^{\eta,\theta}_{\rho_0}(\rho)$$

where $\mathsf{TPlays}^{\eta,\theta}_{\rho_0}$ is the $\mathbb{P}^{\eta,\theta}_{\rho_0}$-measurable set of plays starting in the last configuration of ρ_0, conforming to η and θ and ending in the target.

Proof. Since η is proper, the probability measure of the set of plays that do not reach the target is 0. By definition of the expectation, we thus have

$$\mathbb{E}^{\eta,\theta}_{\rho_0} = \int_{\rho} \mathbf{P}(\rho) \, \mathrm{d}\mathbb{P}^{\eta,\theta}_{\rho_0}(\rho) = \int_{\rho} \mathbf{P}(\rho) \, \chi_{\mathsf{TPlays}^{\eta,\theta}_{\rho_0}}(\rho) \, \mathrm{d}\mathbb{P}^{\eta,\theta}_{\rho_0}(\rho) \,.$$

We conclude by showing that $\mathsf{TPlays}^{\eta,\theta}_{\rho_0}$ is countable. To do so, notice that the number of finite maximal paths from the last location of ρ_0 is countable (as we restrict ourselves to plays that reach the target). Moreover, since the distributions on delays have finite support, for each finite path, the number of possible delays

consistent with strategies is finite. Thus, along a finite path, at each step, the current player can extend the play with a finite number of choices. In total, $\mathsf{TPlays}_{\rho_0}^{\eta,\theta}$ only contains a countable number of plays. □

6. Probabilities are Useless in the Presence of Memory

In contrast with strategic games for which the probabilities increase the power of strategies for players, in WTGs, the probabilities offer the same guarantees as the deterministic strategies. Our contribution consists in proving this: the stochastic values are bounded by the smooth deterministic ones. Moreover, we also establish the equality between the deterministic value and the stochastic and smooth deterministic ones in some classes of WTGs.

Table of contents

6.1. Max always has a deterministic best-response 152
6.2. Stochastic values are bounded by the smooth deterministic ones . . 157
6.3. Equality between the stochastic and deterministic values 159

The chapter is devoted to our first result about stochastic values: the combination of stochastic choices and memory in strategies does not bring more power than just the memory.

Theorem 6.1. *Let $\mathcal{G}$ be a WTG. Then, for all configurations (ℓ, ν), we have*

$$\underline{\mathsf{sdVal}}(\ell,\nu) \leq \underline{\mathsf{Val}}(\ell,\nu) \leq \overline{\mathsf{Val}}(\ell,\nu) \leq \overline{\mathsf{sdVal}}(\ell,\nu)\,.$$

Moreover, if $\mathcal{G}$ is divergent, one-clock or without clocks (i.e. $\mathcal{G}$ is an SPG), then values are determined and equal to the deterministic ones, i.e.

$$\underline{\mathsf{sdVal}}(\ell,\nu) = \underline{\mathsf{Val}}(\ell,\nu) = \mathsf{dVal}(\ell,\nu) = \overline{\mathsf{Val}}(\ell,\nu) = \overline{\mathsf{sdVal}}(\ell,\nu)\,.$$

The main ingredient for the proof is that when Min plays with a proper strategy, Max always has a deterministic *best-response* strategy. Moreover, to obtain the equality when $\mathcal{G}$ is divergent, one-clock or an SPG, we prove that Min (and Max) has almost-optimal proper smooth deterministic strategies w.r.t. the deterministic value.

6.1. Max always has a deterministic best-response

We begin by explaining how we build a smooth deterministic best-response w.r.t. the (memoryless) upper-value for Max against all proper strategies of Min. This section is dedicated to proving the following proposition.

Proposition 6.2. *Let $\eta \in \mathsf{Strat}^{\mathsf{p}}_{\mathsf{Min}}$ and $\varepsilon > 0$. There exists a deterministic strategy $\tau \in \mathsf{dStrat}_{\mathsf{Max}}$ such that for all finite plays ρ, $\mathbb{E}^{\eta,\tau}_{\rho} \geq \mathsf{Val}^{\eta}(\rho) - \varepsilon$. If $\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}$ is memoryless, then there exists a deterministic strategy $\tau \in \mathsf{sdStrat}_{\mathsf{Max}}$ such that for all configurations (ℓ, ν), $\mathbb{E}^{\eta,\tau}_{\ell,\nu} \geq \mathsf{mVal}^{\eta}(\ell, \nu) - \varepsilon$.*

To make the proof, we first consider a (stochastic) strategy θ of Max such that for all finite plays ρ_0, $\mathbb{E}^{\eta,\theta}_{\rho_0} \geq \mathsf{Val}^{\eta}(\rho_0) - \frac{\varepsilon}{2}$, for a fixed $\varepsilon > 0$, which exists by definition of the (stochastic) value. We show the existence of a smooth deterministic strategy $\tau \in \mathsf{sdStrat}_{\mathsf{Max}}$ for Max such that for all finite plays ρ_0, we have $\mathbb{E}^{\eta,\tau}_{\rho_0} \geq \mathbb{E}^{\eta,\theta}_{\rho_0} - \frac{\varepsilon}{2^{|\rho_0|}}$, which allows us to conclude that $\mathbb{E}^{\eta,\tau}_{\rho_0} \geq \mathsf{Val}^{\eta}(\rho_0) - \varepsilon$.

To do that, we distinguish the cases where η is deterministic or is not.

The deterministic case We suppose $\eta \in \mathsf{sdStrat}^{\mathsf{p}}_{\mathsf{Min}}$ is a proper smooth deterministic strategy. By contradiction, we suppose that for all $\tau \in \mathsf{sdStrat}_{\mathsf{Max}}$, $\mathsf{wt}(\mathsf{Play}(\rho_0, \eta, \tau)) \leq \mathbb{E}^{\eta,\theta}_{\rho_0} - \frac{\varepsilon}{2^{|\rho_0|}}$. Thus, by definition of the expectation,

$$\mathbb{E}^{\eta,\theta}_{\rho_0} = \int_{\rho} \mathbf{P}(\rho)\, \mathrm{d}\mathbb{P}^{\eta,\theta}_{\rho_0} \leq \mathbb{E}^{\eta,\theta}_{\rho_0} - \frac{\varepsilon}{2^{|\rho_0|}}\,.$$

We obtain a contradiction.

The non-deterministic case We suppose that η satisfies Hypothesis 2. This case is more tricky, but we rely on a useful tool: a Bellman-like fixpoint equation fulfilled by the expected payoff. To state it, we pack all expectations following a pair (η, θ) of strategies in a single mapping $\mathbb{E}^{\eta,\theta} \colon \rho \in \mathsf{FPlays} \mapsto \mathbb{E}^{\eta,\theta}_{\rho} \in \mathbb{R}$. The Bellman-like fixpoint operator $\mathcal{H}^{\eta,\theta}$ is a function relating mappings $X \colon \mathsf{FPlays} \to \mathbb{R}$ such that $\mathcal{H}^{\eta,\theta} \colon (\mathsf{FPlays} \to \mathbb{R}) \to (\mathsf{FPlays} \to \mathbb{R})$ is partially defined for $X \colon \mathsf{FPlays} \to \mathbb{R}$, and $\rho \in \mathsf{FPlays}$ by

$$\begin{cases} 0 & \text{if } \rho \text{ ends in } L_T\,; \\ \sum_{\delta} \int_{I(\rho,\delta)} \eta_{\triangle}(\rho)(\delta)\big(t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + X(\rho[\delta,t])\big)\, \mathrm{d}\eta_{\mathbb{R}^+}(\rho,\delta)(t) & \text{if } \rho \in \mathsf{FPlays}_{\mathsf{Min}}\,; \\ \sum_{\delta} \int_{I(\rho,\delta)} \theta_{\triangle}(\rho)(\delta)\big(t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + X(\rho[\delta,t])\big)\, \mathrm{d}\theta_{\mathbb{R}^+}(\rho,\delta)(t) & \text{if } \rho \in \mathsf{FPlays}_{\mathsf{Max}}\,. \end{cases} \tag{6.1}$$

This operator is only partially defined since it makes sense only for functions X that are measurable w.r.t. delays along the finite play. In the following, we prove that $\mathbb{E}^{\eta,\theta}$ is a function for which $\mathcal{H}^{\eta,\theta}$ is well-defined, and it is a fixpoint of $\mathcal{H}^{\eta,\theta}$.

Lemma 6.3. *Let $\eta \in \mathsf{Strat}^{\mathsf{p}}_{\mathsf{Min}}$ satisfying Hypothesis 2 and $\theta \in \mathsf{Strat}_{\mathsf{Max}}$. The mapping $\mathbb{E}^{\eta,\theta}\colon \rho \in \mathsf{FPlays} \mapsto \mathbb{E}^{\eta,\theta}_{\rho}$ is a fixpoint of the operator $\mathcal{H}^{\eta,\theta}\colon (\mathsf{FPlays} \to \mathbb{R}) \to (\mathsf{FPlays} \to \mathbb{R})$.*

Proof. Let η and θ be two strategies, and $\rho_0 \in \mathsf{FPlays}$. If ρ_0 ends in L_T, then

$$\mathcal{H}^{\eta,\theta}(\mathbb{E}^{\eta,\theta})(\rho_0) = 0 = \mathbb{E}^{\eta,\theta}_{\rho_0}\,.$$

Otherwise, we let ξ be the strategy η or θ depending on whether $\rho_0 \in \mathsf{FPlays}_{\mathsf{Min}}$ or $\rho_0 \in \mathsf{FPlays}_{\mathsf{Max}}$ and π_0 be the path that ρ_0 follows. By Lemma 5.18,

$$\mathbb{E}^{\eta,\theta}_{\rho_0} = \sum_{\pi \in \mathsf{TPaths}_{\rho_0}} \mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\pi)\,.$$

In particular, by decomposing the path (that is possible, since ρ_0 does not end in L_T), we have $\mathbb{E}^{\eta,\theta}_{\rho_0} = \sum_{\delta}\sum_{\pi \mid \delta\pi \in \mathsf{TPaths}_{\rho_0}} \mathbb{E}^{\eta,\theta}_{\rho_0}(\pi_0\delta\pi)$. Thus, by definition of the expectation of a path, $\mathbb{E}^{\eta,\theta}_{\rho_0}$ is equal to

$$\sum_{\delta}\sum_{\substack{\pi\\ \delta\pi \in \mathsf{TPaths}_{\rho_0}}}\int_{I(\rho_0,\delta)} \xi_{\Delta}(\rho_0)(\delta)\big[(t\,\mathsf{wt}(\ell)+\mathsf{wt}(\delta))\mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\delta\pi) + \mathbb{E}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\delta\pi)\big]\,\mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta)(t)\,.$$

Since η satisfies Hypothesis 2 for bounds α and m, then by Lemma 5.15 and Lemma 5.17-2 we obtain that

$$\mathbb{E}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\delta\pi) \leq (1-\alpha)^{\lfloor |\pi|/m \rfloor}\,|\pi|\,W_{\mathsf{e}}$$

and

$$(t\,\mathsf{wt}(\ell)+\mathsf{wt}(\delta))\mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\delta\pi) \leq (1-\alpha)^{\lfloor |\pi|/m \rfloor} W_{\mathsf{e}}\,.$$

In particular, $(t\,\mathsf{wt}(\ell)+\mathsf{wt}(\delta))\mathbb{P}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\delta\pi) + \mathbb{E}^{\eta,\theta}_{\rho_0[\delta,t]}(\pi_0\delta\pi)$ is bounded by $(1-\alpha)^{\lfloor |\pi|/m \rfloor}\,(|\pi|+1)\,W_{\mathsf{e}}$. Thus, by dominated convergence, again by Lemma 5.18, and by definition of $\mathcal{H}^{\eta,\theta}$, we obtain that $\mathbb{E}^{\eta,\theta}_{\rho_0}$ is equal to

$$\sum_{\delta}\int_{I(\rho_0,\delta)} \xi_{\Delta}(\rho_0)(\delta)\big(t\,\mathsf{wt}(\ell)+\mathsf{wt}(\delta)+\mathbb{E}^{\eta,\theta}_{\rho_0[\delta,t]}\big)\,\mathrm{d}\xi_{\mathbb{R}^+}(\rho_0,\delta)(t) = \mathcal{H}^{\eta,\theta}(\mathbb{E}^{\eta,\theta})(\rho_0).$$

Thus, $\mathbb{E}^{\eta,\theta}$ is a fixpoint of $\mathcal{H}^{\eta,\theta}$. □

With this fixpoint equation in mind, we are equipped to prove Proposition 6.2: Max has a deterministic best-response strategy against a proper strategy of Min

when η satisfies Hypothesis 2. We will (again) use the argsup$^\varepsilon$ operator defined for all mappings $f\colon A \to \mathbb{R}$ and $B \subseteq A$ by $\mathrm{argsup}^\varepsilon_B f = \{a \in B \mid f(a) \geq \sup_B f - \varepsilon\}$. For all $\rho \in \mathsf{FPlays}_{\mathsf{Max}}$ ending in (ℓ, ν), we let

$$\tau(\rho) \in \mathrm{argsup}^{\varepsilon/2^{|\rho|+1}}_{(\ell,\nu)\xrightarrow{\delta,t}(\ell',\nu')} \left(t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + \mathbb{E}^{\eta,\theta}_{\rho[\delta,t]}\right).$$

We use Kuratowski and Ryll-Nardzewski's measurable selection theorem [KR65] to ensure that τ can be chosen in $\mathsf{sdStrat}_{\mathsf{Max}}$. This theorem states that if ψ is a mapping from a measurable space taking values in the set of non-empty closed subsets of a Borel set (for us, pairs (δ, t) of transitions and real delays) that is *weakly measurable*, then ψ has a selection, i.e. a measurable mapping τ such that for all elements ω of the measurable space, $\tau(\omega) \in \psi(\omega)$. Weak measurability amounts to saying that for every open subset U, we have $\{\omega \mid \psi(\omega) \cap U \neq \emptyset\}$ that is measurable.

Indeed, the mapping ψ defined for all plays ρ by

$$\psi(\rho) = \{(\delta, t) \in \mathbb{R} \mid t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + \mathbb{E}^{\eta,\theta}_{\rho[\delta,t]} \geq K - \varepsilon\}$$

where K is the supremum of the function $t \mapsto t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + \mathbb{E}^{\eta,\theta}_{\rho[\delta,t]}$ can be shown to be weakly measurable: for all open set U of $\mathbb{R}$, the set $\{\rho \mid \psi(\rho) \cap U \neq \emptyset\}$ can be generated by a countable union of cylinders, since we have a countable number of paths, and delays are constrained by U (that is measurable) and a supremum (that is also measurable). The selection theorem thus allows us to define a measurable strategy $\tau \in \mathsf{sdStrat}_{\mathsf{Max}}$.

We also define, for all $i \in \mathbb{N}$, the strategy $\xi_i \in \mathsf{Strat}_{\mathsf{Max}}$ consisting in playing τ during the first i steps of the play, and θ for the remaining ones: since τ and θ satisfy Hypothesis 1, the combination ξ_i does too. When i increases, strategy ξ_i looks more and more like τ. In particular, $\mathbb{E}^{\eta,\xi_i}_\rho$ tends to $\mathbb{E}^{\eta,\tau}_\rho$ when i tends to $+\infty$. To prove it formally, we use the fact that η satisfies Hypothesis 2, and we will show (with α and m given by Hypothesis 2):

$$\forall \rho_0 \in \mathsf{FPlays} \qquad \lim_{i\to+\infty} |\mathbb{E}^{\eta,\xi_i}_{\rho_0} - \mathbb{E}^{\eta,\tau}_{\rho_0}| = 0\,. \tag{6.2}$$

Moreover, independantly, we relate expectation defined by θ and ξ_i by proving that

$$\forall i \in \mathbb{N} \quad \forall \rho_0 \in \mathsf{FPlays} \qquad \mathbb{E}^{\eta,\xi_i}_{\rho_0} \geq E^{\eta,\theta}_{\rho_0} - \frac{\varepsilon}{2^{|\rho_0|}}\,. \tag{6.3}$$

When taking the limit when i tends to $+\infty$, we obtain that for all finite plays ρ_0, $\mathbb{E}^{\eta,\tau}_{\rho_0} \geq \mathbb{E}^{\eta,\theta}_{\rho_0} - \frac{\varepsilon}{2^{|\rho_0|}}$. In particular, $\mathbb{E}^{\eta,\tau}_{\rho_0} \geq \mathsf{Val}^\eta(\rho_0) - \varepsilon$. In the case where η is memoryless since memoryless strategies of Max are stochastic strategies of Max, we have, for all configurations (ℓ, ν), $\mathsf{Val}^\eta(\ell, \nu) \geq \mathsf{mVal}^\eta(\ell, \nu)$. Thus, by applying

the previous the result in case where ρ is only a configuration, we deduce that

$$\mathbb{E}^{\eta,\tau}_{\ell,\nu} \geq \mathsf{Val}^{\eta}(\ell,\nu) - \varepsilon \geq \mathsf{mVal}^{\eta}(\ell,\nu) - \varepsilon\,.$$

To conclude the proof, we need to show properties (6.2) and (6.3).

Proof of property (6.2) We consider the expectation under ξ_i versus the one under τ. In particular, by decomposing the sum given for the expectation in Lemma 5.18 according to the length of the path, we have

$$|\mathbb{E}^{\eta,\xi_i}_{\rho_0} - \mathbb{E}^{\eta,\tau}_{\rho_0}| = \Big|\sum_{j=0}^{\infty}\sum_{\pi\in\mathsf{TPaths}^j_\rho}\mathbb{E}^{\eta,\xi_i}_{\rho_0}(\pi_0\pi) - \sum_{j=0}^{\infty}\sum_{\pi\in\mathsf{TPaths}^j_\rho}\mathbb{E}^{\eta,\tau}_{\rho_0}(\pi_0\pi)\Big|$$

where π_0 is the path followed by ρ_0. Letting $k_i = \max(0, i - |\rho_0|)$ be the number of steps after ρ_0 where ξ_i and τ are equal, and noticing that ξ_i and τ behave the same until the step k_i from ρ_0, we deduce that

$$\begin{aligned}|\mathbb{E}^{\eta,\xi_i}_{\rho_0} - \mathbb{E}^{\eta,\tau}_{\rho_0}| &= \Big|\sum_{j=k_i+1}^{\infty}\sum_{\pi\in\mathsf{TPaths}^j_{\rho_0}}\mathbb{E}^{\eta,\xi_i}_{\rho_0}(\pi_0\pi) - \sum_{j=k_i+1}^{\infty}\sum_{\pi\in\mathsf{TPaths}^j_{\rho_0}}\mathbb{E}^{\eta,\tau}_{\rho_0}(\pi_0\pi)\Big| \\ &\leq \sum_{j=k_i+1}^{\infty}\sum_{\pi\in\mathsf{TPaths}^j_{\rho_0}}\big(|\mathbb{E}^{\eta,\xi_i}_{\rho_0}(\pi_0\pi)| + |\mathbb{E}^{\eta,\tau}_{\rho_0}(\pi_0\pi)|\big)\,.\end{aligned}$$

Thus, by Lemma 5.17-2 and Lemma 5.15, we have

$$|\mathbb{E}^{\eta,\xi_i}_{\rho_0} - \mathbb{E}^{\eta,\tau}_{\rho_0}| \leq \sum_{j=k_i+1}^{\infty} 2\,j\,W_{\mathsf{e}}(1-\alpha)^{\lfloor j/m\rfloor}\,.$$

Moreover, since $x \mapsto (1-\alpha)^x$ is decreasing, we obtain that

$$|\mathbb{E}^{\eta,\xi_i}_{\rho_0} - \mathbb{E}^{\eta,\tau}_{\rho_0}| \leq \sum_{j=k+1}^{\infty} 2\,j\,W_{\mathsf{e}}(1-\alpha)^{j/m+1}\,.$$

Noticing that the formal power series $\sum x\beta^{x-1}$ is the formal derivative of the formal power series $\sum \beta^x$, whose sum we can compute, we conclude that

$$|\mathbb{E}^{\eta,\xi_i}_{\rho_0} - \mathbb{E}^{\eta,\tau}_{\rho_0}| \leq 2\,W_{\mathsf{e}}\,\frac{(1-\alpha)^{(k_i+2)/m}\big(k_i(1-(1-\alpha)^{1/m})+1\big)}{(1-(1-\alpha)^{1/m})^2}$$

that tends to 0 when i (and thus k_i) tends to $+\infty$.

Proof of property (6.3) First, we notice that this inequality trivially holds when $|\rho_0| \geq i$, since then strategies ξ_i and θ coincide for the rest of the play. It remains to show that the property holds for all i and plays ρ_0 such that $|\rho_0| \leq i$. We proceed by induction on $i - |\rho_0|$. If $i - |\rho_0| = 0$, we have said before that the inequality trivially holds. Otherwise, we distinguish three cases according to the last location of ρ_0. If ρ_0 ends in L_T, we have

$$\mathbb{E}^{\eta,\xi_i}_{\rho_0} = 0 \geq 0 - \frac{\varepsilon}{2^{|\rho_0|}} = \mathbb{E}^{\eta,\theta}_{\rho_0} - \frac{\varepsilon}{2^{|\rho_0|}}\,.$$

If $\rho_0 \in \mathsf{FPlays}_{\mathsf{Min}}$, by Lemma 6.3,

$$\begin{aligned}\mathbb{E}^{\eta,\xi_i}_{\rho_0} &= \mathcal{H}^{\eta,\xi_i}(\mathbb{E}^{\eta,\xi_i})(\rho_0)\\ &= \sum_{\delta} \int_{I(\rho_0,\delta)} \eta_{\Delta}(\rho_0)(\delta)\left[t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + \mathbb{E}^{\eta,\xi_i}_{\rho_0[\delta,t]}\right] \mathrm{d}\eta_{\mathbb{R}^+}(\rho_0,\delta)(t)\,.\end{aligned}$$

By induction hypothesis, we deduce that

$$\mathbb{E}^{\eta,\xi_i}_{\rho_0} \geq \sum_{\delta} \int_{I(\rho_0,\delta)} \eta_{\Delta}(\rho_0)(\delta)\Big(t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + \mathbb{E}^{\eta,\theta}_{\rho_0[\delta,t]} - \frac{\varepsilon}{2^{|\rho_0|+1}}\Big) \mathrm{d}\eta_{\mathbb{R}^+}(\rho_0,\delta)(t).$$

Thus, by linearity of the integral, we have

$$\begin{aligned}\mathbb{E}^{\eta,\xi_i}_{\rho_0} \geq \sum_{\delta} \int_{I(\rho_0,\delta)} \eta_{\Delta}(\rho_0)(\delta)\Big(t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + \mathbb{E}^{\eta,\theta}_{\rho_0[\delta,t]}\Big) \mathrm{d}\eta_{\mathbb{R}^+}(\rho_0,\delta)(t)\\ - \frac{\varepsilon}{2^{|\rho_0|+1}} \sum_{\delta} \int_{I(\rho_0,\delta)} \eta_{\Delta}(\rho_0)(\delta)\, \mathrm{d}\eta_{\mathbb{R}^+}(\rho_0,\delta)(t).\end{aligned}$$

Since η is a distribution, we obtain that

$$\mathbb{E}^{\eta,\xi_i}_{\rho_0} \geq \mathcal{H}^{\eta,\theta}(\mathbb{E}^{\eta,\theta})(\rho_0) - \frac{\varepsilon}{2^{|\rho_0|+1}} \times 1 \geq \mathcal{H}^{\eta,\theta}(\mathbb{E}^{\eta,\theta})(\rho_0) - \frac{\varepsilon}{2^{|\rho_0|}}\,.$$

If $\rho_0 \in \mathsf{FPlays}_{\mathsf{Max}}$, Lemma 5.18 gives again

$$\mathbb{E}^{\eta,\xi_i}_{\rho_0} = \sum_{\delta} \sum_{\pi \mid \delta\pi \in \mathsf{TPaths}_{\rho_0}} \mathbb{E}^{\eta,\xi_i}_{\rho_0}(\pi_0\delta\pi)\,.$$

Since $|\rho_0| \leq i$, the next step for Max is dictated by the strategy τ, that is deter-

ministic. Letting $(\delta_0, t_0) = \tau(\rho_0)$, we thus have

$$\mathbb{E}^{\eta,\xi_i}_{\rho_0} = (t_0\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta_0)) \sum_{\substack{\pi \\ \delta_0\pi \in \mathsf{TPaths}_{\rho_0}}} \mathbb{P}^{\eta,\xi_i}_{\rho_0[\delta_0,t_0]}(\pi_0\delta_0\pi) + \sum_{\substack{\pi \\ \delta_0\pi \in \mathsf{TPaths}_{\rho_0}}} \mathbb{E}^{\eta,\xi_i}_{\rho_0[\delta_0,t_0]}(\pi_0\delta_0\pi)\,.$$

Moreover, since η is proper, we know that

$$\sum_{\pi \mid \delta_0\pi \in \mathsf{TPaths}_{\rho_0}} \mathbb{P}^{\eta,\xi_i}_{\rho_0[\delta_0,t_0]}(\mathsf{TPaths}_{\rho_0[\delta_0,t_0]}) = 1\,,$$

and we deduce that $\mathbb{E}^{\eta,\xi_i}_{\rho_0} = t_0\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta_0) + \mathbb{E}^{\eta,\xi_i}_{\rho_0[\delta_0,t_0]}$. By the induction hypothesis, we obtain that

$$\mathbb{E}^{\eta,\xi_i}_{\rho_0} \geq t_0\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta_0) + \mathbb{E}^{\eta,\theta}_{\rho_0[\delta_0,t_0]} - \frac{\varepsilon}{2^{|\rho_0|+1}}\,.$$

Thus, by definition of τ and letting (ℓ, ν) be the last configuration of ρ_0, we get

$$\mathbb{E}^{\eta,\xi_i}_{\rho_0} \geq \sup_{(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')} \left(t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + \mathbb{E}^{\eta,\theta}_{\rho_0[\delta,t]}\right) - \frac{\varepsilon}{2^{|\rho_0|+1}} - \frac{\varepsilon}{2^{|\rho_0|+1}}.$$

Since θ is a distribution, we have

$$\begin{aligned}\mathbb{E}^{\eta,\xi_i}_{\rho_0} &\geq \sum_{\delta} \int_{I(\rho_0,\delta)} \theta_{\Delta}(\rho_0)(\delta)\left(t\,\mathsf{wt}(\ell) + \mathsf{wt}(\delta) + \mathbb{E}^{\eta,\theta}_{\rho_0[\delta,t]}\right) \mathrm{d}\theta_{\mathbb{R}^+}(\rho_0,\delta)(t) - \frac{\varepsilon}{2^{|\rho_0|}} \\ &= \mathcal{H}^{\eta,\theta}(\mathbb{E}^{\eta,\theta})(\rho_0) - \frac{\varepsilon}{2^{|\rho_0|}}.\end{aligned}$$

We conclude the proof in all cases, by Lemma 6.3:

$$\mathbb{E}^{\eta,\xi_i}_{\rho_0} \geq \mathcal{H}^{\eta,\theta}(\mathbb{E}^{\eta,\theta})(\rho_0) - \frac{\varepsilon}{2^{|\rho_0|}} = \mathbb{E}^{\eta,\theta}_{\rho_0} - \frac{\varepsilon}{2^{|\rho_0|}}.$$

6.2. Stochastic values are bounded by the smooth deterministic ones

We are now able to come back to the proof of Theorem 6.1 and prove the sequence of inequality (in the general case). The first inequality shows that the stochastic values are at most equal to the smooth deterministic upper-value:

Lemma 6.4. *In all WTGs, for all configurations* (ℓ, ν), $\overline{\mathsf{Val}}(\ell, \nu) \leq \overline{\mathsf{sdVal}}(\ell, \nu)$.

Proof. Let σ be a proper smooth deterministic strategy for Min. Let $\varepsilon > 0$. By

Proposition 6.2, there exists $\tau \in \mathsf{sdStrat}_{\mathsf{Max}}$ such that

$$\mathbf{P}\big(\mathsf{Play}((\ell,\nu),\sigma,\tau)\big) = \mathbb{E}^{\sigma,\tau}_{\ell,\nu} \geq \mathsf{Val}^{\sigma}(\ell,\nu) - \varepsilon\,.$$

Therefore

$$\sup_{\tau\in\mathsf{sdStrat}_{\mathsf{Max}}} \mathbf{P}\big(\mathsf{Play}((\ell,\nu),\sigma,\tau)\big) \geq \mathsf{Val}^{\sigma}(\ell,\nu) - \varepsilon \geq \mathsf{Val}^{\sigma}(\ell,\nu)$$

since this holds for all ε. By considering the infimum over all proper smooth deterministic strategies, we obtain

$$\overline{\mathsf{sdVal}}_{\ell,\nu} = \inf_{\sigma\in\mathsf{sdStrat}^{\mathsf{p}}_{\mathsf{Min}}} \sup_{\tau\in\mathsf{sdStrat}_{\mathsf{Max}}} \mathbf{P}\big(\mathsf{Play}((\ell,\nu),\sigma,\tau)\big) \geq \inf_{\sigma\in\mathsf{sdStrat}^{\mathsf{p}}_{\mathsf{Min}}} \mathsf{Val}^{\sigma}(\ell,\nu)\,.$$

Since $\mathsf{sdStrat}^{\mathsf{p}}_{\mathsf{Min}} \subseteq \mathsf{Strat}^{\mathsf{p}}_{\mathsf{Min}}$, the infimum over all proper strategies of Min is at most the infimum over deterministic strategies:

$$\overline{\mathsf{sdVal}}(\ell,\nu) \geq \inf_{\eta\in\mathsf{Strat}^{\mathsf{p}}_{\mathsf{Min}}} \mathsf{Val}^{\eta}(\ell,\nu) = \overline{\mathsf{Val}}(\ell,\nu)\,. \qquad \square$$

The second one shows that the smooth deterministic lower-value is at most equal to the stochastic values.

Lemma 6.5. *In all WTGs, for all configurations* (ℓ,ν), $\underline{\mathsf{sdVal}}(\ell,\nu) \leq \underline{\mathsf{Val}}(\ell,\nu)$.

Proof. Let $\varepsilon > 0$. If $\underline{\mathsf{sdVal}}(\ell,\nu) = -\infty$, the result is trivial. Otherwise, by definition, Max has a deterministic ε-optimal strategy τ for the smooth deterministic lower-value. Against any proper smooth deterministic strategy of Min, from (ℓ,ν), it guarantees a weight at least $\underline{\mathsf{sdVal}}(\ell,\nu) - \varepsilon$.

Let η be any proper strategy of Min. Every play conforming to η and τ has a weight at least $\underline{\mathsf{sdVal}}(\ell,\nu) - \varepsilon$ (since it is conforming to τ), so that, by definition of the expectation,

$$\mathbb{E}^{\eta,\tau}_{\ell,\nu} = \int_{\rho} \mathbf{P}(\rho)\,\mathrm{d}\mathbb{P}^{\eta,\tau}_{\ell,\nu}(\rho) \geq \underline{\mathsf{sdVal}}(\ell,\nu) - \varepsilon\,.$$

Since this inequality holds for all proper strategies of Min, we deduce that

$$\inf_{\eta\in\mathsf{Strat}^{\mathsf{p}}_{\mathsf{Min}}} \mathbb{E}^{\eta,\tau}_{\ell,\nu} \geq \underline{\mathsf{sdVal}}(\ell,\nu) - \varepsilon\,.$$

By taking the supremum over all strategies of Max (that contain the smooth deterministic strategy τ), we have

$$\sup_{\theta\in\mathsf{Strat}_{\mathsf{Max}}} \inf_{\eta\in\mathsf{Strat}^{\mathsf{p}}_{\mathsf{Min}}} \mathbb{E}^{\eta,\tau}_{\ell,\nu} \geq \underline{\mathsf{sdVal}}(\ell,\nu) - \varepsilon\,.$$

In particular, $\underline{\mathsf{Val}}(\ell,\nu) \geq \underline{\mathsf{sdVal}}(\ell,\nu) - \varepsilon$. To conclude, we remark that this inequality holds for all $\varepsilon > 0$, and thus also when ε is set to 0. □

Combined with the fact that $\underline{\mathsf{Val}}(\ell,\nu) \leq \overline{\mathsf{Val}}(\ell,\nu)$, we indeed proved the inequalities of Theorem 6.1.

6.3. Equality between the stochastic and deterministic values

To conclude the proof of Theorem 6.1, we need to prove the equality between values in the case of divergent WTGs, one-clock WTGs and SPGs. To do it, we give a necessary condition to obtain equality: the existence of almost-optimal smooth deterministic strategies w.r.t. the deterministic value.

Lemma 6.6. *In all WTGs for which there exist* $\sigma \in \mathsf{sdStrat}^{\mathsf{p}}_{\mathsf{Min}}$ *and* $\tau \in \mathsf{sdStrat}_{\mathsf{Max}}$ *two almost-optimal strategies w.r.t. the deterministic value, for all configurations* (ℓ,ν)*, we have*

$$\mathsf{dVal}(\ell,\nu) = \underline{\mathsf{sdVal}}(\ell,\nu) = \overline{\mathsf{sdVal}}(\ell,\nu) = \underline{\mathsf{Val}}(\ell,\nu) = \overline{\mathsf{Val}}(\ell,\nu)\,.$$

Proof. Let $\varepsilon > 0$, there exists $\sigma \in \mathsf{sdStrat}^{\mathsf{p}}_{\mathsf{Min}}$ and $\tau \in \mathsf{sdStrat}_{\mathsf{Max}}$ such that

$$\mathsf{dVal}^{\sigma}(\ell,\nu) - \varepsilon \leq \mathsf{dVal}(\ell,\nu) \leq \mathsf{dVal}^{\tau}(\ell,\nu) + \varepsilon$$

by hypothesis. First, since $\mathsf{sdStrat}_{\mathsf{Max}} \subseteq \mathsf{dStrat}_{\mathsf{Max}}$, we deduce that

$$\begin{aligned}\overline{\mathsf{sdVal}}(\ell,\nu) &\leq \sup_{\tau\in\mathsf{sdStrat}_{\mathsf{Max}}} \mathbf{P}\big(\mathsf{Play}((\ell,\nu),\sigma,\tau)\big)\\ &\leq \sup_{\tau\in\mathsf{dStrat}_{\mathsf{Max}}} \mathbf{P}\big(\mathsf{Play}((\ell,\nu),\sigma,\tau)\big) = \mathsf{dVal}^{\sigma}(\ell,\nu) - \varepsilon\,.\end{aligned}$$

Next, with the same reasoning, since $\mathsf{sdStrat}^{\mathsf{p}}_{\mathsf{Min}} \subseteq \mathsf{dStrat}_{\mathsf{Min}}$, we deduce that

$$\begin{aligned}\mathsf{dVal}^{\tau}(\ell,\nu) - \varepsilon &= \inf_{\sigma\in\mathsf{dStrat}_{\mathsf{Min}}} \mathbf{P}\big(\mathsf{Play}((\ell,\nu),\sigma,\tau)\big)\\ &\leq \inf_{\sigma\in\mathsf{sdStrat}^{\mathsf{p}}_{\mathsf{Min}}} \mathbf{P}\big(\mathsf{Play}((\ell,\nu),\sigma,\tau)\big) \leq \underline{\mathsf{sdVal}}(\ell,\nu)\,.\end{aligned}$$

Thus, $\underline{\mathsf{sdVal}}(\ell,\nu) = \overline{\mathsf{sdVal}}(\ell,\nu)$, since $\underline{\mathsf{sdVal}}(\ell,\nu) \leq \overline{\mathsf{sdVal}}(\ell,\nu)$. Finally, we conclude the proof by applying the first part of Theorem 6.1 (i.e. the sequence of inequalities):

$$\underline{\mathsf{sdVal}}(\ell,\nu) \leq \underline{\mathsf{Val}}(\ell,\nu) \leq \overline{\mathsf{Val}}(\ell,\nu) \leq \overline{\mathsf{sdVal}}(\ell,\nu) = \underline{\mathsf{sdVal}}(\ell,\nu)\,.$$ □

Finally, we prove that divergent WTGs, one-clock WTGs and SPGs satisfy the condition of the previous lemma. Thus, we conclude the proof of Theorem 6.1.

The case of SPGs Since SPGs are special cases of WTGs, the definition of stochastic strategies and values from WTGs can be adapted. In this context, all distributions over transitions are finite; thus Hypothesis 1 always holds. Moreover, all strategies are discrete, and if the strategies of Min are supposed to fulfil Hypothesis 2, various values exist. In particular, smooth deterministic values are defined only by a restriction over strategies of Min, i.e. $\mathsf{sdStrat}_{\mathsf{Max}} = \mathsf{dStrat}_{\mathsf{Max}}$.

With the existence of a switching optimal strategy with respect to the deterministic value for Min (by Theorem 3.6), we obtain the existence of smooth deterministic strategies that satisfy Hypothesis 2 (we take m the bound over the length of all plays conforming to the switching strategy, and $\alpha \in (0, 1)$ since the probability to reach the target is equal to 1). Thus, by Proposition 5.14, we obtain the existence of proper smooth deterministic optimal strategies with respect to the deterministic value for Min in all SPGs. Finally, from Lemma 6.6, we deduce that SPGs are determined with respect to the stochastic value, and the stochastic and (smooth) deterministic values are equal.

The case of divergent WTGs To prove that divergent WTGs satisfy the necessary condition of Lemma 6.6, we use the construction of almost-optimal strategies for Min and Max in such WTGs given in Section 3.2.

For Min, we note that the switching strategy $\sigma = \langle \sigma^1, \sigma^2, \kappa \rangle$ defined in the proof of Proposition 3.10 is a proper smooth deterministic strategy, i.e. satisfies Hypotheses 1 and 2. Indeed, it is easy to choose σ^2 that is an attractor strategy smooth enough so that it fulfils Hypothesis 1. The strategy σ^1 needs more care. In the $-\infty$ case, σ^1 is also an attractor strategy to a Büchi objective and it is easy to choose σ^1 smooth enough so that it fulfils Hypothesis 1. Otherwise, knowing that the mapping $\nu \mapsto \mathsf{dVal}(\ell, \nu)$ is piecewise affine by Theorem 1.34, it is possible to choose σ^1 (via the argsup operator) so that it fulfils the measurability (even piecewise continuity) conditions of Hypothesis 1. More precisely, we can consider it to take the same kind of decision for all configurations of the same cell: same transition, and either no delay or a delay jumping to the same border of the cell. Thus, σ is defined to be a smooth deterministic strategy. Next, we note that all plays conforming to σ have a length bounded by a constant (i.e. $\kappa\,|Q|$ where Q is the set of regions state of the WTGs). In particular, we have $\sigma \in \mathsf{sdStrat}^{\mathsf{P}}_{\mathsf{Min}}$.

According to Max's point of view, we note that the memoryless almost-optimal strategy defined in Section 3.2.2 can be chosen smooth enough by the same process as σ^1. Thus, from Lemma 6.6, we deduce that divergent WTGs are determined w.r.t. the stochastic value, and the stochastic and (smooth) deterministic values

are equal.

The case of one-clock WTGs To prove that one-clock WTGs satisfy the necessary condition of Lemma 6.6, we use the constructions of the closure and the unfolding of the WTG given in Chapter 4.

For Min, we first consider the unfolding of the WTG. Since the deterministic value of a one-clock WTG is a fixpoint of the operator $\mathcal{F}$ defined in Chapter 2, in the unfolding we can define an almost-optimal strategy σ for Min with the same process as σ^1 in the case of divergent WTGs. Moreover, since the unfolding is an acyclic WTGs and the deterministic value function is piecewise affine over regions (by Theorem 1.32), we know that $\sigma \in \mathsf{sdStrat}^{\mathsf{p}}_{\mathsf{Min}}$. Now, by function proj (defined in Section 4.3.2) that does not modify the length of plays, we can define σ' be as an almost-optimal proper smooth deterministic strategy in the closure of the WTG. Finally, by applying the same process as in [Bri+22, Lemma 6.1], we can define an almost-optimal smooth deterministic strategy from σ': by letting K be the bound on the length of plays, we know a bound on the length of all plays conforming to σ, we just perturb by ε/K each non-feasible decision given by σ. Since decisions are uniform over each cell, this transformation does not add any memory, i.e. the length of plays does not increase. Thus, this strategy also satisfies Hypothesis 2, and the strategy is proper smooth deterministic by Proposition 5.14.

We note that the memoryless almost-optimal strategy for Max defined in Section 4.2 can be chosen smooth enough by the same processes as σ^1 in the closure of the WTG. Thus, by applying the same process as in [Bri+22, Lemma 6.1], we define a smooth deterministic strategy in the WTG. In this case, the length of plays is not a constraint, so we can choose the new delay according to the length of the prefix to which we apply the strategy (as illustrated in Example 4.6). Again, from Lemma 6.6, we deduce that one-clock WTGs are determined w.r.t. the stochastic value, and the stochastic and (smooth) deterministic values are equal.

7. Trading Memory for Probabilities

Our main motivation is to emulate the memory with probability in strategies used to play almost-optimally in WTGs. The main contribution of this part is the tradeoff between memory and probabilities in divergent WTGs and SPGs, i.e. in these classes of WTGs, we prove the equality between the memoryless values and the deterministic one. Moreover, this equality gives a characterisation of memoryless optimality in SPGs. Finally, we prove that this tradeoff does not hold for all WTGs by giving an example of a one-clock WTG in which this equality does not hold.

Table of contents

7.1. Memoryless value in divergent weighted timed games 165
7.1.1. Definition of the proper memoryless strategy η^p 167
7.1.2. η^p is almost-optimal for the deterministic value 171
7.2. Memoryless value in shortest path games 175
7.3. Characterisation of memoryless optimality in shortest path games . 183

It remains to study the memoryless values and compare them with the deterministic value. We will be able to obtain the equality between the memoryless values and the deterministic one in some classes of WTGs that we present in this chapter.

Theorem 7.1. *The memoryless value is determined and equal to the deterministic value in divergent WTGs, and SPGs.*

Unfortunately, we cannot hope for a similar result as the one above for all WTGs. Indeed, the following example shows that all WTGs are not determined with respect to memoryless value.

Example 7.2. We consider the one-clock WTG depicted in Figure 7.1, and show that $\overline{\mathsf{mVal}}(\ell_0, 0) = 0 > -9 \geq \underline{\mathsf{mVal}}(\ell_0, 0)$.

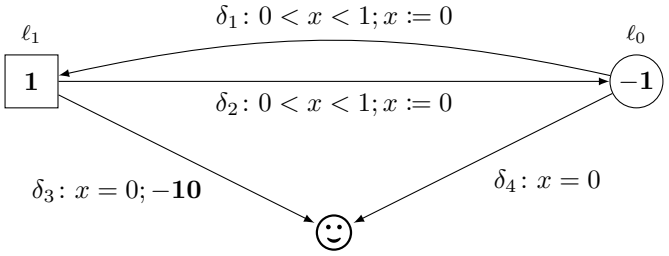


Figure 7.1 – A one-clock WTG where $\overline{\mathsf{mVal}}(\ell_0, 0) > \underline{\mathsf{mVal}}(\ell_0, 0)$.

To do so, we consider a proper memoryless strategy η of Min. First, we remark that there exists $p \in [0, 1)$ such that

$$\eta_{\vartriangle}(\ell_0, 0) = p \times \mathsf{Dirac}_{\delta_1} + (1 - p) \times \mathsf{Dirac}_{\delta_4} .$$

Moreover, since ℓ_0 and ℓ_1 are always reached after a reset, i.e. with the valuation 0, η always chooses the same delay in ℓ_0 that does not impact the next choice of the memoryless strategy of Max. Formally, let $t \in (0, 1)$ be the expected delay in ℓ_0 (from the valuation 0) according to $\eta_{\mathbb{R}^+}(\ell_0, 0)(\delta_1)$, and we define the discrete strategy η' by

$$\eta'(\ell_0, 0) = p \times \mathsf{Dirac}_{\delta_1, t} + (1 - p) \times \mathsf{Dirac}_{\delta_4, 0} .$$

For all memoryless strategies θ of Max, $\mathbb{E}^{\eta,\theta}_{\ell_0,0} = \mathbb{E}^{\eta',\theta}_{\ell_0,0}$. Now, we consider the deterministic memoryless strategy τ of Max defined by $\tau(\ell_1, 0) = (\delta_2, (1 + t)/2)$. Then, we have

$$\mathbb{E}^{\eta',\tau}_{\ell_0,0} = \sum_{i=0}^{+\infty} \frac{1 - t}{2} i p^i = \frac{1 - t}{2} \frac{p}{(1 - p)^2} \geq 0$$

and this expectation is null when $p = 0$, thus $\overline{\mathsf{mVal}}(\ell_0, 0) = 0$.

By a similar reasoning, all memoryless strategies θ of Max are equivalent to a discrete strategy θ' described by the probability $q = \theta_{\vartriangle}(\ell_1, 0) \in [0, 1)$ and the expected delay $t \in (0, 1)$ spent in ℓ_1 (from the valuation 0) according to $\theta_{\mathbb{R}^+}(\ell_1, 0)(\delta_2)$: we have

$$\theta'(\ell_1, 0) = q \times \mathsf{Dirac}_{\delta_2, t} + (1 - q) \times \mathsf{Dirac}_{\delta_3, 0} .$$

We define a memoryless strategy η for Min by letting

$$\eta(\ell_0, 0) = p \times \mathsf{Dirac}_{\delta_1, (1+t)/2} + (1 - p) \times \mathsf{Dirac}_{\delta_4, 0} .$$

By Lemma 6.3, we have $\mathbb{E}^{\eta,\theta'}_{\ell_0,0} = \left(\mathbb{E}^{\eta,\theta'}_{\ell_1,0} - \frac{1+t}{2}\right) \times p$ and $\mathbb{E}^{\eta,\theta'}_{\ell_1,0} = -10(1 - q) + (t +$

$\mathbb{E}^{\eta,\theta'}_{\ell_0,0}) \times q$. In particular, we deduce that

$$\begin{aligned}\mathbb{E}^{\eta,\theta'}_{\ell_0,0} &= \left(-10 \times (1-q) + (\mathbb{E}^{\eta,\theta'}_{\ell_0,0} + t) \times q - \frac{1+t}{2}\right) \times p \\ &= \left(-10(1-q) - \frac{t(1-2q)+1}{2}\right) \frac{p}{1-pq}\,.\end{aligned}$$

We prove that $\underline{\mathsf{mVal}}(\ell_0, 0) \leq -9$ by distinguishing two cases.

- We suppose that $q = 1$, i.e. θ' is a deterministic strategy such that $\theta'(\ell_1, 0) = (\delta_2, t)$. In this case, we have

  $$\mathbb{E}^{\eta,\theta'}_{\ell_0,0} = -\frac{1-t}{2}\,\frac{p}{1-p}\,.$$

  Now, as we control the strategy of Min, we can fix $p = \frac{18}{19-t} < 1$ (since $t < 1$), and we obtain

  $$\mathbb{E}^{\eta,\theta'}_{\ell_0,0} = -\frac{1-t}{2}\,\frac{18}{(19-t)(1-\frac{18}{19-t})} = -\frac{1-t}{2}\,\frac{18}{1-t} = -9\,.$$

- We suppose that $0 \leq q < 1$, we deduce that $t(1-2q)+1 \geq 0$ (since $0 < t < 1$). Thus,

  $$\mathbb{E}^{\eta,\theta'}_{\ell_0,0} \leq -10(1-q) \times \frac{p}{1-pq}\,.$$

  Again, as we control the strategy of Min, we can fix $p = \frac{9}{10-q} < 1$ (since $q < 1$), and

  $$\mathbb{E}^{\eta,\theta'}_{\ell_0,0} \leq -10(1-q)\,\frac{9}{(10-q)(1-\frac{9}{10-q}q)} = -10(1-q)\,\frac{9}{10(1-q)} = -9\,.$$

Thus, we conclude that $\underline{\mathsf{mVal}}(\ell_0, 0) \leq -9 < 0 = \overline{\mathsf{mVal}}(\ell_0, 0)$ as expected. ⌟

Moreover, even if we wanted to restrict our study to one of the memoryless values (upper or lower values), the following example shows a case where memoryless lower and upper values are equal, but different from the deterministic value.

Example 7.3. We consider the one-clock WTG depicted in Figure 7.2. We explain why $\mathsf{dVal}(\ell_1, 0) \neq \overline{\mathsf{mVal}}(\ell_1, 0) = \underline{\mathsf{mVal}}(\ell_1, 0)$. Since Min has a way to obtain the weight 0 by using transition δ_2 (and thus with a memoryless and deterministic strategy), all values $\mathsf{dVal}(\ell_1, 0)$, $\overline{\mathsf{mVal}}(\ell_1, 0)$, $\underline{\mathsf{mVal}}(\ell_1, 0)$ are non-positive.

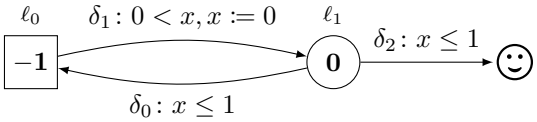


Figure 7.2 – One-clock WTG where $\mathsf{dVal}(\ell_1, 0) \neq \overline{\mathsf{mVal}}(\ell_1, 0) = \underline{\mathsf{mVal}}(\ell_1, 0)$.

We consider the deterministic strategy τ for Max defined, for all plays ρ with (ℓ_0, ν) as the last configuration, by

$$\tau(\rho) = \begin{cases} (\delta_1, \frac{\varepsilon}{2^{|\rho|}}) & \text{if } \nu = 0; \\ (\delta_1, 0) & \text{if } \nu > 0. \end{cases}$$

Since each iteration of the cycle gives a cumulated weight of at most $-\frac{\varepsilon}{2^{|\rho|}}$, then $\mathsf{dVal}^\tau(\ell_1, 0) \geq -\varepsilon$ for all $\varepsilon > 0$. In particular, $\mathsf{dVal}(\ell_1, 0)$ is also non-positive, which implies that $\mathsf{dVal}(\ell_1, 0) = 0$. Notice that the strategy τ uses memory, to compute the length $|\rho|$ of the current play.

Now, we consider the memoryless strategy for Min such that, for $p \in (0, 1)$, chooses transition δ_0 with delay 0 with probability $1 - p$ and transition δ_2 with delay 0 with probability p. When Max follows a memoryless strategy, each time we pass through the configuration $(\ell_0, 0)$, the same expected choice is always a small amount of time before taking transition δ_1. In particular, the two memoryless strategies combined allow one to expect $1/(1-p)$ turns in the cycle, each turn having a fixed negative weight. Thus, Min can secure a weight as little as possible, by making p approaches 1. Thus $\overline{\mathsf{mVal}}(\ell_1, 0) = -\infty$. And since $\underline{\mathsf{mVal}}(\ell_1, 0) \leq \overline{\mathsf{mVal}}(\ell_1, 0)$, we also have $\underline{\mathsf{mVal}}(\ell_1, 0) = -\infty$. ⌟

7.1. Memoryless value in divergent weighted timed games

We study the memoryless values of divergent WTGs showing that memory can be fully emulated with stochastic choices and vice versa. Moreover, we show that divergent WTGs are determined with respect to the memoryless value (as we obtained previously for the stochastic value). In particular, we prove Theorem 7.1 in the case of divergent WTGs by showing that in all divergent WTGs, for all configurations (ℓ, ν), we have

$$\mathsf{dVal}(\ell, \nu) = \overline{\mathsf{mVal}}(\ell, \nu) = \underline{\mathsf{mVal}}(\ell, \nu)\,.$$

This proof is decomposed into two inequalities:

- we show in Lemma 7.4 that $\mathsf{dVal}(\ell, \nu) \leq \underline{\mathsf{mVal}}(\ell, \nu)$, relying on the existence of almost-optimal deterministic and memoryless strategies of Max w.r.t. the deterministic value (Theorem 3.2-2);
- we show in Proposition 7.5 that memoryless stochastic strategies can emulate deterministic ones: $\overline{\mathsf{mVal}}(\ell, \nu) \leq \mathsf{dVal}(\ell, \nu)$.

Combined with the fact that $\underline{\mathsf{mVal}}(\ell, \nu) \leq \overline{\mathsf{mVal}}(\ell, \nu)$, we indeed obtain the desired equalities.

As shown in Example 7.3, the theorem does not hold for all WTGs, and the divergence property is therefore useful (though it might be possible to weaken it). We start by showing that the deterministic value is at most equal to the memoryless lower value:

Lemma 7.4. *In all divergent WTGs, for all configurations* (ℓ, ν), $\mathsf{dVal}(\ell, \nu) \leq \underline{\mathsf{mVal}}(\ell, \nu)$.

Proof. Let $\varepsilon > 0$. By Theorem 3.2-2, Max has a memoryless deterministic ε-optimal strategy τ w.r.t. the deterministic value. Against any deterministic strategy of Min, from (ℓ, ν), it guarantees a weight at least $\mathsf{dVal}(\ell, \nu) - \varepsilon$.

Let η be any memoryless proper strategy of Min. Every play conforming to η and τ has a weight at least $\mathsf{dVal}(\ell, \nu) - \varepsilon$ (since it is conforming to τ), so that, by definition of the expectation,

$$\mathbb{E}^{\eta,\tau}_{\ell,\nu} = \int_{\rho} \mathbf{P}(\rho) \, \mathrm{d}\mathbb{P}^{\eta,\tau}_{\ell,\nu}(\rho) \geq \mathsf{dVal}(\ell, \nu) - \varepsilon \,.$$

Since this inequality holds for all proper strategies of Min, we deduce that

$$\inf_{\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}} \mathbb{E}^{\eta,\tau}_{\ell,\nu} \geq \mathsf{dVal}(\ell, \nu) - \varepsilon \,.$$

By taking the supremum over all memoryless strategies of Max (that contain the memoryless and deterministic strategy τ), we have

$$\sup_{\theta \in \mathsf{mStrat}_{\mathsf{Max}}} \inf_{\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu} \geq \mathsf{dVal}(\ell, \nu) - \varepsilon \,.$$

In particular, $\underline{\mathsf{mVal}}(\ell, \nu) \geq \mathsf{dVal}(\ell, \nu) - \varepsilon$. To conclude, we remark that this inequality holds for all $\varepsilon > 0$, and thus also when ε is set to 0. □

The most technical inequality is the remaining one, showing that we can simulate deterministic strategies with memoryless strategies:

Proposition 7.5. *In all divergent WTGs, for all configurations* (ν, ℓ),

$$\overline{\mathsf{mVal}}(\ell, \nu) \leq \mathsf{dVal}(\ell, \nu)\,.$$

We prove this result in the rest of this section. To do so, we build a memoryless strategy of Min at least as good as a deterministic strategy. By Theorem 3.2-1, we can start from a switching strategy for Min. For $N \in \mathbb{N}$ and $\varepsilon > 0$, we thus consider a switching strategy $\sigma = (\sigma^1, \sigma^2, \kappa)$ of deterministic value $\mathsf{dVal}^\sigma(\ell, \nu) \leq \max(-N, \mathsf{dVal}(\ell, \nu)) + \varepsilon$, and simulate it with a memoryless strategy for Min, denoted η^p, with a probability parameter $p \in (0, 1)$. This new strategy is a probabilistic superposition of the two memoryless deterministic strategies σ^1 and σ^2.

7.1.1. Definition of the proper memoryless strategy η^p

The definition of $\eta^p(\ell, \nu)$, with $\ell \in L_{\mathsf{Min}}$, depends on the sign of the SCC[1] containing the location (ℓ, r) of the region game $\mathcal{R}(\mathcal{G})$ with r the region of ν. In a positive SCC, Min always chooses to play σ^1, thus looking for a negative cycle in the next SCCs (in topological order) if any, i.e. we let $\eta^p(\ell, \nu) = \mathsf{Dirac}_{\sigma^1(\ell,\nu)}$. In a negative SCC, Min chooses to play σ^1 with probability p, and σ^2 with probability $1 - p$, i.e. we let

$$\eta^p(\ell, \nu) = p \times \mathsf{Dirac}_{\sigma^1(\ell,\nu)} + (1 - p) \times \mathsf{Dirac}_{\sigma^2(\ell,\nu)}\,.$$

These choices can be decomposed as first choosing a transition and then a delay as follows: letting $(\delta_1, t_1) = \sigma^1(\ell, \nu)$, and $(\delta_2, t_2) = \sigma^2(\ell, \nu)$, we define

$$\eta^p_\Delta(\ell, \nu) = \begin{cases} \mathsf{Dirac}_{\delta_1} & \text{if } \ell \text{ is in a positive SCC;} \\ p \times \mathsf{Dirac}_{\delta_1} + (1 - p) \times \mathsf{Dirac}_{\delta_2} & \text{otherwise;} \end{cases}$$

and $\eta^p_{\mathbb{R}^+}(\ell, \nu)(\delta)$ is equal to

$$\begin{cases} \mathsf{Dirac}_{t_1} & \text{if } \ell \text{ is in a positive SCC;} \\ p \times \mathsf{Dirac}_{t_1} + (1 - p) \times \mathsf{Dirac}_{t_2} & \text{if } \ell \text{ is in a negative SCC, and } \delta_1 = \delta_2; \\ \mathsf{Dirac}_{t_1} & \text{if } \ell \text{ is in a negative SCC, and } \delta = \delta_1; \\ \mathsf{Dirac}_{t_2} & \text{if } \ell \text{ is in a negative SCC, and } \delta = \delta_2. \end{cases}$$

Section 6.3 ensures that strategies σ^1 and σ^2 satisfy Hypothesis 1. Thus, the

[1] Since $\mathcal{G}$ is divergent, by [BMR17, Proposition 5], the region game of $\mathcal{G}$ is split into SCC where all cyclic paths are only negative weights (called negative SCC) or only positive weights (called positive SCC).

superposition η^p also satisfies this hypothesis. Moreover, we use the sufficient condition in Hypothesis 2 to show that η^p is also proper:

Lemma 7.6. *For all $p \in (0,1)$, the strategy $\eta^p \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}$, i.e. it satisfies Hypothesis 2.*

Proof. We use Proposition 5.14, and thus only prove that η^p satisfies Hypothesis 2, with $m = |Q|$ and $\alpha = (1-p)^{m+1}$ where Q is the set of region states. Let $\theta \in \mathsf{Strat}_{\mathsf{Max}}$, and ρ_0 a finite play. We prove that $\mathbb{P}^{\eta^p,\theta}_{\rho_0}(\bigcup_{n\leq m} \mathsf{TPlays}^n_{\rho_0}) \geq \alpha$. Not reaching the target in at most m steps is equivalent to having a prefix of length $m+1$, so that

$$\mathbb{P}^{\eta^p,\theta}_{\rho_0}\Big(\bigcup_{n\leq m} \mathsf{TPlays}^n_{\rho_0}\Big) = 1 - \mathbb{P}^{\eta^p,\theta}_{\rho_0}\Big(\bigcup_{\pi||\pi|=m+1} \mathsf{Cyl}_{\rho_0}(\pi)\Big)\,.$$

We thus prove that $\mathbb{P}^{\eta^p,\theta}_{\rho_0}(\bigcup_{\pi||\pi|=m+1} \mathsf{Cyl}_{\rho_0}(\pi)) \leq 1-\alpha$.

Since $m = |Q|$, a play is in $\bigcup_{\pi||\pi|=m+1} \mathsf{Cyl}_{\rho_0}(\pi)$ if and only if it is of the form $\rho_0 \cdot \rho \cdot \rho'$ with $|\rho| = m+1$: in particular, ρ contains a cyclic path in the region game, so cannot be conforming to the attractor strategy σ_2. As a consequence, ρ goes through a transition played by σ_1, chosen with probability p in η^p. Given $k \geq 1$, calling $S^k_{\rho_0}$ the set of plays $\rho_0 \cdot \rho \cdot \rho'$ with ρ of length k containing a transition of probability p chosen by η^p, we have

$$\mathbb{P}^{\eta^p,\theta}_{\rho_0}\Big(\bigcup_{\pi||\pi|=m+1} \mathsf{Cyl}_{\rho_0}(\pi)\Big) = \mathbb{P}^{\eta^p,\theta}_{\rho_0}(S^{m+1}_{\rho_0})\,.$$

We conclude by showing that $\mathbb{P}^{\eta^p,\theta}_{\rho_0}(S^k_{\rho_0}) \leq 1-(1-p)^k$ holds for all $k \geq 1$ by induction. We let π_0 be the path followed by ρ_0.

When $k=1$, the transition of every play in $S^1_{\rho_0}$ after the common prefix ρ_0 is chosen by Min with a probability p. Thus, $\rho_0 \in \mathsf{FPlays}_{\mathsf{Min}}$,

$$\eta^p(\rho_0) = p \times \mathsf{Dirac}_{(\delta_0,t_0)} + (1-p) \times \mathsf{Dirac}_{(\delta'_0,t'_0)}$$

with $(\delta_0,t_0) \neq (\delta'_0,t'_0)$ and the transition of every play in $S^1_{\rho_0}$ after the common prefix ρ_0 is (δ_0,t_0). Indeed, $S^1_{\rho_0} = \mathsf{Cyl}_{\rho_0}(\pi_0\cdot\delta_0, \mathbb{R}^{|\pi_0|}_{\geq 0} \times \{t_0\})$. As a consequence,

$$\begin{aligned}
\mathbb{P}^{\eta^p,\theta}_{\rho_0}(S^1_{\rho_0}) &= \mathbb{P}^{\eta^p,\theta}_{\rho_0}\big(\mathsf{Cyl}_{\rho_0}(\pi_0\cdot\delta_0, \mathbb{R}^{|\pi_0|}_{\geq 0} \times \{t_0\})\big) \\
&= \eta^p(\rho_0)(\delta_0,t_0) \times \mathbb{P}^{\eta^p,\theta}_{\rho_0\delta_0,t_0}\big(\mathsf{Cyl}_{\rho_0[\delta_0,t_0]}(\pi_0\cdot\delta_0, \mathbb{R}^{|\pi_0|}_{\geq 0} \times \{t_0\})\big) \\
&= p \times 1 = 1-(1-p)^1\,.
\end{aligned}$$

When $k \geq 2$, we distinguish three cases.

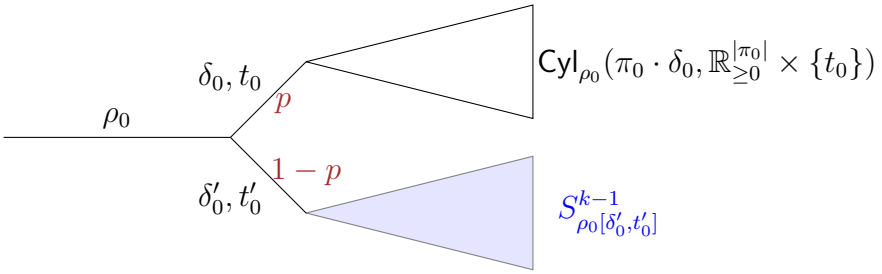


Figure 7.3 – The partition of $S^k_{\rho_0}$ when $\rho_0 \in \mathsf{FPlays}_{\mathsf{Min}}$.

- If $\rho_0 \in \mathsf{FPlays}_{\mathsf{Min}}$ and $\eta^p(\rho_0) = p \times \mathsf{Dirac}_{(\delta_0,t_0)} + (1-p) \times \mathsf{Dirac}_{(\delta'_0,t'_0)}$ with $(\delta_0, t_0) \neq (\delta'_0, t'_0)$, we have (as illustrated in Figure 7.3)

$$S^k_{\rho_0} = \mathsf{Cyl}_{\rho_0}(\pi_0 \cdot \delta_0, \mathbb{R}_{\geq 0}^{|\pi_0|} \times \{t_0\}) \uplus S^{k-1}_{\rho_0[\delta'_0,t'_0]}\,.$$

  As before, we have $\mathbb{P}^{\eta^p,\theta}_{\rho_0}\big(\mathsf{Cyl}_{\rho_0}(\pi_0 \cdot \delta_0, \mathbb{R}_{\geq 0}^{|\pi_0|} \times \{t_0\})\big) = p$. We show that

$$\mathbb{P}^{\eta^p,\theta}_{\rho_0}\big(S^{k-1}_{\rho_0[\delta'_0,t'_0]}\big) = (1-p) \times \mathbb{P}^{\eta^p,\theta}_{\rho_0[\delta'_0,t'_0]}\big(S^{k-1}_{\rho_0[\delta'_0,t'_0]}\big) \tag{7.1}$$

  and, by induction hypothesis, we deduce that

$$\mathbb{P}^{\eta^p,\theta}_{\rho_0[\delta'_0,t'_0]}\big(S^{k-1}_{\rho_0[\delta'_0,t'_0]}\big) \leq 1 - (1-p)^{k-1}\,.$$

  In conclusion, we obtain

$$\begin{aligned}\mathbb{P}^{\eta^p,\theta}_{\rho_0}\big(S^k_{\rho_0}\big) &= p + (1-p) \times \mathbb{P}^{\eta^p,\theta}_{\rho_0[\delta'_0,t'_0]}\big(S^{k-1}_{\rho_0[\delta'_0,t'_0]}\big)\\ &\leq p + (1-p)(1-(1-p)^{k-1}) = 1-(1-p)^k\,.\end{aligned}$$

  Now, we show (7.1): to do it, we must come back to the definition of the probabilities with respect to cylinders. First, $S^{k-1}_{\rho_0[\delta'_0,t'_0]}$ can be decomposed a union of cylinders over some paths $\pi_0 \cdot \delta'_0 \cdot \pi$ with $|\pi| = k-1$. For such a path π, let C_π be the set of sequences of delays of plays appearing in $S^{k-1}_{\rho_0[\delta'_0,t'_0]}$ so that

$$S^{k-1}_{\rho_0[\delta'_0,t'_0]} = \bigcup_{|\pi|=k-1} \mathsf{Cyl}_{\rho_0}(\pi_0 \cdot \delta'_0 \cdot \pi, C_\pi) = \bigcup_{|\pi|=k-1} \mathsf{Cyl}_{\rho_0[\delta'_0,t'_0]}(\pi_0 \cdot \delta'_0 \cdot \pi, C_\pi)\,.$$

  The set C_π is Lebesgue-measurable since it can be obtained as a finite unions and intersections of the constraints appearing in $\eta^p_{\mathbb{R}^+}$ (and thus in σ_1 or σ_2) or $\theta_{\mathbb{R}^+}$, that are all Lebesgue-measurable by Hypothesis 1 (the explanation of

this decomposition is given in the following remark). Thus,

$$\begin{aligned}\mathbb{P}^{\eta^p,\theta}_{\rho_0}\big(S^{k-1}_{\rho_0[\delta'_0,t'_0]}\big) &= \sum_{|\pi|=k-1} \mathbb{P}^{\eta^p,\theta}_{\rho_0}\big(\mathsf{Cyl}_{\rho_0}(\pi_0\cdot\delta'_0\cdot\pi, C_\pi)\big)\\ &= \sum_{|\pi|=k-1} \eta^p(\rho_0)(\delta'_0,t'_0)\times \mathbb{P}^{\eta^p,\theta}_{\rho_0[\delta'_0,t'_0]}\big(\mathsf{Cyl}_{\rho_0[\delta'_0,t'_0]}(\pi_0\cdot\delta'_0\cdot\pi, C_\pi)\big)\\ &= (1-p)\times \sum_{|\pi|=k-1} \mathbb{P}^{\eta^p,\theta}_{\rho_0[\delta'_0,t'_0]}\big(\mathsf{Cyl}_{\rho_0[\delta'_0,t'_0]}(\pi_0\cdot\delta'_0\cdot\pi, C_\pi)\big)\\ &= (1-p)\times \mathbb{P}^{\eta^p,\theta}_{\rho_0[\delta'_0,t'_0]}\big(S^{k-1}_{\rho_0[\delta'_0,t'_0]}\big)\,.\end{aligned}$$

- If $\rho_0 \in \mathsf{FPlays}_{\mathsf{Min}}$ and $\eta^p(\rho_0) = \mathsf{Dirac}_{\delta_0,t_0}$, then $S^k_{\rho_0} = S^{k-1}_{\rho_0[\delta_0,t_0]}$. As in (7.1), we can obtain that

$$\mathbb{P}^{\eta^p,\theta}_{\rho_0}\big(S^{k-1}_{\rho_0[\delta_0,t_0]}\big) = \eta^p(\rho_0)(\delta_0,t_0)\times \mathbb{P}^{\eta^p,\theta}_{\rho_0[\delta_0,t_0]}\big(S^{k-1}_{\rho_0[\delta_0,t_0]}\big)\,.$$

  Then, we have directly

$$\mathbb{P}^{\eta^p,\theta}_{\rho_0}\big(S^k_{\rho_0}\big) = \mathbb{P}^{\eta^p,\theta}_{\rho_0[\delta_0,t_0]}\big(S^{k-1}_{\rho_0[\delta_0,t_0]}\big) \leq 1-(1-p)^{k-1} \leq 1-(1-p)^k\,.$$

- If $\rho_0 \in \mathsf{FPlays}_{\mathsf{Max}}$, then

$$S^k_{\rho_0} = \bigcup_{(\delta_0,t_0)\in \mathrm{supp}(\theta(\rho_0))} S^{k-1}_{\rho_0[\delta_0,t_0]}\,.$$

  Again by decomposing on cylinders, we have that

$$\mathbb{P}^{\eta^p,\theta}_{\rho_0}\big(S^k_{\rho_0}\big) = \sum_{\delta_0}\int_{t_0} \theta_\Delta(\rho_0)(\delta_0)\mathbb{P}^{\eta^p,\theta}_{\rho_0[\delta_0,t_0]}\big(S^{k-1}_{\rho_0[\delta_0,t_0]}\big)\,\mathrm{d}\theta_{\mathbb{R}^+}(\rho_0,\delta_0)(t_0)\,.$$

  By induction hypothesis, we obtain

$$\begin{aligned}\mathbb{P}^{\eta^p,\theta}_{\rho_0}\big(S^k_{\rho_0}\big) &\leq (1-(1-p)^{k-1})\times \sum_{\delta_0}\int_{t_0} \theta_\Delta(\rho_0)(\delta_0)\,\mathrm{d}\theta_{\mathbb{R}^+}(\rho_0,\delta_0)(t_0)\\ &\leq 1-(1-p)^{k-1} \leq 1-(1-p)^k\,.\end{aligned}$$

□

Remark. We want to justify why the set C_π of delays that appear in a path π of $S^k_{\rho_0}$, is a constraint, i.e. can be obtained as a finite union and intersection of the constraints in $\eta^p_{\mathbb{R}^+}$ and $\theta_{\mathbb{R}^+}$.

We consider a finite path $\pi = \pi_0\cdot\delta_0\cdots\delta_k$ that appears in $S^k_{\rho_0}$, i.e. containing a transition chosen by η^p with probability p. A play following π and conform-

ing to η^p and θ satisfies some constraints over each transition. If δ_k belongs to Min, then the set of delays such that $\rho_0[\delta_0, t_0] \cdots [\delta_{k-1}, t_{k-1}]$ satisfies a (Lebesgue-measurable) constraint C_{δ_k} given by Hypothesis 1 over η^p (as the inverse image of the strategy applying on $\rho_0[\delta_0, t_0] \cdots [\delta_{k-1}, t_{k-1}]$). The case where δ_k belongs to Max is analogous.

Now, let δ_i be the transition belonging to Min with the constraint $C_{\delta_{i+1} \cdots \delta_k}$ with $0 \leq i \leq k$. The constraint $C_{\delta_i \cdots \delta_k}$ is given by the inverse image of η^p on $\rho_0[\delta_0, t_0] \cdots [\delta_{i-1}, t_{i-1}]$ (that gives a measurable set of delay) and the intersection with $C_{\delta_{i+1} \cdots \delta_k}$ (we consider only delays satisfying the next constraints). Thus, by induction, we can define the constraint C_π by a finite (since π is finite) intersection of the constraints in $\eta^p_{\mathbb{R}^+}$ and $\theta_{\mathbb{R}^+}$ (that are measurable by Hypothesis 1).

To conclude, we remark that in π, we need to choose the position of the transition of proba p (when σ_1 and σ_2 choose the same transition). In particular, there exists at most k available possible positions, and C_{δ_k} is given by the finite union over all possible positions. ◁

7.1.2. η^p is almost-optimal for the deterministic value

To show that $\overline{\mathsf{mVal}}(\ell, \nu) \leq \mathsf{dVal}(\ell, \nu)$ for all configurations (ℓ, ν), we prove that

$$\mathsf{mVal}^{\eta^p}(\ell, \nu) \leq \max(-N, \mathsf{dVal}(\ell, \nu)) + 3\varepsilon$$

for all configurations (ℓ, ν) and for p *close enough to* 1: we conclude by taking the limit when N tends to $+\infty$ and ε tends to 0. We get that inequality by showing the following result, paired with the fact that $\mathsf{dVal}^\sigma(\ell, \nu) \leq \max(-N, \mathsf{dVal}(\ell, \nu)) + \varepsilon$.

Lemma 7.7. *Let $\mathcal{G}$ be a divergent WTG, for all configurations (ℓ, ν) and $\varepsilon > 0$ small enough, there exists $\tilde{p} \in (0, 1)$ so that for all $p \in [\tilde{p}, 1)$,*

$$\mathsf{mVal}^{\eta^p}(\ell, \nu) \leq \mathsf{dVal}^\sigma(\ell, \nu) + 2\varepsilon.$$

We conclude this section with the proof of this result. Since η^p is memoryless and satisfies Hypothesis 2 (by Lemma 7.6) we can use Proposition 6.2 (with $\varepsilon/2$) to obtain the existence of a deterministic strategy $\tau \in \mathsf{dStrat}_{\mathsf{Max}}$ such that for all configurations (ℓ, ν), $\mathbb{E}^{\eta^p, \tau}_{\ell, \nu} \geq \mathsf{mVal}^{\eta^p}(\ell, \nu) - \varepsilon/2$. The proof then consists in computing the bound $\tilde{p}$ on the probabilities such that for all $p \in [\tilde{p}, 1)$, $\mathbb{E}^{\eta^p, \tau}_{\ell, \nu} \leq \mathsf{dVal}^\sigma(\ell, \nu) + 3\varepsilon/2$. We conclude by combining both inequalities.

Notice that η^p and τ are both discrete strategies, i.e. we can apply Lemma 5.21: for all configurations (ℓ, ν),

$$\mathbb{E}^{\eta^p, \tau}_{\ell, \nu} = \sum_{\rho \in \mathsf{TPlays}^{\eta^p, \tau}_{\ell, \nu}} \mathbf{P}(\rho) \times \mathbb{P}^{\eta^p, \tau}_{\ell, \nu}(\rho)$$

where $\mathsf{TPlays}^{\eta^p,\tau}_{\ell,\nu}$ is the $\mathbb{P}^{\eta^p,\tau}_{\ell,\nu}$-measurable set of plays starting in (ℓ,ν), conforming to η^p and θ and ending in the target.

The case where the whole region game only contains positive SCCs is easy since η^p chooses the transition and delay given by σ^1 with probability 1. By divergence, $\mathcal{G}$ then contains no negative cycles. A play conforming to η^p is also conforming to the deterministic strategy σ^1, so it must be acyclic. In particular, there exists only one play ρ conforming to η^p and τ. This one is also conforming to σ and thus reaches the target with a cumulated weight $\mathsf{wt}_\Sigma(\rho) = \mathbb{E}^{\eta^p,\tau}_{\ell,\nu} \leq \mathsf{dVal}^\sigma(\ell,\nu)$ as expected.

Now, suppose that the region graph contains at least a negative SCC. Thus, we let $c > 0$ be the maximal size of an elementary cycle of the region game (that visits each pair (ℓ, r) at most once) and $w^- > 0$ be the opposite of the maximal cumulated weight of an elementary negative cycle in $\mathcal{R}(\mathcal{G})$ (necessarily bounded by $W_\mathsf{e}\,|Q|$ where Q is the set of region states).

We partition the set $\mathsf{TPlays}^{\eta^p,\tau}_{\ell,\nu}$ into subsets $\Pi_{i,j}$ of plays according to the number i of edges $(\ell,\nu) \xrightarrow{\delta,t} (\ell',\nu')$ chosen by Min with a probability $\eta^p(\ell,\nu)(\delta,t) = 1-p$, and their length j (we always have $i \leq j$). The partition is depicted in Figure 7.4:

- $\Pi_{\mathbb{N},\geq\kappa}$, depicted in blue, contains all plays with a length greater than κ (the switching threshold);
- $\Pi_{0,\leq\kappa}$, depicted in yellow, contains all plays without edges of probability $1-p$, with a length at most κ;
- $\widetilde{\Pi}$, depicted in red, contains the rest of the plays.

In particular, we have

$$\mathbb{E}^{\eta^p,\tau}_{\ell,\nu} = \underbrace{\sum_{\rho\in\Pi_{0,\leq\kappa}} \mathbf{P}(\rho)\,\mathbb{P}^{\eta^p,\tau}_{\ell,\nu}(\rho)}_{\gamma_{0,\leq\kappa}} + \underbrace{\sum_{\rho\in\Pi_{\mathbb{N},\geq\kappa}} \mathbf{P}(\rho)\,\mathbb{P}^{\eta^p,\tau}_{\ell,\nu}(\rho)}_{\gamma_{\mathbb{N},\geq\kappa}} + \underbrace{\sum_{\rho\in\widetilde{\Pi}} \mathbf{P}(\rho)\,\mathbb{P}^{\eta^p,\tau}_{\ell,\nu}(\rho)}_{\widetilde{\gamma}}\,. \tag{7.2}$$

We now compute and bound the three expectations $\gamma_{0,\leq\kappa}$, $\gamma_{\mathbb{N},\geq\kappa}$ and $\widetilde{\gamma}$.

The red zone is such that $\widetilde{\gamma} \leq \varepsilon/4$ All plays in $\widetilde{\Pi}$ have a length at most κ: so the cumulated weight of all such plays is at most $\kappa\,W_\mathsf{e}$. So, we have

$$\widetilde{\gamma} = \sum_{\rho\in\widetilde{\Pi}} \mathbf{P}(\rho)\,\mathbb{P}^{\eta^p,\tau}_{\ell,\nu}(\rho) \leq \sum_{\rho\in\widetilde{\Pi}} \kappa\,W_\mathsf{e}\,\mathbb{P}^{\eta^p,\tau}_{\ell,\nu}(\rho) = \kappa\,W_\mathsf{e}\,\mathbb{P}^{\eta^p,\tau}_{\ell,\nu}(\widetilde{\Pi}).$$

But all plays $\rho \in \bigcup_{j\leq K} \Pi_{i,j}$ with $i \leq \kappa$ take i edges of probability $1-p$. In particular, by bounding all other probabilities by 1, and since there are at most 2^κ

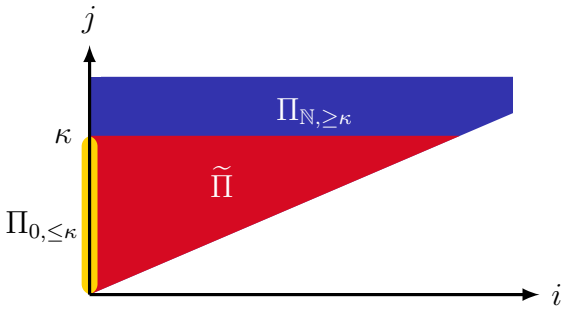


Figure 7.4 – Partition of the plays $\mathsf{TPlays}^{\eta^p,\tau}_{\ell,\nu}$

plays in $\bigcup_{j\leq K}\Pi_{i,j}$ (since τ is deterministic and the support of all $\eta^p(\ell,\nu)$ contains at most two edges), we obtain (using that $1-(1-p)^K\leq 1$)

$$\mathbb{P}^{\eta^p,\tau}_{\ell,\nu}(\widetilde{\Pi}) \leq 2^\kappa \sum_{i=1}^{\kappa-1}(1-p)^i = 2^\kappa \frac{(1-p)(1-(1-p)^\kappa)}{p} \leq 2^\kappa \frac{1-p}{p} \qquad (7.3)$$
$$\xrightarrow[p\to 1]{} 0$$

Finally, if we suppose that

$$p \geq \frac{1}{1+\varepsilon/(4\times 2^\kappa \kappa\, W_{\mathsf{e}})}$$

we obtain as desired

$$\widetilde{\gamma} \leq 2^\kappa \kappa\, W_{\mathsf{e}} \frac{1-p}{p} \leq \frac{\varepsilon}{4}.$$

The yellow and blue zones are such that $\gamma_{0,\leq\kappa}+\gamma_{\mathbb{N},\geq\kappa}\leq \mathsf{dVal}^\sigma(\ell,\nu)+5\varepsilon/4$ All plays in $\Pi_{0,\leq\kappa}$ reach the target without taking any edge of probability $1-p$, so they are conforming to σ^1. In the case where $\mathsf{dVal}(\ell,\nu)=-\infty$, $\Pi_{0,\leq\kappa}=\emptyset$ and $\gamma_{0,\leq\kappa}=0$, since no play conforming to σ^1 from (ℓ,ν) reaches the target. In this case, Min can stay in a cycle with a negative cumulated weight as long as wanted. Now, if $\mathsf{dVal}(\ell,\nu)$ is finite, by Lemma 3.11-1, the cumulated weight of a play in $\Pi_{0,\leq\kappa}$ is at most $\mathsf{dVal}(\ell,\nu)+\kappa\,\varepsilon/\kappa=\mathsf{dVal}(\ell,\nu)+\varepsilon$. As $\mathsf{dVal}(\ell,\nu)=\inf_{\sigma\in\mathsf{dStrat}_{\mathsf{Min}}}\mathsf{dVal}^\sigma(\ell,\nu)\leq \mathsf{dVal}^\sigma(\ell,\nu)$, in both cases, we can write

$$\gamma_{0,\leq\kappa}\leq (\mathsf{dVal}^\sigma(\ell,\nu)+\varepsilon)\ \mathbb{P}^{\eta^p,\tau}_{\ell,\nu}(\Pi_{0,\leq\kappa})\,.$$

Let ρ be a play in $\Pi_{i,j}$ with $0\leq i$ and $j\geq\kappa$. Since η^p only allows cycles in negative SCCs, all region cycles in ρ have a cumulated weight at most -1. By

definition of κ and the proof of Proposition 3.10,

$$\mathbf{P}(\rho) \leq \mathsf{dVal}^{\sigma}(\ell,\nu) \leq \mathsf{dVal}^{\sigma}(\ell,\nu) + \varepsilon\,.$$

By summing up the contribution of yellow and blue zones, we get

$$\gamma_{0,\leq\kappa} + \gamma_{\mathbb{N},\geq\kappa} \leq (\mathsf{dVal}^{\sigma}(\ell,\nu) + \varepsilon)\ \mathbb{P}^{\eta^p,\tau}_{\ell,\nu}(\Pi_{0,\leq\kappa} \cup \Pi_{\mathbb{N},\geq\kappa}). \tag{7.4}$$

To conclude, we distinguish two cases.

- We suppose that $\mathsf{dVal}^{\sigma}(\ell,\nu) \geq -\frac{5\varepsilon}{4}$. By having $\mathbb{P}^{\eta^p,\tau}_{\ell,\nu}(\Pi_{0,\leq\kappa} \cup \Pi_{\mathbb{N},\geq\kappa}) \leq 1$, we get

$$\gamma_{0,\leq\kappa} + \gamma_{\mathbb{N},\geq\kappa} \leq \left(\mathsf{dVal}^{\sigma}(\ell,\nu) + \frac{5\varepsilon}{4}\right) \mathbb{P}^{\eta^p,\tau}_{\ell,\nu}(\Pi_{0,\leq\kappa} \cup \Pi_{\mathbb{N},\geq\kappa}) \leq \mathsf{dVal}^{\sigma}(\ell,\nu) + \frac{5\varepsilon}{4}\,.$$

- We suppose that $\mathsf{dVal}^{\sigma}_{\ell,\nu} < -\frac{5\varepsilon}{4}$, and then, by (7.3),

$$\mathbb{P}^{\eta^p,\tau}_{\ell,\nu}(\Pi_{0,\leq\kappa} \cup \Pi_{\mathbb{N},\geq\kappa}) = 1 - \mathbb{P}^{\eta^p,\tau}_{\ell,\nu}(\widetilde{\Pi}) \geq 1 - 2^{\kappa}\frac{1-p}{p} \xrightarrow[p\to 1]{} 1\,.$$

  Finally, if we suppose that

$$p \geq \frac{2^{\kappa}}{2^{\kappa} + 1 - \frac{\mathsf{dVal}^{\sigma}(\ell,\nu)+5\varepsilon/4}{\mathsf{dVal}^{\sigma}(\ell,\nu)+\varepsilon}} \in (0,1)$$

  then

$$\mathbb{P}^{\eta^p,\tau}_{\ell,\nu}(\Pi_{0,\leq\kappa} \cup \Pi_{\mathbb{N},\geq\kappa}) \geq \frac{\mathsf{dVal}^{\sigma}(\ell,\nu) + 5\varepsilon/4}{\mathsf{dVal}^{\sigma}(\ell,\nu) + \varepsilon}$$

  and, by negativity of $\mathsf{dVal}^{\sigma}_{\ell,\nu} + \varepsilon$, we can rewrite (7.4) as

$$\gamma_{0,\leq\kappa} + \gamma_{\mathbb{N},\geq\kappa} \leq (\mathsf{dVal}^{\sigma}(\ell,\nu) + \varepsilon)\,\frac{\mathsf{dVal}^{\sigma}(\ell,\nu) + 5\varepsilon/4}{\mathsf{dVal}^{\sigma}(\ell,\nu) + \varepsilon} = \mathsf{dVal}^{\sigma}(\ell,\nu) + \frac{5\varepsilon}{4}\,.$$

In all cases, we have $\gamma_{0,\leq\kappa} + \gamma_{\mathbb{N},\geq\kappa} \leq \mathsf{dVal}^{\sigma}(\ell,\nu) + 5\varepsilon/4$.

Gathering all constraints on p We gather all the lower bounds over p that we need in the proof, letting

$$\tilde{p} = \begin{cases} \max\left(\frac{1}{1+\varepsilon/(4\times 2^{\kappa}\kappa\, W_{\mathsf{e}})}, \frac{1}{2}\right) & \text{if } \mathsf{dVal}^{\sigma}(\ell,\nu) \geq -\frac{5\varepsilon}{4} \\ \max\left(\frac{1}{1+\varepsilon/(4\times 2^{\kappa}\kappa\, W_{\mathsf{e}})}, \frac{1}{2}, \frac{2^{\kappa}}{2^{\kappa}+1-\frac{\mathsf{dVal}^{\sigma}(\ell,\nu)+5\varepsilon/4}{\mathsf{dVal}^{\sigma}(\ell,\nu)+\varepsilon}}\right) & \text{otherwise} \end{cases}$$

Then, for $p \in (\tilde{p}, 1)$, we have $\mathbb{E}_{\ell,\nu}^{\eta^p,\tau} \leq \mathsf{dVal}^\sigma(\ell,\nu) + 3\varepsilon/2$. Since $\tilde{p}$ does not depend on τ, we conclude that for $p \in (\tilde{p}, 1)$, we have $\mathsf{mVal}^{\eta^p}(\ell,\nu) \leq \mathsf{dVal}^\sigma(\ell,\nu) + 2\varepsilon$ as expected.

7.2. Memoryless value in shortest path games

In this section, we restrict ourselves to SPGs that are WTGs using no clocks. In particular, we recall that the definition of stochastic strategies and values from WTGs can be adapted. In this context, all distributions over transitions are finite; thus, Hypothesis 1 always holds. Moreover, all strategies are discrete, and if the strategies of Min are supposed to fulfil Hypothesis 2, the various values exist and Lemma 5.21 applies.

In SPGs, we are able to extend the proof given in the previous section without imposing the divergence condition. In particular, we prove Theorem 7.1 in the case of SPGs by showing that in all SPGs, for all vertices ℓ, we have

$$\mathsf{dVal}(\ell) = \overline{\mathsf{mVal}}(\ell) = \underline{\mathsf{mVal}}(\ell)\,.$$

The proof strategy follows the same path as in the divergent case:

- we show in Lemma 7.8 that $\mathsf{dVal}(\ell) \leq \underline{\mathsf{mVal}}(\ell)$, by adapting the proof of Lemma 7.4 relying on the existence of an optimal deterministic and memoryless strategy of Max w.r.t. the deterministic value (by Theorem 1.31-2);
- we show in Proposition 7.9 that memoryless stochastic strategies can emulate deterministic ones: $\overline{\mathsf{mVal}}(\ell) \leq \mathsf{dVal}(\ell)$ (with similar but more involved techniques as in Proposition 7.5).

Combined with the fact that $\underline{\mathsf{mVal}}(\ell) \leq \overline{\mathsf{mVal}}(\ell)$, we indeed obtain the desired equalities.

We start by adapting the proof of Lemma 7.4:

Lemma 7.8. *In all SPGs, for all vertices ℓ, $\mathsf{dVal}(\ell) \leq \underline{\mathsf{mVal}}(\ell)$.*

Proof. By Theorem 1.31-2, Max has a memoryless and deterministic optimal strategy τ w.r.t. the deterministic value, i.e. against any deterministic strategy of Min, from the vertex ℓ, it guarantees a weight at least $\mathsf{dVal}(\ell)$.

The rest of the proof is entirely similar, but reproduced here for ease. Let η be any memoryless proper strategy of Min. Every play conforming to η and τ has a weight at least $\mathsf{dVal}(\ell)$ (since it is conforming to τ), so that, by definition of the expectation,

$$\mathbb{E}_\ell^{\eta,\tau} = \int_\rho \mathbf{P}(\rho)\,\mathrm{d}\mathbb{P}_\ell^{\eta,\tau}(\rho) \geq \mathsf{dVal}(\ell)\,.$$

We notice that this integral is indeed a finite sum here, since the strategies are necessarily discrete. Moreover, the inequality holds for all proper strategies of Min, we deduce that

$$\inf_{\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}} \mathbb{E}^{\eta,\tau}_{\ell} \geq \mathsf{dVal}(\ell)\,.$$

By taking the supremum over all memoryless strategies of Max (that contain the memoryless and deterministic strategy τ), we have

$$\sup_{\theta \in \mathsf{mStrat}_{\mathsf{Max}}} \inf_{\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}} \mathbb{E}^{\eta,\theta}_{\ell} \geq \mathsf{dVal}(\ell)\,.$$

In particular, $\underline{\mathsf{mVal}}(\ell) \geq \mathsf{dVal}(\ell)$, as expected. □

The rest of this section is devoted to the proof of the final piece of the proof:

Proposition 7.9. *Let $\mathcal{G} = \langle V_{\mathsf{Max}}, V_{\mathsf{Min}}, V_T, E, \mathsf{wt}\rangle$ be a SPG. Then, for all vertices ℓ, $\overline{\mathsf{mVal}}(\ell) \leq \mathsf{dVal}(\ell)$.*

In an SPG, once we fix a memoryless strategy $\eta \in \mathsf{mStrat}_{\mathsf{Min}}$, we obtain a *Markov decision process* (MDP) where the other player must still choose how to react. An MDP is a tuple $\langle V, A, P\rangle$ where V is a set of vertices, A is a set of actions, and $P\colon V \times A \to \mathsf{Dist}(V)$ is a partial function mapping to some pair of vertices and actions a distribution of probabilities over the successor vertices. In our context, we let $\mathcal{G}^\eta$ be the MDP with the same set V of vertices as $\mathcal{G}$, actions $A = V \cup \{\bot\}$ being either successor vertices of the game or an additional action $\bot$ denoting the random choice of η, and a probability distribution P defined by:

- if $\ell \in V_{\mathsf{Max}}$, $P(\ell, \ell')$ is only defined if $(\ell, \ell') \in E$ in which case $P(\ell, \ell') = \mathsf{Dirac}_{\ell'}$, and $P(\ell, \bot)$ is also undefined;
- if $\ell \in V_{\mathsf{Min}}$, $P(\ell, \bot) = \eta(\ell)$, and $P(\ell, \ell')$ is undefined for all $\ell' \in V$.

In drawings of MDPs (and also of Markov chains, later), we show weights as trivially transferred from the game graph.

Example 7.10. In the left of Figure II.1, the MDP obtain from an SPG depicted in Figure 1.8, by picking as a memoryless strategy for Min the one choosing to go to ℓ_{Max} with probability $p \in (0,1)$ and to the target vertex with probability $1-p$.

Another more complex example is given in Figure 7.5 where the memoryless strategy for Min consists, in vertex ℓ_1, to choose successor ℓ_0 with probability $p \in (0,1)$ and successor ℓ_2 with probability $1-p$, and in vertex ℓ_3, to choose successor ℓ_1 with the same probability p and the target vertex with probability $1-p$. ⌟

In such an MDP, when player Max has chosen a strategy, there will remain no "choices" to make, and we will thus end up in a *Markov chain*. A Markov chain (*MC*) is a tuple $\mathcal{M} = \langle V, P \rangle$ where V is a set of vertices, and $P \colon V \to \mathsf{Dist}(V)$ associates to each vertex a distribution of probabilities over the successor vertices. In our context, for all memoryless strategies $\theta \in \mathsf{mStrat}_{\mathsf{Max}}$, we let $\mathcal{G}^{\eta,\theta}$ the MC obtained from the MDP $\mathcal{G}^{\eta}$ by following strategy θ and action $\bot$. Formally, it consists of the same set V of vertices as $\mathcal{G}$, and mapping P associating to a vertex $\ell \in V_{\mathsf{Min}}$, $P(\ell) = \eta(\ell)$ and to a vertex $\ell \in V_{\mathsf{Max}}$, $P(\ell) = \theta(\ell)$.

Example 7.11. On the right of Figure II.1 is depicted the MC obtained when Max decides to go to ℓ_{Min} with probability $q \in [0,1]$ and to the target vertex with probability $1-q$. ⌟

By [BK08, Section 10.5.1][2], the memoryless value $\mathsf{mVal}^{\eta}(\ell)$ is finite if and only if strategy η ensures the reachability of a target vertex with probability 1, no matter how the opponent plays. Thus, for memoryless strategies, Hypothesis 2 can be restricted to only checking that for all vertices ℓ_0 and strategies $\theta \in \mathsf{Strat}_{\mathsf{Max}}$, $\mathbb{P}^{\eta,\theta}_{\ell_0}(\mathsf{TPlays}) = 1$. From Lemma 6.3, we obtain a Bellman-like equation as follows, once simplified since we restrict ourselves to memoryless strategies:

$$\mathbb{E}^{\eta,\theta}_{\ell} = \begin{cases} 0 & \text{if } \ell \in V_T\,; \\ \sum_{(\ell,\ell')\in E} \eta(\ell,\ell') \times (\mathsf{wt}(\ell,\ell') + \mathbb{E}^{\eta,\theta}_{\ell'}) & \text{if } v \in V_{\mathsf{Min}}\,; \\ \sum_{(\ell,\ell')\in E} \theta(\ell,\ell') \times (\mathsf{wt}(\ell,\ell') + \mathbb{E}^{\eta,\theta}_{\ell'}) & \text{if } \ell \in V_{\mathsf{Max}}\,. \end{cases} \tag{7.5}$$

Example 7.12. For the game of Figure II.1, we let η and θ the memoryless strategies that result in the MC on the right. Letting $x = \mathbb{E}^{\eta,\theta}_{\ell_{\mathsf{Min}}}$ and $y = \mathbb{E}^{\eta,\theta}_{\ell_{\mathsf{Max}}}$, the system (7.5) rewrites as $x = (1-p)\times 0 + p \times y$ and $y = q \times (-1+x) + (1-q)\times(-10)$. We thus have

$$x = \frac{p\,(9q-10)}{1-pq}\,.$$

Two cases happen, depending on the value of p: if $p < 9/10$, then Max maximises x by choosing $q = 1$, while choosing $q = 0$ when $p \geq 9/10$. In all cases, Max will therefore play deterministically: if $p < 9/10$, the expected payoff from ℓ_{Min} will then be $\mathsf{mVal}^{\eta}(\ell_{\mathsf{Min}}) = -p/(1-p)$; if $p \geq 9/10$, it will be $\mathsf{mVal}^{\eta}(\ell_{\mathsf{Min}}) = -10\,p$. This memoryless value is always greater than the (deterministic) optimum -10 that Min was able to achieve with memory since we must keep $1-p > 0$ to ensure reaching the target with probability 1. We thus obtain $\overline{\mathsf{mVal}}(\ell_{\mathsf{Min}}) = \overline{\mathsf{mVal}}(\ell_{\mathsf{Max}}) = -10$ as before. There are no optimal strategies for Min, but an ε-optimal one consisting in choosing probability $q \geq 1 - \varepsilon/10$. ⌟

[2] In [BK08, Section 10.5.1], the authors only work with non-negative weights. However, in this case, the proof is identical to the general case.

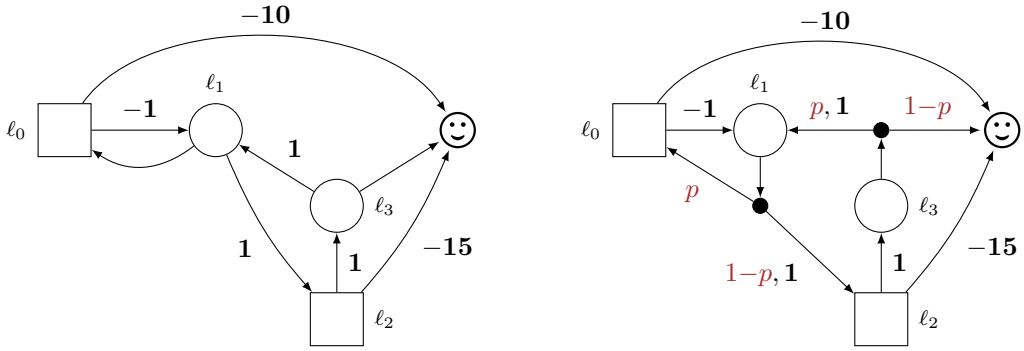


Figure 7.5 – On the left, a more complex example of SPG. On the right, the MDP associated with a randomised strategy of Min with a parametric probability $p \in (0, 1)$.

The fact that Max can play optimally with a deterministic strategy in the MDP $\mathcal{G}^\eta$ is not specific to this example. Indeed, in an MDP $\mathcal{G}^\eta$ such that $\mathbb{P}_\ell^{\eta,\theta}(\mathsf{TPlays}) = 1$ for all θ, Max cannot avoid reaching the target vertices, and must then ensure the most expensive play possible. Considering the MDP $\tilde{\mathcal{G}}^\eta$ obtained by multiplying all the weights in the graph by -1, the objective of Max becomes a shortest-path objective. We can then deduce from [BT91] that Max has an optimal deterministic memoryless strategy: the same applies in the original MDP $\mathcal{G}^\eta$.

Lemma 7.13. *In the MDP $\mathcal{G}^\eta$ such that $\mathbb{P}_\ell^{\eta,\theta}(\mathsf{TPlays}) = 1$ for all $\theta \in \mathsf{Strat}_{\mathsf{Max}}$,* Max *has an optimal deterministic memoryless strategy.*

For SPGs, we note that the divergence is not required to define optimal switching strategy w.r.t. the deterministic. In particular, we consider a switching strategy σ defined in Theorem 3.6, and define the memoryless strategy η^p, with $p \in (0, 1)$ as follows: for $\ell \in V$, letting $\sigma^1(\ell) = e_1$ and $\sigma^2(\ell) = e_2$, we let

$$\eta^p(\ell) = \begin{cases} \mathsf{Dirac}_{\delta_1} & \text{if } \ell \text{ is in a positive SCC} \\ p \times \mathsf{Dirac}_{\delta_1} + (1-p) \times \mathsf{Dirac}_{\delta_2} & \text{otherwise} \end{cases}$$

We conclude as before by showing the following result paired with the fact that $\mathsf{dVal}^\sigma(\ell) \leq \max(-n, \mathsf{dVal}(\ell))$:

Lemma 7.14. *Let $\mathcal{G}$ be a SPG. Then, for all vertices ℓ and $\varepsilon > 0$ small enough, there exists $\tilde{p} \in (0, 1)$ so that for all $p \in [\tilde{p}, 1)$,*

$$\mathsf{mVal}^{\eta^p}(\ell) \leq \mathsf{dVal}^\sigma(\ell) + 2\varepsilon\,.$$

We conclude this section with the proof of this result. Unlike in the proof of Proposition 7.5, some SCCs of SPGs may contain positive and negative cycles

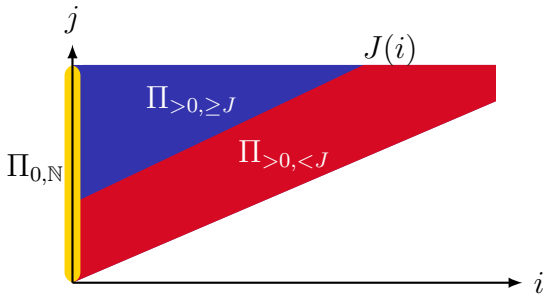


Figure 7.6 – A new partition of plays Π in a SPG.

(e.g. the SPG depicted on the left of Figure 7.5): we will thus refine our partition to compute the expected values.

First, by Lemma 7.6, we still know that η^p is proper, i.e. $\mathbb{P}_\ell^{\eta^p,\tau}(\mathsf{TPlays}) = 1$. We can thus use Lemma 7.13 to obtain the existence of a deterministic memoryless strategy τ such that $\mathbb{E}_\ell^{\eta^p,\tau} \geq \mathsf{mVal}^{\eta^p}(\ell) - \varepsilon/2$.

We let $c > 0$ be the maximal size of an elementary cycle (that visits a vertex at most once) in $\mathcal{G}$, $w^- > 0$ be the opposite of the maximal weight of an elementary negative cycle in $\mathcal{G}$, and $w^+ \geq 0$ be the maximal weight of an elementary non-negative cycle in $\mathcal{G}$ (or 0 if such cycle does not exist).

We now find the bound $\tilde{p}$ such that $\mathbb{E}_\ell^{\eta^p,\tau} \leq \mathsf{dVal}^\sigma(\ell) + 3\frac{\varepsilon}{2}$ to conclude. To do so, we consider a new partition of the set Π of plays starting in ℓ, conforming to η^p and τ, and reaching the target, as depicted in Figure 7.6: as before the partition relies on the number i of transitions of probability $1-p$ a play goes through, and its length j:

- $\Pi_{0,\mathbb{N}}$, depicted in yellow, contains all plays with no edges of probability $1-p$;
- $\Pi_{>0,\geq J}$, depicted in blue, contains all plays with $i \geq 1$ edges of probability $1-p$, and a length at least $J(i) = \lfloor ia + b \rfloor$ with

$$a = c\left(1 + \frac{w^+}{w^-}\right) \quad \text{and} \quad b = \frac{|\mathsf{dVal}^\sigma(\ell)| + |V|\,W_{\mathsf{tr}} + w^-}{w^-}c + |V|$$

- $\Pi_{>0,<J}$, depicted in red, is the rest of the plays.

We let $\gamma_{0,\mathbb{N}}$ (respectively, $\gamma_{>0,\geq J}$, and $\gamma_{>0,<J}$) be the expectation $\mathbb{E}_\ell^{\eta^p,\tau}$ restricted to plays in $\Pi_{0,\mathbb{N}}$ (respectively, $\Pi_{>0,\geq J}$, and $\Pi_{>0,<J}$). By Lemma 5.21,

$$\mathsf{mVal}^{\eta^p,\tau}(\ell) = \mathbb{E}_\ell^{\eta^p,\tau} = \sum_\rho \mathbf{P}(\rho) \times \mathbb{P}_\ell^{\eta,\tau}(\rho) = \gamma_{0,\mathbb{N}} + \gamma_{>0,\geq J} + \gamma_{>0,<J} \tag{7.6}$$

Like in the proof of Lemma 7.7, we thus separately compute and bound the three terms of (7.6).

Control of the weight of plays in $\Pi_{i,j}$ Unlike in the divergent case, here a play conforming to η^p may follow a cyclic play with a negative weight. In particular, we bound the weight of a play depending on the number of edges $1-p$ it goes through. Let ρ be a play in $\Pi_{i,j}$, with $1 \leq i$ and $j \geq i$: it goes through i edges of probability $1-p$.

Each cycle conforming to η^p and τ with a non-negative cumulative weight contains at least one transition taken by η^p with probability $1-p$ (otherwise, all its edges would be of probability p or 1, and the cycle would be conforming to the fake-optimal strategy σ^1, implying that the cumulative weight would be negative).

In particular, ρ contains at most i elementary cycles of non-negative cumulated weight (at most w^+). The total length of these cycles is at most ic. Once we have removed these cycles from the play, it remains a play of length at least $j-ic$. By a repeated pumping argument, it still contains at least $\left\lfloor \frac{j-ic-|V|}{c} \right\rfloor$ elementary cycles, that all have a negative cumulated weight (at most $-w^-$). The remaining part, once removed the last negative cycles it contains, has length at most $|V|$, and thus a cumulative weight at most $|V|\,W_{\mathsf{tr}}$. In summary the cumulative weight of every play in $\Pi_{i,j}$ is at most

$$iw^+ + \left\lfloor \frac{j-ic-|V|}{c} \right\rfloor (-w^-) + |V|\,W_{\mathsf{tr}}\,. \tag{7.7}$$

Red zone is such that $\gamma_{>0,<J} \leq \varepsilon/2$. Let ρ be a play in $\Pi_{i,j}$, with $i \geq 1$ and $j < J(i)$. By (7.7), its cumulative weight is at most

$$iw^+ + \left\lfloor \frac{j-ic-|V|}{c} \right\rfloor (-w^-) + |V|\,W_{\mathsf{tr}} \leq iw^+ + |V|\,W_{\mathsf{tr}}\,.$$

So, we can decompose the expectation $\gamma_{>0,<J}$ as follows:

$$\gamma_{>0,<J} = \sum_{\rho \in \Pi_{>0,<J}} \mathbf{P}(\rho)\,\mathbb{P}_\ell^{\eta^p,\tau}(\rho) \leq \sum_{i=1}^{+\infty} (iw^+ + |V|\,W_{\mathsf{tr}})\,\mathbb{P}_\ell^{\eta^p,\tau}(\Pi_{i,<J(i)})\,. \tag{7.8}$$

Moreover, the probability of a play in $\Pi_{i,<J(i)}$, given by the i edges of probability $(1-p)$ and the $j-i$ edges with a probability bounded by 1, is at most $(1-p)^i$. Since the number of plays in $\Pi_{i,<J(i)}$ is bounded by $2^{J(i)}$ (for each of the at most $J(i)$ steps, Min has at most two choices in its distribution, while Max plays a

deterministic strategy), we have

$$\mathbb{P}_\ell^{\eta^p,\tau}(\Pi_{i,<J(i)}) \leq (1-p)^i 2^{J(i)}\,. \tag{7.9}$$

We rewrite (7.8) as

$$\begin{aligned}\gamma_{>0,<J} &\leq \sum_{i=1}^{+\infty}(iw^+ + |V|\,W_{\mathsf{tr}})(1-p)^i\,2^{J(i)} \leq \sum_{i=1}^{+\infty}(iw^+ + |V|\,W_{\mathsf{tr}})(1-p)^i\,2^{ai+b}\\ &\leq w^+2^b\sum_{i=1}^{+\infty} i((1-p)2^a)^i + |V|\,W_{\mathsf{tr}}\,2^b\sum_{i=1}^{+\infty}((1-p)2^a)^i\end{aligned}$$

these sums converging as soon as we consider $p \geq 1-\frac{1}{2^a}$. We finally obtain

$$\gamma_{>0,<J} \leq w^+\,2^b\frac{2^a(1-p)}{(1-2^a(1-p))^2} + |V|\,W_{\mathsf{tr}}\,2^b\frac{2^a(1-p)}{1-2^a(1-p)}\,.$$

We consider a stronger assumption on p, namely that $p \geq 1-\frac{1}{2^{a+1}}$. Then, we know that $1 \leq \frac{1}{1-2^a(1-p)} \leq 2$, so that we rewrite the previous inequality as

$$\gamma_{>0,<J} \leq w^+2^{b+a+2}(1-p) + |V|\,W_{\mathsf{tr}}\,2^{b+a+1}\,(1-p)\,.$$

By choosing p such that

$$p \geq 1 - \frac{\varepsilon}{2^{b+a+2}(|V|\,W_{\mathsf{tr}} + 2w^+)}$$

we obtain as desired $\gamma_{>0,<J} \leq \varepsilon/2$.

Yellow and blue zones are such that $\gamma_{0,\mathbb{N}} + \gamma_{>0,\geq J} \leq \mathsf{dVal}_\ell^\sigma + \varepsilon$. We first upper-bound the cumulative weight of all plays of these two zones. On the one hand, all plays of $\Pi_{0,\mathbb{N}}$ reach the target without edges of probability $1-p$, i.e. by conforming to σ^1. By fake-optimality of σ^1, their cumulative weight is upper-bounded by $\mathsf{dVal}^\sigma(\ell)$. On the other hand, by (7.7), all plays ρ of $\Pi_{i,j}$, with $0 \leq i < I$ and

$j \geq J(i)$, have a cumulative weight at most

$$\begin{aligned}
\mathbf{P}(\rho) &\leq iw^+ + \left\lfloor \frac{j - ic - |V|}{c} \right\rfloor (-w^-) + |V|\, W_{\mathsf{tr}} \\
&\leq iw^+ + \frac{J(i) - ic - |V|}{c}(-w^-) + |V|\, W_{\mathsf{tr}} \\
&= iw^+ + \frac{ai + \frac{|\mathsf{dVal}^\sigma(\ell)| + |V|\, W_{\mathsf{tr}} + w^-}{w^-} c + |V| - ic - |V|}{c}(-w^-) + |V|\, W_{\mathsf{tr}} \\
&= iw^+ + \left(i\left(1 + \frac{w^+}{w^-}\right) + \frac{|\mathsf{dVal}^\sigma(\ell)| + |V|\, W_{\mathsf{tr}}}{w^-} - i \right)(-w^-) + |V|\, W_{\mathsf{tr}} \\
&= iw^+ - iw^+ - |\mathsf{dVal}^\sigma(\ell)| - |V|\, W_{\mathsf{tr}} + |V|\, W_{\mathsf{tr}} = -|\mathsf{dVal}^\sigma_\ell(\ell)| \leq \mathsf{dVal}^\sigma(\ell)\,.
\end{aligned}$$

Therefore, all plays in the yellow and blue zones have a cumulative weight bounded by $\mathsf{dVal}^\sigma_\ell$. This implies

$$\begin{aligned}
\gamma_{0,\mathbb{N}} + \gamma_{>0,\geq J} &\leq \sum_{\rho \in \Pi_{0,\mathbb{N}}} \mathsf{dVal}^\sigma(\ell)\, \mathbb{P}^{\eta^p,\tau}_\ell(\rho) + \sum_{\rho \in \Pi_{>0,\geq J}} \mathsf{dVal}^\sigma(\ell)\, \mathbb{P}^{\eta^p,\tau}_\ell(\rho) \\
&\leq \mathsf{dVal}^\sigma(\ell)\, \mathbb{P}^{\eta^p,\tau}_\ell\left(\Pi_{0,\mathbb{N}} \cup \Pi_{>0,\geq J}\right).
\end{aligned}$$

Depending on the sign of $\mathsf{dVal}^\sigma_\ell$, we can conclude:

- If $\mathsf{dVal}^\sigma(\ell) \geq 0$, then upper-bounding the probability $\mathbb{P}^{\eta^p,\tau}_\ell\left(\Pi_{0,\mathbb{N}} \cup \Pi_{>0,\geq J}\right)$ by 1, suffices to get $\gamma_{0,\mathbb{N}} + \gamma_{>0,\geq J(i)} \leq \mathsf{dVal}^\sigma(\ell)$.
- If $\mathsf{dVal}^\sigma(\ell) < 0$, then, by the bound (7.9) found for the red zone, we have

  $$\begin{aligned}
  \mathbb{P}^{\eta^p,\tau}_\ell\left(\Pi_{0,\mathbb{N}} \cup \Pi_{>0,\geq J}\right) &= 1 - \mathbb{P}^{\eta^p,\tau}_\ell\left(\Pi_{>0,<J}\right) \\
  &\geq 1 - \sum_{i=1}^{\infty} (1-p)^i\, 2^{ai+b} \\
  &= 1 - \frac{2^{a+b}(1-p)}{1-(1-p)2^a} \\
  &\geq 1 - 2^{a+b+1}\,(1-p) \qquad \text{(since } 1/(1-(1-p)2^a) \leq 2\text{)}.
  \end{aligned}$$

  This allows us to obtain

  $$\gamma_{0,\mathbb{N}} + \gamma_{>0,\geq J} \leq \mathsf{dVal}^\sigma(\ell)\,(1 - 2^{a+b+1}(1-p)).$$

  In case we have moreover

  $$p \geq 1 - \frac{\varepsilon}{2^{a+b+1}|\mathsf{dVal}^\sigma(\ell)|}$$

we finally obtain $\gamma_{0,\mathbb{N}} + \gamma_{>0,\geq J} \leq \mathsf{dVal}^\sigma(\ell) + \varepsilon$ as expected.

Lower bound over p If we gather all the lower bounds over p that we need in the proof, we get that:

- if $\mathsf{dVal}^\sigma(\ell) \geq 0$, we must have

$$p \geq \max\left(1 - \frac{1}{2^{a+1}}, 1 - \frac{\varepsilon}{2^{b+a+2}(|V|\, W_{\mathsf{tr}} + 2w^+)}\right)$$

- if $\mathsf{dVal}^\sigma(\ell) < 0$, we must have

$$p \geq \max\left(1 - \frac{1}{2^{a+1}}, 1 - \frac{\varepsilon}{2^{b+a+2}(|V|\, W_{\mathsf{tr}} + 2w^+)}, 1 - \frac{\varepsilon}{2^{a+b+1}|\mathsf{dVal}^\sigma_(\ell)|}\right)$$

with ε small enough so that this bound is less than 1. This ends the proof that for all vertices ℓ, $\overline{\mathsf{mVal}}(\ell) \leq \mathsf{dVal}(\ell)$.

7.3. Characterisation of memoryless optimality in shortest path games

All SPGs admit an optimal deterministic strategy for both players: however, as we have seen in Example 1.30, Min may require memory to play optimally. In this case, Min does not have an optimal memoryless (randomised) strategy, but only has ε-optimal ones, for all $\varepsilon > 0$. But some SPGs indeed admit optimal memoryless strategies for Min: the strategy η^p described in the proof of Lemma 7.14 is indeed optimal in SPGs not containing negative cycles, for instance. We characterise here the SPGs in which Min admits an optimal memoryless strategy. For sure, Min does not have an optimal strategy if there is some vertex ℓ of value $\mathsf{dVal}(\ell) = -\infty$. Thus, **we only consider, in this section, SPGs without vertices ℓ with a deterministic value $-\infty$.**

We first recall the computations performed in [Bri+17] to compute deterministic values $\mathsf{dVal}(\ell)$. As for divergent WTGs, it consists of a value iteration based on the operator $\mathcal{F}\colon (\mathbb{Z} \cup \{+\infty\})^V \to (\mathbb{Z} \cup \{+\infty\})^V$ be the untimed version than one defined in Chapter 2, i.e. for all $X = (X_\ell)_{\ell \in V} \in (\mathbb{Z} \cup \{+\infty\})^V$ and all vertices $\ell \in V$ we let

$$\mathcal{F}(X)_\ell = \begin{cases} 0 & \text{if } \ell \in V_T\,; \\ \min_{(\ell,\ell')\in E} \left[\mathsf{wt}(\ell,\ell') + X_{\ell'}\right] & \text{if } \ell \in V_{\mathsf{Min}}\,; \\ \max_{(\ell,\ell')\in E} \left[\mathsf{wt}(\ell,\ell') + X_{\ell'}\right] & \text{if } \ell \in V_{\mathsf{Max}}\,. \end{cases}$$

We let $f_\ell^{(0)} = 0$ if $\ell \in V_T$ and $+\infty$ otherwise. By monotony of $\mathcal{F}$, the sequence $(f^{(i)} = \mathcal{F}^i(f^{(0)}))_{i\in\mathbb{N}}$ is non-increasing. It is proved to be stationary, and convergent towards $(\mathsf{dVal}(\ell))_{\ell\in V}$, the smallest fixpoint of $\mathcal{F}$. The pseudo-polynomial complexity of computing the values of SPGs comes from the fact that this sequence may become stationary after a pseudo-polynomial (and not polynomial) number of steps: the game of Figure 1.8 is one of the typical examples.

We introduce a new notion, being *the most permissive strategy* of Min at each step $i \geq 0$ of the computation. It maps each vertex $\ell \in V_{\mathsf{Min}}$ to the set

$$\widetilde{E}^{(i)}(\ell) = \left\{(\ell,\ell') \in E \mid \mathsf{wt}(\ell,\ell') + f_{\ell'}^{(i-1)} = f_\ell^{(i)}\right\}$$

of vertices that Min can choose. For each such most permissive strategy $\widetilde{E}^{(i)}$, we let $\widetilde{\mathcal{G}}^{(i)}$ be the SPG where we remove all edges (ℓ,ℓ') with $\ell \in V_{\mathsf{Min}}$ and $(\ell,\ell') \notin \widetilde{E}^{(i)}(\ell)$. This allows us to state the following result:

Theorem 7.15. *The following assertions are equivalent:*

1. Min *has an optimal memoryless deterministic strategy in* $\mathcal{G}$ *(for* dVal*);*
2. Min *has an optimal memoryless (randomised) strategy in* $\mathcal{G}$ *(for* $\overline{\mathsf{mVal}}$*);*
3. $f_\ell^{(|V|-1)} = f_\ell^{(|V|)} = \mathsf{dVal}_\ell$ *for all vertices* ℓ *(this means that the sequence* $(f^{(i)})$ *is stationary as soon as step* $|V|-1$*), and* Min *can guarantee that* V_T *is reached from all vertices in the game* $\widetilde{\mathcal{G}}^{(|V|-1)}$*.*

Remark. This characterisation of the existence of optimal memoryless strategy is testable in polynomial time since it is enough to compute vectors $f^{(|V|-1)}$ and $f^{(|V|)}$, check their equality, compute the sets $\widetilde{E}^{(|V|-1)}(\ell)$ (this can be done while computing $f^{(|V|)}$), and, finally, check whether Min can guarantee that the target in $\widetilde{\mathcal{G}}^{(|V|-1)}$ is reached by an attractor computation. ◁

We prove Theorem 7.15 in the rest of this section. First, we note that the implication 1 ⇒ 2 is trivial by the result of Theorem 7.1.

Implication 3 ⇒ 1 We consider any memoryless deterministic strategy σ that guarantees that Min reached V_T from all vertices in the game graph $\widetilde{\mathcal{G}}^{(|V|-1)}$. Then, for all vertices ℓ, we show by induction on n, that each play ρ from ℓ that reaches the target in at most n steps and is conforming to σ, has a payoff $\mathbf{P}(\rho) \leq \mathsf{dVal}(\ell)$. This is trivial for $n = 0$. If $\rho = \ell \xrightarrow{e} \rho'$ with ρ' starting in ℓ', then

$$\mathbf{P}(\rho) = \mathsf{wt}(e) + \mathbf{P}(\rho') \leq \mathsf{wt}(e) + \mathsf{dVal}(\ell') = \mathsf{wt}(e) + f_\ell^{(|V|-1)}\,.$$

- If $\ell \in V_{\mathsf{Max}}$, we have

$$\mathbf{P}(\rho) \leq \mathsf{wt}(e) + f_\ell^{(|V|-1)} \leq f_\ell^{(|V|)} = \mathsf{dVal}(\ell)\,.$$

- If $\ell \in V_{\mathsf{Min}}$, since $\ell' \in \widetilde{E}^{(|V|-1)}(\ell)$,

$$\mathbf{P}(\rho) = f_\ell^{(|V|)} = \mathsf{dVal}(\ell)\,.$$

This ends the proof by induction. To conclude that property 1 holds, since σ guarantees that the target is reached, all plays conforming to it reach the target in less than $|V|$ steps, which proves that $\mathsf{dVal}^\sigma(\ell) \leq \mathsf{dVal}(\ell)$, showing that σ is optimal.

Implication 1 $\Rightarrow$ 3 We consider an optimal deterministic memoryless strategy σ^*, such that for all vertices ℓ, $\mathsf{dVal}^{\sigma^*}(\ell) = \mathsf{dVal}(\ell)$.

First, we show that $f_\ell^{(|V|-1)} = \mathsf{dVal}(\ell)$ for all vertices ℓ. For that, consider the deterministic strategy τ of Max defined for all finite plays ρ having $n \leq |V|$ vertices, ending in a vertex $\ell \in V_{\mathsf{Max}}$, by $\tau(\rho) = (\ell, \ell')$ such that $\mathsf{wt}(\ell, \ell') + f_{\ell'}^{(|V|-1-n)} = f_\ell^{(|V|-n)}$. For longer finite plays, we define τ arbitrarily. Then, let ρ be the play from ℓ conforming to σ^* and τ. Since σ^* ensures reaching the target and is memoryless and deterministic, ρ reaches the target in at most $|V|-1$ steps. Let $\rho = \ell_0 \xrightarrow{e_0} \ell_1 \cdots \xrightarrow{e_{k-1}} \ell_k$ with $k \leq |V|$. Let us show that $\mathbf{P}(\rho) \geq f_{\ell_0}^{(|V|-1)}$. We prove by induction on $0 \leq j \leq k$ that

$$\sum_{i=j}^{k-1} \mathsf{wt}(e_i) \geq f_{\ell_j}^{(|V|-1-j)}\,.$$

When $j = k-1$, the result is trivial since the sum is

$$0 = f_{\ell_k}^{(0)} \geq f_{\ell_k}^{(|V|-1-(k-1))}$$

by using that $(f_{\ell_k}^{(i)})_i$ is non-increasing. Otherwise, by induction hypothesis

$$\sum_{i=j}^{k-1} \mathsf{wt}(e_i) \geq \mathsf{wt}(e_j) + f_{\ell_{j+1}}^{(|V|-1-(j+1))}\,.$$

- If $\ell_j \in V_{\mathsf{Max}}$, ℓ_{j+1} is chosen by τ so that

$$\mathsf{wt}(e_j) + f_{\ell_{j+1}}^{(|V|-1-(j+1))} = f_{\ell_j}^{(|V|-1-j)}\,.$$

- If $\ell \in V_{\mathsf{Min}}$, by definition of $\mathcal{F}$,

$$\mathsf{wt}(e_j) + f_{\ell_{j+1}}^{(|V|-1-(j+1))} \geq f_{\ell_j}^{(|V|-1-j)}.$$

We can conclude in all cases, so that $f_\ell^{(|V|-1)} = \mathsf{dVal}(\ell)$ for all vertices ℓ.

Then, we show that Min can guarantee that V_T is reached from all vertices in the SPG $\widetilde{\mathcal{G}}^{(|V|-1)}$. Let us suppose that this is not the case. Then, there exists a set V' of vertices of $\widetilde{\mathcal{G}}^{(|V|-1)}$ in which Max can trap Min forever: for all $\ell' \in V' \cap V_{\mathsf{Min}}$, $\widetilde{E}^{(|V|-1)}(\ell') \subseteq V'$, and for all $\ell' \in V' \cap V_{\mathsf{Max}}$, $E(\ell) \cap E' \neq \emptyset$. Since σ^* guarantees that the target is reached, there exists $\ell \in V' \cap V_{\mathsf{Min}}$ such that $\sigma^*(\ell) = \ell' \notin V'$: then $\mathsf{wt}(\ell, \ell') + \mathsf{dVal}_{\ell'} > \mathsf{dVal}(\ell)$ (here we use that $\mathsf{dVal}(\ell) = f_\ell^{(|V|-1)} = f_\ell^{(|V|)}$). We consider an optimal deterministic memoryless strategy τ^* of Max in $\mathcal{G}$ (which we know exists by Theorem 1.31-2). Then, the play ρ from ℓ conforming to σ^* and τ^* starts by taking the edge (ℓ, ℓ') and continues with a play ρ'. By optimality, we know that $\mathbf{P}(\rho) = \mathsf{dVal}(\ell)$ and $\mathbf{P}(\rho') = \mathsf{dVal}(\ell')$. However,

$$\mathbf{P}(\rho) = \mathsf{wt}(\ell, \ell') + \mathbf{P}(\rho') = \mathsf{wt}(\ell, \ell') + \mathsf{dVal}(\ell') > \mathsf{dVal}(\ell)$$

which raises a contradiction.

Implication 2 $\Rightarrow$ 1 We finish with a constructive proof that actually allows one to build an optimal memoryless deterministic strategy when it exists. We consider thus an optimal memoryless strategy η^* for the memoryless value. We build a memoryless deterministic strategy σ^*, and show that it is also optimal. The construction of σ^* is performed in two steps.

We first select transitions that minimise the value of η^* at horizon 1. Formally, for all vertices ℓ, we define

$$\widetilde{\Delta}(\ell) = \operatorname*{argmin}_{(\ell,\ell') \in \mathrm{supp}(\eta(\ell))} [\mathsf{wt}(\ell, \ell') + \mathsf{mVal}^{\eta}(\ell')] \tag{7.10}$$

In a first approximation, we thus fix $\sigma^*(\ell)$ to be any edge of $\widetilde{E}(\ell)$, for all vertices ℓ. The next example shows that σ^* may not be optimal.

Example 7.16. We consider the SPG of Figure 7.7, with the memoryless strategy η^* defined by $\eta^*(\ell_0) = \mathsf{Dirac}_{(\ell_0,\ell_1)}$ and $\eta^*(\ell_1) = (1-p) \times \mathsf{Dirac}_{(\ell_1,\ell_0)} + p \times \mathsf{Dirac}_{(\ell_1,\☺)}$, where $p \in (0, 1)$. Then, we can check that $\mathsf{mVal}^{\eta^*}(\ell_0) =$ and $\mathsf{mVal}^{\eta^*}(\ell_1) = -1$, which implies that η^* is optimal.

Now, we have $\widetilde{E}(\ell_0) = \{(\ell_0, \ell_1)\}$ and $\widetilde{E}(\ell_1) = \{(\ell_1, \ell_0), (\ell_1, \☺)\}$. However, considering the strategy σ^* defined by $\sigma^*(\ell_0) = (\ell_0, \ell_1)$ and $\sigma^*(\ell_1) = (\ell_1, \ell_0)$, results in a determinstic value $\mathsf{dVal}^{\sigma^*}(\ell_0) = \mathsf{dVal}^{\sigma^*}(\ell_1) = +\infty$: thus σ^* is not optimal. ⌟

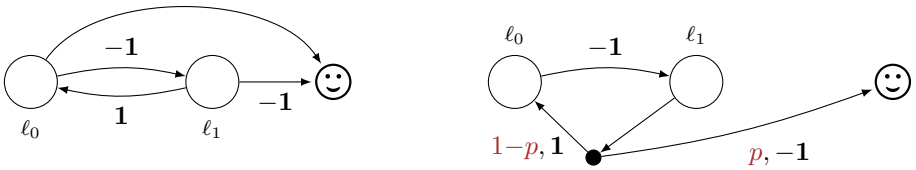


Figure 7.7 – An SPG (on the left) and the Markov chain was obtained by fixing an optimal memoryless strategy, with $p \in (0, 1)$.

We employ techniques already used in [CAH04] to refine the definition of η^*. For each vertex ℓ in the game, we let $d(\ell)$ be the attractor distance of ℓ to the target in the finite MDP $\mathcal{G}^\eta$, i.e. the smallest number of steps such that Min has a path of this length reaching the target, no matter how is playing Max (notice that this may be different from the attractor distance given in the whole SPG since some edges are taken with probability 0 in η). For all vertices ℓ, $d(\ell) < +\infty$ since otherwise (η being a memoryless strategy) the finite MDP $\mathcal{G}^\eta$ would contain a bottom strongly connected component without target states, which would contradict the fact that η ensures reaching V_T with probability 1. We then let, for all vertices $\ell \in V_{\mathsf{Min}}$, $\sigma^*(\ell)$ be any edge in $\operatorname{argmin}_{(\ell,\ell')\in\widetilde{E}(\rho)} d(\ell')$.

Example 7.17. On the same example as above, using the fact that $d(☺) = 0$, $d(\ell_1) = 1$ and $d(\ell_0) = 2$ (since the edge $(\ell_0, ☺)$ is not present in $\mathcal{G}^\eta$), we have no choice but to let σ^* be defined by $\sigma^1(\ell_0) = (\ell_0, \ell_1)$ and $\sigma^1(\ell_1) = (\ell_1, ☺)$, which is indeed an optimal deterministic (and memoryless) strategy. ⌟

Lemma 7.18. *The memoryless deterministic strategy σ^* defined above is optimal.*

Proof. We show this by proving independently that:

1. each finite play ρ conforming to σ^* from ℓ and reaching the target has a payoff at most $\mathsf{mVal}^\eta(\ell)$;
2. no play conforming to σ^* can contain a cycle.

From the second property, we are thus certain that all plays conforming to σ^* reach the target. The first property then allows us to conclude that σ^* is optimal.

1. We prove the property by induction on the length of finite plays ρ conforming to σ^* that reaches the target, for all initial vertices ℓ. If ρ has length 0, this means that $\ell \in V_T$, in which case $\mathbf{P}(\rho) = 0 = \mathsf{mVal}^{\eta^*}(\ell)$. We consider then a play $\rho = \ell \xrightarrow{\delta} \rho'$ such that $|\rho| \geq 1$ and ρ' starting by the vertex ℓ', so that $\mathbf{P}(\rho) = \mathsf{wt}(\delta) + \mathbf{P}(\rho')$. By induction hypothesis, $\mathbf{P}(\rho') \leq \mathsf{mVal}^{\eta^*}(\ell')$, so that

$$\mathbf{P}(\rho) \leq \mathsf{wt}(\delta) + \mathsf{mVal}^{\eta^*}(\ell')\,.$$

We suppose first that $\ell \in V_{\mathsf{Max}}$. By Lemma 7.13, we consider a deterministic best-response strategy τ of Max against η^*. Then $\mathbb{E}_u^{\eta^*,\tau} = \mathsf{mVal}^{\eta^*}(u)$ for all $u \in V_{\mathsf{Max}}$, and by (7.5), letting $(u,u') = \tau(u)$,

$$\mathbb{E}_u^{\eta^*,\tau} = \mathsf{wt}(u,u') + \mathbb{E}_{u'}^{\eta^*,\tau}\,.$$

We thus know that $\mathsf{wt}(u,u') + \mathbb{E}_{u'}^{\eta^*,\tau} = \mathsf{mVal}^{\eta^*}(u)$. In particular, for all vertices $u \in V_{\mathsf{Max}}$ and edge $(u,u') \in E$,

$$\mathsf{wt}(u,u') + \mathsf{mVal}^{\eta^*}(u') \leq \mathsf{mVal}^{\eta^*}(u) \tag{7.11}$$

In particular, $\mathbf{P}(\rho) \leq \mathsf{wt}(\ell,\ell') + \mathsf{mVal}^{\eta^*}(\ell') \leq \mathsf{mVal}^{\eta^*}(\ell)$.

If $\ell \in V_{\mathsf{Min}}$, then $e = (\ell,\ell') \in \widetilde{E}(\ell)$ so that $\mathsf{wt}(e) + \mathsf{mVal}^{\eta^*}(\ell')$ is minimum over all possible $(\ell,\ell') \in \mathrm{supp}(\eta^*(\ell))$. The system (7.5) implies that

$$\mathbb{E}_\ell^{\eta^*,\theta} = \sum_{(\ell,\ell'')\in E} \eta^*(\ell)(\ell,\ell'') \times \big(\mathsf{wt}(\ell,\ell'') + \mathbb{E}_{\ell''}^{\eta^*,\theta}\big).$$

for all memoryless strategies θ of Max. In particular, for the best-response deterministic and memoryless strategy θ given by Lemma 7.13,

$$\begin{aligned}\mathsf{mVal}^{\eta^*}(\ell) = \mathbb{E}_\ell^{\eta^*,\theta} &= \sum_{(\ell,\ell'')\in \mathrm{supp}(\eta^*(\ell))} \eta^*(\ell)(\ell,\ell'') \times (\mathsf{wt}(\ell,\ell'') + \mathsf{mVal}^{\eta^*}(\ell'')) \\ &\geq \mathsf{wt}(\ell,\ell') + \mathsf{mVal}^{\eta^*}(\ell')\end{aligned} \tag{7.12}$$

so that we also get $\mathbf{P}(\rho) \leq \mathsf{mVal}^{\eta^*}(\ell)$.

2. Suppose that a cycle $\ell_1\ell_2\cdots\ell_k\ell_1$ conforms to σ^*, with ℓ_1 a vertex of minimal distance $d(\ell_1)$. We can choose ℓ_1 such that it belongs to Min, since the distance is computed by an attractor for Min (and thus all successors of a vertex of Max must have a smaller distance). By minimality of $d(\ell_1)$ among the vertices of the cycle, $d(\ell_2) \geq d(\ell_1)$. Moreover, by the attractor computation, there exists $(\ell_1,\ell') \in E(\ell_1)$ such that $d(\ell') = d(\ell_1) - 1 < d(\ell_1)$. By definition of σ^*, we know that $(\ell_1,\ell') \notin \widetilde{E}(\ell_1)$, so that

$$\mathsf{wt}(\ell_1,\ell') + \mathsf{mVal}^{\eta^*}(\ell') > \mathsf{wt}(\ell_1,\ell_2) + \mathsf{mVal}^{\eta^*}(\ell_2)\,.$$

By (7.12), we know that in this case

$$\mathsf{mVal}^{\eta^*}(\ell_1) > \mathsf{wt}(\ell_1,\ell_2) + \mathsf{mVal}^{\eta^*}(\ell_2)\,.$$

By optimality of η^*, this rewrites in

$$\overline{\mathsf{mVal}}(\ell_1) > \mathsf{wt}(\ell_1, \ell_2) + \overline{\mathsf{mVal}}(\ell_2)\,.$$

By Theorem 7.1 applying in the case of SPGs, this also rewrites in

$$\mathsf{dVal}(\ell_1) > \mathsf{wt}(\ell_1, \ell_2) + \mathsf{dVal}(\ell_2) \geq \mathcal{F}\big((\mathsf{dVal}(\ell))_{\ell \in V}\big)(\ell_1)$$

(since $\ell_1 \in V_{\mathsf{Min}}$): this contradicts the fact that the vector $(\mathsf{dVal}(\ell))_{\ell \in V}$ is a fixpoint of $\mathcal{F}$. □

Discussion

This part has studied the trade-off between memoryless and deterministic strategies, showing that Min guarantees the same value when restricted to these two kinds of strategies, or when Min can play with both memory and randomisation. To achieve this goal, we introduced a new class of strategies, the stochastic strategies, from which we deduced many new values. In particular, we developed good hypotheses on stochastic strategies to obtain mathematical coherence for these strategies.

Trading memory for probabilities in all WTGs

The determinacy and the equality between values hold both in divergent WTGs, and SPGs (i.e. WTGs with no clocks). We aim at extending our study to more general WTGs, though we have studied counter-examples (Examples 7.2 and 7.3) to the determinacy result with respect to the memoryless value, and thus of the equality between memoryless and deterministic values.

Direct extension of our result In our work, two results naturally ask for an extension: the determinacy and the equality with the deterministic value for the stochastic values (Theorem 6.1) and the memoryless values (Theorem 7.1).

When we consider the memoryless values, we know that the generalisation of our result is impossible for all WTGs (as illustrated by counter-examples). However, we could still hope to weaken slightly the necessary condition to obtain it. For instance, we can consider the class of *almost-divergent* WTGs (adding the possibility for the executions following a region cycle to have weight *exactly* 0), used in [BJM15; BMR18] to obtain an approximation schema of the deterministic value function. In particular, the existence of almost-optimal switching strategies for Min and almost-optimal memoryless strategies for Max are crucial arguments to extend our emulation result.

When we consider the stochastic values, we do not have a counter-example for the non-determinacy or the inequality of values. However, the general proof seems tricky. In Lemma 6.6, we exploit the existence of almost-optimal smooth deterministic strategies w.r.t. the deterministic value for both players (that seems reasonable) to conclude the determinacy and the equality of the deterministic value

with the stochastic value. However, even if we prove this fact in some classes of WTGs (divergent, one-clock and SPGs), it remains open in all WTGs. Indeed, proving the existence of an almost-optimal strategy satisfying Hypothesis 1, i.e. the measurability of the memory of the strategies, for both players implies that the deterministic value is smooth enough. For instance, to apply Kuratowski and Ryll-Nardzewski's measurable selection theorem [KR65], as in Proposition 6.2, the deterministic value function needs to be at least Lebesgue-measurable, which we cannot confirm or deny for all WTGs. Moreover, to define a smooth deterministic strategy for Min, the strategy must be proper: we morally need to prove that Min wins by going quickly to the target. However, we know that, in some WTGs, Max must deal with some convergence phenomena in its strategies (see Example 4.6), and we do not know if Min can deal with such phenomena with its strategies.

Asymmetric definition for the memoryless value Reducing the memory used by Max decreases its power: we can not emulate the memory of strategies of Max by the probabilities in all WTGs as illustrated in counter-examples (Examples 7.2 and 7.3). However, from a controller synthesis perspective, we do not want to restrict the set of strategies used by the environment (i.e. Max). In particular, it could be more meaningful to study the value obtained by Min when being forced to play with a restricted set of strategies (i.e. memoryless or deterministic) while not enforcing any restrictions on strategies used by Max. Thus, by breaking the symmetry in the definitions of the values (as the memoryless value), we hope to obtain an extension of our result to other classes of WTGs.

When we focus on an asymmetric (smooth) deterministic value, the result of Proposition 6.2, on the smooth deterministic best-response of Max against any stochastic strategies of Min, ensures that this value is identical to the smooth deterministic value in all WTGs.

A difference may appear when we consider the memoryless value. In particular, we introduce an asymmetric memoryless value such that we do not constrain strategies for Max. Formally, in all WTGs and for all configurations (ℓ, ν), we fix

$$\overline{\mathsf{amVal}}(\ell,\nu) = \inf_{\eta\in\mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}} \sup_{\theta\in\mathsf{Strat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu} \quad \text{and} \quad \underline{\mathsf{amVal}}(\ell,\nu) = \sup_{\theta\in\mathsf{Strat}_{\mathsf{Max}}} \inf_{\eta\in\mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu}\,.$$

As expected, these asymmetric memoryless values increase the power of Max as set by the memoryless and stochastic values:

Lemma II.1. *In all WTGs, for all configurations (ℓ, ν), we have*

$$\underline{\mathsf{mVal}}(\ell,\nu) \le \underline{\mathsf{amVal}}(\ell,\nu)\,, \quad \overline{\mathsf{mVal}}(\ell,\nu) \le \overline{\mathsf{amVal}}(\ell,\nu) \quad \textit{and}$$
$$\underline{\mathsf{Val}}(\ell,\nu) \le \underline{\mathsf{amVal}}(\ell,\nu)\,, \quad \overline{\mathsf{Val}}(\ell,\nu) \le \overline{\mathsf{amVal}}(\ell,\nu)\,.$$

Proof. We consider a WTG and one of their configurations (ℓ, ν). The lower-bound defined by the memoryless values exploits the inclusion over strategies of Max. In particular, since $\mathsf{mStrat}_{\mathsf{Max}} \subseteq \mathsf{Strat}_{\mathsf{Max}}$, we have

$$\underline{\mathsf{mVal}}_{\ell,\nu} = \sup_{\theta \in \mathsf{mStrat}_{\mathsf{Max}}} \inf_{\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu} \leq \sup_{\theta \in \mathsf{Strat}_{\mathsf{Max}}} \inf_{\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu} = \underline{\mathsf{amVal}}_{\ell,\nu} .$$

Moreover, we note that for all $\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}$, $\sup_{\theta \in \mathsf{mStrat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu} \leq \sup_{\theta \in \mathsf{Strat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu}$. Thus, we deduce that

$$\overline{\mathsf{mVal}}(\ell, \nu) = \inf_{\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}} \sup_{\eta \in \mathsf{mStrat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu} \leq \inf_{\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}} \sup_{\eta \in \mathsf{Strat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu} = \overline{\mathsf{amVal}}(\ell, \nu) .$$

Analogously, the lower-bound defined by the (stochastic) values exploits the inclusion over strategies of Min. In particular, since $\mathsf{mStrat}_{\mathsf{Min}} \subseteq \mathsf{Strat}_{\mathsf{Min}}$, we have

$$\overline{\mathsf{Val}}(\ell, \nu) = \inf_{\eta \in \mathsf{Strat}^{\mathsf{p}}_{\mathsf{Min}}} \sup_{\eta \in \mathsf{mStrat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu} \leq \inf_{\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}} \sup_{\eta \in \mathsf{Strat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu} = \overline{\mathsf{amVal}}(\ell, \nu) .$$

Again, we remark that for all $\theta \in \mathsf{Strat}_{\mathsf{Max}}$, $\inf_{\eta \in \mathsf{Strat}_{\mathsf{Min}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu} \leq \inf_{\eta \in \mathsf{mStrat}_{\mathsf{Min}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu}$. Thus, we deduce that

$$\underline{\mathsf{Val}}(\ell, \nu) = \sup_{\theta \in \mathsf{Strat}_{\mathsf{Max}}} \inf_{\eta \in \mathsf{Strat}_{\mathsf{Min}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu} \leq \sup_{\theta \in \mathsf{Strat}_{\mathsf{Max}}} \inf_{\eta \in \mathsf{mStrat}_{\mathsf{Min}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu} = \underline{\mathsf{amVal}}(\ell, \nu) . \qquad \square$$

Now, when we consider only divergent WTGs or SPGs (i.e. classes of WTGs where Min can emulate the memory by probabilities), the power of Min does not decrease. Indeed, Min can emulate the memory of their strategies whatever Max does (and not only against memoryless strategies of Max):

Theorem II.2. *The asymmetric memoryless value is determined and equal to the deterministic value in divergent WTGs, and SPGs.*

Proof. We independently prove the result for the two classes of WTGs.

The case of divergent WTGs: Let $\mathcal{G}$ be a divergent WTG, and (ℓ, ν) be a configuration of $\mathcal{G}$. By Lemma II.1, we know that $\underline{\mathsf{mVal}}(\ell, \nu) \leq \underline{\mathsf{amVal}}(\ell, \nu)$. In particular, to conclude the proof in this case, we need to prove that $\overline{\mathsf{amVal}}(\ell, \nu) \leq \mathsf{dVal}(\ell, \nu)$. Indeed, in this case, since $\underline{\mathsf{mVal}}(\ell, \nu) = \mathsf{dVal}(\ell, \nu)$ (by Theorem 7.1), we obtain that

$$\underline{\mathsf{mVal}}(\ell, \nu) \leq \underline{\mathsf{amVal}}(\ell, \nu) \leq \mathsf{dVal}(\ell, \nu) = \underline{\mathsf{mVal}}(\ell, \nu) .$$

By Proposition 6.2, we note that Max has a deterministic best-response strategy against all proper stochastic strategies for Min w.r.t. the stochastic value. More precisely, for all $\eta \in \mathsf{Strat}^{\mathsf{p}}_{\mathsf{Min}}$ and $\varepsilon > 0$, there exists a deterministic strategy

$\tau \in \mathsf{dStrat}_{\mathsf{Max}}$ such that for all configurations (ℓ, ν),

$$\mathsf{Val}^{\eta}(\ell,\nu) - \varepsilon = \sup_{\theta \in \mathsf{Strat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu} - \varepsilon \leq \mathbb{E}^{\eta,\tau}_{\ell,\nu} \leq \sup_{\tau \in \mathsf{dStrat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\tau}_{\ell,\nu}\,.$$

In particular, since these inequalities hold for all $\varepsilon > 0$, we deduce that

$$\overline{\mathsf{amVal}}(\ell,\nu) = \inf_{\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}} \sup_{\theta \in \mathsf{Strat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\theta}_{\ell,\nu} \leq \inf_{\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}} \sup_{\tau \in \mathsf{dStrat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\tau}_{\ell,\nu}\,.$$

In the proof of Proposition 7.5, we show that for all $\varepsilon > 0$, there exists $\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}$ such that for all $\tau \in \mathsf{dStrat}_{\mathsf{Max}}$, $\mathbb{E}^{\eta,\tau}_{\ell,\nu} \leq \mathsf{dVal}(\ell,\nu) + \varepsilon$. In particular, we deduce that

$$\overline{\mathsf{amVal}}(\ell,\nu) \leq \inf_{\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}} \sup_{\tau \in \mathsf{dStrat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\tau}_{\ell,\nu} \leq \sup_{\tau \in \mathsf{dStrat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\tau}_{\ell,\nu} \leq \mathsf{dVal}(\ell,\nu) + \varepsilon\,.$$

Since the inequality holds for all $\varepsilon > 0$, we deduce that $\overline{\mathsf{amVal}}(\ell,\nu) \leq \mathsf{dVal}(\ell,\nu)$.

The case of SPGs: We consider $\mathcal{G}$ be a SPG and ℓ one of its locations. By Lemma 7.13, against all proper memoryless strategies for Min, Max has a memoryless deterministic best-response w.r.t. the stochastic value, i.e. for all vertices ℓ, we deduce that

$$\sup_{\theta \in \mathsf{Strat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\theta}_{\ell} \leq \sup_{\tau \in \mathsf{dStrat}_{\mathsf{Max}} \cap \mathsf{mStrat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\tau}_{\ell}\,.$$

More precisely, we deduce that

$$\overline{\mathsf{amVal}}(\ell) = \inf_{\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}} \sup_{\theta \in \mathsf{Strat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\theta}_{\ell} \leq \inf_{\eta \in \mathsf{mStrat}^{\mathsf{p}}_{\mathsf{Min}}} \sup_{\theta \in \mathsf{mStrat}_{\mathsf{Max}} \cap \mathsf{dStrat}_{\mathsf{Max}}} \mathbb{E}^{\eta,\theta}_{\ell}\,.$$

Thus, we deduce by inclusion of strategies that $\overline{\mathsf{amVal}}(\ell) \leq \overline{\mathsf{mVal}}(\ell)$. Moreover, Lemma II.1, we conclude that

$$\underline{\mathsf{mVal}}(\ell) \leq \underline{\mathsf{amVal}}(\ell) \leq \overline{\mathsf{amVal}}(\ell) \leq \overline{\mathsf{mVal}}(\ell)\,.$$

and we conclude by Theorem 7.1. □

Finally, our aim with this new memoryless value is the extension of Theorem II.2, at least in the case of one-clock WTGs. This new value is a good candidate since it solves our problem in Example 7.3. Indeed, in the game depicted in Figure 7.2, with a deterministic strategy that using memory, Max can guarantee, for all $\varepsilon > 0$, that the target location is reached with at most $-\varepsilon$ (whatever Min does). In particular, Min has an almost-optimal memoryless strategy: directly reaching the target location. Thus, asymmetric memoryless values are equal to deterministic ones. However, we do not know how to extend the proof to all one-clock WTGs.

Indeed, we need to encode a deterministic strategy for Min in a memoryless strategy (like in Proposition 7.5). To do it, our method exploits almost-optimal switching strategies for Min w.r.t. the deterministic value, while its existence in all one-clock WTGs remains open.

Implementability of randomised strategies Another question concerns the implementation of the randomised strategies: even if they use no memory, they still need to know the precise current clock valuation and may play a precise delay.

First, previous work [CHP08b] on (non-weighted) timed games aimed at removing this need for precision over delay, by using stochastic strategies where the delays are chosen with probability distributions that do not require exact knowledge of the clocks measurements. In our setting, we aim to further study the implementation of the randomised strategies of Min in WTGs, e.g. by requiring them to be robust against small imprecisions (as explored in the next part).

Second, to remove the precision over the valuations, we want to define strategies that are uniform over regions. In other words, we want to define strategies that make the same choice from all valuations in the region (or a finite partition of each region). We hope that probabilities help us to smooth choices along the region (instead of choosing the best one, we can choose an average).

Trading memory in concurrent WTG A natural extension of the WTGs is concurrent WTGs. In this case, we recall that, at each turn, both players choose a decision that is applied when its delay is minimal. Already, in concurrent finite games, stochastic strategies are needed in this context to guarantee the existence of a Nash equilibrium [AHK98], or to guarantee the existence of optimal (memoryless) strategies for the reachability objective [dM04, Theorem 1]. Thus, even if introducing stochastic strategies is natural, studying the trade-off between memory and probabilities is more complex in concurrent WTGs. Indeed, in concurrent reachability timed games, Min needs infinite memory to win with deterministic strategies [Alf+03] and finite memory is necessary to win with randomised strategies [CHP08a].

Characterisation of the memory used to win

A recent series of works [Bou+21; Bou+22; Koz22; BRV23] look for a characterisation of the memory needed for each player to win in a finite two-player game according to some properties over the objective of the controller. For instance, we know a characterisation to decide when a prefix-independent objective is positionally determined [GZ05], in the sense that both players have a winning memoryless strategy. However, these works focus on deterministic strategies and we aim to

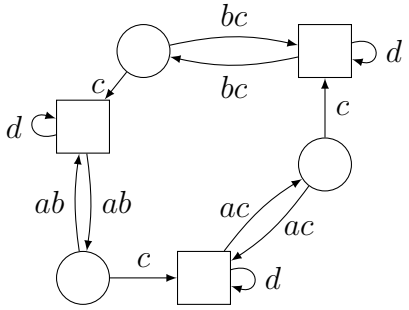


Figure II.2 – A Müller game in which the controller (depicted by squares) needs memory to win.

extend this series by considering stochastic strategies too. In [CAH04], authors propose a first study in this sense for stochastic games under qualitative objectives like parity or Rabin objectives.

Under some objectives (as illustrated in the following examples), probabilities do not remove (or reduce) the memory used by each player. A natural question is to know under which properties on the payoff we can have a tradeoff between memory and probabilities. To be more general, under which properties on the payoff, we know that at least one player can play without memory (but with probabilities).

Müller games Müller games (without probabilities in the arena) are the first example of classes where memoryless stochastic strategies do not suffice for the controller. More precisely, a Müller game which is an extension of parity games, is an infinite duration two-player game in which the controller wins if he can produce an infinite play such that the colours seen along the play verify a regular property. When we consider deterministic strategies, the memory used by the controller is given by *Zielonka's tree* [Zie98]. However, even if we allow both players to play with stochastic strategies, the controller may need memory to win, as illustrated in the following example.

Example II.3. We consider the Müller game depicted in Figure II.2 over the set of colours $\mathcal{C} = \{a, b, c, d\}$, with the following objective $\mathcal{F} = \{A \subseteq \mathcal{C} \mid |A| \text{ is even}\}$. In this game, the controller (depicted by squares) has a winning deterministic strategy with one bit of memory (in all generality, we need one bit for each strongly connected component).

However, he does not have a memoryless winning strategy even if we allow him to use probabilities: let's try to define such a strategy. Since he must see infinitely often two colours, he can not play d with probability 1 in any vertex: otherwise, the play will see only one colour infinitely often. Now, if he chooses the colour d with probability 0 in all vertices, then the environment has a (deterministic) memoryless

winning strategy: he always plays the colour c. In this case, only three colours have been seen infinitely often (the colour d is never used). Finally, if there exists a vertex in which the controller chooses the colour d with a non-null probability, then the environment has also a (deterministic) memoryless winning strategy: in the following vertex instead of playing the colour c, he comes back to the previous vertex. In this case, only three colours can be seen since the environment avoids the controller changing the cycle. Thus, when we consider strategies with probability, the controller also needs one bit of memory to remember if the previously seen letter is d (and in this case, he chooses to go in the cycle), or if it is c (and in this case, he chooses d). ⌟

Even if probabilities can not emulate all memory used by the controller, we conjecture that probabilities reduce the memory to half of the size of Zielonka's tree, i.e. stochastic strategies reduce half the memory used by the controller. Our intuition is that probabilities allow us to emulate memory inside a level of the tree, but not the memory to change the level inside the tree.

Finding a polynomial time algorithm for SPGs

In finite game theory, a long-standing open question consists on looking after a polynomial time algorithm to solve mean-payoff games (and thus parity games). Moreover, even if SPGs are close to mean-payoff ones (there exists a polynomial reduction between then [Bri+17]), computing deterministic value in SPGs is done in pseudo-polynomial time (Theorem 1.31). Indeed, SPGs are solved with the value iteration technique (see Chapter 2) that converges into pseudo-polynomial iterations. However, in the previous chapter, we give a (polynomial time) characterisation to decide if value iteration converges in polynomial time. Now, to try to solve SPGs in polynomial time, we want to investigate other classes of algorithms.

The theoretical result obtained in this part, and more precisely the value restricted by using only randomisation and no memory could pave the way to new kinds of computation (or approximation) algorithms, possibly with a better complexity; using *strategy improvement algorithms*. To solve linear programming two classes of algorithm exist: simplex methods (in exponential time) by only looking at the borders of the set of solutions, and *interior-point method*s (in polynomial time) that look inside the set of solutions. Since the value iteration paradigm is close to simplex methods, we aim to develop some links between strategy improvement algorithms and the interior-point paradigm. Indeed, in our context, we can define interior points as non-deterministic randomised strategies, while the corners of the simplex are indeed deterministic strategies. Thus, having a way to dynamically improve from a randomised strategy to a better one, in the same way

as the interior-point method proposes a way to move in the interior of the set of valid points of the linear program, could enable new types of algorithms.

Part III.

Robust Values

Introduction

We used probabilities to remove the structural memory of strategies but not the memory induced by the infinite precision of clocks. Indeed, to play almost optimally, Min wants to play close to a non-reachable optimal decision (due to a guard or a future action of Max). For instance, as illustrated in Example 1.26, Min must play decisions with a delay $1/2 - \varepsilon$ where ε is as small as necessary. Moreover, this infinite precision of clocks relies on the real-time execution of the system: each transition is taken immediately after having spent the delay. This behaviour inherited of timed automata is not only unrealistic, but also not implementable.

Example of using robustness for modelling a simple real-time system We consider a camera that captures a frame while an encoder treats a frame every 40 milliseconds. We assume that the camera directly sends the frame to the encoder that uses a bounded buffer to store incoming frames. By abstracting the computations, we concentrate on execution times and the scheduling of these tasks. Our objective is to model a system that can treat an infinite number of tasks. Intuitively, to succeed, the encoder should not be too late treating the frames. Since both periods are the same, the system can realise this specification when the encoder treats the frame immediately after the recording. However, this behaviour does not account for slightly different times of execution between the camera and the encoder.

In particular, even if the "exact" behaviour of the system is correct, we want to check its robustness, i.e. if the system may continue forever (without buffer overflow) when some perturbations may happen in the execution of tasks of the camera or the encoder. Consider a simple perturbation which consists in increasing or decreasing by a small amount the execution times. For instance, the camera may work slightly faster than expected by sending frames with a period of $40 - \varepsilon$, while the encoder is still treating each frame in 40 milliseconds. In particular, the encoder is now slower than the camera: it will eventually skip frames or have a buffer overflow. The system is not robust since it no longer satisfies the specification (being able to run indefinitely) for all $\varepsilon > 0$.

Robustness in weighted timed games From the implementation context, a natural way to introduce robustness in the model is the definition of semantics that

enlarge guards with a parameter. Hence, we naturally enlarge the set of decisions that satisfies the guard by considering the shifting in execution time induced by the implementation. Instead of using parametric timed automata [AHV93], we focus on the robustness as an interpretation by a two-player timed game where one player can only influence the choice between the expected exact behaviour and the realistic execution that we call the *perturbation* [BMS15; Bus+19; San+13].

In timed games, this notion of robustness is easily extensible by allowing the environment, i.e. Max, to control its own choices and the perturbations (according to a given bound) on delays chosen by the controller, i.e. Min. In particular, we use asymmetric semantics between both players since the power of Max increases: controlling perturbations allows him to avoid certain regions or, on the contrary, to force their reachability. The moment when the guard is tested (before or after the perturbation) defines two robust semantics [CHP11; SBM11; BMS13; SBM14; BMS15; Guh+15] called *excessive* and *conservative* semantics. Under conservative semantics, perturbed delays (i.e. delays with the perturbation) must satisfy guards, whereas excessive semantics checks the guard before the perturbation. Intuitively, excessive semantics allows some new behaviours in the system (some regions become reachable), whereas the conservative semantic restricts the set of behaviours (some regions become unreachable). Formal definitions of these semantics, with a general definition of robust semantics, are introduced in Chapter 8.

Values guaranteed by Min under robust semantics Our objective is to study the impact of robust semantics on the value of the game by introducing three new robust values. When we fix the parameter, we suppose that it is negligible regarding the constants in the system: hard coding it in the system induces a state space explosion which is not easy to manage. To deal with the space explosion, we suppose that the value of the parameter is not known a priori: we study the impact of the parameter on guarantees obtained on the system. In particular, this approach allows us to identify the behaviour of the system for arbitrarily small values of the parameter.

We fix the parameter $\mathfrak{p} > 0$ of the robust semantics, i.e. the set of possible perturbations is the interval $[-\mathfrak{p}, \mathfrak{p}]$.

- We first define the *fixed-parameter robust value*, denoted $\mathsf{rVal}^{\mathfrak{p}}$, when both players play under the robust semantics of the parameter $\mathfrak{p}$.
- Another way to define a robust value when the parameter is fixed, is the definition of the *half-fixed-parameter robust value* introduced in [BMS13] (under the name *limit value*) and denoted here by $\overline{1/2\mathsf{rVal}}^{\mathfrak{p}}$. The half-fixed-parameter robust value consists in the value defined when Min plays under the robust semantics of parameter $\mathfrak{p}$, whereas Max plays under the robust semantics of

Value problems		conservative	excessive
fixed-parameter robust value $\mathsf{rVal}^{\mathfrak{p}}$	WTG	Undecidable [BMS13]	
	Divergent WTG	3-EXPTIME Theorem 9.28	Open
	1-clock WTG (non-neg)	Decidable Theorem 9.32	Decidable [Guh+15]
half-fixed-parameter robust value $\overline{1/2\mathsf{rVal}}^{\mathfrak{p}}$	WTG	Undecidable [BMS13]	
	WTA	Open	Undecidable [BMS13]
limit robust value $\overline{\mathsf{rVal}}$	WTG	Undecidable [San13a]	Undecidable [San13a; Guh+15]
	WTA	PSPACE-complete [BMS13]	Undecidable [San13a]
	Acyclic WTG	Decidable (Theorem 10.4)	
	Divergent WTG (non-neg)	Decidable Theorem 10.16	Open

Table III.1 – Solving robust value problems under conservative and excessive semantics.

parameter $\mathfrak{p}' \in (0, \mathfrak{p}]$ that tends to 0. In particular, the half-fixed-parameter robust value computes the value when Min resists all perturbations in $[-\mathfrak{p}, \mathfrak{p}]$ and Max only infinitesimally perturbs.

- Finally, we study the value obtained when the parameter $\mathfrak{p}$ is not fixed for both players. In particular, we define the *limit robust value*, denoted $\overline{\mathsf{rVal}}$, by taking the limit of the fixed-parameter robust value when $\mathfrak{p}$ tends to 0. This studies the system when the perturbation is infinitesimal for both players. Chapter 8 formally introduces these values and establishes links between them.

Robust value problems A hope with the introduction of robust semantics is the decrease of the difficulty of solving deterministic value problems in WTGs. However, for general WTGs, it was proved that robust value problems (one by robust values defined) are undecidable. In particular, we note that the undecidability of the limit robust value problem is given by the equality between the limit robust value and the limit of the half-fixed-parameter robust value (see Proposition 8.13). Therefore, we want to investigate the decidability of the robust value problems for classes of WTGs on which the deterministic value problem is known as decidable (i.e. acyclic WTGs, divergent WTGs and one-clock WTGs). Table III.1

		conservative	excessive
fixed-parameter robust reachability problem		EXPTIME-complete [CHP11; ORS14]	EXPTIME-complete Theorem 9.25
limit robust reachability problem	timed automata	PSPACE-complete [San+13]	EXPTIME-complete [BMS15]
	timed games	EXPTIME-complete Theorem 9.27	EXPTIME-complete [BMS15]

Table III.2 – Solving robust reachability problems for Min.

summarises known results and our contributions to robust value problems for conservative and excessive semantics.

We start with a qualitative study, i.e. *robust reachability problems*, to decide when robust values are equal to $+\infty$. The *fixed-parameter robust reachability problem* is used to decide when the fixed-parameter robust value is equal to $+\infty$. Indeed, the fixed-parameter robust reachability problem tests when Min can reach a target location when the parameter $\mathfrak{p}$ is fixed. This problem was known EXPTIME-complete under the conservative semantics [CHP11; ORS14]. By encoding the excessive semantics into the exact semantics, we prove that the problem is also EXPTIME-complete under the excessive semantics. To decide when the limit robust value are equal to $+\infty$, we use the *limit robust reachability problem*. It decides if there exists a parameter $\mathfrak{p}$ such that Min wins the fixed-parameter robust reachability problem. The limit robust reachability problem was known EXPTIME-complete under the excessive semantics in timed games and timed automata [BMS15]. In our work, we extend the results under the conservative semantics by encoding the conservative semantics into the excessive one. In particular, we prove that the problem is also EXPTIME-complete under the conservative semantics. Moreover, we remark that both problems can be solved with the same complexity that reinforces the interest of the limit robust reachability problem. In Chapter 9, we present problems on reachability timed games under robust semantics and Table III.2 summarises the complexity of each problem.

To push the decidability frontier of robust value problems, we use two techniques. First, in Chapter 9, we present polynomial reductions between some robust value problems. Thus, we can obtain the decidability for some classes of WTGs of the robust value problems: for the one-clock WTGs and divergent WTGs under the conservative semantics. Finally, we conclude this part with Chapter 10, by adapting the value iteration algorithm (from [ABM04]) with *shrunk DBMs*[1] (from [SBM11])

[1]Shrunk DBM is a parametrised (with $\mathfrak{p}$) extension of the difference bound matrices (DBMs). A DBM is a data structure to model regions (and, more generally, zones) used to implement and solve reachability timed games.

to prove that the limit robust values can be computed in acyclic games and strictly non-Zeno WTGs (i.e. divergent WTGs with only non-negative weights).

Other notions of robustness in timed automata We used a notion of robustness to deal with the precision of clocks. In the literature (a summary is given in [Mar11]), other directions are also being studied to deal with other limitations of timed automata, like Zeno behaviours[2].

A natural way to deal with the precision of the clocks is a (finite) approximation of the values of clocks. *Discrete semantics*, introduced in [AH93], supposes that synchronous clocks are only integer (or multiple of a fixed granularity) values. In this case, some algorithmic questions are easier: undecidable problems (e.g. equivalence of timed automata) become decidable[3]. The real-time properties of this semantics depend on the chosen granularity, but knowing the existence of granularity such that real-time properties of the systems are preserved is a difficult question [HMP91; KAY10].

To deal with the immediacy of transitions and Zeno behaviours, we want to define a semantics that restricts the set of possible executions in the timed automata by selecting robust executions among the possible ones for classical timed automata, e.g. that cannot generate Zeno behaviours. For instance, such semantics may consider only accepting plays that still produce an accepting play when delays are perturbed. To formalise this intuition, two semantics have been proposed. First, the tube semantics (introduced in [GHJ97]) removes every play such that neighbour plays (according to a topology) are not dense. Intuitively, this semantics opens the guards of the timed automaton. An extension of this idea with probabilistic semantics was also proposed [Bai+07]. In particular, under probabilistic semantics, plays are accepted with a certain probability (given by the system). Thus, a measure over the set of plays is defined to appreciate the "quality" of the execution. In this case, the measure of probability defines the set of non-desired runs with a null measure.

Finally, to deal with the infinite precision of clocks, *enlarging semantics* (introduced in [WDR05]) induce some imprecision on the values of the clocks. Intuitively, these semantics desynchronise the clock's speed that evolves according to a piecewise affine function. These robust semantics may introduce some new (desirable) behaviour in the limit of guards, but also some bad behaviours. To limit that, enlarging semantics are considered only under a *robust safety checking* that checks, if all runs under a given robust semantics, satisfy a given property (that can describe the bad behaviour).

[2] An infinite play has a *Zeno behaviour* if the total time elapsing along the play is finite.
[3] Reducing complexity of the problems is another motivation for robustness.

8. Playing Robustly in Weighted Timed Games

To introduce robustness in WTGs, we use different robust semantics to model the behaviour of the system according to perturbations. In particular, we adapt useful notions from WTGs in the case of robust semantics, e.g. plays, path, strategies, etc. We aim to study the guarantee that Min has in WTGs under robust semantics: we thus extend the notion of deterministic value into robust values.

Table of contents

8.1. Robust semantics in weighted timed games 204
8.1.1. Plays and paths 206
8.1.2. Robust strategies 209
8.2. Values guaranteed by Min under robust semantics 211
8.2.1. Limits of robust values are equal 214
8.2.2. Robust value problems 215

Let $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt}\rangle$ be a WTG. We define a generic notion of robust semantics for every (reasonable) robust semantics that we can instantiate. This will allow us to recover conservative and excessive semantics given in the literature. In particular, the definition of a robust semantics $\mathsf{rob}_{\mathfrak{p}}$ with parameter $\mathfrak{p} > 0$ consists of the definition of two ingredients: the interval of possible perturbations for Max according to $\mathfrak{p}$, and the set of possible decisions from a configuration of Min. In the rest of this part, we call the semantics defined in Definition 1.25, the *exact semantics* of WTGs, and we denote it by exact.

8.1. Robust semantics in weighted timed games

From the modelling perspective, the perturbations model the small errors of physical systems on the real value of clocks. In particular, under the robust semantics of parameter $\mathfrak{p} > 0$, the perturbations of Max should remain close to 0. Moreover,

a robust semantics also filters the feasible decisions of Min by adding constraints on delays. For instance, conservative semantics only chooses feasible decisions such that they remain feasible after applying every possible perturbation chosen by Max. In particular, robust semantics does not add decisions from classical configurations.

Definition 8.1. A *robust semantics* $\mathsf{rob}_{\mathfrak{p}}$ of parameter $\mathfrak{p} > 0$ is a pair $(\Delta_{\mathfrak{p}}, \models^{\ell,\nu}_{\mathsf{rob}_{\mathfrak{p}}})$ where $\Delta_{\mathfrak{p}} \subseteq \mathbb{R}_{\geq 0}$ is an interval, and $\models^{\ell,\nu}_{\mathsf{rob}_{\mathfrak{p}}}$ is a relation defining *robust decisions* between configurations of Min and feasible decisions such that

1. $\Delta_{\mathfrak{p}}$ always contains 0 and has a length equal to $2\mathfrak{p}$;
2. by letting (δ, t) be a decision from (ℓ, ν), if $t \models^{\ell,\nu}_{\mathsf{rob}_{\mathfrak{p}}} \delta$, then $t \models^{\ell,\nu}_{\mathsf{exact}} \delta$.

Now, given a robust semantics, we define the corresponding infinite transition system associated with a WTG.

Definition 8.2. Let $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt}\rangle$ be a WTG. The *robust semantics* of $\mathcal{G}$ under $\mathsf{rob}_{\mathfrak{p}}$ is a transition system $[\![\mathcal{G}]\!]^{\mathsf{rob}} = \langle S, E, \mathsf{wt}\rangle$ defined such that

- $S = S_{\mathsf{Min}} \uplus S_{\mathsf{Max}} \uplus S_T$ is the set of *states* such that

  $$S_{\mathsf{Min}} \subseteq \mathsf{Conf}^{\mathcal{G}}_{\mathsf{Min}}\,, \quad S_T \subseteq \mathsf{Conf}^{\mathcal{G}}_{T} \quad \text{and} \quad S_{\mathsf{Max}} \subseteq \mathsf{Conf}^{\mathcal{G}}_{\mathsf{Max}} \cup (\mathsf{Conf}^{\mathcal{G}}_{\mathsf{Min}} \times \mathbb{R}_{\geq 0} \times \Delta)\,.$$

  Moreover, every state $(\ell, \nu, \delta, t) \in (\mathsf{Conf}^{\mathcal{G}}_{\mathsf{Min}} \times \mathbb{R}_{\geq 0} \times \Delta)$ is said *perturbed*;

- $E = E_{\mathsf{Min}} \uplus E_{\mathsf{Max}} \uplus E_{\mathsf{per}}$ is the set of edges such that

  $$E_{\mathsf{Max}} = \left\{ \left((\ell, \nu) \xrightarrow{\delta, t} (\ell', \nu') \right) \mid \ell \in L_{\mathsf{Max}}, t \models^{\ell,\nu}_{\mathsf{exact}} \delta \text{ and } \nu' = (\nu + t)[Y := 0] \right\}$$
  $$E_{\mathsf{Min}} = \left\{ \left((\ell, \nu) \xrightarrow{\delta, t} (\ell, \nu, \delta, t) \right) \mid \ell \in L_{\mathsf{Min}} \text{ and } t \models^{\ell,\nu}_{\mathsf{rob}_{\mathfrak{p}}} \delta \right\}$$
  $$E_{\mathsf{per}} = \left\{ \left((\ell, \nu, \delta, t) \xrightarrow{\delta, p} (\ell', \nu') \right) \mid p \in \Delta_{\mathfrak{p}} \text{ and } \nu' = (\nu + t + p)[Y := 0] \right\}$$

  where $\delta = (\ell, g, Y, \ell') \in \Delta$;

- $\mathsf{wt}\colon S \cup E \to \mathbb{Z}$ is a weight function such that for all states $s \in S$ with $s = (\ell, \nu)$ or $s = (\ell, \nu, \delta, t)$ and all edges $e \in E$, we fix

  $$\mathsf{wt}(s) = \mathsf{wt}(\ell) \quad \text{and} \quad \mathsf{wt}(e) = \begin{cases} \mathsf{wt}(\delta) & \text{if } e = (s \xrightarrow{\delta, t} s') \text{ with } s \in \mathsf{Conf}^{\mathcal{G}}; \\ 0 & \text{otherwise.} \end{cases}$$

This definition illustrates the abscence of symmetry between both players. Under a robust semantics, Min must only use robust decisions whereas Max uses

feasible decisions (like in exact semantics) as well as perturbations in $\Delta_{\mathfrak{p}}$ from the perturbed states. In the following, under a robust semantics, all feasible decisions of Max will be also called *robust*. Moreover, by letting $s \in S$ be a state of a robust semantics, we denote by $E(s)$ the set of *outgoing* edges from s in the robust semantics.

We also study robust semantics when the perturbation bound parameter varies (for instance, when this parameter bound tends to 0). In this case, we use the notion of a valid class of robust semantics.

Definition 8.3. A class of robust semantics $(\mathsf{rob}_{\mathfrak{p}})_{\mathfrak{p}}$ is *valid* if it satisfies the following constraints: for all $0 < \mathfrak{p}' < \mathfrak{p}$, $\Delta_{\mathfrak{p}'} \subset \Delta_{\mathfrak{p}}$; and for all decisions (δ, t) such that $t \models^{\ell,\nu}_{\mathsf{rob}_{\mathfrak{p}}} \delta$, we have $t \models^{\ell,\nu}_{\mathsf{rob}_{\mathfrak{p}'}} \delta$.

Conservative and excessive semantics In [BMS13; BMS15; Guh+15], authors focus on two classes of robust semantics: centred excessive semantics and centred conservative semantics. These semantics are defined by instantiating the parameters of the skeleton, i.e. the notion of robust decisions as well as the interval of the perturbations regarding $\mathfrak{p}$. First, a robust semantics is *centred* if $\Delta_{\mathfrak{p}} = [-\mathfrak{p}, \mathfrak{p}]$, or *translated* when $\Delta_{\mathfrak{p}} = [0, 2\,\mathfrak{p}]$.

A robust semantics is called *excessive*, denoted by $\mathsf{excess}_{\mathfrak{p}}$, when guards are tested before the perturbation. Moreover, to avoid going back in time (as in all robust semantics), Min must guarantee that the delay remains positive after all possible perturbations of Max, i.e. a feasible decision (δ, t) is robust if for all $p \in \Delta_{\mathfrak{p}}$, $t + p \geq 0$. Formally, $\mathsf{excess}_{\mathfrak{p}} = (\Delta_{\mathfrak{p}}, \models^{\ell,\nu}_{\mathsf{excess}_{\mathfrak{p}}})$ such that, for all configurations (ℓ, ν) and decisions (δ, t), we have

$$t \models^{\ell,\nu}_{\mathsf{excess}_{\mathfrak{p}}} \delta \Leftrightarrow t \geq -\min(\Delta_{\mathfrak{p}}) \wedge t \models^{\ell,\nu}_{\mathsf{exact}} \delta\,. \tag{8.1}$$

Now, a robust semantics is called *conservative*, denoted by cons, when guards are tested after the perturbation, i.e. a decision (δ, t) is robust if, for all perturbations $p \in \Delta_{\mathfrak{p}}$, the decision $(\delta, t + p)$ is feasible. Moreover, since $\Delta_{\mathfrak{p}}$ is an interval as the set of delays that satisfy a guard, testing the minimal and maximal values of $\Delta_{\mathfrak{p}}$ suffices. Formally, $\mathsf{cons}_{\mathfrak{p}} = (\Delta_{\mathfrak{p}}, \models^{\ell,\nu}_{\mathsf{cons}_{\mathfrak{p}}})$ such that, for all configurations (ℓ, ν) and all feasible decisions (δ, t), we have

$$t \models^{\ell,\nu}_{\mathsf{cons}_{\mathfrak{p}}} \delta \Leftrightarrow t \geq -\min(\Delta_{\mathfrak{p}}) \wedge t + \min(\Delta_{\mathfrak{p}}) \models^{\ell,\nu}_{\mathsf{exact}} \delta \wedge t + \max(\Delta_{\mathfrak{p}}) \models^{\ell,\nu}_{\mathsf{exact}} \delta\,. \tag{8.2}$$

8.1.1. Plays and paths

Let $\mathsf{rob}_{\mathfrak{p}}$ be a robust semantics of parameter $\mathfrak{p} > 0$ in a WTG $\mathcal{G}$. Like under the exact semantics, a *play* of $\mathcal{G}$ under $\mathsf{rob}_{\mathfrak{p}}$ is a sequence of edges in $[\![\mathcal{G}]\!]^{\mathsf{rob}_{\mathfrak{p}}} = \langle S, E, \mathsf{wt}\rangle$

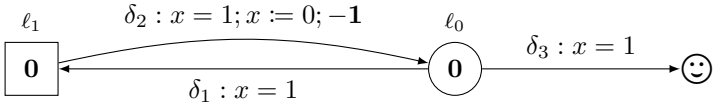


Figure 8.1 – An example of WTG where we apply different robust semantics.

starting in a configuration of $\mathcal{G}$. Let ρ be a play under $\mathsf{rob}_{\mathfrak{p}}$, we extend the function last such that $\mathsf{last}(\rho) \in S$, we define the length of ρ as the number of edges along the play, and we define the cumulative weight of ρ and its payoff in the expected way. Moreover, for modelling reasons, we only consider plays (starting and) ending in a configuration of $\mathcal{G}$. In particular, we say that a play ρ is *well-formed* when $\mathsf{last}(\rho) \in \mathsf{Conf}^{\mathcal{G}}$, otherwise ρ is said *ill-formed*.

Example 8.4. We consider the WTG depicted in Figure 8.1, in which the set of plays differs according to the (robust) semantics considered. For instance, a (well-formed) play under the exact semantics is

$$\rho_{\mathrm{det}} = (\ell_0, 0) \xrightarrow{\delta_1, 1} (\ell_1, 1) \xrightarrow{\delta_2, 0} (\ell_0, 0) \xrightarrow{\delta_1, 1} (\ell_1, 1) \xrightarrow{\delta_2, 0} (\ell_0, 0) \xrightarrow{\delta_3, 1} (\text{☺}, 1)$$

whose payoff is $\mathbf{P}(\rho_{\mathrm{det}}) = -2$.

In the centred excessive semantics of parameter $\mathfrak{p} > 0$, when Min chooses to fire δ_1, Max can enforce to reach the configuration $(\ell_1, 1 + \mathfrak{p})$ from which there are no feasible decisions. In particular, if Min chooses to fire δ_1, Max can enforce to reach a deadlock to avoid the target location yielding the payoff $+\infty$. To obtain a finite payoff, Min needs to choose δ_3, giving rise to the following play:

$$\rho_{\mathsf{excess}} = (\ell_0, 0) \xrightarrow{\delta_3, 1} (\ell_0, 0, \delta_3, 1) \xrightarrow{\delta_3, +\mathfrak{p}} (\text{☺}, 1 + \mathfrak{p})$$

whose payoff is $\mathbf{P}(\rho_{\mathsf{excess}}) = 0$, and length is $|\rho_{\mathsf{excess}}| = 2$. We observe that Max could choose another perturbation in $\Delta_{\mathfrak{p}}$.

Finally, in the centred conservative semantics, there exist no robust decision from $(\ell_0, 0)$ since all guards of transitions starting from ℓ_0 are singleton intervals. Thus, the unique play under the centred conservative semantics is $\rho_{\mathsf{cons}} = (\ell_0, 0)$ with $\mathbf{P}(\rho_{\mathsf{cons}}) = +\infty$. ⌟

As illustrated in Example 8.4, a play is always defined regarding a WTG and a semantics. Indeed, in the same WTG, plays can differ regarding the semantics considered. In particular, we denote by $\mathsf{FPlays}_{\mathcal{G}}^{\mathsf{rob}_{\mathfrak{p}}}$ (resp. $\mathsf{FPlays}_{\mathcal{G}}^{\mathsf{exact}}$) the set of plays of $\mathcal{G}$ under $\mathsf{rob}_{\mathfrak{p}}$ (resp. exact). When the WTG is clear from the context, we can omit the subscript $\mathcal{G}$. Moreover, we denote by $\mathsf{FPlays}_{\mathcal{G},\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}}}$ (resp. $\mathsf{FPlays}_{\mathcal{G},\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}}}$) the set of plays ending in a state of Min (resp. Max) in the semantics.

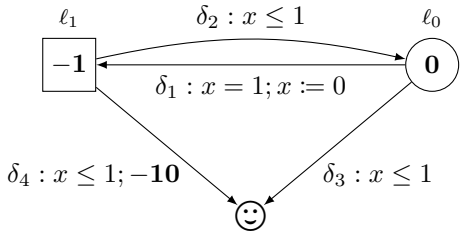


Figure 8.2 – A WTG in which the set of robust strategies for Min depends of the considered robust semantics.

Fortunately, we can define a link between robust and exact semantics. Under a conservative semantics, we note that from a robust decision (δ, t) and a perturbation p of Max, the generating decision $(\delta, t + p)$ remains feasible. Thus, we easily obtain the following property:

Lemma 8.5. *Let ρ be a well-formed play under $\mathsf{cons}_{\mathfrak{p}}$. There exists a play $\rho' \in \mathsf{FPlays}^{\mathsf{exact}}$ such that $\mathit{last}(\rho) = \mathit{last}(\rho')$ and $\mathsf{wt}_\Sigma(\rho) = \mathsf{wt}_\Sigma(\rho')$.*

Unfortunately, under excessive semantics, we can not guarantee that the decision generated from a robust decision and a perturbation of Max remains feasible:

Example 8.6. We consider the WTG, $\mathcal{G}$, depicted in Figure 8.1. The play $(\ell_0, 1)[\delta_3, 0]$ is a play in $\mathcal{G}$ under the exact semantics. However, there exist no plays from $(\ell_0, 1)$ under the centred excessive semantics of parameter $\mathfrak{p} > 0$ in $\mathcal{G}$. Indeed, since under such a semantics, Min must play a delay at least $\mathfrak{p}$ (to avoid negative perturbed delays), a play may exist only if the decision $(\delta_3, \mathfrak{p})$ is feasible which does not hold in the configuration $(\ell_0, 1)$. ⌟

Conversely, a play under the exact semantics is not a play under a robust semantics since some feasible decisions are not robust. Moreover, since robust decisions may not exist from a configuration where feasible decisions exist, we can not transform exact plays with a robust play by only modifying the delay with a possible perturbation (see Example 8.4).

Finally, a *path* in a WTG $\mathcal{G}$ under a robust semantics is a sequence of transitions of $\mathcal{G}$, i.e. it is the projection of a play on its sequence of locations (without time). Let ρ be a finite well-formed play in $\mathcal{G}$ under a robust semantics. We inductively define the path in $\mathcal{G}$ followed by ρ, denoted by $\pi(\rho)$ as follows:

$$\pi(\rho) = \begin{cases} \ell & \text{if } \rho = (\ell, \nu); \\ \ell \xrightarrow{\delta} \pi(\rho') & \text{if } \rho = (\ell, \nu) \xrightarrow{\delta,t} \rho'; \\ \pi(\rho') & \text{if } \rho = (\ell, \nu, \delta, t) \xrightarrow{\delta,t} \rho'. \end{cases} \tag{8.3}$$

Example 8.7. The play ρ_{excess} defined in Example 8.4 under the centred excessive semantics of parameter $\mathfrak{p}$ follows the path $\ell_0 \xrightarrow{\delta_3} \smiley$. ⌟

8.1.2. Robust strategies

We recall that, under the exact semantics, a strategy is a mapping from finite plays to a feasible decision at the last configuration of the finite play. Here, we extend this definition to robust semantics. In particular, a strategy for Min (resp. Max) in a WTG, under a robust semantics, is a mapping from finite plays (in the WTG under the robust semantics) ending in a configuration of Min (resp. state[1] of Max) to a robust decision for Min (resp. Max) from the last state of the play.

Unlike in the case of the exact semantics where we suppose that deadlocks are not admitted, in one of his configurations, Min may be unable to play a robust decision because none exist (even if there exists a feasible decision). In particular, we suppose that robust strategies for Min are only partial mappings[2].

Definition 8.8. Let $\mathcal{G}$ be a WTG under a robust semantics $\mathsf{rob}_{\mathfrak{p}}$ where $[\![\mathcal{G}]\!]^{\mathsf{rob}_{\mathfrak{p}}} = \langle S, E, \mathsf{wt}\rangle$. A *robust strategy* under $\mathsf{rob}_{\mathfrak{p}}$ is a mapping $\chi\colon \mathsf{FPlays}_{\mathcal{G}}^{\mathsf{rob}_{\mathfrak{p}}} \to E$ such that $\chi(\rho) \in E(\mathsf{last}(\rho))$. More precisely, a robust strategy for Min, denoted χ, is a *partial* mapping from $\mathsf{FPlays}_{\mathcal{G},\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}}}$, whereas a robust strategy for Max, denoted ζ, is a *total* mapping from $\mathsf{FPlays}_{\mathcal{G},\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}}}$. The set of strategies of Min (resp. Max) under $\mathsf{rob}_{\mathfrak{p}}$ is denoted by $\mathsf{rStrat}_{\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}}}(\mathcal{G})$ (resp. $\mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}}}(\mathcal{G})$).

We explain the asymmetric definition between the strategies of Min and Max by the objective of both players. Indeed, if the strategy of Min does not choose a robust decision, i.e. the strategy stops the play without reaching a target location, then Min loses and obtains $+\infty$ as a payoff which is the worst situation for Min. However, it is in the interest of Max to choose to play no (robust) decision even if there exist feasible ones, since in this case, Max wins by obtaining $+\infty$ as a payoff (that is the best situation for Max).

Example 8.9. We consider the WTG depicted in Figure 8.2 under $\mathsf{excess}_{\mathfrak{p}}$. For instance, a robust strategy for Min is the function χ such that for all finite play ρ ending in (ℓ_0, ν) with $\nu < 1$ maps $\chi(\rho) = (\delta_3, 1-\nu)$. However, function χ' defined for all finite play ρ ending in (ℓ_0, ν) with $\nu < 1$ by $\chi'(\rho) = (\delta_3, 0)$ is not a robust strategy for Min: the decision $(\delta_3, 0)$ is never robust under $\mathsf{excess}_{\mathfrak{p}}$ (since the perturbed delay must be non-negative). Moreover, in $(\ell_0, 1)$ no robust strategy for

[1] Under a robust semantics, the set of states of the semantics that belongs to Max, i.e. S_{Max}, are the configurations of Max *and* the perturbed states.

[2] In [BMS13], this deadlock problem in strategies does not exist, as the strategies for Min in concurrent WTGs can always use $\perp$ to indicate that the player lets Max plays.

Min can be defined since the unique feasible decision from $(\ell_0, 1)$ is $(\delta_3, 0)$ which is not robust. ⌟

Let $\mathsf{rob}_{\mathfrak{p}}$ be a robust semantics in a WTG $\mathcal{G}$ and (ℓ, ν) be a configuration of $\mathcal{G}$. We extend the notion of play conforming to robust strategies: we fix $\mathsf{Play}((\ell,\nu),\chi,\zeta)$ be the play of $\mathsf{FPlays}_{\mathcal{G}}^{\mathsf{rob}_{\mathfrak{p}}}$ conforming to $\chi \in \mathsf{rStrat}_{\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}}}$ and $\zeta \in \mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}}}$ two robust strategies. Moreover, we can give a link between sets of robust strategies in a class of robust semantics. For Min, we use the partiality of mappings to prove that the set of robust strategies increases when the parameter of the semantics decreases. For Max, this property does not hold (since robust strategies are total mappings) but we prove that a robust strategy can always be extended when the parameter of the semantics increases.

Lemma 8.10. *Let $\mathcal{G}$ be a WTG under a class of robust semantics $(\mathsf{rob}_{\mathfrak{p}})_{\mathfrak{p}}$. We consider two parameters $\mathfrak{p}, \mathfrak{p}'$ such that $\mathfrak{p} \geq \mathfrak{p}' > 0$, then we have*

1. $\mathsf{rStrat}_{\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}}} \subseteq \mathsf{rStrat}_{\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}'}}$;
2. *by letting $\chi \in \mathsf{rStrat}_{\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}}}$ be a robust strategy for Min, we have*

$$\sup_{\zeta \in \mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta)) \geq \sup_{\zeta \in \mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}'}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta)) \,.$$

Proof. 1. Let $\chi \in \mathsf{rStrat}_{\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}}}$ be a robust strategy of Min in $\mathcal{G}$, and $\rho \in \mathsf{FPlays}^{\mathsf{rob}_{\mathfrak{p}'}}$ be a finite play in $\mathcal{G}$ under $\mathsf{rob}_{\mathfrak{p}'}$ such that $\rho \in \mathsf{FPlays}^{\mathsf{rob}_{\mathfrak{p}}}$. Otherwise, $\rho \notin \mathsf{FPlays}^{\mathsf{rob}_{\mathfrak{p}}}$, and $\chi(\rho)$ is not defined. Now, by definition of the robust strategy, we have $\chi(\rho) = (\delta, t)$ with $t \models^{\mathsf{last}(\rho)}_{\mathsf{rob}_{\mathfrak{p}}} \delta$. We conclude that $t \models^{\mathsf{last}(\rho)}_{\mathsf{rob}_{\mathfrak{p}'}} \delta$ (by the definition of a class of robust semantics), and thus $\chi \in \mathsf{rStrat}_{\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}'}}$.

2. The proof consists in proving the following claim:

$$\forall \zeta' \in \mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}'}} \exists \zeta \in \mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}}}, \mathsf{Play}((\ell,\nu),\chi,\zeta) = \mathsf{Play}((\ell,\nu),\chi,\zeta') \qquad (\star)$$

Indeed, we can conclude since, for all $\zeta' \in \mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}'}}$, we have

$$\sup_{\zeta \in \mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta)) \geq \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta)) = \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta')) \,.$$

Now, by letting $\zeta' \in \mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}'}}$, we prove $(\star)$ by defining $\zeta \in \mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}}}$ as an

extension of ζ', i.e. for all finite play $\rho \in \mathsf{FPlays}^{\mathsf{rob}_\mathfrak{p}}_{\mathsf{Max}}$, we fix

$$\zeta(\rho) = \begin{cases} \zeta'(\rho) & \text{if } \rho \in \mathsf{FPlays}^{\mathsf{rob}_{\mathfrak{p}'}}_{\mathsf{Max}}; \\ e & \text{where } e \in E(\mathsf{last}(\rho)), \text{ otherwise;} \end{cases}$$

where E is the set of edges of $\mathcal{G}$ under $\mathsf{rob}_\mathfrak{p}$. In particular, we conclude by induction on the length of the play induced by χ and ζ. The property is trivial when $\rho = (\ell, \nu)$. Otherwise, we fix $\rho = \rho_1[\delta, t] \in \mathsf{FPlays}^{\mathsf{rob}_\mathfrak{p}}$ be the play conforming to χ and ζ of length i from (ℓ, ν), and $\rho' = \rho'_1[\delta', t'] \in \mathsf{FPlays}^{\mathsf{rob}_{\mathfrak{p}'}}$ be the play conforming to χ and ζ' of length i from (ℓ, ν) (we note that such a play exists since $\chi \in \mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}'}}_{\mathsf{Min}}$, by the previous property). Thus, by the hypothesis of induction, $\rho_1 = \rho'_1$, thus we need to prove that $(\delta, t) = (\delta', t')$. In particular, if $\mathsf{last}(\rho_1) \in \mathsf{Conf}_{\mathsf{Min}}$, then the property holds since we apply the same strategy in both cases. Otherwise, since $\rho_1 = \rho'_1 \in \mathsf{FPlays}^{\mathsf{rob}_{\mathfrak{p}'}}_{\mathsf{Max}}$, we conclude by definition of ζ. □

Finally, we can extend the notion of winning strategies under a reachability objective in the context of robust semantics. By letting $\mathcal{A}$ be an RTG under a robust semantics $\mathsf{rob}_\mathfrak{p}$ and $(\ell, \nu) \in \mathsf{Conf}^{\mathcal{A}}$, a robust strategy χ for Min is *winning* if for all robust strategies $\zeta \in \mathsf{rStrat}^{\mathcal{A}}_{\mathsf{Max}}(\mathsf{rob}_\mathfrak{p})$, $\mathsf{last}(\mathsf{Play}((\ell, \nu), \chi, \zeta)) \in S_T$ where $\mathsf{Play}((\ell, \nu), \chi, \zeta)$ is the unique (well-formed) play in $\mathsf{FPlays}^{\mathsf{rob}_\mathfrak{p}}$ for (ℓ, ν) and conforming to χ and ζ.

8.2. Values guaranteed by Min under robust semantics

We aim to study the impact of robust semantics in the weight that Min can guarantee whatever Max does: to do that we define some *robust values*. There exist different ways to deal with the parameter: when both players play under the robust semantics of the parameter $\mathfrak{p} > 0$, when Min plays under the robust semantics of parameter $\mathfrak{p}$, whereas Max plays under the robust semantics of parameter $\mathfrak{p}' \in (0, \mathfrak{p}]$ that tends to 0 and, finally, when the parameter $\mathfrak{p}$ is not fixed for both players. Each option induces a new value that deserves to be studied.

Fixed-parameter robust value Let $\mathcal{G}$ be a WTG under a robust semantics $\mathsf{rob}_\mathfrak{p}$ of parameter $\mathfrak{p} > 0$. We define its *fixed-parameter robust value* (according to $\mathsf{rob}_\mathfrak{p}$), denoted by $\overline{\mathsf{rVal}}^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}}_{\mathcal{G}}$ (we note that $\mathsf{rob}_\mathfrak{p}$ or $\mathcal{G}$ may be skipped when they are clear from the context), for all configurations (ℓ, ν) of $\mathcal{G}$ (and not for all states of the

semantics) by

$$\overline{\mathsf{rVal}}_{\mathcal{G}}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}(\ell,\nu) = \inf_{\chi\in\mathsf{rStrat}_{\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}}}(\mathcal{G})} \sup_{\zeta\in\mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}}}(\mathcal{G})} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta))\,.$$

Since a fixed-parameter robust semantics defines a *quantitative reachability game*[3], where configurations of Max also contain the robust states, we obtain that the fixed-parameter robust value is determined, by applying [Bri+22, Theorem 1].

Proposition 8.11. *Let $\mathcal{G}$ be a WTG under a robust semantics $\mathsf{rob}_{\mathfrak{p}}$ of parameter $\mathfrak{p} > 0$. Then, the fixed-parameter robust value is determined, i.e.*

$$\overline{\mathsf{rVal}}_{\mathcal{G}}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}(\ell,\nu) = \sup_{\zeta\in\mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}}}(\mathcal{G})} \inf_{\chi\in\mathsf{rStrat}_{\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}}}(\mathcal{G})} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta))\,.$$

In the following, we will denote it by $\mathsf{rVal}_{\mathcal{G}}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}$.

Moreover, we say that a robust strategy $\chi \in \mathsf{rStrat}_{\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}}}$ of Min is *ε-optimal* with respect to $\mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}$ if, for all configurations (ℓ,ν),

$$\sup_{\zeta\in\mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}}}(\mathcal{G})} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta)) \leq \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}(\ell,\nu) + \varepsilon\,.$$

Half-fixed-parameter robust value In [BMS13], another robust value is defined by considering a fixed perturbation bound for Min, and a limit (to 0) on the perturbation bound for Max. Formally, we define the *half-fixed-parameter robust value* of a class of robust semantics $(\mathsf{rob}_{\mathfrak{p}})_{\mathfrak{p}}$ under a parameter $\mathfrak{p} > 0$, for all configurations (ℓ,ν) of $\mathcal{G}$ by

$$\overline{{}^{1}\!/_{2}\mathsf{rVal}}_{\mathcal{G}}^{\mathfrak{p},\mathsf{rob}}(\ell,\nu) = \inf_{\chi\in\mathsf{rStrat}_{\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}}}(\mathcal{G})} \lim_{\substack{\mathfrak{p}'\to 0\\ 0<\mathfrak{p}'<\mathfrak{p}}} \sup_{\zeta\in\mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}'}}(\mathcal{G})} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta))\,.$$

Moreover, we say that a robust strategy $\chi \in \mathsf{rStrat}_{\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}}}$ of Min is *ε-optimal* with respect to $\overline{{}^{1}\!/_{2}\mathsf{rVal}}^{\mathfrak{p},\mathsf{rob}}$ if, for all configurations (ℓ,ν),

$$\lim_{\substack{\mathfrak{p}'\to 0\\ 0<\mathfrak{p}'<\mathfrak{p}}} \sup_{\zeta\in\mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}'}}(\mathcal{G})} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta)) \leq \overline{{}^{1}\!/_{2}\mathsf{rVal}}^{\mathfrak{p},\mathsf{rob}}(\ell,\nu) + \varepsilon\,.$$

[3] A quantitative reachability game introduced in [Bri+15] is an abstract model to formally define the semantics of quantitative (infinite) games.

Value under infinitesimal perturbations We consider the limit of previous values when the perturbation bound parameter tends to 0: given a class of robust semantics $(\mathsf{rob}_\mathfrak{p})_\mathfrak{p}$, we define the *limit robust value* $\overline{\mathsf{rVal}}^{\mathsf{rob}}_{\mathcal{G}}$ and the *limit of the half-fixed-parameter robust value* $\overline{1/2\mathsf{rVal}}^{\mathsf{rob}}_{\mathcal{G}}$ (already introduced in [BMS13]), for all configurations (ℓ,ν) of $\mathcal{G}$, by

$$\overline{\mathsf{rVal}}^{\mathsf{rob}}_{\mathcal{G}}(\ell,\nu) = \lim_{\substack{\mathfrak{p}\to 0\\ \mathfrak{p}>0}} \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}}_{\mathcal{G}}(\ell,\nu) \quad \text{and} \quad \overline{1/2\mathsf{rVal}}^{\mathsf{rob}}_{\mathcal{G}}(\ell,\nu) = \lim_{\substack{\mathfrak{p}\to 0\\ \mathfrak{p}>0}} \overline{1/2\mathsf{rVal}}^{\mathfrak{p},\mathsf{rob}}_{\mathcal{G}}(\ell,\nu)\,.$$

The limit robust value is quite natural to explore. However, this value is defined as a limit of functions (the fixed-parameter robust values). Thus, its (local) existence needs to be checked:

Lemma 8.12. *Let $\mathcal{G}$ be a WTG under class of robust semantics $(\mathsf{rob}_\mathfrak{p})_\mathfrak{p}$. Then, for all configurations (ℓ,ν) of $\mathcal{G}$, $\overline{\mathsf{rVal}}^{\mathsf{rob}}_{\mathcal{G}}(\ell,\nu)$ exists.*

Proof. Given a configuration (ℓ,ν), the proof focuses on the existence of a limit to 0 of the function $\mathfrak{p}\mapsto \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}}(\ell,\nu)$. More precisely, we prove that this function is non-increasing since a non-increasing function in $\mathbb{R}_\infty$ always admits a limit.

Let $\chi\in\mathsf{rStrat}^{\mathsf{rob}_\mathfrak{p}}_{\mathsf{Min}}$ be a strategy under the robust semantics $\mathsf{rob}_\mathfrak{p}$ (and $\mathsf{rob}_{\mathfrak{p}'}$, by Lemma 8.10-1) for Min. Now, since

$$\sup_{\zeta\in\mathsf{rStrat}^{\mathsf{rob}_\mathfrak{p}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta)) \geq \sup_{\zeta\in\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}'}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta))$$

(by Lemma 8.10-2), we conclude, by applying the infimum over strategies in $\mathsf{rStrat}^{\mathsf{rob}_\mathfrak{p}}_{\mathsf{Min}} \subseteq \mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}'}}_{\mathsf{Min}}$ (by Lemma 8.10-1) for Min:

$$\begin{aligned}\mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}}_{\mathcal{G}}(\ell,\nu) &= \inf_{\chi\in\mathsf{rStrat}^{\mathsf{rob}_\mathfrak{p}}_{\mathsf{Min}}} \sup_{\zeta\in\mathsf{rStrat}^{\mathsf{rob}_\mathfrak{p}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta))\\ &\geq \inf_{\chi\in\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}'}}_{\mathsf{Min}}} \sup_{\zeta\in\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}'}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta)) = \mathsf{rVal}^{\mathfrak{p}',\mathsf{rob}_{\mathfrak{p}'}}_{\mathcal{G}}(\ell,\nu)\,. \qquad \square\end{aligned}$$

Moreover, in Section 8.2.1, we study the link between all robust values according to a class of robust semantics and a permutation bound. In particular, by letting $\mathcal{G}$ be a WTG, for all classes of robust semantics $(\mathsf{rob}_\mathfrak{p})_\mathfrak{p}$ and parameter $\mathfrak{p}>0$, we have

$$\overline{1/2\mathsf{rVal}}^{\mathsf{rob}}_{\mathcal{G}} \underbrace{=}_{\text{Prop. 8.13}} \overline{\mathsf{rVal}}^{\mathsf{rob}}_{\mathcal{G}} \underbrace{\leq}_{\text{Lem. 8.14}} \overline{1/2\mathsf{rVal}}^{\mathfrak{p},\mathsf{rob}}_{\mathcal{G}} \underbrace{\leq}_{\text{Lem. 8.15}} \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}}_{\mathcal{G}}\,.$$

Finally, we say that a robust strategy $\chi\in\mathsf{rStrat}^{\mathsf{rob}_\mathfrak{p}}_{\mathsf{Min}}$ of Min is *ε-optimal* with

respect to $\overline{\mathsf{rVal}}^{\mathsf{rob}}$ if, for all configurations (ℓ, ν),

$$\sup_{\zeta \in \mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_\mathfrak{p}}(\mathcal{G})} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta)) \leq \overline{\mathsf{rVal}}^{\mathsf{rob}}(\ell,\nu) + \varepsilon\,.$$

8.2.1. Limits of fixed-parameter robust value and half-fixed-parameter robust value are equal

As introduced in the beginning of Section 8.2, many notions of robust values coexist: fixed-parameter robust value and half-fixed-parameter robust value. However, when we consider the limit of both classes of values, we prove that resulting values are equal:

Proposition 8.13. *Let $\mathcal{G}$ be a WTG under a class of robust semantics $(\mathsf{rob}_\mathfrak{p})_\mathfrak{p}$. Then, for all configurations (ℓ,ν) of $\mathcal{G}$, we have*

$$\overline{1/2\mathsf{rVal}}_{\mathcal{G}}^{\mathsf{rob}}(\ell,\nu) = \overline{\mathsf{rVal}}_{\mathcal{G}}^{\mathsf{rob}}(\ell,\nu)\,.$$

This section is devoted to the proof of this result. Let (ℓ,ν) be a configuration of $\mathcal{G}$ and $\mathfrak{p} > 0$ be a parameter, we prove the following inequalities:

$$\overline{\mathsf{rVal}}_{\mathcal{G}}^{\mathsf{rob}}(\ell,\nu) \leq \overline{1/2\mathsf{rVal}}_{\mathcal{G}}^{\mathfrak{p},\mathsf{rob}}(\ell,\nu) \leq \mathsf{rVal}_{\mathcal{G}}^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}}(\ell,\nu)\,.$$

Indeed, by applying the limit over $\mathfrak{p}'$ in these inequalities, we can conclude that

$$\overline{\mathsf{rVal}}_{\mathcal{G}}^{\mathsf{rob}}(\ell,\nu) \leq \overline{1/2\mathsf{rVal}}_{\mathcal{G}}^{\mathsf{rob}}(\ell,\nu) \leq \overline{\mathsf{rVal}}_{\mathcal{G}}^{\mathsf{rob}}(\ell,\nu)$$

i.e. $\overline{\mathsf{rVal}}_{\mathcal{G}}^{\mathsf{rob}}(\ell,\nu) = \overline{1/2\mathsf{rVal}}_{\mathcal{G}}^{\mathsf{rob}}(\ell,\nu)$. It remains to prove both inequalities to conclude the proof. We begin by the first inequality:

Lemma 8.14. *Let $\mathcal{G}$ be a WTG under a class of robust semantics $(\mathsf{rob}_\mathfrak{p})_\mathfrak{p}$ and $\mathfrak{p} > 0$ be a parameter. Then, for all configurations (ℓ,ν), we have*

$$\overline{\mathsf{rVal}}_{\mathcal{G}}^{\mathsf{rob}}(\ell,\nu) \leq \overline{1/2\mathsf{rVal}}_{\mathcal{G}}^{\mathfrak{p},\mathsf{rob}}(\ell,\nu)\,.$$

Proof. Let (ℓ,ν) be a configuration and $\mathfrak{p}' \in [0,\mathfrak{p}]$ be a parameter. We consider a robust strategy $\chi \in \mathsf{rStrat}_{\mathsf{Min}}^{\mathsf{rob}_\mathfrak{p}}$ for Min such that $\chi \in \mathsf{rStrat}_{\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}'}}$ (by Lemma 8.10-1). In particular, we deduce that

$$\mathsf{rVal}_{\mathcal{G}}^{\mathfrak{p}',\mathsf{rob}_{\mathfrak{p}'}}(\ell,\nu) \leq \sup_{\zeta \in \mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}'}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta))\,.$$

By applying the limit when $\mathfrak{p}$ tends to 0, we have

$$\overline{\mathsf{rVal}}^{\mathsf{rob}}_{\mathcal{G}}(\ell,\nu) = \lim_{\substack{\mathfrak{p}'\to 0\\ 0<\mathfrak{p}'<\mathfrak{p}}} \mathsf{rVal}^{\mathfrak{p}',\mathsf{rob}_{\mathfrak{p}'}}_{\mathcal{G}}(\ell,\nu) \leq \lim_{\substack{\mathfrak{p}'\to 0\\ 0<\mathfrak{p}'<\mathfrak{p}}} \sup_{\zeta\in\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}'}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta))$$

Finally, since $\chi \in \mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Min}}$ is arbitrary, we conclude that

$$\overline{\mathsf{rVal}}^{\mathsf{rob}}_{\mathcal{G}}(\ell,\nu) \leq \inf_{\chi\in\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Min}}} \lim_{\substack{\mathfrak{p}'\to 0\\ 0<\mathfrak{p}'<\mathfrak{p}}} \sup_{\zeta\in\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}'}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta)) = \overline{1/2\mathsf{rVal}}^{\mathfrak{p},\mathsf{rob}}_{\mathcal{G}}(\ell,\nu)\,.$$

□

Now, we prove the second inequality:

Lemma 8.15. *Let $\mathcal{G}$ be a WTG under a class of robust semantics $(\mathsf{rob}_{\mathfrak{p}})_{\mathfrak{p}}$ and $\mathfrak{p} > 0$ be a parameter. Then, for all configurations (ℓ,ν), we have*

$$\overline{1/2\mathsf{rVal}}^{\mathfrak{p},\mathsf{rob}}_{\mathcal{G}}(\ell,\nu) \leq \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}_{\mathcal{G}}(\ell,\nu)\,.$$

Proof. Let (ℓ,ν) be a configuration and $\mathfrak{p}' \in [0,\mathfrak{p}]$ be a parameter. We consider a robust strategy $\chi \in \mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Min}} \subseteq \mathsf{rStrat}^{\mathsf{rob}'_{\mathfrak{p}}}_{\mathsf{Min}}$ of Min (by Lemma 8.10-1). Moreover, we know that $\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}'}}_{\mathsf{Max}} \subseteq \mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Max}}$ (by Lemma 8.10-2), thus we deduce that

$$\sup_{\zeta\in\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}'}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta)) \leq \sup_{\zeta\in\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta))\,.$$

In particular, by applying the limit on $\mathfrak{p}'$ in the inequality, we deduce that

$$\lim_{\substack{\mathfrak{p}'\to 0\\ 0<\mathfrak{p}'<\mathfrak{p}}} \sup_{\zeta\in\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}'}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta)) \leq \sup_{\zeta\in\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta))\,.$$

Finally, since this inequality holds for all strategies χ of Min in $\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Min}}$, we conclude that

$$\begin{aligned}\overline{1/2\mathsf{rVal}}^{\mathfrak{p},\mathsf{rob}}_{\mathcal{G}}(\ell,\nu) &= \inf_{\chi\in\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Min}}} \lim_{\substack{\mathfrak{p}'\to 0\\ 0<\mathfrak{p}'<\mathfrak{p}}} \sup_{\zeta\in\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}'}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta))\\ &\leq \inf_{\chi\in\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Min}}} \sup_{\zeta\in\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta)) = \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}_{\mathcal{G}}(\ell,\nu)\,.\end{aligned}$$

□

8.2.2. Robust value problems

We extend the deterministic value problem with the robust values defined in Section 8.2: each robust value that we consider induces different robust value problems

illustrating the different ways to use the perturbation bound. In this section, we present each sort of problem and summarise the knowledge about them (a summary is given in Table III.1).

Fixed-parameter robust value problems We begin with the robust value problem defined by the fixed-parameter robust value, i.e. the robust value problem under fixed robust semantics.

Definition 8.16. Given a WTG $\mathcal{G}$ under a robust semantics $\mathsf{rob}_{\mathfrak{p}}$ of parameter $\mathfrak{p} \in \mathbb{Q}$, an initial configuration (ℓ, ν), and a threshold $\lambda \in \mathbb{Q} \cup \{-\infty, +\infty\}$, the *fixed-parameter robust value problem* consists in deciding if $\mathsf{rVal}_{\mathcal{G}}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}(\ell, \nu) \leq \lambda$.

Unfortunately, like under the exact semantics, this problem is undecidable for WTGs under excessive and conservative semantics. Indeed, in [BMS13, Theorem 4], the undecidability is shown when $\mathfrak{p} \leq 1/3$, for WTGs with non-negative weights and ten clocks by a reduction to the undecidable problem of the halting problem in two-counter machines.

To get around the undecidability, we will consider some restrictions over WTGs. In particular, we consider classes of WTGs where the deterministic value problem is known as decidable: we want to check if the decidability remains for these classes of WTGs. In [Guh+15, Theorem 8], authors focus on one-clock WTGs (with urgent location) and they prove that $\mathsf{rVal}^{\mathfrak{p},\mathsf{excess}_{\mathfrak{p}}}$ is computable for a given excessive semantics by exhibiting in the timed automata, the robust states given by the excessive semantics without adding a clock. In Section 9.4.2, we extend this result to the conservative semantics by using a reduction between conservative and excessive semantics that only adds urgent locations.

Another useful restriction is given by divergent WTGs. By reducing robust semantics into exact semantics, we can solve the obtained WTG under the exact semantics when it is divergent. However, we need to be careful since the divergence is not preserved when we change the semantics: a WTG under the exact semantics may contain cyclic region paths with a null weight (for all parameters) under an excessive semantics (as illustrated in Example 9.31). In particular, the divergence of the game is not tested over exact semantics but on the robust semantics that we consider. This proof is discussed in Section 9.1.

Half-fixed-parameter robust value problems The second robust value problem that we consider, introduced in [BMS13] under the name "limit strategy strict upper-bound problem", is defined with the half-fixed-parameter robust value. In particular, this problem studies the value when Min resists all perturbations in $\Delta_{\mathfrak{p}}$ (where $\mathfrak{p}$ is the fixed parameter of the robust semantics) and Max only infinitesimally perturbs.

Definition 8.17. Given a WTG $\mathcal{G}$ under a class of robust semantics $(\mathsf{rob}_\mathfrak{p})_\mathfrak{p}$, a parameter $\mathfrak{p} \in \mathbb{Q}$, an initial configuration (ℓ, ν), and a threshold $\lambda \in \mathbb{Q} \cup \{-\infty, +\infty\}$, the *half-fixed-parameter robust value problem* consists in deciding if $\overline{1/2\mathsf{rVal}}^{\mathfrak{p},\mathsf{rob}}_{\mathcal{G}}(\ell, \nu) \leq \lambda$.

Unfortunately, with an analogous reduction to the one used for the previous problem, this problem is undecidable for (general) WTGs under excessive and conservative semantics [BMS13, Theorem 3]. Moreover, in [BMS13, Theorem 1], authors also prove (with the same reduction) that the problem, when $\mathfrak{p} \leq 1/3$, is undecidable for WTA with non-negative weights and ten clocks under the excessive semantics. Regarding conservative semantics, the question remains open for WTA. However, we note that, since Min must play far enough of borders of guards, the *corner-point abstraction*[4] needs (at least) to be adapted.

Limit robust value problems The last robust value problem that we consider is defined with the limit robust value, i.e. when perturbations are supposed infinitesimal.

Definition 8.18. Given a WTG $\mathcal{G}$ under a class of robust semantics $(\mathsf{rob}_\mathfrak{p})_\mathfrak{p}$, an initial configuration (ℓ, ν), and a threshold $\lambda \in \mathbb{Q} \cup \{-\infty, +\infty\}$, the *limit robust value problem* consists in deciding if $\overline{\mathsf{rVal}}^{\mathsf{rob}}_{\mathcal{G}}(\ell, \nu) \leq \lambda$.

Since fixed-parameter robust value problems are known as undecidable when $\mathfrak{p} \leq 1/3$, the limit robust value problems remain undecidable for (general) WTGs under conservative and excessive semantics, and WTA under excessive semantics. However, in this case, the problem becomes PSPACE-complete for WTAs under conservative semantics [BMS13, Theorem 2]. In particular, in this case (unlike for the previous problem), authors remark that playing in a WTA under a conservative semantic is equivalent (for the limit robust value) to playing into its corner-point abstraction where all punctual corners are removed (that correspond to guards with at least one diagonal constraint). This is explained because the parameter can be chosen to guarantee that Min stays far enough from the border of the guards.

Finally, in Chapter 10, we give a symbolic algorithm to compute the limit robust value in acyclic WTGs under conservative and excessive semantics. Indeed, we adapt the value iteration algorithm from [ABM04] with shrunk cells based on shrunk DBM introduced in [SBM14].

[4] The corner-point abstraction, introduced in [Beh+01; Lar+01] to study the deterministic value in WTA with non-negative weights, is a refinement of regions that also store more precise information about where we are inside each region. Intuitively, each region is split according to a given granularity, and points refer to the behaviours "use to" corners of regions, i.e. coordinates that are a multiple of the granularity. Then, the corner-point abstraction is an equivalent (for the deterministic value) SPG such that each state stores the location, the region and the corner.

9. Reductions between Robust and Exact Semantics

Through the definition of robust semantics, natural links between the different semantics of WTGs emerge and can be exploited to answer certain open questions. Intuitively, playing in a WTG under a certain robust semantics is equivalent to playing in a WTG under the exact semantics where robust states are hard-coded in the WTG. In particular, the well-known results about the decidability of the deterministic value may be applied. However, this techniqueque adds one clock to the WTG. Our first contribution consists in proving the decidability of the fixed-parameter robust value problem in divergent WTGs under a conservative semantics. Our second contribution extends the decidability of the fixed-parameter robust value in one-clock WTGs under a conservative semantics with a reduction to the excessive semantics. Finally, as introducing robust value functions makes sense only if we can decide when Min can guarantee that a target location is reached (i.e. when the value will not be $+\infty$), our last contribution consists in determining the exact complexity of the robust reachability problems by using the previous reductions (see Table III.2).

Table of contents

9.1. A robust semantics simulated by the exact semantics 219
9.1.1. Translated robust semantics preserves robust values 220
9.1.2. Translated robust semantics encoded into the exact semantics 225
9.2. A conservative semantics encoded in an excessive semantics 231
9.3. Applications to robust reachability problems 237
9.3.1. Fixed-parameter robust reachability problem 238
9.3.2. Limit robust reachability problem 239
9.4. Application to robust value problems 240
9.4.1. The case of divergent weighted timed games 241
9.4.2. The case of one-clock weighted timed games 243

In this chapter, we have a double objective: closing open questions about robust reachability problems that decide robust value problems for the threshold $+\infty$ and proving the decidability of fixed-parameter robust value problems under a conservative semantics in one-clock WTGs with only non-negative weights or divergent WTGs. To do it we use two natural encodings: a robust semantics into the exact one by hard-coding robust states in the WTG, and a conservative into an excessive semantics by checking the delay chosen by Min after the perturbations. In particular, we establish the following statement:

Theorem 9.1.
- *The fixed-parameter robust reachability problem under an excessive semantics and the limit robust reachability problem under a conservative semantics are* EXPTIME*-complete.*
- *The fixed-parameter robust value problem under the conservative semantics is decidable for one-clock (with only non-negative weights) or divergent WTGs.*

9.1. A robust semantics simulated by the exact semantics

We start with the more natural encoding: a robust semantics into the exact one. In particular, we define a new WTG where robust states become a location. We note that the parameter of robustness is used for the new guards, therefore it is crucial to fix it. However, encoding robust states into a location is not always possible: perturbations may be negative but timed automata only use non-negative delays. Hence, to make this transformation, we need to translate the semantics to remove possible negative perturbations.

Theorem 9.2. *Let $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt}\rangle$ be a WTG under a robust semantics $\mathsf{rob}_{\mathfrak{p}}$ of parameter $\mathfrak{p} \in \mathbb{Q}_{>0}$ such that by letting δ be a transition, the set of delays $\{t \mid t \models^{\ell,\nu}_{\mathsf{rob}_{\mathfrak{p}}} \delta\}$ is an interval of $\mathbb{R}_{\geq 0}$. Then, there exists a WTG denoted by $\mathcal{G}^{\mathfrak{p}}$ such that for all configurations (ℓ, ν) of $\mathcal{G}$ we have*

$$\mathsf{dVal}_{\mathcal{G}^{\mathfrak{p}}}(\ell, \nu) = \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}_{\mathcal{G}}(\ell, \nu)\,.$$

We present the proof in two steps in the rest of this section. First, we prove that there exists a translated robust semantics that preserves the fixed-parameter robust value of $\mathcal{G}$. Then, we can define $\mathcal{G}^{\mathfrak{p}}$ from the previous translated robust semantics.

9.1.1. Translated robust semantics preserves the fixed-parameter robust value

To define an encoding a robust semantics into the exact one, we need to guarantee that perturbations of Max have only non-negative values. In particular, we use the translated robust semantics while preserving the original fixed-parameter robust value.

Proposition 9.3. *Let $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt}\rangle$ be a WTG under a robust semantics $\mathsf{rob}_{\mathfrak{p}}$ with the parameter $\mathfrak{p} > 0$. There exists a translated robust semantics $\mathsf{rob}'_{\mathfrak{p}}$ of parameter $\mathfrak{p}$ such that for all configurations (ℓ, ν), we have*

$$\mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}_{\mathcal{G}}(\ell,\nu) = \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}'_{\mathfrak{p}}}_{\mathcal{G}}(\ell,\nu)\,.$$

The proof consists in the definition of a translated robust semantics from $\mathsf{rob}_{\mathfrak{p}}$. Intuitively, we want that the robust decisions under $\mathsf{rob}'_{\mathfrak{p}}$ correspond to a robust decision under $\mathsf{rob}_{\mathfrak{p}}$ where the delay is shifted by the smallest possible perturbation. Formally, by letting $a \in [0, 2\,\mathfrak{p}]$ such that $\Delta_{\mathfrak{p}} = [-a, -a + 2p]$, we fix $\mathsf{rob}'_{\mathfrak{p}}$ be the translated robust semantics such that $\Delta'_{\mathfrak{p}} = [0, 2\,\mathfrak{p}]$, and for all feasible decisions (δ, t) from a configuration (ℓ, ν) of Min in $\mathcal{G}$, we have

$$t \models^{\ell,\nu}_{\mathsf{rob}'} \delta \qquad \text{if } (t+a) \models^{\ell,\nu}_{\mathsf{rob}} \delta\,. \tag{9.1}$$

Example 9.4. The translated excessive semantics of parameter $\mathfrak{p} > 0$ is a robust semantics such that (δ, t) is a robust decision from (ℓ, ν), if for all inequalities $a \le x \le b$ that define the guard of δ, $\nu(x) + t$ satisfies $a \le \nu(x) + t \le b - \mathfrak{p}$. ⌟

We can not guarantee that the new robust semantics is in the same class as $\mathsf{rob}_{\mathfrak{p}}$. In particular, from excessive semantics, the defined translated semantics is not excessive in the sense of the semantics given in (8.1). Indeed, transitions where the guard contains a constraint of form $x = 0$ admit a robust decision under the translated excessive semantics but not under an excessive semantics that is not translated. However, conservative semantics are preserved by this transformation since it only translates to the interval of a possible delays. In other words, all conservative semantics are equivalent (in the sense of their robust value) to the translated conservative semantics.

Given an initial configuration (ℓ, ν), we reason by double inequalities.

First inequality: $\mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}(\ell,\nu) \ge \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}'_{\mathfrak{p}}}(\ell,\nu)$. We exhibit a robust strategy under $\mathsf{rob}'_{\mathfrak{p}}$ as least as good as a (given) robust strategy under $\mathsf{rob}_{\mathfrak{p}}$. Formally, we consider a robust strategy $\chi \in \mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Min}}$ for Min under $\mathsf{rob}_{\mathfrak{p}}$, and we define a robust

strategy $\chi' \in \mathsf{rStrat}^{\mathsf{rob}'_{\mathfrak{p}}}_{\mathsf{Min}}$ for Min under $\mathsf{rob}'_{\mathfrak{p}}$ such that

$$\sup_{\zeta \in \mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta)) \geq \sup_{\zeta' \in \mathsf{rStrat}^{\mathsf{rob}'_{\mathfrak{p}}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi',\zeta'))\,.$$

The definition of χ' is induced by a function between plays under $\mathsf{rob}'_{\mathfrak{p}}$ to $\mathsf{rob}_{\mathfrak{p}}$ such that this function preserves the weight of plays. Intuitively, this function implements the property (9.1) about translated semantics. Formally, we define the function $\mathsf{proj}\colon \mathsf{FPlays}^{\mathsf{rob}'_{\mathfrak{p}}} \to \mathsf{FPlays}^{\mathsf{rob}_{\mathfrak{p}}}$ by induction on the length of plays such that for all finite plays $\rho \in \mathsf{FPlays}^{\mathsf{rob}'_{\mathfrak{p}}}$, we have

$$\mathsf{proj}(\rho) = \begin{cases} (\ell,\nu) & \text{if } \rho = (\ell,\nu); \\ \mathsf{proj}(\rho') \xrightarrow{\delta,t} (\ell,\nu) & \text{if } \rho = \rho' \xrightarrow{\delta,t} (\ell,\nu) \text{ and } \mathsf{last}(\rho') \in \mathsf{Conf}_{\mathsf{Max}}; \\ \mathsf{proj}(\rho') \xrightarrow{\delta,t+a} (\ell,\nu,\delta,t+a) & \text{if } \rho = \rho' \xrightarrow{\delta,t} (\ell,\nu,\delta,t); \\ \mathsf{proj}(\rho') \xrightarrow{\delta,p-a} (\ell,\nu) & \text{if } \rho = \rho' \xrightarrow{\delta,p} (\ell,\nu) \text{ and } \mathsf{last}(\rho') \notin \mathsf{Conf}. \end{cases}$$

Moreover, proj fulfils the following property:

Lemma 9.5. *For all finite plays $\rho \in \mathsf{FPlays}^{\mathsf{rob}'_{\mathfrak{p}}}$, we have:*

1. $\mathsf{proj}(\rho) \in \mathsf{FPlays}^{\mathsf{rob}_{\mathfrak{p}}}$ *and* $\mathit{last}(\rho) \in \mathsf{Conf}$ *if and only if* $\mathit{last}(\mathsf{proj}(\rho)) \in \mathsf{Conf}$. *Moreover, when* $\mathit{last}(\rho) \in \mathsf{Conf}$ *we have* $\mathit{last}(\mathsf{proj}(\rho)) = \mathit{last}(\rho)$*;*

2. *if ρ is well-formed, i.e.* $\mathit{last}(\rho) \in \mathsf{Conf}$*, then* $\mathsf{wt}_\Sigma(\rho) = \mathsf{wt}_\Sigma(\mathsf{proj}(\rho))$.

Proof. 1. By definition of proj.

2. We reason by induction on the length of a well-formed ρ, i.e. such that $\mathsf{last}(\rho) \in \mathsf{Conf}$. If $\rho = (\ell,\nu)$, then the property trivially holds. Otherwise, we let $\rho = \rho'[\delta,t]$. Thus, we conclude by distinguishing cases according to $\mathsf{last}(\rho')$. First, we remark that $\mathsf{last}(\rho') \notin \mathsf{Conf}_{\mathsf{Min}}$. Otherwise, that will contradict $\mathsf{last}(\rho) \in \mathsf{Conf}$.

   Now, we suppose that $\mathsf{last}(\rho') \in \mathsf{Conf}_{\mathsf{Max}}$, and we have $\mathsf{proj}(\rho) = \mathsf{proj}(\rho')[\delta,t]$. In particular, by hypothesis of induction applied on ρ' that is well-formed, we have $\mathsf{wt}_\Sigma(\mathsf{proj}(\rho')) = \mathsf{wt}(\rho')$. Moreover, since $\mathsf{last}(\rho') = \mathsf{last}(\mathsf{proj}(\rho'))$ (by property 1), we deduce that

$$\begin{aligned} \mathsf{wt}_\Sigma(\mathsf{proj}(\rho)) &= \mathsf{wt}_\Sigma(\mathsf{proj}(\rho')) + t\,\mathsf{wt}(\mathsf{last}(\mathsf{proj}(\rho'))) + \mathsf{wt}(\delta) \\ &= \mathsf{wt}_\Sigma(\rho') + t\,\mathsf{wt}(\mathsf{last}(\rho')) + \mathsf{wt}(\delta) = \mathsf{wt}_\Sigma(\rho)\,. \end{aligned}$$

Finally, we suppose that $\mathsf{last}(\rho') = (\ell, \nu, \delta, t') \notin \mathsf{Conf}$, and we have $\mathsf{proj}(\rho) = \mathsf{proj}(\rho')[\delta, t-a]$ such that

$$\mathsf{wt}_\Sigma(\mathsf{proj}(\rho)) = \mathsf{wt}_\Sigma(\mathsf{proj}(\rho')) + (t-a)\,\mathsf{wt}(\mathsf{last}(\mathsf{proj}(\rho'))) + 0\,. \tag{9.2}$$

Since $\mathsf{last}(\rho') \notin \mathsf{Conf}$, we can not apply the hypothesis of induction on ρ'. However, we can rewrite $\rho' = \rho'' \xrightarrow{\delta,t'} (\ell, \nu, \delta, t')$ where $\mathsf{last}(\rho'') = (\ell, \nu) \in \mathsf{Conf}_{\mathsf{Min}}$ (since ρ' begins by a configuration). In particular,

$$\mathsf{proj}(\rho') = \mathsf{proj}(\rho'') \xrightarrow{\delta,t'+a} (\ell, \nu, \delta, t'+a)\,,$$

and we obtain that

$$\begin{aligned}\mathsf{wt}_\Sigma(\mathsf{proj}(\rho')) &= \mathsf{wt}_\Sigma(\mathsf{proj}(\rho'')) + (t'+a)\,\mathsf{wt}(\mathsf{last}(\mathsf{proj}(\rho''))) + \mathsf{wt}(\delta)\\ &= \mathsf{wt}_\Sigma(\rho'') + (t'+a)\,\mathsf{wt}(\mathsf{last}(\rho'')) + \mathsf{wt}(\delta)\end{aligned}$$

since $\mathsf{wt}_\Sigma(\mathsf{proj}(\rho'')) = \mathsf{wt}(\rho'')$ (by hypothesis of induction applied on ρ'') and $\mathsf{last}(\rho'') = \mathsf{last}(\mathsf{proj}(\rho''))$ (by property 1). By combining the previous equation with (9.2), we conclude that

$$\begin{aligned}\mathsf{wt}_\Sigma(\mathsf{proj}(\rho)) &= \mathsf{wt}_\Sigma(\rho'') + (t'+a)\,\mathsf{wt}(\mathsf{last}(\rho'')) + (t-a)\,\mathsf{wt}(\mathsf{last}(\rho'')) + \mathsf{wt}(\delta)\\ &= \mathsf{wt}_\Sigma(\rho'') + (t'+t)\;\mathsf{wt}(\mathsf{last}(\rho')) + \mathsf{wt}(\delta) = \mathsf{wt}_\Sigma(\rho)\,.\end{aligned}$$ □

Now, we define $\chi' \in \mathsf{rStrat}^{\mathsf{rob}'_{\mathsf{p}}}_{\mathsf{Min}}$ such that for all $\rho \in \mathsf{FPlays}^{\mathsf{rob}'_{\mathsf{p}}}$

$$\chi'(\rho) = (\delta, t-a) \qquad \text{if } \chi\big(\mathsf{proj}(\rho)\big) = (\delta, t)\,.$$

By (9.1), we note that it is a robust decision for Min under $\mathsf{rob}'_{\mathsf{p}}$ since (δ, t) is a robust decision from $\mathsf{last}(\rho)$ under $\mathsf{rob}_{\mathsf{p}}$ (by Lemma 9.5-1). Moreover, since the definition of χ' relies on proj, it is no surprise that:

Lemma 9.6. *Let $\rho \in \mathsf{FPlays}^{\mathsf{rob}'_{\mathsf{p}}}$ be a play conforming to χ'. Then, $\mathsf{proj}(\rho)$ is conforming to χ.*

Proof. We reason by induction on the length of ρ. If $\rho = (\ell, \nu)$ then $\mathsf{proj}(\rho) = (\ell, \nu)$, and the property trivially holds. Otherwise, we can write $\rho = \rho'[\delta, t]$ such that $\mathsf{proj}(\rho')$ is conforming to χ by the hypothesis of induction. In particular, if $\mathsf{last}(\rho') \notin \mathsf{Conf}_{\mathsf{Min}}$ then $\mathsf{last}(\mathsf{proj}(\rho')) \notin \mathsf{Conf}_{\mathsf{Min}}$ (by Lemma 9.5-1) and $\mathsf{proj}(\rho)$ is conforming to χ.

Now, we suppose that $\mathsf{last}(\rho') \in \mathsf{Conf}_{\mathsf{Min}}$, and $\mathsf{proj}(\rho) = \mathsf{proj}(\rho')[\delta, t+a]$. Since ρ is conforming to χ', then $\chi'(\rho') = (\delta, t)$. We conclude by definition of χ', since $\chi(\mathsf{proj}(\rho')) = (\delta, t+a)$. □

We consider a well-formed play $\rho \in \mathsf{FPlays}^{\mathsf{rob}'_{\mathfrak{p}}}$ conforming to χ', and we prove that the play $\mathsf{proj}(\rho) \in \mathsf{FPlays}^{\mathsf{rob}'_{\mathfrak{p}}}$ conforming to χ (by Lemma 9.6) satisfies $\mathbf{P}(\rho) \leq \mathbf{P}(\mathsf{proj}(\rho))$. Indeed, if ρ reaches a target location, then $\mathsf{proj}(\rho)$ reaches the same target location by applying Lemma 9.5-1. In particular, by Lemma 9.5-2, we conclude that $\mathbf{P}(\rho) \leq \mathbf{P}(\mathsf{proj}(\rho))$. Otherwise, ρ does not reach a target location and $\mathbf{P}(\rho) = +\infty$. By using Lemma 9.5-2, we conclude that $\mathsf{proj}(\rho)$ does not reach a target location. Otherwise, ρ would reach the same last configuration (since it is well-formed), i.e. $\mathbf{P}(\mathsf{proj}(\rho)) = +\infty$.

Finally, we have shown that for all plays ρ conforming to χ', we can define a play $\mathsf{proj}(\rho)$ conforming to χ (by Lemma 9.6) such that $\mathbf{P}(\rho) \leq \mathbf{P}(\mathsf{proj}(\rho))$. In particular, we obtain that

$$\sup_{\zeta \in \mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta)) \geq \sup_{\zeta' \in \mathsf{rStrat}^{\mathsf{rob}'_{\mathfrak{p}}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi',\zeta')) \geq \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}'_{\mathfrak{p}}}(\ell,\nu)\,.$$

Since, this inequality holds for all χ, we obtain the claimed inequality.

Second inequality: $\mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}_{\mathcal{G}}(\ell,\nu) \leq \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}'_{\mathfrak{p}}}_{\mathcal{G}}(\ell,\nu)$. Conversely, we exhibit a strategy under $\mathsf{rob}_{\mathfrak{p}}$ as least as good as a (given) strategy under $\mathsf{rob}'_{\mathfrak{p}}$. Formally, we consider a strategy $\chi' \in \mathsf{rStrat}^{\mathsf{rob}'_{\mathfrak{p}}}_{\mathsf{Min}}$ for Min under $\mathsf{rob}'_{\mathfrak{p}}$, and we define a strategy $\chi \in \mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Min}}$ for Min under $\mathsf{rob}_{\mathfrak{p}}$ such that $\mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}},\chi}_{\mathcal{G}}(\ell,\nu) \leq \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}'_{\mathfrak{p}},\chi'}_{\mathcal{G}}(\ell,\nu)$. In particular, we use the function $\mathsf{inj}\colon \mathsf{FPlays}^{\mathsf{rob}_{\mathfrak{p}}} \to \mathsf{FPlays}^{\mathsf{rob}'_{\mathfrak{p}}}$ defined by induction on the length of plays such that for all finite plays $\rho \in \mathsf{FPlays}^{\mathsf{rob}_{\mathfrak{p}}}$, we have

$$\mathsf{inj}(\rho) = \begin{cases} (\ell,\nu) & \text{if } \rho = (\ell,\nu) \\ \mathsf{inj}(\rho') \xrightarrow{\delta,t} (\ell,\nu) & \text{if } \rho = \rho' \xrightarrow{\delta,t} (\ell,\nu) \text{ and } \mathsf{last}(\rho') \in \mathsf{Conf}_{\mathsf{Max}} \\ \mathsf{inj}(\rho') \xrightarrow{\delta,t-a} (\ell,\nu,\delta,t-a) & \text{if } \rho = \rho' \xrightarrow{\delta,t} (\ell,\nu,\delta,t) \\ \mathsf{inj}(\rho') \xrightarrow{\delta,p+a} s & \text{if } \rho = \rho' \xrightarrow{\delta,p} s \text{ and } \mathsf{last}(\rho') \notin \mathsf{Conf} \end{cases}$$

Moreover, inj satisfies the properties:

Lemma 9.7. *For all finite plays $\rho \in \mathsf{FPlays}^{\mathsf{rob}_{\mathfrak{p}}}$, we have:*

1. $\mathsf{inj}(\rho) \in \mathsf{FPlays}^{\mathsf{rob}'_{\mathfrak{p}}}$ *and* $\mathit{last}(\rho) \in \mathsf{Conf}$ *if and only if* $\mathit{last}(\mathsf{inj}(\rho)) \in \mathsf{Conf}$. *Moreover, when* $\mathit{last}(\rho) \in \mathsf{Conf}$ *we have* $\mathit{last}(\mathsf{inj}(\rho)) = \mathit{last}(\rho)$*;*
2. inj *is the inverse of* proj*;*
3. *if ρ is well-formed, i.e.* $\mathit{last}(\rho) \in \mathsf{Conf}$*, then* $\mathsf{wt}_\Sigma(\rho) = \mathsf{wt}_\Sigma(\mathsf{inj}(\rho))$.

Proof. 1. By definition of proj.

2. Let $\rho \in \mathsf{FPlays}^{\mathsf{rob}_\mathfrak{p}}$, we prove that $\mathsf{proj}(\mathsf{inj}(\rho)) = \rho$ by induction on the length of ρ. If $\rho = (\ell, \nu)$, then $\mathsf{proj}(\mathsf{inj}(\rho)) = (\ell, \nu)$. Otherwise, we rewrite by $\rho = \rho'[\delta, t]$ such that $\mathsf{proj}(\mathsf{inj}(\rho')) = \rho'$ by the hypothesis of induction. We conclude by distinguishing cases according to $\mathsf{last}(\rho')$. If $\mathsf{last}(\rho') \in \mathsf{Conf}_{\mathsf{Max}}$, then

$$\mathsf{proj}(\mathsf{inj}(\rho)) = \mathsf{proj}\big(\mathsf{inj}(\rho')[\delta, t]\big) = \mathsf{proj}(\mathsf{inj}(\rho'))[\delta, t] = \rho'[\delta, t]\,.$$

If $\mathsf{last}(\rho') \in \mathsf{Conf}_{\mathsf{Min}}$, then

$$\mathsf{proj}(\mathsf{inj}(\rho)) = \mathsf{proj}\big(\mathsf{inj}(\rho')[\delta, t-a]\big) = \mathsf{proj}(\mathsf{inj}(\rho'))[\delta, t-a+a] = \rho'[\delta, t]\,.$$

Finally, if $\mathsf{last}(\rho') \notin \mathsf{Conf}$, we have

$$\mathsf{proj}(\mathsf{inj}(\rho)) = \mathsf{proj}\big(\mathsf{inj}(\rho')[\delta, t+a]\big) = \mathsf{proj}(\mathsf{inj}(\rho'))[\delta, t+a-a] = \rho'[\delta, t]\,.$$

Conversely, via an analogous proof, we obtain that for all $\rho \in \mathsf{FPlays}^{\mathsf{rob}'_\mathfrak{p}}$ we have $\mathsf{inj}(\mathsf{proj}(\rho)) = \rho$ by induction on the length of ρ.

3. Let $\rho \in \mathsf{FPlays}^{\mathsf{rob}_\mathfrak{p}}$, we have $\mathsf{wt}_\Sigma(\mathsf{proj}(\mathsf{inj}(\rho))) = \mathsf{wt}_\Sigma(\mathsf{inj}(\rho))$ by Lemma 9.5-2. We conclude that $\mathsf{wt}_\Sigma(\rho) = \mathsf{wt}_\Sigma(\mathsf{inj}(\rho))$ by property 2. □

Now, we define $\chi \in \mathsf{rStrat}^{\mathsf{rob}_\mathfrak{p}}_{\mathsf{Min}}$ such that for all $\rho \in \mathsf{FPlays}^{\mathsf{rob}_\mathfrak{p}}$

$$\chi(\rho) = (\delta, t+a) \qquad \text{if } \chi'\big(\mathsf{inj}(\rho)\big) = (\delta, t)\,.$$

By (9.1), we note that it is a robust decision for Min under $\mathsf{rob}_\mathfrak{p}$ since (δ, t) is a robust decision from $\mathsf{last}(\rho)$ under $\mathsf{rob}'_\mathfrak{p}$ (by Lemma 9.7-1). Moreover, since the definition of χ' relies on inj, it is no surprise that:

Lemma 9.8. *Let $\rho \in \mathsf{FPlays}^{\mathsf{rob}_\mathfrak{p}}$ be a play conforming to χ. Then, $\mathsf{inj}(\rho)$ is conforming to χ'.*

Proof. We reason by induction on the length of ρ and as in Lemma 9.6, the interesting case is when $\rho = \rho'[\delta, t]$ where $\mathsf{last}(\rho') \in \mathsf{Conf}_{\mathsf{Min}}$ and $\mathsf{inj}(\rho')$ is conforming to χ' by the hypothesis of induction. In this case, $\mathsf{inj}(\rho) = \mathsf{inj}(\rho')[\delta, t-a]$. Since ρ is conforming to χ, then we have $\chi(\rho') = (\delta, t)$. We conclude by definition of χ, since $\chi'(\mathsf{inj}(\rho')) = (\delta, t-a)$. □

We consider a well-formed play $\rho \in \mathsf{FPlays}^{\mathsf{rob}_\mathfrak{p}}$ conforming to χ, and we prove that the play $\mathsf{inj}(\rho) \in \mathsf{FPlays}^{\mathsf{rob}'_\mathfrak{p}}$ conforming to χ' (by Lemma 9.8) satisfies $\mathbf{P}(\rho) \leq \mathbf{P}(\mathsf{inj}(\rho))$. Indeed, if ρ reaches a target location, then $\mathsf{inj}(\rho)$ reaches the same target location by applying Lemma 9.7-1. In particular, by Lemma 9.7-3, we conclude that $\mathbf{P}(\rho) \leq \mathbf{P}(\mathsf{inj}(\rho))$. Otherwise, ρ does not reach a target location and $\mathsf{inj}(\rho)$ too by Lemma 9.7-1. Thus, $\mathbf{P}(\rho) = +\infty = \mathbf{P}(\mathsf{proj}(\rho))$.

Finally, we have shown that for all plays ρ conforming to χ', we can define a play $\mathsf{inj}(\rho)$ conforming to χ (by Lemma 9.8) such that $\mathbf{P}(\rho) \leq \mathbf{P}(\mathsf{inj}(\rho))$. In particular, we conclude since the following inequality holds for all χ,

$$\mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}}(\ell,\nu) \leq \sup_{\zeta \in \mathsf{rStrat}^{\mathsf{rob}_\mathfrak{p}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta)) \leq \sup_{\zeta' \in \mathsf{rStrat}^{\mathsf{rob}'_\mathfrak{p}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi',\zeta'))\,.$$

9.1.2. Translated robust semantics encoded into the exact semantics

We define the reduction that allows us to compute the fixed-parameter robust value with the deterministic value. From Proposition 9.3, we give a relation between the translated robust semantics given by (9.1), and the exact one that preserves the fixed-parameter robust values. Thus, to conclude the proof of Theorem 9.2, it remains to prove the following proposition.

Proposition 9.9. *Let $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt}\rangle$ be a WTG under a translated robust semantics $\mathsf{rob}_\mathfrak{p}$ of parameter $\mathfrak{p} \in \mathbb{Q}_{>0}$ such that by letting δ be a transition, the set of delays $\{t \mid t \models^{\ell,\nu}_{\mathsf{rob}_\mathfrak{p}} \delta\}$ is an interval of $\mathbb{R}_{\geq 0}$. Then, there exists a WTG denoted by $\mathcal{G}^\mathfrak{p}$ such that for all configurations of $\mathcal{G}$, (ℓ,ν), we have*

$$\mathsf{dVal}_{\mathcal{G}^\mathfrak{p}}(\ell,\nu) = \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}}_{\mathcal{G}}(\ell,\nu)\,.$$

The rest of this section is dedicated to the proof of this proposition. We define $\mathcal{G}^\mathfrak{p}$ according to $\mathcal{G}$ and $\mathsf{rob}_\mathfrak{p}$ by exhibiting virtual states from $[\![\mathcal{G}]\!]^{\mathsf{rob}_\mathfrak{p}}$ as locations[1] into $\mathcal{G}^\mathfrak{p}$ and modifying guards of $\mathcal{G}$ to keep only robust decisions under $\mathsf{rob}_\mathfrak{p}$ in $\mathcal{G}^\mathfrak{p}$. Formally, we fix a function $\mathsf{pert}^{\mathsf{rob}}_\mathfrak{p} \colon \mathsf{Guards}_{\mathcal{G}}(\mathcal{X}) \to \mathsf{Guards}_{\mathcal{G}^\mathfrak{p}}(\mathcal{X})$ such that given a guard g of $\mathcal{G}$, the function[2] defines a new guard such that if a delay t satisfies $\mathsf{pert}^{\mathsf{rob}}_\mathfrak{p}(g)$, then t is robust under $\mathsf{rob}_\mathfrak{p}$ for all transitions with a guard g.

Example 9.10. When we focus on translated excessive semantics and translated conservative semantics defined by (9.1), we can give an explicit formula for the function pert. Let $g = I_1 \wedge \cdots \wedge I_n$ be a guard where each constraint is of the form $I_i = a_i \leq x \leq b_i$ (for $x \in \mathcal{X}$). Let $I_i = a_i \leq x \leq b_i$ one such constraint, we define $\mathsf{cons}(I_i) = a_i \leq x \leq b_i - 2\mathfrak{p}$, and $\mathsf{excess}(I_i) = a_i \leq x \leq b_i - \mathfrak{p}$ be the constraints on robust moves for the conservative and excessive semantics respectively. From this

[1] This idea was already used in [BMS13] to prove the result of undecidability for the excessive robust value problem on WTA under the excessive semantics.

[2] We note that, without hypothesis on the robust semantics, this function may not exist (robust delays may not be defined with constraints on clocks). However, for conservative and excessive semantics, this function exists since the set of delays into a robust decision for a fixed transition always is an interval (see (8.1) and (8.2)).

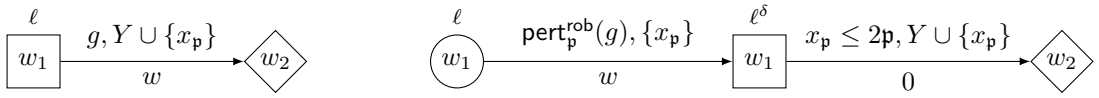


Figure 9.1 – Gadgets used to define $\mathcal{G}^{\mathfrak{p}}$ from $\mathcal{G}$ where $w, w_1, w_2 \in \mathbb{Z}$ denote the weights from $\mathcal{G}$. On the left, the transition obtained from (ℓ, g, Y, ℓ') with $\ell \in L_{\mathsf{Max}}$. On the right, the sequence of transitions obtained from (ℓ, g, Y, ℓ') with $\ell \in L_{\mathsf{Min}}$.

new constraints, we define both functions $\mathsf{pert}_{\mathfrak{p}}^{\mathsf{cons}} : \mathsf{Guards}_{\mathcal{G}}(\mathcal{X}) \to \mathsf{Guards}_{\mathcal{G}^{\mathfrak{p}}}(\mathcal{X})$ (resp. $\mathsf{pert}_{\mathfrak{p}}^{\mathsf{excess}}$ and $\mathsf{pert}_{\mathfrak{p}}^{\mathsf{excess}_t}$) for the conservative (resp. excessive) semantics by $\mathsf{pert}_{\mathfrak{p}}^{\mathsf{cons}}(g) = \mathsf{cons}(I_1) \wedge \cdots \wedge \mathsf{cons}(I_n)$ (resp. $\mathsf{pert}_{\mathfrak{p}}^{\mathsf{excess}}(g) = g$ and $\mathsf{pert}_{\mathfrak{p}}^{\mathsf{excess}_t}(g) = \mathsf{excess}(I_1) \wedge \cdots \wedge \mathsf{excess}(I_n)$). ⌟

Now, we can formally define the new WTG $\mathcal{G}^{\mathfrak{p}}$.

Definition 9.11. Let $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt}\rangle$ be a WTG under $\mathsf{rob}_{\mathfrak{p}}$. Its *interpretation under the exact semantics* is given by the WTG $\mathcal{G}^{\mathfrak{p}} = \langle L_{\mathsf{Min}}, L^{\mathfrak{p}}_{\mathsf{Max}}, L_T, L_u, \mathcal{X} \cup \{x_{\mathfrak{p}}\}, \Delta^{\mathfrak{p}}, \mathsf{wt}^{\mathfrak{p}}\rangle$ defined by

- $L^{\mathfrak{p}} = L_{\mathsf{Min}} \uplus L^{\mathfrak{p}}_{\mathsf{Max}} \uplus L_T$, the set of locations where

$$L^{\mathfrak{p}}_{\mathsf{Max}} = L_{\mathsf{Max}} \cup \{\ell^{\delta} \mid \ell \in L_{\mathsf{Min}} \text{ and } \delta = (\ell, g, Y, \ell') \in \Delta\}$$

- $\Delta^{\mathfrak{p}} = \Delta^{\mathfrak{p}}_{\mathsf{Min}} \uplus \Delta^{\mathfrak{p}}_{\mathsf{Max}} \uplus \Delta^{\mathfrak{p}}_{\mathsf{rob}}$, the set of transitions such that

$$\begin{aligned}
\Delta^{\mathfrak{p}}_{\mathsf{Max}} &= \{(\ell, g, Y \cup \{x_{\mathfrak{p}}\}, \ell') \mid (\ell, g, Y, \ell') \in \Delta \text{ with } \ell \in L_{\mathsf{Max}}\} \\
\Delta^{\mathfrak{p}}_{\mathsf{Min}} &= \left\{(\ell, \mathsf{pert}^{\mathsf{rob}}_{\mathfrak{p}}(g), \{x_{\mathfrak{p}}\}, \ell^{\delta}) \mid \delta = (\ell, g, Y, \ell') \in \Delta \text{ with } \ell \in L_{\mathsf{Min}}\right\} \\
\Delta^{\mathfrak{p}}_{\mathsf{rob}} &= \left\{(\ell^{\delta}, x_{\mathfrak{p}} \leq 2\mathfrak{p}, Y \cup \{x_{\mathfrak{p}}\}, \ell') \mid \delta = (\ell, g, Y, \ell') \in \Delta\right\}
\end{aligned}$$

- $\mathsf{wt}^{\mathfrak{p}} : L^{\mathfrak{p}} \cup \Delta^{\mathfrak{p}} \to \mathbb{Z}$, the weight function defined for all locations $\ell \in L^{\mathfrak{p}}$ by

$$\mathsf{wt}^{\mathfrak{p}}(\ell) = \begin{cases} \mathsf{wt}(\ell) & \text{if } \ell \in L \\ \mathsf{wt}(\ell') & \text{if } \ell = (\ell')^{\delta} \in L^{\mathfrak{p}}_{\mathsf{Max}} \setminus L_{\mathsf{Max}} \end{cases}$$

  and for all transitions $\delta = (\ell, g, Y, \ell') \in \Delta^{\mathfrak{p}}$ by

$$\mathsf{wt}^{\mathfrak{p}}(\delta) = \begin{cases} \mathsf{wt}(\delta) & \text{if } \ell \in L_{\mathsf{Max}} \\ \mathsf{wt}(\delta') & \text{if } \ell \in L_{\mathsf{Min}} \text{ and } \ell' = \ell^{\delta'} \\ 0 & \text{if } \ell \in L^{\mathfrak{p}}_{\mathsf{Max}} \setminus L_{\mathsf{Max}}. \end{cases}$$

The gadget of this transformation is given in Figure 9.1. We note that the fresh clock $x_{\mathfrak{p}}$ in $\mathcal{G}^{\mathfrak{p}}$ encodes the perturbations chosen by Max under the translated robust semantics. Moreover, for all transitions δ of $\mathcal{G}$, we denote by $\mathsf{pert}^{\mathsf{rob}}_{\mathfrak{p}}(\delta)$ the transition defined in $\mathcal{G}^{\mathfrak{p}}$ from δ. We denote by $\delta^{\mathfrak{p}}$ all transitions in $\Delta^{\mathfrak{p}}_{\mathsf{rob}}$, i.e. all transitions that correspond to a perturbation of Max. Finally, we note that this transformation can be computed in polynomial time.

Lemma 9.12. *Let $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt}\rangle$ be a WTG under a conservative or excessive semantics $\mathsf{rob}_{\mathfrak{p}}$ of parameter $\mathfrak{p} \in \mathbb{Q}_{>0}$. Then, the number of locqtions of $\mathcal{G}^{\mathfrak{p}}$ is polynomial in $|L|$, its number of clocks is equal to $|\mathcal{X}|+1$, and constants of guards are linear in the ones of $\mathcal{G}$ and $\mathfrak{p}$.*

We prove that $\mathcal{G}^{\mathfrak{p}}$ is correct since it is a WTG such that for all configurations $(\ell,\nu) \in \mathsf{Conf}^{\mathcal{G}}$, we have $\mathsf{dVal}_{\mathcal{G}^{\mathfrak{p}}}(\ell,\nu) = \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}_{\mathcal{G}}(\ell,\nu)$. In particular, we remark that a robust decision in $\mathcal{G}$ (under $\mathsf{rob}_{\mathfrak{p}}$) induces a feasible decision in $\mathcal{G}^{\mathfrak{p}}$ (and, reciprocally). In particular, we prove that we can adapt the strategies in a WTG into strategies in the second one. Now, given an initial configuration $(\ell,\nu) \in \mathsf{Conf}^{\mathcal{G}}$, we reason by double inequality to prove the equality of values.

First inequality: $\mathsf{dVal}_{\mathcal{G}^{\mathfrak{p}}}(\ell,\nu) \geq \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}_{\mathcal{G}}(\ell,\nu)$. To prove this inequality, we exhibit a strategy for Min under $\mathsf{rob}_{\mathfrak{p}}$ in $\mathcal{G}$ as least as good as a (given) strategy $\sigma \in \mathsf{dStrat}_{\mathsf{Min}}$ under the exact semantics in $\mathcal{G}^{\mathfrak{p}}$. We thus define a robust strategy $\chi \in \mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Min}}$ for Min under $\mathsf{rob}_{\mathfrak{p}}$ such that

$$\mathsf{dVal}^{\sigma}_{\mathcal{G}^{\mathfrak{p}}}(\ell,\nu) \geq \sup_{\zeta \in \mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Max}}(\mathcal{G})} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta))\,.$$

Again, to define this strategy, we define the (partial) function $\mathsf{proj} : \mathsf{FPlays}^{\mathsf{exact}}_{\mathcal{G}^{\mathfrak{p}}} \to \mathsf{FPlays}^{\mathsf{excess}_{\mathfrak{p}}}_{\mathcal{G}}$ by induction on the length of plays such that for all finite plays $\rho \in \mathsf{FPlays}^{\mathsf{exact}}_{\mathcal{G}^{\mathfrak{p}}}$ starting in a configuration of $\mathcal{G}$, we let $\mathsf{proj}(\rho)$ be

$$\begin{cases} (\ell, \nu_{|\mathcal{X}}) & \text{if } \rho = (\ell,\nu) \text{ with } \ell \in L; \\ \mathsf{proj}(\rho') \xrightarrow{\delta,t} (\ell, \nu_{|\mathcal{X}}) & \text{if } \rho = \rho' \xrightarrow{\mathsf{pert}^{\mathsf{rob}}_{\mathfrak{p}}(\delta),t} (\ell,\nu) \text{ and } \mathsf{last}(\rho') \in \mathsf{Conf}^{\mathcal{G}}_{\mathsf{Max}}; \\ \mathsf{proj}(\rho') \xrightarrow{\delta,t} (\ell, \nu_{|\mathcal{X}}) & \text{if } \rho = \rho' \xrightarrow{\delta^{\mathfrak{p}},t} (\ell,\nu) \text{ and } \mathsf{last}(\rho') = (\ell^{\delta}_1,\nu); \\ \mathsf{proj}(\rho') \xrightarrow{\delta,t} (\ell, \nu_{|\mathcal{X}} - t, \delta, t) & \text{if } \rho = \rho' \xrightarrow{\mathsf{pert}^{\mathsf{rob}}_{\mathfrak{p}}(\delta),t} (\ell^{\delta},\nu) \text{ and } \mathsf{last}(\rho') \in \mathsf{Conf}^{\mathcal{G}}_{\mathsf{Min}}; \end{cases}$$

where $\nu_{|\mathcal{X}}$ is the restriction of the valuation to the set of clocks of $\mathcal{G}$. Moreover, we remark that proj fulfils the following properties:

Lemma 9.13. *Let $\rho \in \mathsf{FPlays}^{\mathsf{exact}}_{\mathcal{G}^{\mathfrak{p}}}$ starting in a configuration of $\mathcal{G}$. Then, we have*

1. $\mathsf{proj}(\rho) \in \mathsf{FPlays}_{\mathcal{G}^\mathfrak{p}}^{\mathsf{excess}_\mathfrak{p}}$ *and* $\mathit{last}(\rho) \in \mathsf{Conf}^{\mathcal{G}}$ *if and only if* $\mathit{last}(\mathsf{proj}(\rho)) \in \mathsf{Conf}^{\mathcal{G}}$. *Moreover, if* $\mathit{last}(\rho) \in \mathsf{Conf}^{\mathcal{G}}$, *then* $\mathit{last}(\rho) = \mathit{last}(\mathsf{proj}(\rho))$;

2. $\mathsf{wt}_\Sigma(\rho) = \mathsf{wt}_\Sigma(\mathsf{proj}(\rho))$.

Proof. 1. By definition of proj.

2. We reason by induction on the length of ρ starting in a configuration of $\mathcal{G}$. If $\rho = (\ell, \nu)$, then the property trivially holds. Otherwise, ρ contains at least one edge.

   First, we suppose that $\rho = \rho'[\mathsf{pert}_\mathfrak{p}^{\mathsf{rob}}(\delta), t]$, i.e. $\mathsf{last}(\rho') \in \mathsf{Conf}^{\mathcal{G}}$. By definition of proj, then we have $\mathsf{proj}(\rho) = \mathsf{proj}(\rho')[\delta, t]$ (regardless of where $\mathsf{last}(\rho')$ belongs) and

   $$\mathsf{wt}_\Sigma(\mathsf{proj}(\rho)) = \mathsf{wt}_\Sigma(\mathsf{proj}(\rho')) + t\,\mathsf{wt}(\mathsf{last}(\mathsf{proj}(\rho'))) + \mathsf{wt}(\mathsf{pert}_\mathfrak{p}^{\mathsf{rob}}(\delta))\,.$$

   We conclude since $\mathsf{wt}_\Sigma(\mathsf{proj}(\rho')) = \mathsf{wt}_\Sigma(\rho')$ (by the induction hypothesis applied on ρ'), $\mathsf{wt}(\mathsf{pert}_\mathfrak{p}(\delta)) = \mathsf{wt}(\delta)$ (by Definition 9.11) and $\mathsf{wt}(\mathsf{last}(\rho')) = \mathsf{wt}(\mathsf{last}(\mathsf{proj}(\rho')))$ (by item 1):

   $$\mathsf{wt}_\Sigma(\mathsf{proj}(\rho)) = \mathsf{wt}_\Sigma(\rho') + t\,\mathsf{wt}(\mathsf{last}(\rho')) + \mathsf{wt}(\delta) = \mathsf{wt}_\Sigma(\rho)\,.$$

   Finally, we suppose that $\rho = \rho'[\delta^\mathfrak{p}, t]$, i.e. $\mathsf{last}(\rho') \notin \mathsf{Conf}^{\mathcal{G}}$. In particular, we have $\mathsf{proj}(\rho) = \mathsf{proj}(\rho')[\delta, t]$. Moreover, since $\mathsf{wt}_\Sigma(\mathsf{proj}(\rho')) = \mathsf{wt}_\Sigma(\rho')$ (by the induction hypothesis applied on ρ'), $\mathsf{wt}(\mathsf{pert}_\mathfrak{p}(\delta)) = \mathsf{wt}(\delta)$ (by Definition 9.11) and $\mathsf{wt}(\mathsf{last}(\rho')) = \mathsf{wt}(\mathsf{last}(\mathsf{proj}(\rho')))$ (by item 1), we conclude that $\mathsf{wt}_\Sigma(\mathsf{proj}(\rho)) = \mathsf{wt}(\rho)$. □

Now, we define $\sigma \in \mathsf{dStrat}_{\mathsf{Min}}(\mathcal{G}^\mathfrak{p})$ such that for all $\rho \in \mathsf{FPlays}_{\mathcal{G}^\mathfrak{p}}^{\mathsf{exact}}$, we have

$$\sigma(\rho) = (\mathsf{pert}_\mathfrak{p}(\delta), t) \qquad \text{if } \chi(\mathsf{proj}(\rho)) = (\delta, t).$$

We note that σ chooses a feasible decision since robust decisions in $\mathcal{G}$ are feasible in $\mathcal{G}^\mathfrak{p}$. Moreover, since the definition of σ relies on proj, it is no surprise that:

Lemma 9.14. *Let* $\rho \in \mathsf{FPlays}_{\mathcal{G}^\mathfrak{p}}^{\mathsf{exact}}$ *be a play conforming to* σ. *Then,* $\mathsf{proj}(\rho)$ *is conforming to* χ.

Proof. We reason by induction on the length of ρ. If $\rho = (\ell, \nu) \in \mathsf{Conf}^{\mathcal{G}}$, then $\mathsf{proj}(\rho) = (\ell, \nu_{|\mathcal{X}})$, and the property trivially holds. Otherwise, we can rewrite $\rho = \rho'[\delta, t]$ such that $\mathsf{proj}(\rho')$ is conforming to χ (by the hypothesis of induction). To conclude, we need to look after when $\mathsf{last}(\rho') \in \mathsf{Conf}_{\mathsf{Min}}^{\mathcal{G}^\mathfrak{p}}$. In this case, $\mathsf{proj}(\rho) = \mathsf{proj}(\rho')[(\delta', t)]$ where δ' is a transition such that $\delta = \mathsf{pert}_\mathfrak{p}(\delta')$. Moreover, since ρ is

conforming to σ, $\sigma(\rho') = (\delta, t)$, and by its definition, we have $\chi(\mathsf{proj}(\rho')) = (\delta', t)$ (as $\mathsf{last}(\rho') \in \mathsf{Conf}^{\mathcal{G}_\mathfrak{p}}_{\mathsf{Min}}$ by Lemma 9.13-1). □

We consider a play $\rho \in \mathsf{FPlays}_{\mathcal{G}_\mathfrak{p}}$ conforming to σ, and we prove that the play $\mathsf{proj}(\rho) \in \mathsf{FPlays}^{\mathsf{rob}_\mathfrak{p}}_{\mathcal{G}}$ is conforming to χ (by Lemma 9.14) and satisfies $\mathbf{P}(\rho) \leq \mathbf{P}(\mathsf{proj}(\rho))$. Indeed, if ρ reaches a target location, then $\mathsf{proj}(\rho)$ reaches the same target location by applying Lemma 9.13-1. In particular, by Lemma 9.13-2, we conclude that $\mathbf{P}(\rho) \leq \mathbf{P}(\mathsf{proj}(\rho))$. Otherwise, ρ does not reach a target location and $\mathsf{proj}(\rho)$ does not reach a target location too (Lemma 9.13-1). In this case, we conclude that $\mathbf{P}(\rho) = +\infty = \mathbf{P}(\mathsf{proj}(\rho))$.

Finally, we have shown that for all plays ρ conforming to σ, we can define a play $\mathsf{proj}(\rho)$ conforming to χ (by Lemma 9.14) such that $\mathbf{P}(\rho) \leq \mathbf{P}(\mathsf{proj}(\rho))$. In particular, we obtain that

$$\mathsf{dVal}^{\sigma}(\ell, \nu) \geq \sup_{\zeta \in \mathsf{rStrat}^{\mathsf{rob}_\mathfrak{p}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell, \nu), \chi, \zeta)) \geq \mathsf{rVal}^{\mathfrak{p}, \mathsf{rob}_\mathfrak{p}}_{\mathcal{G}}(\ell, \nu)\,.$$

Since, this inequality holds for all deterministic strategies, we obtain the claimed inequality.

Second inequality: $\mathsf{dVal}_{\mathcal{G}_\mathfrak{p}}(\ell, \nu) \leq \mathsf{rVal}^{\mathfrak{p}, \mathsf{rob}_\mathfrak{p}}_{\mathcal{G}}(\ell, \nu)$**.** Conversely, we exhibit a strategy under the exact semantics as least as good as a (given) strategy $\chi \in \mathsf{rStrat}^{\mathsf{rob}_\mathfrak{p}}_{\mathsf{Min}}$ under $\mathsf{rob}_\mathfrak{p}$. We thus define a deterministic strategy $\sigma \in \mathsf{dStrat}_{\mathsf{Min}}$ for Min such that

$$\mathsf{dVal}^{\sigma}_{\mathcal{G}_\mathfrak{p}}(\ell, \nu) \leq \sup_{\zeta \in \mathsf{rStrat}^{\mathsf{rob}_\mathfrak{p}}_{\mathsf{Max}}(\mathcal{G})} \mathbf{P}(\mathsf{Play}((\ell, \nu), \chi, \zeta))\,.$$

As for the first implication, we define the function $\mathsf{inj} : \mathsf{FPlays}^{\mathsf{rob}_\mathfrak{p}}_{\mathcal{G}} \to \mathsf{FPlays}^{\mathsf{exact}}_{\mathcal{G}_\mathfrak{p}}$ by induction on the length of plays such that for all finite plays $\rho \in \mathsf{FPlays}^{\mathsf{rob}_\mathfrak{p}}_{\mathcal{G}}$, we let $\mathsf{inj}(\rho)$ be

$$\begin{cases} (\ell, \nu_\mathfrak{p}) & \text{if } \rho = (\ell, \nu); \\ \mathsf{inj}(\rho') \xrightarrow{\mathsf{pert}^{\mathsf{rob}}_\mathfrak{p}(\delta), t} (\ell, \nu_\mathfrak{p}) & \text{if } \rho = \rho' \xrightarrow{\delta, t} (\ell, \nu) \text{ and } \mathsf{last}(\rho') \in \mathsf{Conf}^{\mathcal{G}}_{\mathsf{Max}}; \\ \mathsf{inj}(\rho') \xrightarrow{\delta_\mathfrak{p}, t} (\ell, \nu_\mathfrak{p}) & \text{if } \rho = \rho' \xrightarrow{\delta, t} (\ell, \nu) \text{ and } \mathsf{last}(\rho') \notin \mathsf{Conf}^{\mathcal{G}}; \\ \mathsf{inj}(\rho') \xrightarrow{\mathsf{pert}^{\mathsf{rob}}_\mathfrak{p}(\delta), t} (\ell^{\delta}, (\nu + t)_\mathfrak{p}) & \text{if } \rho = \rho' \xrightarrow{\delta, t} (\ell, \nu, \delta, t); \end{cases}$$

where for all valuations ν of clocks in $\mathcal{X}$ we note $\nu_{\mathfrak{p}}$ a valuation of clocks[3] in $\mathcal{G}^{\mathfrak{p}}$ such that for all clocks $x \in \mathcal{X}^{\mathfrak{p}}$, $\nu_{\mathfrak{p}}(x) = \nu(x)$ if $x \neq x_{\mathfrak{p}}$, and $\nu_{\mathfrak{p}}(x_{\mathfrak{p}}) = 0$. The function inj fulfils the following properties:

Lemma 9.15. *Let $\rho \in \mathsf{FPlays}_{\mathcal{G}}^{\mathsf{rob}_{\mathfrak{p}}}$ be a play under $\mathsf{rob}_{\mathfrak{p}}$. Then, we have*

1. *$\mathsf{inj}(\rho) \in \mathsf{FPlays}_{\mathcal{G}^{\mathfrak{p}}}^{\mathsf{exact}}$ and $\mathit{last}(\rho) \in \mathsf{Conf}^{\mathcal{G}}$ if and only if $\mathit{last}(\mathsf{inj}(\rho)) \in \mathsf{Conf}^{\mathcal{G}}$. Moreover, if $\mathit{last}(\rho) \in \mathsf{Conf}^{\mathcal{G}}$, then $\mathit{last}(\rho) = \mathit{last}(\mathsf{inj}(\rho))$;*
2. *$\mathsf{wt}_{\Sigma}(\rho) = \mathsf{wt}_{\Sigma}(\mathsf{inj}(\rho))$.*

Proof. 1. By definition of inj.

2. We reason by induction on the length of ρ. If $\rho = (\ell, \nu)$ then the property is trivial. Otherwise, we let $\rho = \rho'[\delta, t]$ and we distinguish two cases according to $\mathsf{last}(\rho')$. If $\mathsf{last}(\rho') \in \mathsf{Conf}^{\mathcal{G}}_{\mathsf{Min}}$, then we have $\mathsf{inj}(\rho) = \mathsf{inj}(\rho')[\mathsf{pert}^{\mathsf{rob}}_{\mathfrak{p}}(\delta), t]$. Moreover, we also know that $\mathsf{wt}_{\Sigma}(\mathsf{inj}(\rho')) = \mathsf{wt}_{\Sigma}(\rho')$ (by the induction hypothesis applied on ρ'), $\mathsf{wt}(\mathsf{pert}_{\mathfrak{p}}(\delta)) = \mathsf{wt}(\delta)$ (by Definition 9.11) and $\mathsf{wt}(\mathsf{last}(\rho')) = \mathsf{wt}(\mathsf{last}(\mathsf{inj}(\rho')))$ (by item 1). In particular, we deduce that

$$\begin{aligned}\mathsf{wt}_{\Sigma}(\mathsf{proj}(\rho)) &= \mathsf{wt}_{\Sigma}(\mathsf{inj}(\rho')) + t\,\mathsf{wt}(\mathsf{last}(\mathsf{inj}(\rho'))) + \mathsf{wt}(\mathsf{pert}^{\mathsf{rob}}_{\mathfrak{p}}(\delta)) \\ &= \mathsf{wt}_{\Sigma}(\rho') + t\,\mathsf{wt}(\mathsf{last}(\rho')) + \mathsf{wt}(\delta) = \mathsf{wt}_{\Sigma}(\rho)\,.\end{aligned}$$

   Otherwise, $\mathsf{last}(\rho') \notin \mathsf{Conf}^{\mathcal{G}}_{\mathsf{Min}}$ and $\mathsf{inj}(\rho) = \mathsf{inj}(\rho')[\delta', t]$ where δ' is equal to $\mathsf{pert}^{\mathsf{rob}}_{\mathfrak{p}}(\delta)$ or $\delta^{\mathfrak{p}}$ according to $\mathsf{last}(\rho')$. Again, we have $\mathsf{wt}_{\Sigma}(\mathsf{inj}(\rho')) = \mathsf{wt}_{\Sigma}(\rho')$ (by applying the induction hypothesis on ρ'), $\mathsf{wt}(\delta') = \mathsf{wt}(\delta)$ (by Definition 9.11) and $\mathsf{wt}(\mathsf{last}(\rho')) = \mathsf{wt}(\mathsf{last}(\mathsf{inj}(\rho')))$ (by item 1, if $\mathsf{last}(\rho') \in \mathsf{Conf}^{\mathcal{G}}_{\mathsf{Max}}$, or by definition of rob otherwise). Thus, we conclude that $\mathsf{wt}_{\Sigma}(\rho) = \mathsf{wt}_{\Sigma}(\mathsf{inj}(\rho))$. □

Now, we define $\chi \in \mathsf{rStrat}^{\mathsf{excess}_{\mathfrak{p}}}_{\mathsf{Min}}(\mathcal{G})$ such that for all $\rho \in \mathsf{FPlays}^{\mathsf{excess}_{\mathfrak{p}}}_{\mathcal{G}}$, we have

$$\chi(\rho) = (\delta, t) \qquad \text{if } \sigma(\mathsf{inj}(\rho)) = (\mathsf{pert}_{\mathfrak{p}}(\delta), t)\,.$$

We note that χ chooses a robust decision since feasible decision in $\mathcal{G}^{\mathfrak{p}}$ are robust in $\mathcal{G}$. Moreover, since the definition of χ relies on inj, it is no surprise that:

Lemma 9.16. *Let $\rho \in \mathsf{FPlays}_{\mathcal{G}}^{\mathsf{rob}_{\mathfrak{p}}}$ be a play conforming to χ. Then, $\mathsf{inj}(\rho)$ is conforming to σ.*

[3] To easily define this new valuation, it is important that the clock $x_{\mathfrak{p}}$ was reset at each step, i.e. for all valuations ν to $\mathcal{X}^{\mathfrak{p}}$ we have $\nu(x_{\mathfrak{p}}) = 0$. Otherwise, we need to compute the time elapsed from the last reset (in a transition of Min). In this case, the new valuation does not only depend on the current valuation but also on the previous play.

Proof. We reason by induction on the length of ρ. If $\rho = (\ell, \nu) \in \mathsf{Conf}^{\mathcal{G}}$, then $\mathsf{inj}(\rho) = (\ell, \nu_{\mathfrak{p}})$ and the property trivially holds. Otherwise, we can rewrite $\rho = \rho'[\delta, t]$ such that $\mathsf{inj}(\rho')$ is conforming to σ. By Lemma 9.15-1, the interesting case is when $\mathsf{last}(\rho') \in \mathsf{Conf}^{\mathcal{G}}_{\mathsf{Min}}$, and $\mathsf{inj}(\rho) = \mathsf{inj}(\rho')[\mathsf{pert}_{\mathfrak{p}}(\delta), t]$. Moreover, since ρ is conforming to χ, $\chi(\rho') = (\delta, t)$, and by its definition, we conclude that $\sigma(\mathsf{proj}(\rho')) = (\mathsf{pert}_{\mathfrak{p}}(\delta), t)$. □

We consider a well-formed play $\rho \in \mathsf{FPlays}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathcal{G}}$ conforming to χ, and we prove that the play $\mathsf{proj}(\rho) \in \mathsf{FPlays}^{\mathsf{exact}}$ conforming to σ (by Lemma 9.16) satisfies $\mathbf{P}(\rho) \leq \mathbf{P}(\mathsf{inj}(\rho))$. Indeed, if ρ reaches a target location, then $\mathsf{inj}(\rho)$ reaches the same target location by applying Lemma 9.15-1. In particular, by Lemma 9.15-2, we conclude that $\mathbf{P}(\rho) \leq \mathbf{P}(\mathsf{inj}(\rho))$. Otherwise, ρ does not reach a target location and $\mathsf{proj}(\rho)$ does not reach a target location too (Lemma 9.13-1). In this case, we conclude that $\mathbf{P}(\rho) = +\infty = \mathbf{P}(\mathsf{proj}(\rho))$.

Finally, we have shown that for all plays ρ conforming to χ, we can define a play $\mathsf{inj}(\rho)$ conforming to σ (by Lemma 9.16) such that $\mathbf{P}(\rho) \leq \mathbf{P}(\mathsf{proj}(\rho))$. In particular, we conclude since the following inequality holds for all robust strategies,

$$\mathsf{dVal}(\ell, \nu) \leq \mathsf{dVal}^{\sigma}(\ell, \nu) \leq \sup_{\zeta \in \mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell, \nu), \chi, \zeta)) \,.$$

9.2. A conservative semantics encoded in an excessive semantics

Now, we provide the second encoding that is less natural than the previous one by testing delays chosen by Min after applying the perturbation of Max (and not before like in the conservative semantics). Moreover, unlike in the previous encoding, we do not exploit the parameter of the robust semantics, thus we do not need to fix the parameter.

Theorem 9.17. *Let $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt} \rangle$ be a WTG under the class of conservative semantics $(\mathsf{cons}_{\mathfrak{p}})_{\mathfrak{p}}$. Then, there exists a WTG $\mathcal{G}^{\mathsf{e}}$ under a class of excessive semantics $(\mathsf{excess}_{\mathfrak{p}})_{\mathfrak{p}}$ (with the same $\Delta_{\mathfrak{p}}$) such that for all parameter bounds $\mathfrak{p} > 0$ and all configurations (ℓ, ν), we have*

$$\mathsf{rVal}^{\mathfrak{p},\mathsf{excess}_{\mathfrak{p}}}_{\mathcal{G}^{\mathsf{e}}}(\ell, \nu) = \mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}_{\mathcal{G}}(\ell, \nu) \,.$$

The rest of this section is devoted to the proof of this result, which consists in building a WTG under an excessive semantics that preserves all fixed-parameter robust values from a given WTG under a conservative semantics. Intuitively, a conservative semantics imposes that the guards on transitions of Min are tested

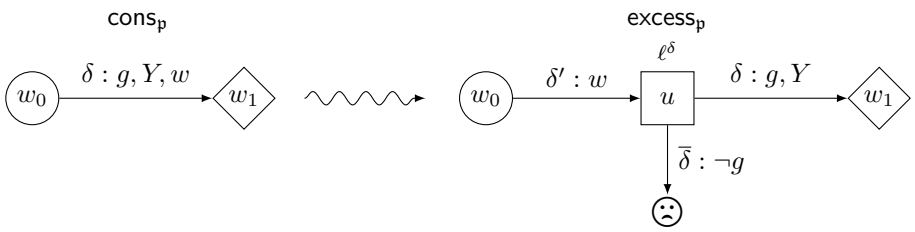


Figure 9.2 – Gadget used to define $\mathcal{G}$ from $\mathcal{G}^{\mathsf{e}}$ where the location that contains u (and not a weight) is an urgent location and $w, w_0, w_1 \in \mathbb{Z}$ denote the weights from $\mathcal{G}$.

after applying the perturbation. The reduction exploits this idea: all transitions of Min in $\mathcal{G}$ will be modified by adding a new urgent location of Max (to forbid modification of the delay) that will check the delay chosen by Min (after perturbation). If Min plays a decision that is not robust w.r.t. the conservative semantics (that is possible since we ask for a robust decision w.r.t. the excessive one), then Max wins by reaching ☹ (see Figure 9.2). Otherwise, Min chooses a robust decision w.r.t. the conservative semantics and Max must apply the original transition. Formally, we define $\mathcal{G}^{\mathsf{e}}$ as follows:

Definition 9.18. Let $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt}\rangle$ be a WTG and $\Delta_{\mathsf{Min}} = \{\delta \mid \delta = (\ell, g, Y, \ell') \in \Delta \text{ with } \ell \in L_{\mathsf{Min}}\}$. We let

$$\mathcal{G}^{\mathsf{e}} = \langle L_{\mathsf{Min}} \cup \{☹\}, L_{\mathsf{Max}} \cup \{\ell^\delta \mid \delta \in \Delta_{\mathsf{Min}}\}, L_T, L_u \cup \{\ell^\delta \mid \delta \in \Delta_{\mathsf{Min}}\}, \mathcal{X}, \Delta^{\mathsf{e}}, \mathsf{wt}^{\mathsf{e}}\rangle$$

be such that $\Delta^{\mathsf{e}} = \Delta^{\mathsf{e}}_{\mathsf{Min}} \uplus \Delta^{\mathsf{e}}_{\mathsf{Max}} \uplus \Delta^{\mathsf{e}}_{\mathsf{rob}_\mathfrak{p}} \uplus \{(☹, \text{true}, \emptyset, ☹)\}$ with

$$\begin{aligned}
\Delta^{\mathsf{e}}_{\mathsf{Max}} &= \{(\ell, g, Y, \ell') \mid (\ell, g, Y, \ell') \in \Delta \text{ with } \ell \in L_{\mathsf{Max}}\} \\
\Delta^{\mathsf{e}}_{\mathsf{Min}} &= \{\delta' = (\ell, \text{true}, \emptyset, \ell^\delta) \mid \delta = (\ell, g, Y, \ell') \in \Delta_{\mathsf{Min}}\} \\
\Delta^{\mathsf{e}}_{\mathsf{rob}_\mathfrak{p}} &= \{(\ell^\delta, g, Y, \ell') \mid \delta = (\ell, g, Y, \ell') \in \Delta_{\mathsf{Min}}\} \\
&\quad \cup \{(\ell^\delta, \neg g, \emptyset, ☹) \mid \delta = (\ell, g, Y, \ell') \in \Delta_{\mathsf{Min}}\}
\end{aligned}$$

where $\neg g$ is the complement[4] of g, and $\mathsf{wt}^{\mathsf{e}} : L^{\mathsf{e}} \cup \Delta^{\mathsf{e}} \to \mathbb{Z}$ be the weight function

[4]We recall that a guard is a conjunction of atomic constraints of the form $x \bowtie a$ with $\bowtie \in \{\leq, <, >, \geq\}$. Thus, the complement of a guard is defined by the disjunction of the negation of each atomic constraint that defines the guards. However, the complement of a guard is not a guard (since it is a conjunction of atomic constraints). To avoid this pitfall, the complement of a guard defines a set of (at most $2|\mathcal{X}|$) transitions defined by each atomic constraint of the disjunction. For instance, the transition with the guard $a \leq x < b$ introduces two complement transitions with the guard $x < a$ and $b \leq x$.

defined for all locations $\ell \in L^{\mathsf{e}}$ and for all transitions $\delta = (\ell, g, Y, \ell') \in \Delta^{\mathsf{e}}$ by

$$\mathsf{wt}^{\mathsf{e}}(\ell) = \begin{cases} \mathsf{wt}(\ell) & \text{if } \ell \in L \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad \mathsf{wt}^{\mathsf{e}}(\delta) = \begin{cases} \mathsf{wt}(\delta) & \text{if } \ell \in L_{\mathsf{Max}} \\ \mathsf{wt}(\delta') & \text{if } \ell \in L_{\mathsf{Min}} \text{ and } \ell' = \ell^{\delta'} \\ 0 & \text{if } \ell \in L^{\mathsf{e}}_{\mathsf{Max}} \setminus L_{\mathsf{Max}}. \end{cases}$$

The gadget used for this transformation is depicted in Figure 9.2. For technical reasons, all deadlocks in L_T are supposed to be winning for Min. In particular, we note that ☹ is not a deadlock but belongs to Min. We note that this transformation can be computed in polynomial time.

Lemma 9.19. *Let $\mathcal{G}$ be a WTG under a conservative semantics. Then, the number of locations of $\mathcal{G}^{\mathsf{e}}$ is polynomial in the number of locations of $\mathcal{G}$.*

Now, we prove the correctness of this construction. As expected, the definition of $\mathcal{G}^{\mathsf{e}}$ does not depend on the parameter: the definition is not parametrized by the perturbation bound and works for all of them. Let $\mathfrak{p} > 0$ be a parameter and an initial configuration (ℓ, ν), we prove the correctness of this construction by showing that for all parameters $\mathfrak{p} > 0$,

$$\mathsf{rVal}^{\mathfrak{p},\mathsf{excess}_{\mathfrak{p}}}_{\mathcal{G}^{\mathsf{e}}}(\ell, \nu) = \mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}_{\mathcal{G}}(\ell, \nu)\,.$$

We rely on the fact that a decision is robust under $\mathsf{cons}_{\mathfrak{p}}$ in $\mathcal{G}$ if and only if it is robust under $\mathsf{excess}_{\mathfrak{p}}$ and it guarantees that Min avoids the location ☹ in $\mathcal{G}^{\mathsf{e}}$.

First inequality: $\mathsf{rVal}^{\mathfrak{p},\mathsf{excess}_{\mathfrak{p}}}_{\mathcal{G}^{\mathsf{e}}}(\ell, \nu) \leq \mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}_{\mathcal{G}}(\ell, \nu)$. To prove this inequality, we exhibit a strategy under $\mathsf{excess}_{\mathfrak{p}}$ as least as good as a (given) strategy under $\mathsf{cons}_{\mathfrak{p}}$. Formally, we consider $\chi \in \mathsf{rStrat}^{\mathsf{cons}_{\mathfrak{p}}}_{\mathsf{Min}}$ be a strategy for Min under $\mathsf{cons}_{\mathfrak{p}}$, and we define $\chi^{\mathsf{e}} \in \mathsf{rStrat}^{\mathsf{excess}_{\mathfrak{p}}}_{\mathsf{Min}}$ be a strategy for Min under $\mathsf{excess}_{\mathfrak{p}}$ such that

$$\sup_{\zeta \in \mathsf{rStrat}^{\mathsf{excess}_{\mathfrak{p}}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell, \nu), \chi^{\mathsf{e}}, \zeta) \leq \sup_{\zeta \in \mathsf{rStrat}^{\mathsf{cons}_{\mathfrak{p}}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell, \nu), \chi, \zeta)\,.$$

To define this strategy, we use the (partial) function $\mathsf{proj} \colon \mathsf{FPlays}^{\mathsf{excess}_{\mathfrak{p}}}_{\mathcal{G}^{\mathsf{e}}} \to \mathsf{FPlays}^{\mathsf{cons}_{\mathfrak{p}}}_{\mathcal{G}}$ on finite plays in $\mathsf{FPlays}^{\mathsf{excess}_{\mathfrak{p}}}_{\mathcal{G}^{\mathsf{e}}}$ starting from (ℓ, ν) and ending in a state of $[\![\mathcal{G}]\!]^{\mathsf{cons}_{\mathfrak{p}}}$. In particular, proj is defined in finite plays that do not reach the location ☹. By construction of $\mathcal{G}^{\mathsf{e}}$, except configurations around ℓ^{δ}, edges along ρ are ones of $[\![\mathcal{G}]\!]^{\mathsf{cons}_{\mathfrak{p}}}$: all decisions applied in ρ are robust for $\mathsf{cons}_{\mathfrak{p}}$, or feasible for Max. Thus, the projection function consists in removing the configurations around ℓ^{δ}. Formally, for all finite well-formed plays $\rho \in \mathsf{FPlays}^{\mathsf{excess}_{\mathfrak{p}}}_{\mathcal{G}^{\mathsf{e}}}$ starting and ending in a

configuration of $[\![\mathcal{G}]\!]^{\mathsf{cons_p}} = \langle S, E\rangle$, we inductively define:

$$\mathsf{proj}(\rho) = \begin{cases} (\ell,\nu) & \text{if } \rho = (\ell,\nu); \\ \mathsf{proj}(\rho_1) \xrightarrow{\delta,t} s & \text{if } \rho = \rho_1 \xrightarrow{\delta,t} s \text{ with } \mathsf{last}(\rho_1) \in S; \\ \mathsf{proj}(\rho_1) \xrightarrow{\delta,p} (\ell',\nu') & \text{if } \rho = \rho_1 \xrightarrow{\delta',p} (\ell^{\delta},\nu) \xrightarrow{\delta,0} (\ell',\nu'). \end{cases}$$

Moreover, we remark that proj fulfils the following property:

Lemma 9.20. *Let $\rho \in \mathsf{FPlays}_{\mathcal{G}^{\mathsf{e}}}^{\mathsf{excess_p}}$ be a well-formed play. Then,*

1. $\mathsf{proj}(\rho) \in \mathsf{FPlays}_{\mathcal{G}}^{\mathsf{cons_p}}$ *and* $\mathit{last}(\rho) = \mathit{last}(\mathsf{proj}(\rho))$*;*
2. $\mathsf{wt}_\Sigma(\rho) = \mathsf{wt}_\Sigma(\mathsf{proj}(\rho))$.

Proof. 1. By definition of proj.

2. We reason by induction on the length of ρ such that $\mathsf{last}(\rho)$ is a state in $[\![\mathcal{G}]\!]^{\mathsf{cons_p}}$. If $\rho = (\ell,\nu)$, then the property trivially holds. Otherwise, we suppose that $\rho = \rho'[\delta,t]$. In particular, we distinguish two cases according to the last configuration of ρ'. If $\mathsf{last}(\rho') \in \mathsf{Conf}^{\mathcal{G}}$, then we have $\mathsf{proj}(\rho) = \mathsf{proj}(\rho')[\delta',t]$. Inparticular, we deduce that

$$\begin{aligned} \mathsf{wt}_\Sigma(\mathsf{proj}(\rho)) &= \mathsf{wt}_\Sigma(\mathsf{proj}(\rho')) + t\,\mathsf{wt}(\mathsf{last}(\mathsf{proj}(\rho'))) + \mathsf{wt}(\delta') \\ &= \mathsf{wt}_\Sigma(\rho') + t\,\mathsf{wt}(\mathsf{last}(\rho')) + \mathsf{wt}(\delta) = \mathsf{wt}_\Sigma(\rho) \end{aligned}$$

since $\mathsf{wt}_\Sigma(\mathsf{proj}(\rho')) = \mathsf{wt}_\Sigma(\rho')$ (by the induction hypothesis applied on ρ'), $\mathsf{wt}(\delta') = \mathsf{wt}(\delta)$ (by Definition 9.18), and $\mathsf{wt}(\mathsf{last}(\rho')) = \mathsf{wt}(\mathsf{last}(\mathsf{proj}(\rho')))$ (by item 1),

Otherwise, $\mathsf{last}(\rho') = \ell^{\delta}$ and we can rewrite ρ by $\rho = \rho'' \xrightarrow{\delta,t} (\ell^{\delta},\nu') \xrightarrow{\delta,0} s$. In particular, we have $\mathsf{proj}(\rho) = \mathsf{proj}(\rho'') \xrightarrow{\delta,t} s$. Moreover, since $\mathsf{wt}_\Sigma(\mathsf{proj}(\rho'')) = \mathsf{wt}_\Sigma(\rho'')$ (by the induction hypothesis applied on ρ'') and $\mathsf{wt}(\mathsf{last}(\rho'')) = \mathsf{wt}(\mathsf{last}(\mathsf{proj}(\rho'')))$ (by item 1), we deduce that $\mathsf{wt}_\Sigma(\mathsf{proj}(\rho)) = \mathsf{wt}(\rho)$. □

Now, we have tools to define χ^{e}: for all finite well-formed plays $\rho \in \mathsf{FPlays}_{\mathcal{G}}^{\mathsf{excess_p}}$ starting in (ℓ,ν) and ending in a configuration of Min, we let

$$\chi^{\mathsf{e}}(\rho) = (\delta', t) \qquad \text{if } \chi(\mathsf{proj}(\rho)) = (\delta, t)\,.$$

We note that this strategy chooses robust decisions for $\mathsf{excess_p}$ since guards in transitions starting in a location of Min in $\mathcal{G}^{\mathsf{e}}$ are equal to true. Moreover, since the definition of χ^{e} relies on proj, it is no surprise that:

Lemma 9.21. *Let $\rho \in \mathsf{FPlays}_{\mathcal{G}^{\mathsf{e}}}^{\mathsf{excess}_{\mathfrak{p}}}$ conforming to χ^{e} such that $\mathsf{proj}(\rho)$ is defined. Then, $\mathsf{proj}(\rho)$ is conforming to χ.*

Proof. We reason by induction on the length of finite plays $\rho \in \mathsf{FPlays}_{\mathcal{G}^{\mathsf{e}}}^{\mathsf{excess}_{\mathfrak{p}}}$ starting in (ℓ, ν) and ending in a configuration of $[\![\mathcal{G}]\!]^{\mathsf{cons}_{\mathfrak{p}}}$ (i.e. $\mathsf{proj}(\rho)$ is defined). If $\rho = (\ell, \nu)$ then $\mathsf{proj}(\rho) = (\ell, \nu)$, and the property trivially holds. Otherwise, we suppose that $\rho = \rho_1[\delta, t]$. Since $\mathsf{proj}(\rho_1)$ is conforming to χ (by the hypothesis of induction) and $\mathsf{last}(\rho_1) = \mathsf{last}(\mathsf{proj}(\rho_1))$ (by Lemma 9.20-1), we conclude that $\mathsf{proj}(\rho)$ is conforming to χ when $\mathsf{last}(\rho_1) \notin \mathsf{Conf}_{\mathsf{Min}}^{\mathcal{G}}$. In particular, we suppose that $\mathsf{last}(\rho_1) \in \mathsf{Conf}_{\mathsf{Min}}^{\mathcal{G}}$, and $\mathsf{proj}(\rho) = \mathsf{proj}(\rho_1)[\delta, t]$. We note that by definition of χ^{e}, the decision chosen by the strategy χ for $\mathsf{proj}(\rho)$ is (δ, t). Thus, $\mathsf{proj}(\rho)$ is conforming to χ. □

We consider $\rho \in \mathsf{FPlays}_{\mathcal{G}^{\mathsf{e}}}^{\mathsf{excess}_{\mathfrak{p}}}$ be a (well-formed) play conforming to χ^{e}, and we prove that there exists a play ρ' in $\mathsf{FPlays}_{\mathcal{G}}^{\mathsf{cons}_{\mathfrak{p}}}$ conforming to χ such that $\mathbf{P}(\rho) \leq \mathbf{P}(\rho')$. To conclude, we distinguish two cases:

- If ρ ends in a configuration of $\mathcal{G}$, then $\mathsf{proj}(\rho) \in \mathsf{FPlays}_{\mathcal{G}}^{\mathsf{cons}_{\mathfrak{p}}}$ exists and is conforming to χ (by Lemma 9.21). If ρ does not reach a target location, then $\mathsf{proj}(\rho)$ does not reach a target location (by Lemma 9.20-1). Thus, in this case $\mathbf{P}(\rho) = +\infty = \mathbf{P}(\mathsf{proj}(\rho))$. Otherwise, ρ reaches a target location and $\mathsf{proj}(\rho)$ reaches the target location (by Lemma 9.20-1). Thus, by Lemma 9.20-2, we conclude that $\mathbf{P}(\rho) \leq \mathbf{P}(\mathsf{proj}(\rho))$.

- Otherwise, $\mathsf{last}(\mathsf{proj}) \notin \mathsf{Conf}^{\mathcal{G}}$, i.e. $\mathsf{proj}(\rho)$ does not exist. Since ρ is well-formed, then either ρ reaches ☹, or $\mathsf{last}(\rho) = (\ell^{\delta}, \nu)$. If ρ reaches ☹, then there exists a decision along ρ that is not robust for the conservative semantics. Now, let (ℓ, ν) be the first configuration along ρ from where Min does not follow the decision of χ. So, the prefix of the play until (ℓ, ν) follows the decisions of χ, i.e. along this play, all decisions of Min are robust for the conservative semantics. Thus, proj is defined on this play, and we obtain a contradiction with the definition of χ^{e}. Otherwise, $\mathsf{last}(\rho) = (\ell^{\delta}, \nu)$ and we rewrite $\rho = \rho' \xrightarrow{\delta, t} (\ell^{\delta}, \nu)$. Since ρ does not reach ☹, we deduce that $\mathsf{last}(\rho') \in \mathsf{Conf}^{\mathcal{G}}$ does not reach a target location. Thus, by applying the previous case, we can fix $\mathsf{proj}(\rho')$ be a play such that $\mathbf{P}(\rho) = +\infty = \mathbf{P}(\mathsf{proj}(\rho'))$.

Finally, we have shown that for all plays ρ conforming to χ^{e}, we can define a play ρ' conforming to χ such that $\mathbf{P}(\rho) \leq \mathbf{P}(\rho')$. In particular, we obtain that

$$\mathsf{rVal}_{\mathcal{G}}^{\mathfrak{p},\mathsf{excess}_{\mathfrak{p}}}(\ell, \nu) \leq \sup_{\zeta \in \mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{excess}_{\mathfrak{p}}}} \mathbf{P}(\mathsf{Play}((\ell, \nu), \chi^{\mathsf{e}}, \zeta) \leq \sup_{\zeta \in \mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{cons}_{\mathfrak{p}}}} \mathbf{P}(\mathsf{Play}((\ell, \nu), \chi, \zeta)\,.$$

Since this inequality holds for all robust strategies under the conservative semantics, we obtain the claimed inequality.

Second inequality: $\mathsf{rVal}^{\mathfrak{p},\mathsf{excess}_\mathfrak{p}}_{\mathcal{G}^\mathsf{e}}(\ell,\nu) \geq \mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_\mathfrak{p}}_{\mathcal{G}}(\ell,\nu)$. We exhibit a strategy under $\mathsf{cons}_\mathfrak{p}$ as least as good as a (given) strategy $\chi \in \mathsf{rStrat}^{\mathsf{excess}_\mathfrak{p}}_{\mathsf{Min}}$ under $\mathsf{excess}_\mathfrak{p}$. We thus define a strategy $\chi^\mathsf{c} \in \mathsf{rStrat}^{\mathsf{cons}_\mathfrak{p}}_{\mathsf{Min}}$ for Min under $\mathsf{cons}_\mathfrak{p}$ such that

$$\sup_{\zeta \in \mathsf{rStrat}^{\mathsf{excess}_\mathfrak{p}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi,\zeta) \geq \sup_{\zeta \in \mathsf{rStrat}^{\mathsf{cons}_\mathfrak{p}}_{\mathsf{Max}}} \mathbf{P}(\mathsf{Play}((\ell,\nu),\chi^\mathsf{c},\zeta)\,.$$

To do it, we define a function that encodes plays in $\mathcal{G}$ under $\mathsf{cons}_\mathfrak{p}$ into plays in $\mathcal{G}^\mathsf{e}$ under $\mathsf{excess}_\mathfrak{p}$. Since $\Delta_\mathfrak{p}$ is the same for both semantics (by hypothesis), all robust decisions for the conservative semantics in $\mathcal{G}$ are robust decisions for the excessive semantics in $\mathcal{G}^\mathfrak{p}$ where all guards from a location of Min contain no constraints. In other words, a decision is robust in $\mathcal{G}^\mathfrak{p}$ if the delay is greater than $\min \Delta_\mathfrak{p}$ that is always the case for delays in robust decisions in $\mathcal{G}$ under the conservative semantics. Formally, we define, by induction on the length of finite plays, $\mathsf{inj} : \mathsf{FPlays}^{\mathsf{cons}_\mathfrak{p}}_{\mathcal{G}} \to \mathsf{FPlays}^{\mathsf{excess}_\mathfrak{p}}_{\mathcal{G}^\mathsf{e}}$ such that for all finite plays $\rho \in \mathsf{FPlays}^{\mathsf{cons}_\mathfrak{p}}_{\mathcal{G}}$, we have $\mathsf{inj}(\rho)$ be equal to

$$\begin{cases} s & \text{if } \rho = s; \\ \mathsf{inj}(\rho_1) \xrightarrow{\delta,t} s & \text{if } \rho = \rho_1 \xrightarrow{\delta,t} s \text{ with } \mathsf{last}(\rho_1) \in \mathsf{Conf}^\mathcal{G}; \\ \mathsf{inj}(\rho_1) \xrightarrow{\delta',p} (\ell^\delta,\nu) \xrightarrow{\delta,0} s & \text{if } \rho = \rho_1 \xrightarrow{\delta,p} s \text{ with } \mathsf{last}(\rho_1) = (\ell,\nu,\delta,t). \end{cases}$$

We remark that inj fulfils the following property:

Lemma 9.22. *Let $\rho \in \mathsf{FPlays}^{\mathsf{cons}_\mathfrak{p}}_{\mathcal{G}}$ be a well-formed play. Then,*

1. $\mathsf{inj}(\rho) \in \mathsf{FPlays}^{\mathsf{excess}_\mathfrak{p}}_{\mathcal{G}}$ *and* $\mathit{last}(\rho) = \mathit{last}(\mathsf{inj}(\rho))$*;*
2. $\mathsf{wt}_\Sigma(\rho) = \mathsf{wt}_\Sigma(\mathsf{inj}(\rho))$.

Proof. 1. By definition of inj.

2. We reason by induction on the length of ρ. If $\rho = (\ell,\nu)$, then the property trivially holds. Otherwise, we suppose that $\rho = \rho'[\delta,t]$. In particular, we distinguish two cases according to $\mathsf{last}(\rho')$. If $\mathsf{last}(\rho') \in \mathsf{Conf}^\mathcal{G}$, then we have $\mathsf{inj}(\rho) = \mathsf{inj}(\rho')[\delta,t]$. Since $\mathsf{wt}_\Sigma(\mathsf{inj}(\rho')) = \mathsf{wt}_\Sigma(\rho')$ (by the induction hypothesis applied on ρ') and $\mathsf{wt}(\mathsf{last}(\rho')) = \mathsf{wt}(\mathsf{last}(\mathsf{inj}(\rho')))$ (by item 1). In particular, we deduce that $\mathsf{wt}_\Sigma(\mathsf{inj}(\rho)) = \mathsf{wt}_\Sigma(\rho)$.

   Otherwise, $\mathsf{last}(\rho') \notin \mathsf{Conf}^\mathcal{G}$. In particular, we have $\mathsf{inj}(\rho) = \mathsf{inj}(\rho') \xrightarrow{\delta',t} (\ell^\delta,\nu) \xrightarrow{\delta,0} s$. Moreover, since $\mathsf{wt}_\Sigma(\mathsf{inj}(\rho')) = \mathsf{wt}_\Sigma(\rho')$ (by the induction hypothesis applied on ρ'), $\mathsf{wt}(\mathsf{last}(\rho')) = \mathsf{wt}(\mathsf{last}(\mathsf{inj}(\rho')))$ (by item 1) and by definition of $\mathcal{G}^\mathsf{e}$, we deduce that $\mathsf{wt}_\Sigma(\mathsf{inj}(\rho)) = \mathsf{wt}(\rho)$. □

Now, we have tools to define χ^{c}: for all finite plays $\rho \in \mathsf{FPlays}_{\mathcal{G}}^{\mathsf{cons_p}}$, we let

$$\chi^{\mathsf{c}}(\rho) = (\delta, t) \qquad \text{if } \chi(\mathsf{inj}(\rho)) = (\delta', t)$$

such that $\delta' = (\mathsf{last}(\rho), \mathrm{true}, \emptyset, \ell^{\delta})$ and $t \models_{\mathsf{cons_p}}^{\mathsf{last}(\rho)} \delta$. We note, by definition, that this strategy chooses a robust decision for $\mathsf{cons_p}$. Moreover, since the definition of χ^{c} relies on inj, we can show:

Lemma 9.23. *Let $\rho \in \mathsf{FPlays}_{\mathcal{G}}^{\mathsf{cons_p}}$ conforming to χ^{c}. Then, $\mathsf{inj}(\rho)$ is conforming to χ.*

Proof. We reason by induction on the length of finite plays $\rho \in \mathsf{FPlays}_{\mathcal{G}}^{\mathsf{cons_p}}$. If $\rho = s$ then $\mathsf{inj}(\rho) = s$, and the property trivially holds. Otherwise, we suppose that $\rho = \rho_1[\delta, t]$. Since $\mathsf{proj}(\rho_1)$ is conforming to χ (by the hypothesis of induction) and $\mathsf{last}(\rho_1) = \mathsf{last}(\mathsf{inj}(\rho_1))$ (by Lemma 9.22-1), we conclude that $\mathsf{inj}(\rho)$ is conforming to χ when $\mathsf{last}(\rho_1) \notin \mathsf{Conf}_{\mathsf{Min}}^{\mathcal{G}}$. In particular, we suppose that $\mathsf{last}(\rho_1) \in \mathsf{Conf}_{\mathsf{Min}}^{\mathcal{G}}$, $\mathsf{inj}(\rho) = \mathsf{inj}(\rho_1)[\delta, t]$. To conclude, we remark that (δ, t) is robust under the conservative semantics in $\mathcal{G}$ since χ is supposed to win. Otherwise, Max can choose a perturbation such that ☹ was reached, which contradicts that χ is winning. □

We consider a (well-formed) play $\rho \in \mathsf{FPlays}_{\mathcal{G}}^{\mathsf{cons_p}}$ conforming to χ, and we prove that $\mathsf{inj}(\rho) \in \mathsf{FPlays}_{\mathcal{G}}^{\mathsf{cons_p}}$ conforming to χ (by Lemma 9.23) satisfies $\mathbf{P}(\rho) \leq \mathbf{P}(\mathsf{proj}(\rho))$. To conclude, we distinguish two cases. First, we suppose that ρ does not reach a target location, then $\mathsf{proj}(\rho)$ does not reach a target location (by Lemma 9.22-1). Thus, in this case $\mathbf{P}(\rho) = +\infty = \mathbf{P}(\mathsf{proj}(\rho))$. Otherwise, ρ reaches a target location and $\mathsf{proj}(\rho)$ reaches the target location (by Lemma 9.22-1). Thus, by Lemma 9.22-2, we conclude that $\mathbf{P}(\rho) \leq \mathbf{P}(\mathsf{proj}(\rho))$.

Finally, we have shown that for all plays ρ conforming to χ, we can define a play ρ' conforming to χ^{c} such that $\mathbf{P}(\rho) \geq \mathbf{P}(\rho')$. In particular, we obtain that

$$\sup_{\zeta \in \mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{excess_p}}} \mathbf{P}(\mathsf{Play}((\ell, \nu), \chi, \zeta) \geq \sup_{\zeta \in \mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{cons_p}}} \mathbf{P}(\mathsf{Play}((\ell, \nu), \chi^{\mathsf{c}}, \zeta) \geq \mathsf{rVal}_{\mathcal{G}}^{\mathsf{p}, \mathsf{cons_p}}(\ell, \nu)\,.$$

Since this inequality holds for all robust strategies under the excessive semantics, we obtain the claimed inequality.

9.3. Applications to robust reachability problems

Robust reachability may be defined when we know and fix the robust semantics. In particular, the parameter is already known, and the problem is close to the reachability problem under the exact semantics. However, we also want to study reachability under a class of robust semantics with an arbitrary small (non-fixed)

parameter. In this case, we talk about limit robust reachability problem, and we ask about the existence of a parameter such that Min reaches a target location under the robust semantics with parameters less than the found parameter. In this section, we present both problems and summarise their results in Table III.2.

9.3.1. Fixed-parameter robust reachability problem

We consider a robust semantics under a fixed-parameter and ask when Min can reach a target location under this semantics. Formally, we defined this problem as follows:

Definition 9.24. Given an RTG $\mathcal{G}$ and a robust semantics $\mathsf{rob}_{\mathfrak{p}}$ of parameter $\mathfrak{p} \in \mathbb{Q}_{>0}$, the *fixed-parameter robust reachability problem* decides if Min has a winning strategy in $\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Min}}(\mathcal{G})$ for the reachability objective in $\mathcal{G}$.

Under a conservative semantics, to solve this problem[5], authors of [CHP11] translate the timed game into an equivalent one under the exact semantics (we have extended this idea for robust semantics in WTGs in Section 9.1). About the hardness of this problem, authors of [ORS14] give a reduction from the halting problem of *linear-bounded alternating Turing machines*[6] that is an EXPTIME-complete problem [CKS81]. More precisely, they prove that they can define an RTG $\mathcal{A}$ with at least five clocks such that for all parameters $\mathfrak{p} < 1/12$, Min has a winning strategy in $\mathcal{G}$ under a conservatie semantics of parameter $\mathfrak{p}$ if and only if the alternating Turing machine where the tape used is bounded by three times the size of the input, halts.

Under the excessive semantics, we want to apply the two previous encodings to obtain the easiness and the hardness of the problem. To do it, we first remark that a fixed-parameter robust value is equal to $+\infty$ if and only if Min loses the fixed-parameter robust reachability problem. In particular, we deduce that

Theorem 9.25. *The fixed-parameter robust reachability problem under an excessive semantics is* EXPTIME*-complete.*

Proof. Let $\mathcal{A}$ be an RTG under an excessive semantics. We remark that fixed-parameter robust values are a quantitative extension of fixed-parameter robust reachability problems. In particular, Theorem 9.2 guarantees that we can compute

[5] In [CHP11], the authors work on concurrent timed games under a parity objective that is an extension of our timed games.

[6] A linear-bounded alternating Turing machine is an extension of a deterministic Turing machine. An alternating Turing machine considers two types of states: the existential ones that implement a disjunction over their successor, and the universal ones that implement a conjunction. A Turing machine is said linear-bounded if the tape used is linearly bounded in the size of the input.

in polynomial time (by Lemma 9.12) a new RTG under the exact semantics $\mathcal{A}^{\mathfrak{p}}$ from $\mathcal{A}$ such that Min wins in $\mathcal{A}$ if and only if Min wins in $\mathcal{A}^{\mathfrak{p}}$. Now, we conclude that the fixed-parameter robust reachability problem under an excessive semantics can be solved in exponential time by applying Theorem 1.21 in $\mathcal{A}^{\mathfrak{p}}$.

Now, to conclude about the hardness result, by reducing the problem from the problem under the conservative semantics with Theorem 9.17 that provides a polynomial reduction by Lemma 9.19. To conclude, we note that the fixed-parameter robust reachability problem under the conservative semantics is known as EXPTIME-complete by [ORS14]. □

9.3.2. Limit robust reachability problem

Now, we consider an extension of the fixed-parameter robust reachability problem. In particular, given a class of robust semantics (and not only one robust semantics), we ask if there exists robust semantics in the class (i.e. a parameter) such that Min wins the fixed-parameter robust reachability problem under this robust semantics. Formally, we defined this problem as follows:

Definition 9.26. Given an RTG $\mathcal{A}$ and a class of robust semantic $(\mathsf{rob}_{\mathfrak{p}})_{\mathfrak{p}}$, the *limit robust reachability problem* decides if there exists $\mathfrak{p} > 0$ be a parameter such that Min has a winning strategy in $\mathsf{rStrat}^{\mathsf{rob}_{\mathfrak{p}}}_{\mathsf{Min}}(\mathcal{A})$.

The limit robust reachability problem is an existential extension of the fixed-parameter robust reachability problem, so solving this second problem seems more tricky. Moreover, the previous techniques to solve it are not applicable. Since we do not know the parameter $\mathfrak{p}$, the same encoding into the exact semantics does not work as we can not define the guard on the new clock.

In timed automata, the problem under the conservative semantics is not harder than solving the reachability problem in timed automata under the exact semantics (that is PSPACE-complete [AD94]). Indeed, the region abstraction with open regions works alike. However, under the excessive semantics, solving this problem is equivalent to solving the reachability in an RTG. Intuitively, under conservative semantics, the unique player can not see the perturbation of the environment, but this property does not hold under excessive semantics. Indeed, in this case, the environment may reach configurations (and more generality, regions) that were not reachable under the exact semantic: the behaviour of an RTG appears.

In RTGs, the problem is already solved under excessive semantics. Intuitively, a configuration in a region is winning if it is far enough from all borders of regions. In particular, authors of [BMS15] used *shrinking regions*[7] of the timed games to

[7] Shrinking regions are regions in which each border may be restrained with a multiple of $\mathfrak{p}$. For instance, the region described by the set of valuations $\{(x_1, x_2) \mid 0 < x_1 \leq x_2 < 1\}$ may become the shrinking region described by $\{(x_1, x_2) \mid \mathfrak{p} < x_1 \leq x_2 < 1 - 3\,\mathfrak{p}\}$.

	conservative	excessive
WTG	Undecidable [BMS13]	
Divergent WTG	3-EXPTIME and EXPTIME-hard Theorem 9.28	Open
1-clock WTG (non-neg)	Decidable Theorem 9.32	Decidable [Guh+15]

Table 9.1 – Solving fixed-parameter robust value problems under conservative and excessive semantics.

describe winning configurations inside each region and computing a parameter that satisfies the limit robust reachability problem (see [BMS15, Section 4.1.2] for details over the upper-bound of $\mathfrak{p}$).

Under the conservative semantics, we apply the encoding of a conservative semantics into an excessive one, since it does not depend on the parameter. In particular, with results from [ORS14], we deduce:

Theorem 9.27. *The limit robust reachability problem is* EXPTIME*-complete.*

Proof. Let $\mathcal{A}$ be an RTG under a class of conservative semantics $(\mathsf{cons}_{\mathfrak{p}})_{\mathfrak{p}}$. In this case, Theorem 9.17 guarantees that we can compute, from $\mathcal{A}$, in polynomial time (by Lemma 9.19) a new RTG $\mathcal{A}^{\mathsf{e}}$ under the class of excessive semantics $(\mathsf{excess}_{\mathfrak{p}})_{\mathfrak{p}}$ such that, for all parameters $\mathfrak{p}$, Min wins in $\mathcal{A}$ under $\mathsf{cons}_{\mathfrak{p}}$ if and only if Min wins in $\mathcal{A}^{\mathsf{e}}$ under $\mathsf{excess}_{\mathfrak{p}}$. Now, from [BMS15], we know that we can decide in exponential time if there exists $\mathfrak{p} > 0$ such that Min wins in $\mathcal{A}^{\mathsf{e}}$ under $\mathsf{excess}_{\mathfrak{p}}$. Thus, we conclude that we can solve the limit robust reachability problem in $\mathcal{A}$ under a class of conservative semantics in exponential time.

Finally, we conclude about the hardness result by applying the reduction from the halting problem of linear-bounded alternating Turing machines from [ORS14]. □

9.4. Application to robust value problems

The two previous encodings also allow us to close some open questions about fixed-parameter robust value problems under a fixed conservative semantics. More precisely, we use the encodings to provide algorithm to decide this problem in the case of divergent WTGs and one-clock WTGs with only non-negative weights under conservative semantics. Unfortunately, both encodings do not suffice to prove the decidability of the problem in divergent WTGs under an excessive semantics. Table 9.1 summarises these results.

Deciding when a robust value is equal to $+\infty$ Before focusing on the fixed-parameter robust value problem under the conservative semantics, we will exploit qualitative results obtained in the previous section on robust reachability problem to prove that deciding when a robust value (under conservative or excessive semantics) is equal to $+\infty$ is EXPTIME-complete.

Given a robust value, the corresponding robust value problems where $\lambda = +\infty$ are purely qualitative (like for the deterministic value). Indeed, a fixed-parameter robust value is equal to $+\infty$ if and only if Min loses the fixed-parameter robust reachability problem, and limit robust values are equal to $+\infty$ if and only if Min loses the limit robust reachability problem. Hence, the problem is decidable (it is EXPTIME-complete) from the results presented in Table III.2.

However, unlike the deterministic value, we can not remove all configurations such that the considered robust value is equal to $+\infty$. Indeed, in a location, the robust value function is not continuous over the region: a finite robust value may become $+\infty$ when the valuation is too close to a border of the guards. Moreover, adding guards may increase the number of configurations from which Min can not reach a target location, i.e. such that robust values are infinite.

9.4.1. Fixed-parameter robust value problem in divergent weighted timed games under a conservative semantics

We consider divergent WTGs under a conservative semantics and we prove that the fixed-parameter robust value problem is decidable in this case.

Theorem 9.28. *Let $\mathcal{G}$ be a divergent WTG and $\mathsf{cons}_{\mathfrak{p}}$ be a conservative semantics of parameter $\mathfrak{p} > 0$. Then, $\mathsf{rVal}_{\mathcal{G}}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}$ is computable in triply-exponential time and the fixed-parameter robust value problem is* EXPTIME*-hard.*

Like for deciding when a robust value is equal to $+\infty$, the hardness result is given by the EXPTIME-hardness of the fixed-parameter robust reachability problem. Thus, the proof consists on showing that $\mathsf{rVal}_{\mathcal{G}}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}$ is computable in triply-exponential time. From Theorem 9.17, we know the existence of a WTG $\mathcal{G}^{\mathfrak{p}}$ under $\mathsf{excess}_{\mathfrak{p}}$ computable in polynomial time (Lemma 9.12) the robust value as $\mathcal{G}$. In other words, solving the fixed-parameter robust value problem in $\mathcal{G}$ under the given robust semantics $\mathsf{rob}_{\mathfrak{p}}$ is equivalent to solving the deterministic value problem in $\mathcal{G}^{\mathfrak{p}}$. Since we add a clock, we only know that the deterministic value in $\mathcal{G}^{\mathfrak{p}}$ is decidable when it is divergent. We summarise this idea in the following lemma.

Lemma 9.29. *For all WTGs $\mathcal{G}$ under a robust semantics $\mathsf{rob}_{\mathfrak{p}}$ such that $\mathcal{G}^{\mathfrak{p}}$ (defined in Definition 9.11) is divergent, $\mathsf{rVal}_{\mathcal{G}}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}$ is computable in triply-exponential time.*

Proof. First, we know that $\mathcal{G}^{\mathfrak{p}}$ is computable in polynomial time (by Lemma 9.12). By Theorem 1.34, since $\mathcal{G}^{\mathfrak{p}}$ is supposed divergent, we deduce that its deterministic

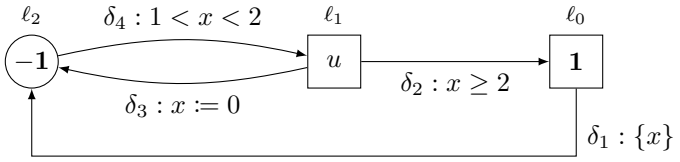


Figure 9.3 – A WTG (without target locations) with ℓ_1 as an urgent location that is divergent under the exact semantics but not under a translated excessive semantics.

value function is computable in triply exponential time. Finally, we conclude since this deterministic value function is equal to the fixed-parameter robust value function (by Proposition 9.9). □

To conclude the proof of Theorem 9.28, we need to prove that divergence property is kept by $\mathcal{G}^{\mathfrak{p}}$ transformation under conservative semantics. Indeed, in this case, we can apply Lemma 9.29 and conclude the proof.

Proposition 9.30. *Let $\mathcal{G}$ be a divergent WTG and $\mathsf{cons}_{\mathfrak{p}}$ be a conservative semantics. Then, $\mathcal{G}^{\mathfrak{p}}$ is divergent.*

Proof. Let ρ be a play in $\mathcal{G}^{\mathfrak{p}}$ under the exact semantic following a cycle in $\mathcal{R}(\mathcal{G}^{\mathfrak{p}})$, and we prove that $\mathsf{wt}_\Sigma(\rho) \notin (-1, 1)$.

First, we prove that plays following a region path[8] in $\mathcal{G}$ under the conservative semantics, are such that $\mathsf{wt}_\Sigma(\rho) \notin (-1, 1)$. Let $\rho \in \mathsf{FPlays}_{\mathcal{G}}^{\mathsf{cons}_{\mathfrak{p}}}$ be such a play. By Lemma 8.5, there exists a play $\rho' \in \mathsf{FPlays}_{\mathcal{G}}^{\mathsf{exact}}$ such that it follows a cyclic path of $\mathcal{R}(\mathcal{G})$ (by Lemma 8.5). As $\mathcal{G}$ is divergent, we know that $\mathsf{wt}_\Sigma(\rho') \notin (-1, 1)$. We conclude, since $\mathsf{wt}_\Sigma(\rho) \notin (-1, 1)$ (by Lemma 8.5).

Now, we use the function inj defined in Section 9.1.2. By letting ρ be a play in $\mathcal{G}^{\mathfrak{p}}$ under the exact semantics following a cycle in $\mathcal{R}(\mathcal{G}^{\mathfrak{p}})$, we deduce that $\mathsf{inj}(\rho) \in \mathsf{FPlays}_{\mathcal{G}}^{\mathsf{rob}_{\mathfrak{p}}}$ follows a cycle in $\mathcal{R}(\mathcal{G})$ since the first and the last locations of ρ and $\mathsf{inj}(\rho)$ are the same (by Lemma 9.15-1). In particular, the last case guarantees that $\mathsf{wt}_\Sigma(\rho) \notin (-1, 1)$. Moreover, by Lemma 9.15-2, we obtain that $\mathsf{wt}(\mathsf{inj}(\rho)) \notin (-1, 1)$. Thus, $\mathcal{G}^{\mathfrak{p}}$ is divergent. □

As a side note, it is tempting to strengthen Proposition 9.30 to ensure that the divergence property of WTGs is also preserved by the excessive semantics. Unfortunately, this is not the case, as shown in the following example.

Example 9.31. We consider the WTG $\mathcal{G}$ depicted in Figure 9.3. Under the exact semantics, only the cyclic paths defined by δ_4 and δ_3 are available. Since all plays that follow this cyclic path have a weight at most -1, $\mathcal{G}$ is divergent.

[8] As under the exact semantics, we can add region information along paths of the WTG to obtain *region paths* under robust semantics.

Now, under a translated excessive semantics of parameter bound $\mathfrak{p}$, the cyclic paths defining by δ_4, δ_2 and δ_1 are also available. However, the weight of the play $(\ell_2, 0) \xrightarrow{\delta_4, 2-\mathfrak{p}} (\ell_2, 0, \delta_4, 2-\mathfrak{p}) \xrightarrow{\delta_4, \mathfrak{p}} (\ell_1, 2) \xrightarrow{\delta_2, 0} (\ell_0, 2) \xrightarrow{\delta_0, 2} (\ell_2, 0)$ that follows a cyclic region path (i.e. $(\ell_2, \{0\}) \xrightarrow{\delta_4} (\ell_1, (1, 2]) \xrightarrow{\delta_2} (\ell_0, [2, +\infty)) \xrightarrow{\delta_1} (\ell_2, \{0\})$) is null. In particular, $\mathcal{G}$ is not divergent under the excessive semantics. ⌟

Remark. As claimed in Lemma 9.29, the fixed-parameter robust value is computable when $\mathcal{G}^{\mathfrak{p}}$ is divergent. Moreover, as we can decide when $\mathcal{G}^{\mathfrak{p}}$ is divergent [BMR17], we can decide if we can compute the robust value function by using this technique. ◁

9.4.2. Fixed-parameter robust value problem in one-clock weighted timed games under a conservative semantics

We consider only one-clock WTGs with only non-negative weights under a centred conservative semantics. In this case, the encoding into the exact semantics adds a clock: we thus do not know how computing the deterministic value in the resulting WTG. Nonetheless, we prove to compute that the fixed-parameter robust value problem under conservative semantics is decidable by using the decidability of the problem for the centred excessive semantics [Guh+15, Theorem 8] and the encoding into excessive semantics.

Theorem 9.32. *Let $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \{x\}, \Delta, \mathsf{wt}\rangle$ be a one-clock WTG under a conservative semantics of parameter $\mathfrak{p} \in \mathbb{Q}_{>0}$. Then, the fixed-parameter robust value problem under a conservative semantics is decidable.*

Proof. Theorem 9.17 guarantees that we can define a one-clock WTG $\mathcal{G}^{\mathsf{e}}$ in polynomial time (by Lemma 9.12), that preserves the fixed-parameter robust value of $\mathcal{G}$ under $\mathsf{cons}_{\mathfrak{p}}$. Moreover, since [Guh+15, Theorem 8] provides an algorithm to solve fixed-parameter robust value problem under $\mathsf{excess}_{\mathfrak{p}}$ in one-clock WTGs with urgent locations, we conclude that the fixed-parameter robust value problem under a conservative semantics is decidable. □

10. Symbolic Computation of the Limit Robust Values

We have introduced different robust value problems, and have extended their decidability of some robust value problems by reduction into classes of WTGs where the value functions is known as computable. The main contribution in this part is an algorithm based on value iteration to symbolically compute the limit robust value in acyclic WTGs under a conservative and a excessive semantics. Moreover, we use this algorithm on acyclic WTGs to extend the computation on strictly non-Zeno WTGs, i.e. divergent WTGs with only non-negative weight, under a conservative semantics.

Table of contents

10.1. Deterministic value function in an acyclic weighted timed game . . 245
10.1.1. Piecewise affine functions encoding by cells 245
10.1.2. Example of computation of deterministic value functions . . 247
10.1.3. Encoding operations of value iteration 252
10.2. Example of computation of robust value functions 253
10.3. Limit robust values in an acyclic weighted timed game 258
10.3.1. Robust operator for translated robust semantics 258
10.3.2. Operations over piecewise affine functions 262
10.4. An extension to the strictly non-Zeno weighted timed games 270

This chapter is devoted to the symbolic computation of the limit robust value under a conservative semantics or the translated excessive semantics.

Theorem 10.1. *Let* $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt}\rangle$ *be a WTG under a class of robust semantics* $(\mathsf{rob}_{\mathfrak{p}})_{\mathfrak{p}>0}$*. Then,*

- *the limit robust value* $\overline{\mathsf{rVal}}^{\mathsf{cons}}$ *is computable if* $\mathcal{G}$ *is acyclic or strictly non-Zeno under a class of conservative semantics;*
- *the limit robust value* $\overline{\mathsf{rVal}}^{\mathsf{excess}}$ *is computable if* $\mathcal{G}$ *is acyclic under a class of translated excessive semantics.*

In acyclic WTGs, we lift the value iteration algorithm (introduced by [ABM04]) used to compute the deterministic value, to obtain an algorithm that computes limit robust values. Moreover, with this method inspired by shrunk DBM (introduced by [SBM14]), we can compute a value of a perturbation $\mathfrak{p} > 0$ for which the limit robust value is valid. In the following, we denote by n the number of clocks such that $\mathcal{X} = \{x_1, \ldots, x_n\}$ and ℓ be a location of $\mathcal{G}$.

10.1. Deterministic value function in an acyclic weighted timed game

In acyclic WTGs, the deterministic value function is computed by applying the value iterations defined by the function $\mathcal{F}$ (defined in Chapter 2). To guarantee the performance of this algorithm, an efficient encoding of the piecewise affine value functions is used.

10.1.1. Piecewise affine functions encoding by cells

We present a simplification of the formalism used in [Bus19] to encode the piecewise affine functions. Intuitively, a piecewise affine function is a mapping represented by a finite partition of $\mathbb{R}^{\mathcal{X}}_{\geq 0}$ and an affine function on each part. An *affine function* $\mathbb{V}_\ell$ is a mapping $\mathbb{V}_\ell \colon \mathbb{R}^{\mathcal{X}}_{\geq 0} \to \mathbb{R}_\infty$ defined for all valuations $\nu \in \mathbb{R}^{\mathcal{X}}_{\geq 0}$ by

$$\mathbb{V}_\ell(\nu) = \sum_{i=0}^{n} a_i \, \nu(x_i) + b \qquad \text{and encoded by} \quad y = \sum_{i=0}^{n} a_i \, x_i + b$$

where for all $1 \leq i \leq n$, $a_i \in \mathbb{Q}$, $b \in \mathbb{Q} \cup \{\pm\infty\}$, and the variable $y \notin \mathcal{X}$ refers to $\mathbb{V}_\ell$. We remark that, with this encoding, the infinite mappings $\nu \mapsto +\infty$ (resp. $\nu \mapsto -\infty$) may be represented by $y = \sum_{i=0}^{n} 0 \times x_i + \infty$ (resp. $y = \sum_{i=0}^{n} 0 \times x_i + (-\infty)$).

Each affine function is specified in a part of a partition of $\mathbb{R}^{\mathcal{X}}_{\geq 0}$ defined by a finite number of convex polyhedra called *cells*. Each cell is defined by a set of affine inequalities, each affine inequality being encoded by

$$\sum_{i=0}^{n} \alpha_i \, x_i + \beta \;\bowtie\; 0$$

where $\alpha_i \in \mathbb{Q}$, for all $1 \leq i \leq n$, $\beta \in \mathbb{Q}$, $\bowtie \in \{<, \leq, =\}$.

Given an inequality I, we say that I is an *affine equality* when $\bowtie$ is $=$, otherwise we can extract, from I, an affine equality denoted by $E(I)$, of identical α_i and β. By letting $\nu \in \mathbb{R}^{\mathcal{X}}$ be a valuation, we say that ν *satisfies* I, and we denote by

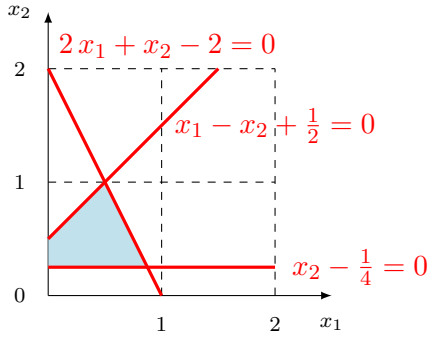


Figure 10.1 – In blue, we depict the cell defined by $\{x_2 - \frac{1}{4} > 0 \wedge 2\,x_1 + x_2 - 2 < 0 \wedge x_1 - x_2 + \frac{1}{2} < 0\}$. In red, we depict its borders.

$\nu \models I$, when $\sum_{i=0}^{n} \alpha_i\, \nu(x_i) + \beta \bowtie 0$ holds. Finally, we denote by $[\![I]\!]$ the set of valuations that satisfy I.

Definition 10.2. A *cell* is a set of valuations $\mathsf{c} \subseteq \mathbb{R}^{\mathcal{X}}$ defined by a conjunction of affine inequalities $I_1 \wedge \cdots \wedge I_m$ such that a valuation $\nu \in c$ if and only if, for all $1 \leq i \leq m$, $\nu \in [\![I_i]\!]$. We denote by $\mathsf{c} = [\![I_1 \wedge \cdots \wedge I_m]\!]$ such a cell.

Given a cell $\mathsf{c} = [\![I_1 \wedge \cdots \wedge I_m]\!]$, its *borders* are given by the affine equalities $E(I_1), \ldots, E(I_m)$. Its *closure* $\overline{\mathsf{c}}$ is obtained by replacing every comparison operator $<$ by $\leq$ in its affine inequalities. For example, a cell is depicted in Figure 10.1. Moreover, we note that the regions are special cells, where borders are of the form $x_i + \beta = 0$ or $x_i - x_j + \beta = 0$ with $x_i, x_j \in \mathcal{X}$.

From a set of affine equations, we define a partition of $\mathbb{R}^{\mathcal{X}}_{\geq 0}$ with cells. Indeed, by letting E be an affine equality of equation $\sum_{i=0}^{n} \alpha_i\, x_i + \beta = 0$, we say that $\mathbb{R}^{\mathcal{X}}$ is partitioned by E into three cells:

- $\mathsf{c}_<$ defined by the set of valuations such that $\sum_{i=0}^{n} \alpha_i\, x_i + \beta < 0$ holds;
- $\mathsf{c}_>$ defined by the set of valuations such that $\sum_{i=0}^{n} \alpha_i\, x_i + \beta > 0$ holds;
- $\mathsf{c}_=$ defined by E, i.e. the set of valuations such that $\sum_{i=0}^{n} \alpha_i\, x_i + \beta = 0$ holds.

Now, given a set of affine equalities $\mathcal{E} = \{E_1, \ldots, E_m\}$, for each equality E_j we define three cells $\mathsf{c}^j_<$, $\mathsf{c}^j_>$ and $\mathsf{c}^j_=$. Since cells are convex polyhedra, the intersection of a finite number of cells remains a cell. Thus, the intersection of cells provided by $\mathcal{E}$ defines a partition of $\mathbb{R}^{\mathcal{X}}_{\geq 0}$ into 3^m cells: each cell of the partition is not intersected with an affine equation of $\mathcal{E}$. Figure 10.1 also provides a partition of $[0, 2]^2$ according to three affine equalities, i.e. according to $\mathcal{E} = \{x_2 - \frac{1}{4} = 0 \wedge 2\,x_1 + x_2 - 2 = 0 \wedge x_1 - x_2 + \frac{1}{2} = 0\}$.

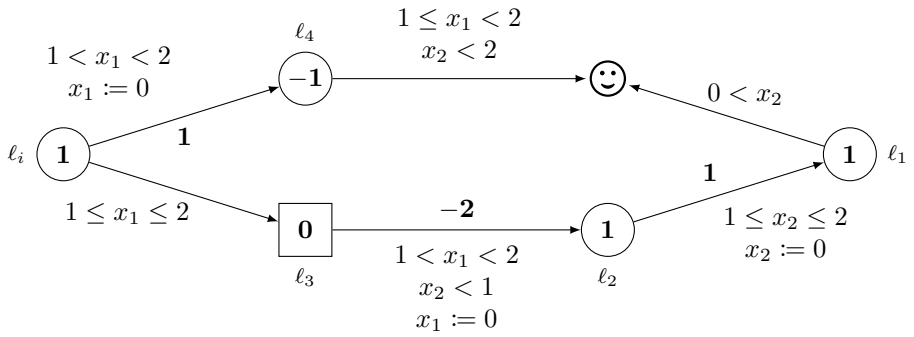


Figure 10.2 – An acyclic WTG with two clocks.

Remark. To obtain a better complexity, in [Bus19], the author used a *nested partition* instead of only the set of cells that we use to define the partition. Indeed, a nested partition encodes a structure of a tree over cells to define the "order" in which cells are defined. ◁

A *partition value function* F over a partition P defined by a set of affine equality is a mapping from the cells of P to an affine value function F. It encodes a mapping from $\mathbb{R}^{\mathcal{X}}_{\ge 0}$ to $\mathbb{R}_\infty$, denoted by $[\![F]\!]$: if $\nu \in \mathsf{c} \subseteq \mathbb{R}^{\mathcal{X}}_{\ge 0}$ is a valuation in a cell defined by P, and F is defined by $y = \sum_{i=0}^{n} a_i\, x_i + b$ over c, then $[\![F]\!](\nu)$, denoted by $[\![F]\!]_\nu$, is equal to $\sum_{i=0}^{n} a_i\, \nu(x_i) + b$.

Definition 10.3. A *piecewise affine value function* $\mathbb{V}_\ell \colon \mathbb{R}^{\mathcal{X}} \to \mathbb{R}_\infty$ for the location ℓ is encoded as a pair (P, F) where P is a partition of domain $\mathbb{R}^{\mathcal{X}}$, and where F is a partition value function defined over P, such that $[\![F]\!] = \mathbb{V}_\ell$.

Finally, we note that cells can be easily manipulated by an algorithm. Indeed, since cells are bounded convex polyhedra, elementary operations over cells (emptiness, intersection and inclusion checking) can be seen as instances of linear programming, and can thus be performed in polynomial time.

10.1.2. Example of computation of deterministic value functions

We consider the acyclic WTG depicted in Figure 10.2 with two clocks $\mathcal{X} = \{x_1, x_2\}$. Our goal is the computation of the deterministic value of ℓ_i in $[0, 2]^2$. To do it, we compute the deterministic value of all other locations with a bottom-up scheme. All cells used throughout this computation are depicted in Figure 10.3.

Deterministic value function in ℓ_1 Since the weight of ℓ_1 is positive, Min intuitively wants to minimise the delay spent in ℓ_1 before reaching the target location.

In particular, from all valuations in $[0,2]^2$, we have

$$\mathsf{dVal}\big(\ell_1, x_1, x_2\big) = \inf_t \big[t \times 1\big] = 0\,.$$

Deterministic value function in ℓ_2 Again since the weight of ℓ_2 is positive, Min wants to minimise the time spent in this location. In particular, there exist two cases according to the valuation of x_2: if $x_2 \geq 1$, then Min can directly leave the location and the deterministic value function is a constant equal to 1; otherwise Min must wait until the valuation of x_2 is equal to 1 and the deterministic value function is equal to $1+(1-x_2) = 2-x_2$. Thus, for all valuations in $[0,2]^2$, we have

$$\mathsf{dVal}\big(\ell_2, x_1, x_2\big) = \inf_{\substack{t \\ 1 \leq x_2 + t \leq 2}} \big[t \times 1\big] + 1 = \begin{cases} 2 - x_2 & \text{if } x_2 < 1; \\ 1 & \text{if } x_2 \geq 1. \end{cases}$$

Deterministic value function in ℓ_3 By studying the guard from ℓ_3 to ℓ_2, we remark that only the blue cell (i.e. when $x_2 < 1$) of ℓ_2 can be reached. More precisely, only the projection of the blue cell on the affine equation $x_1 = 0$ is reached. Thus, the deterministic value function in ℓ_2 is finite only when $x_2 < 1$ and in this case, we have

$$\mathsf{dVal}\big(\ell_3, x_1, x_2\big) = \sup_{\substack{t \\ 1 < x_1 + t < 2 \,\wedge\, x_2 + t < 1}} \mathsf{dVal}(\ell_2, 0, x_2 + t) - 2\,.$$

Now, since the weight of ℓ_3 is null, Max has no pressure to leave (or stay in) this location. Moreover, since the deterministic value function in ℓ_2 in the blue cell decreases when x_2 increases, Max maximises it when $x_2 = 0$, i.e. Max wants to minimise the time spent in ℓ_3 and reach a valuation close enough to 1 for x_1. In particular, we distinguish two cases according to the valuation of x_1.

- We suppose that $x_1 \geq 1$, and Max can directly leave the location to obtain the deterministic value $2 - x_2$ in ℓ_2. In particular, we deduce that the deterministic value function of ℓ_3 when $x_1 \geq 1$ and $x_2 < 1$ is $-x_2$.
- Otherwise, we suppose that $x_1 < 1$ and Max wants to reach ℓ_2 when the valuation of x_1 is equal to 1, i.e. with a delay equal to $1 - x_1$. Since the valuation of x_2 must satisfy $x_2 + 1 - x_1 \leq 1$ to model the guard, we note that the decision is feasible only if $x_2 < x_1$. In this case, the deterministic value function of ℓ_3 is $-2 + 2 - (x_2 + 1 - x_1) = x_1 - x_2 - 1$.

Thus, we conclude that

$$\mathsf{dVal}\big(\ell_3, x_1, x_2\big) = \begin{cases} x_1 - x_2 - 1 & \text{if } x_1 < 1 \text{ and } x_2 < x_1; \\ x_2 & \text{if } x_1 \geq 1 \text{ and } x_2 < 1; \\ +\infty & \text{otherwise.} \end{cases}$$

Deterministic value function in ℓ_4 Since the weight of ℓ_4 is negative, Min intuitively wants to stay in ℓ_4 the most time as possible, i.e. Min wants to take the transition when x_1 and x_2 are close as as possible to 2. Even if the behaviour (i.e. the decision) is analogous whatever the valuation, the deterministic value function depends on it.

- We suppose that $x_2 < x_1$, then x_1 will be the first clock that reaches valuation 2. Thus, by applying the delay $2 - x_1$, $x_2 + 2 - x_1$ satisfies the guard, and the deterministic value function is equal to $-(2 - x_1) = x_1 - 2$.
- Symmetrically, we suppose that $x_1 \leq x_2$ and we apply the delay $2 - x_2$. Unlike in the previous case, the hypothesis $x_1 \leq x_2$ is not sufficient to guarantee that $x_1 + 2 - x_2$ satisfies the guard (since it must be greater than 1). In particular, we must suppose that $x_2 - 1 \leq x_1 \leq x_2$ and, in this case, the deterministic value function is equal to $-(2 - x_2) = x_2 - 2$.

Thus, for all valuations in $[0, 2]^2$, we have

$$\mathsf{dVal}\big(\ell_4, x_1, x_2\big) = \begin{cases} x_1 - 2 & \text{if } x_1 < 2 \text{ and } x_2 < x_1; \\ x_2 - 2 & \text{if } x_2 < 2 \text{ and } x_2 - 1 \leq x_1 \leq x_2; \\ +\infty & \text{otherwise.} \end{cases}$$

Deterministic value function in ℓ_i Finally, we compute the deterministic value function of ℓ_i by remarking that

$$\mathsf{dVal}\big(\ell_i, x_1, x_2\big) = \min\Big(\mathsf{dVal}\big(\ell_i \to \ell_3, x_1, x_2\big), \mathsf{dVal}\big(\ell_i \to \ell_4, x_1, x_2\big)\Big)\,.$$

To conclude, we distinguish two cases according to the choice of transition by Min.

We suppose that Min *chooses to go to* ℓ_4. Since the clock x_1 is reset by the transition between ℓ_i and ℓ_4, the deterministic value is finite even if ℓ_4 is reached on the border $x_1 = 0$ of the green cell. In particular, after applying the delay, we need to guarantee that $x_2 \leq 1$. In this case, we have

$$\mathsf{dVal}\big(\ell_i \to \ell_4, x_1, x_2\big) = \inf_{\substack{t \\ 1 < x_1 + t < 2 \ \wedge \ x_2 + t < 1}} \Big[1 \times t + \mathsf{dVal}(\ell_4, x_1 + t, x_2 + t)\Big] + 1\,.$$

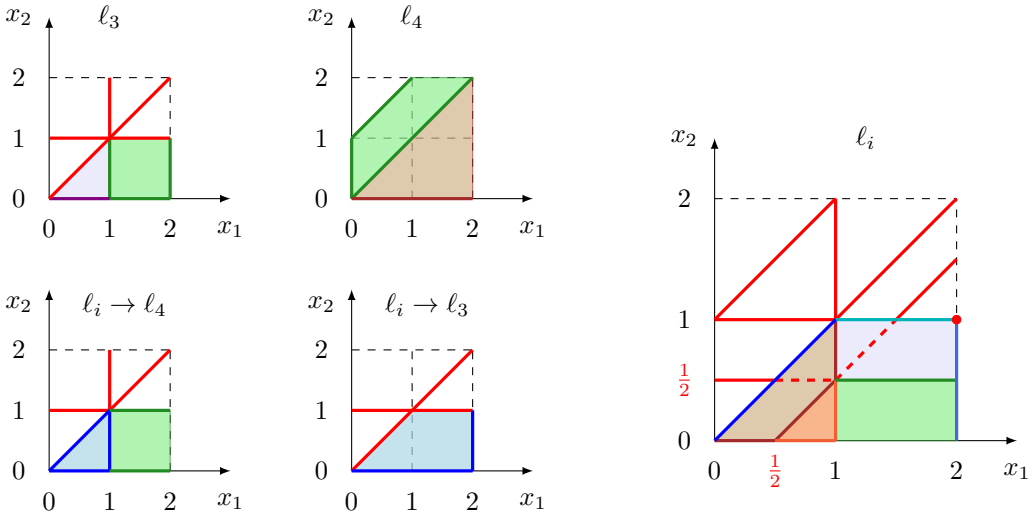


Figure 10.3 – Cells used to encode the deterministic value functions computed for all locations of the acyclic WTG depicted in Figure Figure 10.2. The border of cells with an infinite value is depicted in red. Moreover, we colour cells where the deterministic value function is finite.

We note that Min wants to minimise the time spent in ℓ_0 since its weight is positive, and the deterministic value function of ℓ_4 increases when x_2 increases. Thus, Min waits until the valuation of x_1 is greater than 1, and directly leaves the location.

- We suppose that $x_1 \leq 1$, and we apply $1 - x_1$ as delay. In this case, to guarantee that the valuation of x_2 is less than 1 in ℓ_4, we need to suppose that $x_2 \leq x_1$, and the deterministic value function is equal to $1 \times (1 - x_1) + 1 + (x_2 + 1 - x_1) - 2 = 1 - 2\,x_1 + x_2$.
- We suppose that $1 < x_1$ and Min can directly leave the location. Thus, to guarantee that ℓ_4 is reached with a valuation less than 1 for x_2, we must suppose that $x_2 \leq 1$, and the deterministic value function is equal to $1 + x_2 - 2 = x_2 - 1$.

In particular, for all valuations of $[0, 1]$, we obtain

$$\mathsf{dVal}\big(\ell_i \to \ell_4, x_1, x_2\big) = \begin{cases} 1 - 2\,x_1 + x_2 & \text{if } x_1 \leq 1 \text{ and } x_2 \leq x_1; \\ x_2 - 1 & \text{if } 1 < x_1 < 2 \text{ and } x_2 \leq 1; \\ +\infty & \text{otherwise.} \end{cases}$$

We suppose that Min *chooses to go to* ℓ_3. By studying the guard of the transition between ℓ_i and ℓ_3, Min guarantees a finite deterministic value only if ℓ_3 is reached

in the green cell, i.e. when $x_2 < 1$. Again the deterministic value in ℓ_3 increases when x_2 increases. Thus, Min wants to minimise the time spent in ℓ_i: Min waits until the valuation of x_1 is greater than 1 before leaving the location.

- We suppose that $x_1 < 1$, and we use $1 - x_1$ as delay when $x_2 \leq x_1$ to guarantee that ℓ_3 is reached in the green cell. In this case, the deterministic value function is equal to $(1 - x_1) - (x_2 + 1 - x_1) = -x_2$.
- We suppose that $1 \leq x_1$, and Min directly leaves the location. In particular, we suppose that $x_2 \leq 1$ to guarantee that ℓ_3 is reached in the green cell, and the deterministic value function is equal to $-x_2$.

Thus, for all valuations in $[0, 2]^2$, we have

$$\mathsf{dVal}\big(\ell_i \to \ell_3, x_1, x_2\big) = \begin{cases} -x_2 & \text{if } x_1 \leq 2 \text{ and } x_2 < \min(x_1, 1) \\ +\infty & \text{otherwise} \end{cases}$$

Computation of the minimum. To compute the minimum between these two deterministic value functions; we distinguish five cases according to the considered cells from the analysis of the choice to ℓ_4 and ℓ_3.

- We suppose that $x_1 \leq 1$ and $x_2 = x_1$. In this case, the deterministic value function corresponding to the choice of ℓ_3 is equal to $+\infty$. Thus, the deterministic value function is equal to $\mathsf{dVal}\big(\ell_i \to \ell_4, x_1, x_2\big) = 1 - 2\,x_1 + x_2$.
- We suppose that $1 \leq x_1 < 2$ and $x_2 = 1$. Again, the deterministic value function corresponding to the choice of ℓ_3 is equal to $+\infty$. Thus, the deterministic value function is equal to $\mathsf{dVal}\big(\ell_i \to \ell_4, x_1, x_2\big) = x_2 - 1$.
- We suppose that $x_1 = 2$. Then, the deterministic value function induced by the choice to ℓ_4 is infinite and we deduce that for all valuations $0 \leq x_2 < 1$, $\mathsf{dVal}\big(\ell_i, 2, x_2\big) = -x_2$.
- We suppose that $x_1 \leq 1$ and $x_2 < x_1$, and both deterministic value functions are finite. In particular, we ask when $\mathsf{dVal}\big(\ell_i \to \ell_3, x_1, x_2\big) \leq \mathsf{dVal}\big(\ell_i \to \ell_4, x_1, x_2\big)$, i.e. when $x_2 \leq 1 - 2\,x_1 + x_2$. Thus, we deduce that

$$\mathsf{dVal}\big(\ell_i, x_1, x_2\big) = \begin{cases} \mathsf{dVal}\big(\ell_i \to \ell_3, x_1, x_2\big) = -x_2 & \text{if } x_1 \leq x_2 + \frac{1}{2}; \\ \mathsf{dVal}\big(\ell_i \to \ell_4, x_1, x_2\big) = 1 - 2\,x_1 + x_2 & \text{otherwise.} \end{cases}$$

- We suppose that $x_1 > 1$ and $x_2 < 1$, and both deterministic value functions are finite. In particular, we ask when $\mathsf{dVal}\big(\ell_i \to \ell_3, x_1, x_2\big) \leq \mathsf{dVal}\big(\ell_i \to$

$\ell_4, x_1, x_2\big)$, i.e. when $x_2 \leq x_2 - 1$. Thus, we deduce that

$$\mathsf{dVal}\big(\ell_i, x_1, x_2\big) = \begin{cases} \mathsf{dVal}\big(\ell_i \to \ell_3, x_1, x_2\big) = -x_2 & \text{if } \frac{1}{2} \leq y; \\ \mathsf{dVal}\big(\ell_i \to \ell_4, x_1, x_2\big) = x_2 - 1 & \text{otherwise.} \end{cases}$$

Finally, for all valuations in $[0,1]^2$, we deduce that

$$\mathsf{dVal}\big(\ell_i, x_1, x_2\big) = \begin{cases} -x_2 & \text{if } \begin{cases} x_1 \leq x_2 + \frac{1}{2} \text{ and } x_2 < x_1 \leq 1; \text{ or} \\ 1 < x_1 < 2 \text{ and } \frac{1}{2} < x_2 < 1; \text{ or} \\ x_1 = 2 \text{ and } 0 \leq x_2 < 1; \end{cases} \\ 1 - 2\,x_1 + x_2 & \text{if } \begin{cases} x_2 + \frac{1}{2} \leq x_1 \leq 1 \text{ and } x_2 < x_1; \text{ or} \\ x_1 \leq 1 \text{ and } x_2 = x_1; \end{cases} \\ x_2 - 1 & \text{if } \begin{cases} 1 < x_1 < 2 \text{ and } x_2 \leq \frac{1}{2}; \text{ or} \\ 1 < x_1 < 2 \text{ and } x_2 = 1; \end{cases} \\ +\infty & \text{otherwise.} \end{cases}$$

10.1.3. Encoding operations of value iteration

To implement the computation of $\mathcal{F}(\mathbb{V})$ (defined in Chapter 2), we define intermediate operations over deterministic value functions. Let $\ell \in L$ be a location and $\nu \in [0, \mathsf{M})^n$ be a valuation, we consider the following operations over $\mathbb{V}_\ell$ applying to ν.

- If $Y \subseteq \mathcal{X}$ is a set of clocks, then we define $\mathsf{Unreset}_Y(\mathbb{V}_\ell)\colon [0, \mathsf{M})^n \to \mathbb{R}_\infty$ the piecewise affine function such that

$$\mathsf{Unreset}_Y(\mathbb{V}_\ell)(\nu) = \mathbb{V}_\ell(\nu[Y := 0])\,.$$

- If g is a guard over $\mathcal{X}$, then we define $\mathsf{Guard}_g(\mathbb{V}_\ell)\colon [0, \mathsf{M})^n \to \mathbb{R}_\infty$ the piecewise affine function such that

$$\mathsf{Guard}_g(\mathbb{V}_\ell)(\nu) = \begin{cases} \mathbb{V}_\ell(\nu) & \text{if } \nu \models g \\ -\infty & \text{if } \nu \not\models g \wedge \ell \in L_{\mathsf{Max}} \\ +\infty & \text{if } \nu \not\models g \wedge \ell \in L_{\mathsf{Min}} \end{cases}$$

- If $\ell' \in L$ is a location, then we define $\mathsf{Pre}_{\ell'}(\mathbb{V}_\ell)\colon [0, \mathsf{M})^n \to \mathbb{R}_\infty$ the piecewise

affine function such that, for all valuations $\nu \in [0, \mathsf{M})^n$,

$$\mathsf{Pre}_{\ell'}(\mathbb{V}_\ell)(\nu) = \begin{cases} \sup_{t \le t_\nu}[t\,\mathsf{wt}(\ell') + \mathbb{V}_\ell(\nu + t)] & \text{if } \ell' \in L_{\mathsf{Max}} \\ \inf_{t \le t_\nu}[t\,\mathsf{wt}(\ell') + \mathbb{V}_\ell(\nu + t)] & \text{if } \ell' \in L_{\mathsf{Min}} \end{cases}$$

where $t_\nu = \sup\{t \mid \nu + t \in [0, \mathsf{M})^n\}$ be the greatest possible delay from ν (for all valuations $\nu \in [0, \mathsf{M})^n$).

From these intermediate operations and the encoding of piecewise affine functions, it was shown in [Bus19, Theorem 10.1] that the deterministic value function of $\mathcal{G}$ is computable (in doubly exponential time when the nested partition is used). Our goal is to extend this result when we consider robust semantics in acyclic WTGs by modifying the encoding of piecewise affine functions. Moreover, we need to add a new intermediate operation to model the perturbation of Max.

10.2. Example of computation of robust value functions

We consider the acyclic WTG depicted in Figure 10.2 with two clocks $\mathcal{X} = \{x_1, x_2\}$ under the translated conservative semantic $\mathsf{cons}_{\mathfrak{p}}$ with a parameter $\mathfrak{p} > 0$ and we compute the fixed-parameter robust value function for all valuations in $[0, 2]^2$. To compute this value in ℓ_i, we adapt the computation of the deterministic value function given in Section 10.1.2. All cells used throughout this computation are depicted in Figure 10.4 (we note that the cells are depicted for a fixed $\mathfrak{p}$, otherwise another dimension is needed).

Conservative fixed-parameter robust value function in ℓ_1 Intuitively by the positivity of the weight in ℓ_1, Min wants to play with the minimal delay, i.e. 0 (that is possible since the semantics is supposed translated), whereas Max wants to maximise its perturbation, i.e. $2\,\mathfrak{p}$. Thus, we have

$$\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}\big(\ell_1, x_1, x_2\big) = \inf_t \sup_{p \in [0, 2\mathfrak{p}]} [(t + p) \times 1] = 2\,\mathfrak{p}\,.$$

Conservative fixed-parameter robust value function in ℓ_2 Again, since the weight of ℓ_2 is positive, Min wants to minimise the time spent in this location whereas Max wants to maximise the perturbation, i.e. $2\,\mathfrak{p}$. In particular, from the guard, we remark that if $x_2 \ge 1$, Min can directly leave the location and the conservative fixed-parameter robust value function is equal to $1 + (0 + 2\,\mathfrak{p}) \times 1 + 2\mathfrak{p} = 1 + 4\,\mathfrak{p}$; otherwise, if $x_2 < 1$, Min must wait until the valuation of x_2 is equal to

1 (since the semantics can not negatively perturb the delay, 1 suffice) and the conservative fixed-parameter robust value function is equal to $1 + (1 - x_2 + 2\,\mathfrak{p}) \times 1 + 2\mathfrak{p} = 2 + 4\mathfrak{p} - x_2$. To summarize, for all valuations in $[0, 2]^2$, we have

$$\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_\mathfrak{p}}\big(\ell_2, x_1, x_2\big) = \begin{cases} 2 + 4\,\mathfrak{p} - x_2 & \text{if } x_2 < 1; \\ 1 + 4\,\mathfrak{p} & \text{if } 1 \leq x_2 \leq 2 - 2\,\mathfrak{p}; \\ +\infty & \text{otherwise.} \end{cases}$$

Conservative fixed-parameter robust value function in ℓ_3 Since robust semantics do not perturb the decisions of Max, the analysis remains the same (only the value function may change). In particular, we recall that ℓ_2 can be reached only on the projection of the brown cell on the affine equation $x_1 = 0$ and the conservative fixed-parameter robust value function is equal to

$$\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_\mathfrak{p}}\big(\ell_3, x_1, x_2\big) = \sup_{\substack{t \\ 1<x_1+t<2 \,\wedge\, x_2+t<1}} \mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_\mathfrak{p}}(\ell_2, 0, x_2 + t) - 2\,.$$

Moreover, Max wants to play a delay such that x_1 is close enough to 1 when the transition is applied. In particular, we distinguish two cases according to the valuation of x_1.

- We suppose that $1 < x_1$, and Max can directly leave the location to obtain the weight $2 + 4\,\mathfrak{p} - x_2$ in ℓ_2. In particular, when $x_2 < 1$, we deduce that the conservative fixed-parameter robust value function is equal to $4\,\mathfrak{p} - x_2$.
- Otherwise, we suppose that $x_1 \leq 1$ and Max wants to play the delay $1 - x_1$. We note that this decision is feasible only when $x_2 < x_1$. In this case, the fixed-parameter robust value function is equal to $-2+2+4\mathfrak{p}-(x_2+1-x_1) = 4\mathfrak{p} + x_1 - x_1 - 1$.

Thus, for all valuations in $[0, 2]^2$, we conclude that

$$\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_\mathfrak{p}}\big(\ell_3, x_1, x_2\big) = \begin{cases} 4\mathfrak{p} + x_1 - x_2 - 1 & \text{if } x_1 \leq 1 \text{ and } x_2 < x_1; \\ 4\,\mathfrak{p} - x_2 & \text{if } 1 < x_1 < 2 \text{ and } x_2 < 1; \\ +\infty & \text{otherwise.} \end{cases}$$

We note that the cells involved in this case are identical to ones used for the deterministic value function. In particular, these cells are depicted in Figure 10.3. However, both value functions are not equal.

Conservative fixed-parameter robust value function in ℓ_4 By negativity of the weight of ℓ_4, Max wants to minimise the perturbation, whereas Min wants to

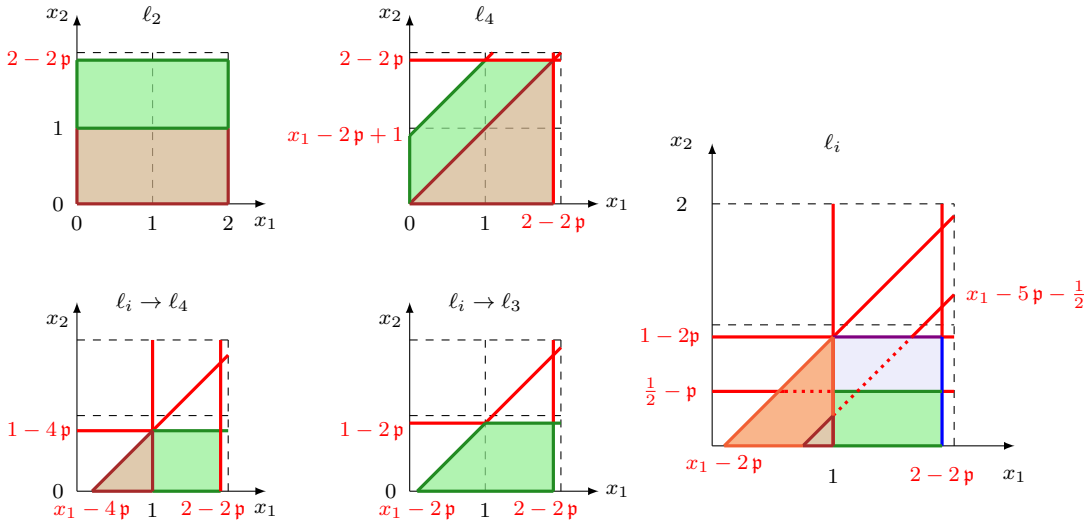


Figure 10.4 – Cells for the conservative fixed-parameter robust value functions computed for all locations of the acyclic WTG depicted in Figure 10.2 when $\mathfrak{p} \leq 1/18$. The cells for the location ℓ_3 are identical to cells for the deterministic value function depicted in Figure 10.3.

maximise the time spent in ℓ_4, i.e. Min waits until a valuation equal to $2-2\mathfrak{p}$ (since the guard must be satisfied for all perturbations, Min can not wait longer). Again, we distinguish two cases according to the order over x_1 and x_2.

- We suppose that $x_2 \leq x_1 \leq 2-2\mathfrak{p}$ and $2-2\mathfrak{p}-x_1$ is a robust delay for Min. Moreover, since Max does not perturb the delay chosen by Min, the conservative fixed-parameter robust value is equal to $-(2-2\mathfrak{p}-x_1)$.
- We suppose that $x_1 < x_2 \leq 2-2\mathfrak{p}$ and, symmetrically, $2-2\mathfrak{p}-x_2$ is a robust delay for Min when $x_2+2\mathfrak{p}-1 \leq x_1 \leq x_2$. Moreover, since Max does not perturb the delay chosen by Min, the conservative fixed-parameter robust value is equal to $-(2-2\mathfrak{p}-x_2)$.

Thus, for all valuations in $[0,2]^2$, we have

$$\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}\big(\ell_4,x_1,x_2\big) = \begin{cases} -(2-2\mathfrak{p}-x_1) & \text{if } x_2 \leq x_1 < 2-2\mathfrak{p}; \\ -(2-2\mathfrak{p}-x_2) & \text{if } x_2+2\mathfrak{p}-1 \leq x_1 < x_2 < 2-2\mathfrak{p}; \\ +\infty & \text{otherwise.} \end{cases}$$

We note that $x_2+2\mathfrak{p}-1 \leq 2$ (i.e. the green cell is not empty) if and only if $\mathfrak{p} \leq 1/2$. For the rest of the computation, we suppose that $\mathfrak{p} \leq 1/2$.

Conservative fixed-parameter robust value function in ℓ_i Finally, we compute the conservative fixed-parameter robust value function of ℓ_i by remarking that

$$\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_\mathfrak{p}}\left(\ell_i, x_1, x_2\right) = \min\left(\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_\mathfrak{p}}\left(\ell_i \to \ell_3, x_1, x_2\right), \mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_\mathfrak{p}}\left(\ell_i \to \ell_4, x_1, x_2\right)\right).$$

To conclude, we distinguish two cases according to the choice of transition by Min.

We suppose that Min *chooses to go to* ℓ_4. Since the clock x_1 is reset by the transition between ℓ_i and ℓ_4 only the green cell on its border $x1 = 0$ can be reached to obtain a finite value function, i.e. ℓ_4 is reached when $x_2 \leq 1 - 2\,\mathfrak{p}$. Now, since the weight of ℓ_i is positive and the conservative fixed-parameter robust value function increases when x_2 increases, Max wants to maximise its perturbation, i.e. $2\,\mathfrak{p}$, whereas Min wants to minimise its delay, i.e. directly leave if $x_1 \geq 1$, or wait until 1 for the valuation of x_1.

- We suppose that $1 < x_1 < 2 - 2\,\mathfrak{p}$ and Min plays the delay 0 when $x_2 \leq 1 - 4\,\mathfrak{p}$. Moreover, the obtained weight is $1 \times 2\mathfrak{p} + 1 - (2 - 2\mathfrak{p} - (x_2 + 2\mathfrak{p})) = x_2 + 6\,\mathfrak{p} - 1$.
- We suppose that $0 \leq x_1 \leq 1$ and Min plays the minimal delay $1 - x_1$ when $x_2 \leq x_1 - 4\,\mathfrak{p}$. In particular, the obtained weight is $1 \times (1 - x_1 + 2\,\mathfrak{p}) + 1 - (2 - 2\mathfrak{p} - (x_2 + 1 + 2\mathfrak{p} - x_1)) = 1 + x_2 + 6\mathfrak{p} - 2\,x_1$.

Thus, for all valuations in $[0, 2]^2$, $\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_\mathfrak{p}}\left(\ell_i \to \ell_4, x_1, x_2\right)$ is equal to

$$\begin{cases} x_2 + 1 + 6\,\mathfrak{p} - 2\,x_1 & \text{if } x_1 \leq 1 \text{ and } x_2 \leq x_1 - 4\,\mathfrak{p}; \\ x_2 - 1 + 6\,\mathfrak{p} & \text{if } 1 < x_1 < 2 - 2\,\mathfrak{p} \text{ and } y \leq 1 - 4\,\mathfrak{p}; \\ +\infty & \text{otherwise.} \end{cases}$$

We suppose that Min *chooses to go to* ℓ_3. According to the guard between ℓ_i and ℓ_3, we know that ℓ_3 is reached with the green cell, i.e. in ℓ_3 (after the perturbation), the valuations of clocks are such that $x_1 \leq 2$ and $x_2 \leq 1$. Again, Min wants to minimise its delay and Max perturbs with $2\,\mathfrak{p}$. Thus, from the guard, we consider two cases according to the valuation of x_1.

- We suppose that $1 \leq x_1 \leq 2 - 2\mathfrak{p}$ and Min plays with the delay 0 when $x_2 \leq 1 - 2\,\mathfrak{p}$. In particular, the conservative fixed-parameter robust value function is equal to $1 \times 2\,\mathfrak{p} + 4\mathfrak{p} - (x_2 + 2\,\mathfrak{p}) = 4\,\mathfrak{p} - x_2$.
- We suppose that $x_1 < 1$ and Min plays with the delay $1 - x_1$ when $x_2 \leq x_1 - 2\,\mathfrak{p}$. Moreover, the conservative fixed-parameter robust value function is equal to $1 \times (1 - x_1 + 2\,\mathfrak{p}) + 4\mathfrak{p} - (x_2 + 1 + 2\mathfrak{p} - x_1) = 4\,\mathfrak{p} - x_2$.

Thus, for all valuations in $[0,2]^2$, $\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}\big(\ell_i \to \ell_3, x_1, x_2\big)$ is equal to

$$\begin{cases} 4\mathfrak{p} - x_2 & \text{if } x_1 \leq 2 - 2\mathfrak{p} \text{ and } x_2 \leq \min(x_1 - 2\,\mathfrak{p}, 1 - 2\,\mathfrak{p}); \\ +\infty & \text{otherwise.} \end{cases}$$

Computation of the minimum. To conclude, we take the minimum between these two value functions: we look for the minimum according to valuations of clocks.

- We suppose that $x_1 \leq 1$ and $x_1 - 4\,\mathfrak{p} \leq x_2 < x_1 - 2\,\mathfrak{p}$. In this case, only $\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}\big(\ell_i \to \ell_3, x_1, x_2\big)$ is finite, i.e. $\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}\big(\ell_i, x_1, x_2\big) = 4\,\mathfrak{p} - x_2$.
- We suppose that $1 \leq x_1 \leq 2 - 2\,\mathfrak{p}$ and $1 - 4\,\mathfrak{p} < x_2 \leq 1 - 2\mathfrak{p}$. Again, only $\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}\big(\ell_i \to \ell_3, x_1, x_2\big)$ is finite, i.e. $\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}\big(\ell_i, x_1, x_2\big) = 4\,\mathfrak{p} - x_2$.
- We suppose that $x_1 = 2 - 2\,\mathfrak{p}$ and $x_2 \leq 1 - 2\mathfrak{p}$ and $\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}\big(\ell_i, x_1, x_2\big) = 4\,\mathfrak{p} - x_2$ by choosing ℓ_3.
- We suppose that $x_1 \leq 1$ and $x_2 \leq x_1 - 4\,\mathfrak{p}$. In this case, both conservative fixed-parameter robust value functions are finite. Thus, we want to decide when $\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}\big(\ell_i \to \ell_4, x_1, x_2\big) \leq \mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}\big(\ell_i \to \ell_3, x_1, x_2\big)$, i.e. when $x_2 + 1 - 6\,\mathfrak{p} - 2\,x_1 \leq 4\,\mathfrak{p} - x_2$. As this inequality holds when $x_2 \leq x_1 + 5\,\mathfrak{p} - 1/2$, we suppose that $\mathfrak{p} \leq 1/18$ to guarantee that the cell defined by $x_1 + 5\,\mathfrak{p} - 1/2 \leq x_1 - 4\,\mathfrak{p}$ is not empty. Thus, in this case, we obtain that $\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}\big(\ell_i, x_1, x_2\big)$ is equal to

$$\begin{cases} x_2 + 1 + 6\,\mathfrak{p} - 2\,x_1 & \text{if } x_2 \leq x_1 + 5\,\mathfrak{p} - \frac{1}{2}; \\ 4\,\mathfrak{p} - x_2 & \text{if } x_1 + 5\,\mathfrak{p} - \frac{1}{2} \leq x_2 \leq x_1 - 4\,\mathfrak{p}; \\ +\infty & \text{otherwise.} \end{cases}$$

- We suppose that $1 < x_1 < 2 - 2\mathfrak{p}$ and $x_2 \leq x_1 - 4\,\mathfrak{p}$. In this case both conservative fixed-parameter robust value functions are finite and we want to decide when $x_2 - 1 + 6\,\mathfrak{p} \leq 4\,\mathfrak{p} - x_2$. Since this inequality holds when $x_2 \leq 1/2 - \mathfrak{p}$ that is possible when $\mathfrak{p} \leq 1/2$ (that is the case by hypothesis). Thus, in this case, we obtain that

$$\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_{\mathfrak{p}}}\big(\ell_i, x_1, x_2\big) = \begin{cases} x_2 - 1 + 6\,\mathfrak{p} & \text{if } x_2 \leq \frac{1}{2} - \mathfrak{p}; \\ 4\,\mathfrak{p} - x_2 & \text{if } \frac{1}{2} - \mathfrak{p} < x_2 \leq x_1 - 4\,\mathfrak{p}; \\ +\infty & \text{otherwise.} \end{cases}$$

Thus, for all valuations in $[0,2]^2$, $\mathsf{rVal}^{\mathfrak{p},\mathsf{cons}_\mathfrak{p}}(\ell_i, x_1, x_2)$ is equal to

$$\begin{cases} x_2 - 1 + 6\,\mathfrak{p} & \text{if } \begin{cases} x_1 \leq 1 \text{ and } x_2 \leq x_1 - 5\,\mathfrak{p} - \frac{1}{2}; \\ 1 < x_1 < 2 - 2\,\mathfrak{p} \text{ and } x_2 \leq \frac{1}{2} - \mathfrak{p}; \end{cases} \\ 4\,\mathfrak{p} - x_2 & \text{if } \begin{cases} x_1 \leq 1 \text{ and } x_1 - 5\,\mathfrak{p} - \frac{1}{2} < x_2 \leq x_1 - 2\,\mathfrak{p}; \\ 1 < x_1 < 2 - 2\,\mathfrak{p} \text{ and } \frac{1}{2} - \mathfrak{p} < x_2 \leq 1 - 2\mathfrak{p}; \\ x_1 = 2 - 2\,\mathfrak{p} \text{ and } x_2 \leq 1 - 2\,\mathfrak{p}; \end{cases} \\ +\infty & \text{otherwise.} \end{cases}$$

10.3. Limit robust values in an acyclic weighted timed game

We present our algorithm to compute the limit robust value in an acyclic WTG under conservative semantics and the translated excessive semantics (that is not equivalent to other excessive semantics, see Section 9.1.1 for details). In particular, we adapt the technique to compute the deterministic value function described in Section 10.1. The main idea is the modification of the data structure used to represent value functions through the computation: we will use parametric (in $\mathfrak{p}$) piecewise affine on $\mathbb{R}^{\mathcal{X}}_{\geq 0}$ and an upper bound constraint over $\mathfrak{p}$.

Theorem 10.4. *Let $\mathcal{G}$ be an acyclic WTG under a class of translated conservative (resp. excessive) semantics $(\mathsf{rob}_\mathfrak{p})_{\mathfrak{p}>0}$. Then, we can compute a parameter $\mathfrak{p} > 0$ and the mapping $\mathfrak{p}' \mapsto \mathsf{rVal}^{\mathfrak{p}',\mathsf{rob}_{\mathfrak{p}'}}$ for all $\mathfrak{p}' \leq \mathfrak{p}$.*

In the rest of this section, we will prove this theorem. Our idea is to compute iteratively $\mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}}$ while maintaining some constraints on $\mathfrak{p}$ with a robust operator analogous to $\mathcal{F}$ defined in Chapter 2.

10.3.1. Robust operator for translated robust semantics

We extend $\mathcal{F}$ to take account of perturbations of Max under robust semantics. Let $\mathsf{rob}_\mathfrak{p}$ be a translated robust semantics of parameter $\mathfrak{p} > 0$ and $X : L \times \mathbb{R}^{\mathcal{X}}_{\geq 0} \to \mathbb{R}_\infty + \mathfrak{p}\,\mathbb{Q}$ be a mapping. We define the following operator over $\mathsf{rob}_\mathfrak{p}$ such that, for

all configurations (ℓ,ν), $\mathcal{F}_{\mathcal{G}}^{\mathsf{rob}_{\mathfrak{p}}}(X)(\ell,\nu)$ is equal to

$$\begin{cases} 0 & \text{if } \ell \in L_T; \\ \sup\limits_{(\ell,\nu)\xrightarrow{t,\delta}(\ell',\nu')} \left[t\,\mathsf{wt}(\ell)+\mathsf{wt}(\delta)+X(\ell',\nu')\right] & \text{if } \ell \in L_{\mathsf{Max}}; \\ \inf\limits_{(\ell,\nu)\xrightarrow{\delta,t}(\ell,\nu,\delta,t)} \sup\limits_{\substack{p\in[0,2\mathfrak{p}] \\ (\ell,\nu,\delta,t)\xrightarrow{\delta,p}(\ell,\nu')}} \left[(t+p)\,\mathsf{wt}(\ell)+\mathsf{wt}(\delta)+X(\ell',\nu')\right] & \text{if } \ell \in L_{\mathsf{Min}}; \end{cases}$$

where $(\ell,\nu)\xrightarrow{\delta,t}(\ell,\nu,\delta,t)$ is an edge of $[\![\mathcal{G}]\!]^{\mathsf{rob}_{\mathfrak{p}}}$. Moreover, by letting $\mathbb{V}_0$ be the affine function defined such that $\mathbb{V}_0(\ell,\nu)=0$ if $\ell\in L_T$ and $\mathbb{V}_0(\ell,\nu)=+\infty$ otherwise, we prove that this operator is useful to compute the fixed-parameter robust value in acyclic WTG.

Proposition 10.5. *Let $\mathcal{G}$ be an acyclic WTG under a translated robust semantics $\mathsf{rob}_{\mathfrak{p}}$ of parameter $\mathfrak{p}>0$. Then, there exists $i\in\mathbb{N}$ such that we have*

$$\mathcal{F}_{\mathcal{G}}^{\mathsf{rob}_{\mathfrak{p}}}(\mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}) = \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}} = \left(\mathcal{F}_{\mathcal{G}}^{\mathsf{rob}_{\mathfrak{p}}}\right)^{i}(\mathbb{V}_0)\,.$$

Proof. We introduce the robust value at horizon i, i.e. the robust value computed only on plays on length at most i. Formally, given two robust strategies $\chi\in\mathsf{rStrat}_{\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}}}$ and $\zeta\in\mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}}}$, we define $\mathsf{Play}^i((\ell,\nu),\chi,\zeta)$ be the maximal play conforming to χ and ζ with a length at most[1] i from (ℓ,ν). In particular, we denote by $\mathsf{rVal}_i^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}$ the *fixed-parameter robust value at horizon i*, defined by

$$\mathsf{rVal}_i^{\mathfrak{p},\mathsf{rob}_{\mathfrak{p}}}(\ell,\nu) = \inf_{\chi\in\mathsf{rStrat}_{\mathsf{Min}}^{\mathsf{rob}_{\mathfrak{p}}}} \sup_{\zeta\in\mathsf{rStrat}_{\mathsf{Max}}^{\mathsf{rob}_{\mathfrak{p}}}} \mathbf{P}(\mathsf{Play}^i((\ell,\nu),\chi,\zeta))\,.$$

Let $i\in\mathbb{N}$ and (ℓ,ν) be a configuration. By grouping all infimum/supremum together[2] in $(\mathcal{F}_{\mathcal{G}}^{\mathsf{rob}_{\mathfrak{p}}})^i(\mathbb{V})(\ell,\nu)$, it can be rewritten as

$$\inf_{\left(f_k\colon((\delta_0,t_0),\dots,(\delta_{k-1},t_{k-1}))\mapsto(\delta_k,t_k)\right)_{\substack{0\le k\le i \\ s_k\in S_{\mathsf{Min}}}}} \sup_{\left(f_k\colon((\delta_0,t_0),\dots,(\delta_{k-1},t_{k-1}))\mapsto(\delta_k,t_k)\right)_{\substack{0\le i\le i \\ s_k\in S_{\mathsf{Max}}}}} \mathsf{wt}_\Sigma(\rho)$$

where ρ is the finite play $s_0 \xrightarrow{(\delta_0,t_0)=f_0} s_1 \xrightarrow{(\delta_1,t_1)=f_1(\delta_0,t_0)} \cdots \xrightarrow{(\delta_i,t_i)=f_i((\delta_0,t_0),\dots,(\delta_{i-1},t_{i-1}))} s_i$. Since the operator $\mathcal{F}^{\mathsf{rob}_{\mathfrak{p}}}$ only used robust decisions under $\mathsf{rob}_{\mathfrak{p}}$ for both players, we notice that the mapping f_k, chosen by the player owning state s_k, describes the

[1] If the play reaches a target location within j steps ($j<i$), we consider that it is an outcome of both strategies within i steps.

[2] In the same way as in the proof of Lemma 4.10-2 about the deterministic value of a path.

robust decision under $\mathsf{rob}_\mathfrak{p}$ for the k-th step as a function of the previously chosen (robust) decisions. In particular, for all $i \in \mathbb{N}$ and configurations (ℓ, ν), we have

$$\mathsf{rVal}_i^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}}(\ell,\nu) = (\mathcal{F}_\mathcal{G}^{\mathsf{rob}_\mathfrak{p}})^i(\mathbb{V})(\ell,\nu)\,. \tag{10.1}$$

Finally, to conclude the proof of Proposition 10.5, we remark that the hypothesis of acyclicity of $\mathcal{G}$ guarantees that there exists $i \in \mathbb{N}$ such that $\mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}} = \mathsf{rVal}_i^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}}$. Indeed, even if (10.1) holds for all WTGs, the convergence in finite time is not guaranteed (see Example I.1 for the deterministic value).

By letting $i = |\mathcal{G}|$, since $\mathcal{G}$ is acyclic, all plays in $\mathcal{G}$ have a length at most i (otherwise, they contain a cycle), and we remark that $\mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}}(\ell,\nu) = \mathsf{rVal}_i^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}}(\ell,\nu)$. In particular, by (10.1), we have $\mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}}(\ell,\nu) = (\mathcal{F}_\mathcal{G}^{\mathsf{rob}_\mathfrak{p}})^i(\mathbb{V}_0)(\ell,\nu)$. To conclude, we note that $\mathcal{F}_\mathcal{G}^{\mathsf{rob}_\mathfrak{p}}(\mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}})(\ell,\nu) = (\mathcal{F}_\mathcal{G}^{\mathsf{rob}_\mathfrak{p}})^{i+1}(\mathbb{V}_0)(\ell,\nu) = \mathsf{rVal}^{\mathfrak{p},\mathsf{rob}_\mathfrak{p}}(\ell,\nu)$. □

By Proposition 10.5, to compute a fixed-parameter robust value, we need to apply $\mathcal{F}^{\mathsf{rob}_\mathfrak{p}}$ repeatedly over a set of piecewise affine functions. As in the case of the deterministic value function, we split this computation into intermediate operators. In particular, we extend the set of operators used in this previous case: we extend the data structure with a parameter $\mathfrak{p}$ (i.e. operators work on function $[0,\mathsf{M})^n \to \mathbb{R}_\infty + \mathfrak{p}\,\mathbb{Q}$) and we add a new operator such that if $\mathfrak{p}$ is a bound over possible perturbations and $\ell' \in L$ be a location, then we define the piecewise affine function $\mathsf{Perturb}_{\ell'}^\mathfrak{p}(\mathbb{V}_\ell)\colon [0,\mathsf{M})^n \to \mathbb{R}_\infty + \mathfrak{p}\,\mathbb{Q}$, for all valuations $\nu \in [0,\mathsf{M})^n$, by

$$\mathsf{Perturb}_{\ell'}^\mathfrak{p}(\mathbb{V}_\ell)(\nu) = \begin{cases} \mathbb{V}_\ell(\nu) & \text{if } \ell' \in L_{\mathsf{Max}};\\ \sup_{p\in[0,2\mathfrak{p}]}[p\,\mathsf{wt}(\ell') + \mathbb{V}_\ell(\nu+p)] & \text{if } \ell' \in L_{\mathsf{Min}}. \end{cases}$$

Unlike the previous results, we restrict ourselves to conservative and excessive semantics to split an application of $\mathcal{F}^{\mathsf{rob}_\mathfrak{p}}$ into several intermediate operators. Intuitively, according to the definition of excessive and conservative semantics, we have two ways to combine intermediate operators to define $\mathcal{F}^{\mathsf{rob}_\mathfrak{p}}$: $\mathsf{Perturb}_\ell^\mathfrak{p}$ will be placed after or before the operator Guard_g. We begin with the definition of the translated conservative semantics.

Lemma 10.6. *Let (ℓ,ν) be a configuration and $\mathbb{V}_\ell\colon \mathbb{R}_{\geq 0}^\mathcal{X} \to \mathbb{R}_\infty + \mathfrak{p}\,\mathbb{Q}$ be a mapping according to ℓ. Then, $\mathcal{F}^{\mathsf{cons}_\mathfrak{p}}(\mathbb{V}_\ell)\colon \mathbb{R}_{\geq 0}^\mathcal{X} \to \mathbb{R}_\infty + \mathfrak{p}\,\mathbb{Q}$ is equal to*

$$\begin{cases} 0 & \text{if } \ell \in L_T\\ \max_{\delta=(\ell,g,Y,\ell')}\left[\mathsf{wt}(\delta) + \mathsf{Pre}_\ell(\mathsf{Perturb}_\ell^\mathfrak{p}(\mathsf{Guard}_g(\mathsf{Unreset}_Y(\mathbb{V}(\ell')))))\right] & \text{if } \ell \in L_{\mathsf{Max}}\\ \min_{\delta=(\ell,g,Y,\ell')}\left[\mathsf{wt}(\delta) + \mathsf{Pre}_\ell(\mathsf{Perturb}_\ell^\mathfrak{p}(\mathsf{Guard}_g(\mathsf{Unreset}_Y(\mathbb{V}(\ell')))))\right] & \text{if } \ell \in L_{\mathsf{Min}} \end{cases}$$

Proof. We suppose $\ell \in L_T$ and for all valuations ν, $\mathcal{F}^{\mathsf{cons}_\mathfrak{p}}(\mathbb{V}_\ell)(\nu) = 0$.

Now, we suppose that $\ell \in L_{\mathsf{Max}}$ and by decomposing the supremum, we have

$$\begin{aligned}\mathcal{F}^{\mathsf{cons}_{\mathfrak{p}}}(\mathbb{V}_\ell)(\nu) &= \max_{\delta=(\ell,g,Y,\ell')} \big(\mathsf{wt}(\delta) + \sup_{(\ell,\nu)\xrightarrow{t,\delta}(\ell',\nu')} [t\,\mathsf{wt}(\ell) + \mathbb{V}_{\ell'}(\nu')]\big) \\ &= \max_{\delta=(\ell,g,Y,\ell')} \big(\mathsf{wt}(\delta) + \sup_{(\ell,\nu)\xrightarrow{t,\delta}(\ell',\nu')} [t\,\mathsf{wt}(\ell) + \mathsf{Unreset}_Y(\mathbb{V}_{\ell'})(\nu+t)]\big)\end{aligned}$$

since $\nu' = (\nu+t)[Y := 0]$. When we fix a transition, choosing a feasible decision (to define an edge in the semantic) is equivalent to choosing t to be a delay such that $(\nu + t) \models g$. In particular, by letting $t_\nu = \sup\{t \mid \nu + t \in [0, \mathsf{M})^n\}$, we conclude that

$$\begin{aligned}\mathcal{F}^{\mathsf{cons}_{\mathfrak{p}}}(\mathbb{V}_\ell)(\nu) &= \max_{\delta=(\ell,g,Y,\ell')} \big(\mathsf{wt}(\delta) + \sup_{t\le t_\nu}[t\,\mathsf{wt}(\ell) + \mathsf{Guard}_g(\mathsf{Unreset}_Y(\mathbb{V}_{\ell'}))(\nu+t)]\big) \\ &= \max_{\delta=(\ell,g,Y,\ell')} \big[\mathsf{wt}(\delta) + \mathsf{Pre}_\ell(\mathsf{Perturb}^{\mathfrak{p}}_\ell(\mathsf{Guard}_g(\mathsf{Unreset}_Y(\mathbb{V}_{\ell'}))))(\nu)\big]\end{aligned}$$

since $\mathsf{Perturb}^{\mathfrak{p}}_\ell(\mathbb{V}_{\ell'}) = \mathbb{V}_{\ell'}$ (as $\ell \in L_{\mathsf{Max}}$).

Finally, we suppose that $\ell \in L_{\mathsf{Min}}$ and by decomposing the infimum, we obtain that $\mathcal{F}^{\mathsf{cons}_{\mathfrak{p}}}(\mathbb{V}_\ell)(\nu)$ is equal to

$$\min_{\delta=(\ell,g,Y,\ell')} \big(\mathsf{wt}(\delta) + \inf_{(\ell,\nu)\xrightarrow{t,\delta}(\ell,\nu,t,\delta)} \sup_{\substack{p\in[0,2\mathfrak{p}] \\ (\ell,\nu,t,\delta)\xrightarrow{p,\delta}(\ell,\nu')}} [(t+p)\,\mathsf{wt}(\ell) + \mathsf{Unreset}_Y(\mathbb{V}_{\ell'})(\nu+t+p)]$$

since $\nu' = (\nu + t + p)[Y := 0]$. In the translated conservative semantics, an edge exists when the guard of the transition holds for all possible perturbations: we rewrite the infimum and the supremum over all edges by a constraint on the delay and the perturbations that we consider before applying the filter defined by the guard, i.e. $\mathcal{F}^{\mathsf{cons}_{\mathfrak{p}}}(\mathbb{V})_\ell(\nu)$ is equal to

$$\min_{\delta=(\ell,g,Y,\ell')} \big(\mathsf{wt}(\delta) + \inf_{t\le t_\nu} \big(t\,\mathsf{wt}(\ell) + \sup_{p\le 2\mathfrak{p}}[p\,\mathsf{wt}(\ell) + \mathsf{Guard}_g(\mathsf{Unreset}_Y(\mathbb{V}_{\ell'}))(\nu+t+p)]\big)\,.$$

Finally, we conclude by definition of operators Pre_ℓ and $\mathsf{Perturb}^{\mathfrak{p}}_\ell$:

$$\mathcal{F}^{\mathsf{cons}_{\mathfrak{p}}}(\mathbb{V})_\ell(\nu) = \min_{\delta=(\ell,g,Y,\ell')} \big[\mathsf{wt}(\delta) + \mathsf{Pre}_\ell(\mathsf{Perturb}^{\mathfrak{p}}_\ell(\mathsf{Guard}_g(\mathsf{Unreset}_Y(\mathbb{V}_{\ell'}))))(\nu)\big]\ .\ \square$$

By analogous reasoning, we split the operator under the excessive semantics. Since under an excessive semantics, the guard is tested before the guard, we just reverse the order of operators $\mathsf{Perturb}^{\mathfrak{p}}_{\mathfrak{p}}$ and Guard_g.

Lemma 10.7. *Let (ℓ, ν) be a configuration and $\mathbb{V}_\ell \colon \mathbb{R}^{\mathcal{X}}_{\ge 0} \to \mathbb{R}_\infty +_{\mathfrak{p}} \mathbb{Q}$ be a mapping*

according to ℓ. *Then,* $\mathcal{F}^{\mathsf{excess}_{\mathfrak{p}}}(\mathbb{V})_\ell \colon \mathbb{R}^{\mathcal{X}}_{\geq 0} \to \mathbb{R}_\infty + \mathfrak{p}\,\mathbb{Q}$ *is equal to*

$$\begin{cases} 0 & \text{if } \ell \in L_T; \\ \max_{\delta=(\ell,g,Y,\ell')} \left[\mathsf{wt}(\delta) + \mathsf{Pre}_\ell(\mathsf{Guard}_g(\mathsf{Perturb}^{\mathfrak{p}}_{\mathfrak{p}}(\mathsf{Unreset}_Y(\mathbb{V}_{\ell'}))))\right] & \text{if } \ell \in L_{\mathsf{Max}}; \\ \min_{\delta=(\ell,g,Y,\ell')} \left[\mathsf{wt}(\delta) + \mathsf{Pre}_\ell(\mathsf{Guard}_g(\mathsf{Perturb}^{\mathfrak{p}}_{\mathfrak{p}}(\mathsf{Unreset}_Y(\mathbb{V}_{\ell'}))))\right] & \text{if } \ell \in L_{\mathsf{Min}}. \end{cases}$$

Proof. The cases where $\ell \in L_{\mathsf{Max}} \cup L_T$ are equivalent to the proof of Lemma 10.6, so we do not detail them here. We now suppose that $\ell \in L_{\mathsf{Min}}$ and we know that $\mathcal{F}^{\mathsf{excess}_{\mathfrak{p}}}(\mathbb{V}_\ell)(\nu)$ is equal to

$$\min_{\delta=(\ell,g,Y,\ell')} \Big(\mathsf{wt}(\delta) + \inf_{(\ell,\nu)\xrightarrow{t,\delta}(\ell,\nu,t,\delta)} \big(t\mathsf{wt}(\ell) + \sup_{\substack{p\in[0,2\mathfrak{p}] \\ (\ell,\nu,t,\delta)\xrightarrow{p,\delta}(\ell,\nu')}} [t\mathsf{wt}(\ell) + \mathsf{Guard}_g(\mathsf{Unreset}_Y(\mathbb{V}_{\ell'}))(\nu+t+p)]\big)\Big)$$

by applying the same reasoning as for the conservative semantics. Under the translated excessive semantics, an edge exists when the guard of the transition holds before the perturbations, i.e. $\mathcal{F}^{\mathsf{excess}_{\mathfrak{p}}}(\mathbb{V}_\ell)(\nu)$ is equal to

$$\min_{\delta=(\ell,g,Y,\ell')} \big(\mathsf{wt}(\delta) + \inf_{t\leq t_\nu} [t\,\mathsf{wt}(\ell) + \mathsf{Guard}_g(\mathsf{Perturb}^{\mathfrak{p}}_\ell(\mathsf{Unreset}_Y(\mathbb{V}_{\ell'})))(\nu+t+p)]\big)\,.$$

We conclude by definition of the operator Pre_ℓ:

$$\mathcal{F}^{\mathsf{excess}_{\mathfrak{p}}}(\mathbb{V})_\ell(\nu) = \min_{\delta=(\ell,g,Y,\ell')} \left[\mathsf{wt}(\delta) + \mathsf{Pre}_\ell(\mathsf{Guard}_g(\mathsf{Perturb}^{\mathfrak{p}}_\ell(\mathsf{Unreset}_Y(\mathbb{V}_{\ell'}))))\right]\,. \quad \square$$

10.3.2. Operations over piecewise affine functions

To conclude the proof of Theorem 10.4, we need to compute $\mathcal{F}^{\mathsf{rob}_{\mathfrak{p}}}(\mathbb{V})$ where $\mathbb{V}$ is represented by a set of parametric piecewise affine functions. In particular, by Lemmas 10.6 and 10.7, we will check if the computation can be done by each intermediate operator. Moreover, we verify that $\mathcal{F}^{\mathsf{rob}_{\mathfrak{p}}}(\mathbb{V})$ is also representable by a set of parametric piecewise affine functions.

Encoding parametric piecewise affine functions We adapt the representation of piecewise affine functions to take account of the perturbation: we represent a *parametric piecewise affine* function by a tuple $\langle P, F, \eta\rangle$ such that P is the partition of the set of possible valuations (defined by a set of parametric equations $\mathcal{E}_P$), F be a set of affine parametric functions, and η be an upper bound over $\mathfrak{p}$. In particular, we adapt the encoding of affine equations and affine functions: we

consider *parametric affine equations* of the form

$$0 = \sum_{i=0}^{n} a_i \, x_i + b + c\mathfrak{p}$$

where $a_i \in \mathbb{Z}$ and $b, c \in \mathbb{Q}$, and *parametric affine functions* of the form

$$y = \sum_{i=0}^{n} \alpha_i \, x_i + \beta + \gamma\mathfrak{p}$$

where $\alpha_i \in \mathbb{Z}$, $\beta, \gamma \in \mathbb{Q}$.

From a set of parametric affine equations, the definition of the partition of $[0, \mathsf{M})^n$ is analogous to the one with classical cells. Since each affine equation induces three cells, the partition is generated by the intersections of these cells. However, given two cells parametrized by $\mathfrak{p}$, its intersection may not exist for all $\mathfrak{p}$. Hence, we want to compute an upper bound over $\mathfrak{p}$, denoted η, to guarantee the existence of a maximal number of cells. In particular, we focus on the evolution of η when we add a new parametric affine equation into the set: we compute a new upper bound to guarantee the existence of cells defined by this new equation intersected with already defined cells.

Lemma 10.8. *Let (P, η) be a partition induced by the set of parametric affine equations $\mathcal{E}$ under an upper bound η on $\mathfrak{p}$, and $E = \sum_{i=1}^{n} a_i \, x_i + b + c\,\mathfrak{p}$ a new parametric affine equation. Then, there exists η' such that there exists a new partition (P', η') under an upper bound η' on $\mathfrak{p}$.*

Proof. Intuitively, the proof consists in finding an upper bound on $\mathfrak{p}$ such that $\mathcal{E} \cup \{E\}$ splits $[0, \mathsf{M})^n$ into cells. To do it, we will look after constraints on each clock to guarantee that the constraint (on this clock) may be bound by the induced constraints from parametric equations of $\mathcal{E}$ on the same clock.

Formally, let $x_i \in \mathcal{X}$ and $E' \in \mathcal{E}$ such that $E' = \sum_{i=1}^{n} a'_i \, x_i + b' + c' \, \mathfrak{p}$ with $a'_i \neq 0$. We want to compare constraints on x_i induced by E and E': in particular, we ask when

$$a'_i \, A + a'_i \, c \, \mathfrak{p} \leq a_i \, A' + a_i \, c' \, \mathfrak{p} \quad \Leftrightarrow \quad a'_i \, A - a_i \, A' \leq (a_i \, c' - a'_i \, c)\mathfrak{p}$$

where $A = \sum_{j \neq i} a_j \, x_j + b$ and $A' = \sum_{j \neq i} a'_j \, x_j + b'$. In particular, we note that this inequality holds when $a'_i \, A - a_i \, A' \leq 0$, and if $a_i \, c' - a'_i \, c < 0$, then the inequality holds only if $\eta' = \min(\eta, \frac{a'_i \, A - a_i \, A'}{a_i \, c' - a'_i \, c})$. We note that the converse inequality is obtained when $a'_i \, A - a_i \, A' \geq 0$ with a new constraint on η' when $a_i \, c' - a'_i \, c > 0$.

Finally, we remark that we have defined η' such that a comparison between each parametric affine equation can be done. Thus, the rest of the partition is given

like for the classical case (see [Bus19]). □

First operations over parametric piecewise affine functions To prove that intermediate operators can deal with parametric piecewise affine functions, we will strongly rely on Lemma 10.8 to adapt the proof from [Bus19]. We begin by studying operators Guard and Unreset.

Lemma 10.9. *Let $\langle P, F, \eta\rangle$ be a parametric piecewise affine function and g be a guard in $\mathcal{G}$. Then, there exists a parametric piecewise affine function $\langle P', F', \eta'\rangle$ such that $[\![F']\!] = \mathsf{Guard}_g([\![F]\!])$.*

Proof. As g is non-diagonal, it is encoded by a cell c_g given by $I_1 \wedge \cdots \wedge I_{2n}$. Intuitively, we will define a refinement P' of P such that all cells of P are included in c_g or disjoint of c_g. Formally, we let P' defined by the following set of equations

$$\mathcal{E}_{P'} = \mathcal{E}_P \ \cup \ \{E(I_i) \mid 0 \leq i \leq 2n\}\,.$$

We note that (P', η') satisfies that all their cells are either included in the cell defined by g, or outside of it. Finally, we define F' such that $[\![F']\!] = \mathsf{Guard}_g([\![F]\!])$. For all cells c, we fix $F'(c) = F(c)$ if c is included in the cell of g, and $\pm\infty$ otherwise (according to which player ℓ belongs to). □

Lemma 10.10. *Let $\langle P, F, \eta\rangle$ be a parametric piecewise affine function and $Y \subseteq \mathcal{X}$ be a set of clocks. Then, there exists a parametric piecewise affine function $\langle P', F', \eta\rangle$ such that $[\![F']\!] = \mathsf{Unreset}_Y([\![F]\!])$.*

Proof. Let I be an affine inequality under the form $\sum_{i=0}^{n} a_i x_i + b + c\mathfrak{p} \bowtie 0$ with $\bowtie \in \{<, >, =\}$, $\mathsf{Unreset}_Y(I)$ be the operator that denotes the affine inequality of form

$$\sum_{i=0}^{n} a'_i\, x_i + b + c\mathfrak{p} \bowtie 0 \qquad \text{where } a'_i = \begin{cases} 0 & \text{if } x_i \in Y; \\ a_i & \text{otherwise.} \end{cases}$$

Let ν be a valuation, then we remark that $\nu \models \mathsf{Unreset}_Y(I)$ if and only if $\nu[Y := 0] \models I$. We extend this construction to cells: given a cell $c = I_1 \wedge \cdots \wedge I_k$, we fix $\mathsf{Unreset}_Y(c) = \mathsf{Unreset}_Y(I_1) \wedge \cdots \wedge \mathsf{Unreset}_Y(I_k)$. Thus, for all valuations ν, $\nu[Y := 0] \models c$ if and only if $\nu \models \mathsf{Unreset}_Y(c)$. Moreover, we note that if c does not intersect the sub-space where all clocks of Y are null, then $\mathsf{Unreset}_Y(c) = \emptyset$.

Now, we consider F that maps every cell c of P to ane affine value function of the form $y = \sum_{i=0}^{n} \alpha_i\, x_i + \beta + \gamma\mathfrak{p}$. Similarly, we define the affine function $\mathsf{Unreset}_Y(F(c))$ such that

$$y = \sum_{i=0}^{n} \alpha'_i\, x_i + \beta + \gamma\mathfrak{p} \qquad \text{where } a'_i = \begin{cases} 0 & \text{if } x_i \in Y; \\ a_i & \text{otherwise.} \end{cases}$$

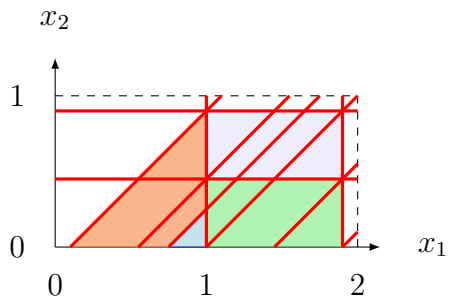


Figure 10.5 – The atomic partition that corresponds to the partition depicted on the right of Figure 10.4

In particular, for all valuations $\nu \in \mathsf{Unreset}_Y(c)$, we obtain that $F(\nu[Y := 0]) = \mathsf{Unreset}_Y(F(c))(\nu)$.

To conclude, we fix $\langle P', F', \eta \rangle$ such that $P' = \{\mathsf{Unreset}_Y(c) \mid c \in P\}$ (we note that if $\mathsf{Unreset}_Y(c) = \emptyset$, the cell is removed) and for all cells c such that $\mathsf{Unreset}_Y(c) \neq \emptyset$, we fix $F'(c) = \mathsf{Unreset}_Y(F(c))$. Thus, $\langle P', F', \eta \rangle$ is a parametric piecewise affine function such that $[\![F']\!] = \mathsf{Unreset}_Y([\![F]\!])$. □

Computation of $\mathsf{Pre}_{\ell'}$ Given a location $\ell' \in L$, we want to compute the operator $\mathsf{Pre}_{\ell'}$ over a parametric piecewise affine function $\mathbb{V}_\ell$ like for piecewise affine functions. Intuitively, this operator computes the time elapse in ℓ' that is beneficial before reaching $\mathbb{V}_\ell$ by focusing on diagonal lines in $\mathbb{R}^n_{\geq 0}$. In particular, a given inequality I is said *diagonal* if $\sum_{i=0}^{|\mathcal{X}|} a_i = 0$: if $\nu \models I$ then $\nu + d \models I$ for all $d \in \mathbb{R}$. A cell is a *tube* if all of its inequalities are diagonal. A cell c is *atomic* if c is contained only in a tube, i.e. if the borders of c do not intersect within the smallest tube that contains c. A partition P is said *atomic* if all their cells are atomic, and a parametric piecewise affine function $\langle P, F, \eta \rangle$ is *atomic* if P is.

Example 10.11. When we compute the conservative fixed-parameter robust value in Section 10.2, the partition of $[0, 2]^2$ that we compute is not atomic (see the partition depicted on the right of Figure 10.4). In Figure 10.5, we represent the atomic partition obtained from it. ⌟

Lemma 10.12. *Let $\langle P, F, \eta \rangle$ be a parametric piecewise affine function. Then, there exists an atomic parametric piecewise affine function $\langle P', F', \eta' \rangle$ such that $[\![F]\!] = [\![F']\!]$.*

Proof. Intuitively, to define P' we split cells of P according to diagonal inequalities. To do it, we introduce the notion of diagonal intersection between two (non-diagonal) equations. We consider two equations $E : \sum_{i=0}^{n} a_i x_i + b + c\mathfrak{p}$ and

$E' : \sum_{i=0}^{n} a'_i\, x_i + b' + c'\mathfrak{p}$. By letting $A = \sum_{i=0}^{n} a_i$ and $A' = \sum_{i=0}^{n} a'_i$, we fix the diagonal intersection of E and E' by

$$E \cap_d E' = \sum_{i=0}^{n} (Aa'_i - A'a_i)x_i + (Ab' - A'b) + (Ac' - A'c)\mathfrak{p}\,.$$

Now, by letting $\mathcal{E}_P$ be the set of equalities that appears in P, we define (P', η') from the set of equalities $\mathcal{E}_P \cup \{E \cap_d E' \mid E, E' \in \mathcal{E}_P\}$. Finally, we remark that $\langle P', F', \eta' \rangle$ is atomic. □

We suppose that $\ell' \in L_{\mathsf{Max}}$ (the case for Min is symmetrical by replacing the supremum with an infimum), and we consider a valuation ν. We recall that

$$\mathsf{Pre}_{\ell'}(\llbracket F \rrbracket)(\nu) = \sup_{t \le t_\nu} [t\, \mathsf{wt}(\ell') + \llbracket F \rrbracket(\nu + t)]\,.$$

For every delay $t \in [0, t_\nu]$, the valuation $\nu + t$ belongs to the diagonal line from ν to t_ν that intersect a finite number of cells of P. We let C_ν this set of cells, and for every cell $c \in C_\nu$, we let $t^c_{\min}$ be the minimum delay such that $\nu + t \in c$ and $t^c_{\max}$ be the maximum one. Now, since $t \mapsto \llbracket F \rrbracket(\nu + t)$ is a (parametric) affine function over c, the maximum is reached on $t^c_{\min}$ or in $t^c_{\max}$. Moreover, these two delays reach a non-diagonal border of the cell (or $t^c_{\min} = 0$).

So, we can focus on the non-diagonal borders of each cell c. If B is a non-diagonal border of c and ν is a valuation, there exists a unique $t_{\nu,B} \in \mathbb{R}$ such that $\nu + t_{\nu,B} \in \llbracket B \rrbracket$: if B is encoded by the following equation $\sum_{i=0}^{n} a_i\, x_i + b + c\mathfrak{p} = 0$ with $A = \sum_{i=0}^{n} a_i$, then

$$t_{\nu,B} = -\frac{1}{A}\Big(\sum_{i=0}^{n} a_i\, \nu(x_i) + b + c\mathfrak{p}\Big)\,.$$

Moreover, if B is reachable with time elapse from ν, we note that $t_{\nu,B} \in [0, t_\nu]$. The supremum of $\mathsf{Pre}_{\ell'}(\llbracket F \rrbracket)(\nu)$ can be rewritten by

$$\max \begin{cases} \llbracket F \rrbracket(\nu) \\ \max_{c \in C_\nu} \max_{B \in B_\nu(c)} [t_{\nu,B}\, \mathsf{wt}(\ell') + \llbracket F(c) \rrbracket(\nu + t_{\nu,B})] \end{cases}$$

where $B_\nu(c)$ is the set of the non-diagonal borders of c, $\llbracket F \rrbracket(\nu)$ corresponds to the delay 0 and $\llbracket F(c) \rrbracket(\nu + t_{\nu,B})$ corresponds to a jump arbitrarily close to B.

Finally, we suppose that $\langle P, F, \eta \rangle$ is atomic. In this case, letting c be a cell, for all valuations $\nu, \nu' \in c$, $C_\nu = C_{\nu'}$ and $B_\nu(c') = B_{\nu'}(c')$ for all $c' \in C_\nu$. In particular, we denote by C_c and $B_c(c')$ those sets. We introduce a new operator $\mathsf{Pre}_{\ell',c,B}$, that

maps a partition value function F to the value function

$$\mathsf{Pre}_{\ell',c,B} \coloneqq \nu \mapsto t_{\nu,B}\,\mathsf{wt}(\ell') + [\![F(c)]\!](\nu + t_{\nu,B})\,.$$

And, in this case, $\mathsf{Pre}_{\ell'}(F) = \max([\![F]\!], \max_{c\in C_\nu} \max_{B\in B_\nu(c)}[\mathsf{Pre}_{\ell',c,B}(F)])$.

Lemma 10.13. *Let $\langle P, F, \eta\rangle$ be an atomic parametric piecewise affine value function. Let $\mathsf{c} \in P$ be a cell of P, ℓ' be a location, $\mathsf{c}' \in C_\mathsf{c}$ and B be the border in $B_\mathsf{c}(\mathsf{c}')$. Then, there exists a parametric affine value function f of equation $y = \sum_{i=0}^n \alpha_i^f\, x_i + \beta^f + \gamma^f \mathfrak{p}$ such that $f = \mathsf{Pre}_{\ell',\mathsf{c}',B}(F)$ on c.*

Proof. Let ν be a valuation of c. As $B \in B_\mathsf{c}(\mathsf{c}')$, then there exists $t_{\nu,B} \in [0, t_\nu]$ such that

$$\mathsf{Pre}_{\ell',\mathsf{c}',B}(F)(\nu) = t_{\nu,B}\,\mathsf{wt}(\ell) + [\![F(\mathsf{c})]\!](\nu + t_{\nu,B})\,.$$

Let $y = \sum_{i=0}^n \alpha_i\, x_i + \beta + \gamma\mathfrak{p}$ be the equation of $F(\mathsf{c})$ with $A = \sum_{i=0}^n \alpha_i$, and $\sum_{i=0}^n a_i\, x_i + b' + c\,\mathfrak{p}$ be the equation of B with $A' = \sum_{i=0}^{|\mathcal{X}|} a'_i$. We obtain the following equalities:

$$t_{\nu,B} = -\frac{1}{A'}\big(\sum_{i=0}^n a_i\,\nu(x_i) + b + c\,\mathfrak{p}\big)\,;$$

$$[\![F(\mathsf{c}')]\!](\nu + t_{\nu,B}) = \sum_{i=0}^n \alpha_i\,\nu(x_i) + At_{\nu,B} + \beta + \gamma\,\mathfrak{p}\,;$$

$$[\![F(\mathsf{c}')]\!](\nu + t_{\nu,B})\,A' = \sum_{i=0}^n (A'\alpha_i - Aa_i)\,\nu(x_i) + (A'\beta - Ab) + (A'\gamma - Ac)\mathfrak{p}\,.$$

In particular, we deduce that $\mathsf{Pre}_{\ell',\mathsf{c}',B}(F)(\nu)$ is equal to

$$\sum_{i=0}^n \underbrace{(A'\alpha_i - Aa_i - a_i\,\mathsf{wt}(\ell'))}_{\alpha^f}\,\nu(x_i) + \underbrace{(A'\beta - Ab - b\,\mathsf{wt}(\ell'))}_{\beta^f} + \underbrace{(A'\gamma - Ac - c\,\mathsf{wt}(\ell'))}_{\gamma^f}\mathfrak{p}\,.$$

and thus $\mathsf{Pre}_{\ell',\mathsf{c}',B}(F)$ is encoded by a parametric affine equation $y = \sum_{i=0}^n \alpha_i^f\, x_i + \beta^f + \gamma^f\mathfrak{p}$. □

We conclude the computation of $\mathsf{Pre}_{\ell'}$ by remarking that $\mathsf{Pre}_{\ell'}$ is a maximum (or a minimum) of a finite number of functions $\mathsf{Pre}_{\ell',\mathsf{c},B}$ that is well computable in our formalism. We compute the minimum (resp. the maximum) with a new operator on parametric piecewise affine functions: the coarsest partition. If P_1 and P_2 are two

partitions over the same domain, then $P_1 \oplus P_2$ is the *coarsest partition* consistent with P_1 and P_2: for each cell c_1 of P_1 and each cell c_2 of P_2, $P_1 \oplus P_2$ contains $\mathsf{c}_1 \cap \mathsf{c}_2$ (we note that a new upper bound for $\mathfrak{p}$ is needed). By Lemma 10.8, this partition is computable by fixing $\mathcal{E}_{P_1} \cup \mathcal{E}_{P_2}$ under a new upper bound η'. Then, we prove that our representation allows us to compute the minimum or the maximum of parametric piecewise affine functions.

Lemma 10.14. *Let $\langle P_i, F_i, \eta_i \rangle_{1 \leq i \leq k}$ be a set of parametric piecewise affine functions defined over the same domain c_P. Then, there exists a parametric piecewise affine function $\langle P, F, \eta \rangle$ over c_P and η' be a new constraint over $\mathfrak{p}$ such that $[\![F]\!] = \min_{1 \leq i \leq k} [\![F_i]\!]$ (respectively, $[\![F]\!] = \max_{1 \leq i \leq k} [\![F_i]\!]$).*

Proof. We fix $P' = P_1 \oplus \cdots \oplus P_k$, and we consider a cell c of P such that $\mathsf{c} = \mathsf{c}_1 \cap \cdots \cap \mathsf{c}_k$ where c_i is a cell of P_i. Let $F_1(\mathsf{c}_1), \ldots, F_k(\mathsf{c}_k)$ be the affine functions of each c_i that are represented by affine equations over $\mathcal{X} \cup \{y, \mathfrak{p}\}$ (denoted by $E_1, \ldots, E_k$). To encode the minimum of $F_i(\mathsf{c}_i)$ on c, we will define $(P_\mathsf{c}, F_\mathsf{c})$ where P_c is a partition of c. Intuitively, this new partition describes when two affine functions E_i and E_j are equals. In particular, we use a new operator $\cap_y$ to define the affine equation that describes this inequality.

Let E and E' be two affine equations of the form

$$0 = \sum_{i=0}^{n} a_i\, x_i + b + c\, \mathfrak{p} \qquad \text{and} \qquad 0 = \sum_{i=0}^{n} a'_i\, x_i + b' + c'\, \mathfrak{p}\,.$$

The set of valuations that are in the intersection, denoted by $[\![E]\!] \cap [\![E']\!]$ is either empty, or it satisfies the following affine equation on $\mathcal{X} \cup \{\mathfrak{p}\}$

$$E \cap_y E' \colon \sum_{i=0}^{n} (a'_i - a_i) x_i + (b' - b) + (c' - c)\mathfrak{p} = 0\,.$$

Now, we define the partition P_c by the set of all such intersections

$$P_\mathsf{c} = \{E_i \cap_y E_j \mid i, j \in [1, k] \wedge [\![E_i]\!] \cap [\![E_j]\!] \neq \emptyset\}$$

with η_c as upper bound, by Lemma 10.8. Now, for every cell c' of P_c there exists $j \in [1, k]$ such that for all valuations $\nu \in \mathsf{c}'$, $[\![F_j]\!](\nu) = \min_{1 \leq i \leq k} [\![F_i]\!](\nu)$. Indeed, by contradiction let suppose that there exists a cell c' of P_c such that for all $j \in [1, k]$, there exists $\nu \in \mathsf{c}'$ such that $[\![F_j]\!](\nu) \neq \min_{1 \leq i \leq k} [\![F_i]\!](\nu)$. In particular, there exists two affine equations E_i and E_j such that $[\![E_i]\!] \cap [\![E_j]\!] \neq \emptyset$ in c'. Thus, since each cell is defined by a set of equations of the form $E_i \cap_y E_j$ when $[\![E_i]\!] \cap [\![E_j]\!] \neq \emptyset$, we obtain a contradiction with the definition of c'.

Finally, we fix $P = \bigcup_{\mathsf{c} \in P'} P_\mathsf{c}$ and for all cells c' in P and η' by Lemma 10.8, we

define F on c' by $F_j(\mathsf{c}_j)$ where j is such that for all valuations $\nu \in \mathsf{c}'$, $[\![F_j]\!](\nu) = \min_{1\leq i\leq k}[\![F_i]\!](\nu)$. □

Computation of $\mathsf{Perturb}^{\mathfrak{p}}_{\ell'}$ Finally, we need to compute $\mathsf{Perturb}^{\mathfrak{p}}_{\ell'}$ for a given location $\ell' \in L$. To do it, we adapt the proof of the operator Pre.

Lemma 10.15. *Let $\langle P, F, \eta\rangle$ be a set of parametric piecewise affine function, $\ell' \in L$ and $0 < \mathfrak{p} < \eta$ be a parameter. Then, there exists a parametric piecewise affine function $\langle P', F', \eta'\rangle$ such that $[\![F']\!] = \mathsf{Perturb}^{\mathfrak{p}}_{\ell'}([\![F]\!])$.*

Proof. Since $\mathsf{Perturb}^{\mathfrak{p}}_{\ell'}(\mathbb{V}_\ell) = \mathbb{V}_\ell$ when $\ell' \in L_{\mathsf{Max}}$, we suppose that $\ell' \in L_{\mathsf{Min}}$, and we consider a valuation ν. In particular, we recall that

$$\mathsf{Perturb}^{\mathfrak{p}}_{\ell'}([\![F]\!])(\nu) = \sup_{p\in[0,2\mathfrak{p}]}\left[p\,\mathsf{wt}(\ell') + [\![F]\!](\nu+p)\right].$$

Again, we use the fact that $p \mapsto [\![F]\!](\nu+p)$ is an affine function to deduce that the supremum of $\mathsf{Perturb}^{\mathfrak{p}}_{\ell'}$ becomes a maximum over the diagonal line from ν to $\nu + 2\mathfrak{p}$, as in the case of operator $\mathsf{Pre}_{\ell'}$. Since the diagonal line can intersect with many cells, we let $C^{\mathfrak{p}}_\nu$ be the set of cells that intersect the diagonal line from ν to $\min(\mathsf{M}, \nu + 2\mathfrak{p})$ where M is the upper bound over valuations of clocks. Now, for every cell $\mathsf{c} \in C^{\mathfrak{p}}_\nu$, we let $p^{c}_{\min}$ be the minimum perturbation such that $\nu + p \in \mathsf{c}$ and $p^{\mathsf{c}}_{\max}$ be the maximum one. Now, since $p \mapsto [\![F]\!](\nu+p)$ is an affine function over c, the maximum is reached on $p^{\mathsf{c}}_{\min}$ or in $p^{\mathsf{c}}_{\max}$. Moreover, these two delays reach a non-diagonal border of the cell (or $p^{\mathsf{c}}_{\min} = 0$).

So, we can focus on the non-diagonal borders of each cell. If B is a non-diagonal border of c and ν is a valuation, there exists a unique $p_{\nu,B} \in \mathbb{R}$ such that $\nu + p_{\nu,B} \in [\![B]\!]$: if B is encoded by the following equation $\sum_{i=0}^{n} a_i\, x_i + b + c\mathfrak{p} = 0$ with $A = \sum_{i=0}^{n} a_i$, then

$$p_{\nu,B} = -\frac{1}{A}\Big(\sum_{i=0}^{n} a_i\,\nu(x_i) + b + c\mathfrak{p}\Big)$$

Moreover, if B is reachable with time elapse from ν, we note that $p_{\nu,B} \in [0, p_\nu]$. The supremum of $\mathsf{Perturb}^{\mathfrak{p}}_{\ell'}([\![F]\!])(\nu)$ can be rewritten by

$$\max\begin{cases}[\![F]\!](\nu)\\ \max_{c\in C^{\mathfrak{p}}_\nu}\max_{B\in B^{\mathfrak{p}}_\nu(c)}[p_{\nu,B}\,\mathsf{wt}(\ell') + [\![F(c)]\!](\nu+p_{\nu,B})]\\ [\![F_1]\!](\nu)\end{cases}$$

where $B^{\mathfrak{p}}_\nu(c)$ is the set of non-diagonal borders of c, $[\![F]\!](\nu)$ corresponds to the delay 0, $[\![F(\mathsf{c})]\!](\nu + p_{\nu,B})$ corresponds to a jump arbitrarily close to $B^{\mathfrak{p}}$, and $[\![F_1]\!]$

corresponds to a jump to $\min(\mathsf{M}-\nu, 2\mathfrak{p})$ defined such that

$$[\![F_1]\!] = \begin{cases} \sum_{i=0}^{c} \alpha_i\, \nu(x_i) + \beta + (2A + \gamma + 2\,\mathsf{wt}(\ell))\mathfrak{p} & \text{if } 2\mathfrak{p} \leq \mathsf{M} - \nu; \\ \sum_{i=0}^{n} \alpha_i\, \nu(x_i) + (\beta + 2A + 2\,\mathsf{wt}(\ell)) + \gamma\mathfrak{p} & \text{otherwise;} \end{cases}$$

when $y = \sum_{i=0}^{n} \alpha_i\, x_i + \beta + \gamma\,\mathfrak{p}$ be the equation of F and $A = \sum_{i=0}^{n} \alpha_i$. We conclude that there exists a parametric piecewise affine function $\langle P', F', \eta' \rangle$ to encode $\mathsf{Perturb}^{\mathfrak{p}}_{\ell'}$ by Lemmas 10.8, 10.13 and 10.14. □

10.4. An extension to the strictly non-Zeno weighted timed games

As we can solve acyclic WTGs, a natural question is to extend this result to other classes of WTGs where the limit robust value can be computed on an unfolding of the game. In this section, we apply this scheme of proof to strictly non-Zeno WTGs which is the class of WTGs that is a divergent WTGs with only non-negative weights, under a class of conservative semantics.

Theorem 10.16. *Let $\mathcal{G} = \langle L_{\mathsf{Min}}, L_{\mathsf{Max}}, L_T, L_u, \mathcal{X}, \Delta, \mathsf{wt} \rangle$ be a strictly non-Zeno WTG under a class of conservative semantics $(\mathsf{cons}_{\mathfrak{p}})_{\mathfrak{p}>0}$. Then, for all locations ℓ, $\nu \mapsto \overline{\mathsf{rVal}}^{\mathsf{cons}}_{\mathcal{G}}(\ell, \nu)$ is computable.*

Let ℓ be a location. We compute the conservative limit robust value function from ℓ with the following method.

1. From the algorithm to solve the parametrized reachability robust problem from ℓ (given in [BMS15]) under the conservative semantics, there exist a parameter $\mathfrak{p} > 0$ and a winning strategy such that all conforming plays from ℓ have an exponential size in $|L|$, that we denote by P.

2. Since Min can guarantee that a target location is reached within P steps, then the weight of this play gives an upper bound of the limit robust value, i.e. it is less than $P \times W_{\mathsf{e}}$.

3. Since we know an upper bound over the limit robust value, Min has no interest to produce plays with too many cyclic region paths: by definition, each cyclic region play is followed by a play of weight at least 1 (since divergence property is preserved by the conservative semantics, see proof of Proposition 9.30). Thus, we define an unfolding of depth $(P \times W_{\mathsf{e}} + 1) \times 2\,|Q|$ where Q is the set of region states of $\mathcal{G}$. In particular, with this unfolding, we are forbidding Min to make more than $P \times W_{\mathsf{e}}$ cyclic paths in the region game before reaching a target location.

4. Finally, we apply the previous section to this unfolding with an initial constraint on the parameter: $\mathfrak{p}$.

The proof of Theorem 10.16 is given by the proof of the correction of this algorithm. In particular, we need to prove that the unfolding preserves the conservative limit robust value. First, we note that all strategies from the unfolding may be implemented inside $\mathcal{G}$. In particular, the limit robust value of the unfolding is greater than the one of $\mathcal{G}$.

Conversely, since $P \times W_{\mathsf{e}}$ is an upper bound of the conservative limit robust value in $\mathcal{G}$, then for all $\varepsilon > 0$, there exists a strategy such that its value is less than $P \times W_{\mathsf{e}} + \varepsilon$ in $\mathcal{G}$. Moreover, since $\mathcal{G}$ is strictly non-Zeno, i.e. all plays that follow a cyclic region path have a weight of at least 1, we know that all plays conforming to this strategy have length[3] at most $(P \times W_{\mathsf{e}} + 1) \times 2\,|Q|$ (otherwise, the play contains at least $(P \times W_{\mathsf{e}} + 1)$ cyclic region paths, and has a weight at least $(P \times W_{\mathsf{e}} + 1)$). In particular, we can apply this strategy inside the unfolding and obtain the same weight, i.e. the same value. Thus, the limit robust value of the unfolding is less than the one of $\mathcal{G}$.

Remark. We note that we prove that value iteration applying to $\mathcal{F}^{\mathsf{cons}_{\mathfrak{p}}}$ in strictly non-Zeno WTGs converges in finite time. ◁

[3] We note that we multiply by $2\,|Q|$ since we need to take into account the perturbed states of the semantics.

Discussion

This part studied the tradeoff between mathematical idealization and a more realistic system. We focused on strategies that resist a small perturbation induced by the environment. Our goal was to study the value guaranteed under different robust semantics that model imprecision induced by the environment. In particular, we have developed a new data structure to apply the value iteration algorithm (developed in [ABM04] in the deterministic case) to solve acyclic WTGs under robust semantics where the value of its parameter is arbitrarily small.

Robust value problems in WTGs

Through this work, we recover the decidability for some classes of WTGs for one of the value problems defined by the robust values. However, the precise complexity of these problems remains open. In particular, when we compute the limit robust value in acyclic WTGs, our algorithm is not the most efficient (in [Bus19], the author uses nested partitions[1] to optimise the complexity). Moreover, we want to extend this result to other classes of WTGs in which the deterministic value is known to be decidable, i.e. one clock or divergent WTGs under both semantics.

Limit robust value problem in acyclic WTGs under excessive semantics In Chapter 10, we prove that the limit robust value is computable in the translated excessive semantics. Translated excessive semantics is not equivalent to all excessive semantics since guards are modified by the translation (see Example 9.4). Indeed, to compute the limit robust value under excessive semantics, we need to adapt the operator that checks the guards. Intuitively, borders induced by guards will be also parametrised by $\mathfrak{p}$ (before the security distance taken by Min to avoid $+\infty$). Formally, we use a new operator such that if g is a guard over $\mathcal{X}$ and $\mathfrak{p}$ is the parameter of the semantics, $\mathsf{Guard}_g^{\mathfrak{p}}(\mathbb{V}_\ell)\colon [0,\mathsf{M})^n \to \mathbb{R}_\infty$ is the piecewise affine

[1] A nested partition encodes a tree that models the order in which the cells are defined.

function defined by

$$\mathsf{Guard}_g^{\mathfrak{p}}(\mathbb{V}_\ell)(\nu) = \begin{cases} \mathbb{V}_\ell(\nu) & \text{if } \nu \models g \wedge \ell \in L_{\mathsf{Max}}; \\ \mathbb{V}_\ell(\nu) & \text{if } \nu \models \mathsf{pert}^{\mathfrak{p}}(g) \wedge \ell \in L_{\mathsf{Min}}; \\ -\infty & \text{if } \nu \not\models g \wedge \ell \in L_{\mathsf{Max}}; \\ +\infty & \text{if } \nu \not\models \mathsf{pert}^{\mathfrak{p}}(g) \wedge \ell \in L_{\mathsf{Min}}; \end{cases}$$

where $\mathsf{pert}^{\mathfrak{p}}(g)$ is inductively defined on atomic constraints such that $\mathsf{pert}^{\mathfrak{p}}(a \leq x \leq b) = a \leq x \leq b - \min \Delta_{\mathfrak{p}}$. By adapting our computation over parametrised piecewise affine value function, we hope that we can compute the limit robust value in acyclic WTGs under excessive semantics.

Limit robust value problem in one-clock WTGs Another class of interest is the one-clock WTGs in which the value iteration algorithm may not converge in finite time (it is already the case in the deterministic case). In particular, we need to find another method. For instance, an idea may be considering the traduction into the exact semantics proposed in Section 9.1 with a parametrized point of view (of parameter $\mathfrak{p}$). In particular, under a class of translated robust semantics, we obtain a parametrized WTG according to $\mathfrak{p}$. Since the reachability problem is known as undecidable [JLR19] for all parametric RTGs[2], we hope that this problem (and its weighted extension) becomes decidable in the special class of parametrized WTGs that are induced by the robust semantics in WTGs.

Fixed-parameter robust value problems in WTA under conservative semantics In [BMS13], authors prove that the corner-point abstraction can be adapted (by only removing singleton corners) to compute fixed-parameter robust values and limit robust values under conservative semantics. However, this technique can not be directly extended for fixed-parameter robust value problems. Indeed, in this case, Min must deal with the parameter $\mathfrak{p}$ (that cannot be modified along the computation) by staying far enough from the border of the guards. Intuitively, only cyclic paths (and their iterations) may be a difficulty, since a WTA is a one-player game in which Max can not push Min in a border of guards. In particular, by combining the corner point abstraction with *flags*[3], we hope that fixed-parameter

[2]Parametric timed automata introduced by [AHV93] are an extension of timed automata in which guards are not defined by atomic constraints with integer but with atomic constraints with affine function according to one (or many) parameter(s). For all classical problems from timed automata like reachability, two versions coexist: solving the problem when the value of the parameter is fixed, and looking the whole set of parameters such that the problem is solved in the automata instanciated with one of these values.

[3]Flags are introduced in [Bus+19] to solve the Büchi objective in timed automaton under the centred conservative semantics. Flags are conditions to denote cyclic paths that can be

robust values and limit robust values may be computable in WTA.

Robust value problems in divergent WTGs An important class of WTGs in the deterministic case is divergent WTGs. Thus, it is natural to look after the decidability of robust value problems in this class of WTGs.

Under conservative semantics, we proved that the fixed-parameter robust value problem is decidable in divergent WTGs, and the limit robust value problem is decidable in strictly non-Zeno WTGs. In particular, the first natural question is about the limit robust value problem in divergent WTGs. In the deterministic case (see [BMR17]), SCCs that contain a negative cyclic path can be solved like positive ones by inverting the membership of location to player Min and Max, and multiplying all weights by -1. Morally, this construction inverts the infimum and the supremum in the definition of $\mathcal{F}$. In particular, a way to extend the decidability of the limit robust value problem in all divergent WTGs would be to multiply the weight by -1 and invert the supremum and infimum in the operator $\mathcal{F}^{\mathsf{rob}}$, instead of inverting the players as in the deterministic case.

Under the excessive semantics, the robust value problems are more difficult to study: excessive semantics induce plays that cannot be transformed into a play under the exact semantics. Thus, under excessive semantics, WTGs may lose their property helping to compute the (robust) values, e.g. a divergent WTG may contain a cyclic path followed by a play with a null weight (see Example 9.31). In particular, the decidability for value problems seems to be difficult to recover under excessive semantics: the limit robust value in one-player games is undecidable, and the classical property of divergence does not suffice to guarantee that fixed-parameter robust values are computable in divergent WTGs. Hence, we conjecture that the fixed-parameter robust value problem in divergent WTGs under excessive semantics is undecidable, which would imply that the limit robust value problem in divergent WTGs under excessive semantics is also undecidable.

Other problems with robust values

Given a robust value, a natural question is the corresponding value problem that induces the problem of the computation of the robust value. However, we can consider other questions with robust values.

Implementation of computation of robust values Some tools are developed to analyse the robustness in the case of the model checking, i.e. in the case of WTA. For instance, we can cite ShrinkTech[4] which analyses reachability with shrunk

iterated under the centred conservative semantics.

[4] http://www.lsv.fr/Software/shrinktech/

DBMs in timed automata [San13b], or IMITATOR[5] which provides a robustness analysis on parametrised timed automata [ÉFE08]. However, to our knowledge, there are no tools that implement robust semantics in WTGs. The first step in the implementation of robust semantics in WTGs may be acyclic one-clock WTGs. Indeed, we have shown that limit robust values are computable in acyclic WTGs (see Chapter 10) with a value iteration algorithm that only adds a new intermediate operator w.r.t. the classical one. In particular, adapting our prototype used to compute the deterministic value in one-clock WTGs (see Section 4.5) may be a reasonable step in the objective to have tools to compute robust values (when it is possible).

The most permissive perturbation bound From the parametric point of view on robustness in WTGs, we want to extend the robust value problems by computing all perturbation bounds for which the fixed-parameter robust value exists (and the corresponding value function). This problem called the permissiveness of timed automata, introduced by [BFM15], was studied in one-clock [BFM15] or acyclic [Cle+20] timed automata and games under the reachability objective. When we consider WTGs, we note that, in the computation of the limit robust values in acyclic WTGs, we consider the computation of fixed-parameter robust values where $\mathfrak{p}$ remains arbitrarily close to 0 to guarantee keeping all possible cells. However, we do not guarantee that the upper bound computed for the perturbation bound is the maximal one to guarantee that the fixed-parameter robust values will be finite.

Comparison with the deterministic value Like in Part II, we could be willing to compare the robust values to deterministic ones. First, when the perturbation bound is fixed, there are few chances that values are equal since the parameter has to appear in the robust value. Moreover, intuitively, at fixed parameters, the robust value is always greater than the deterministic value since Max can never perturb and produce a play under the exact semantics.

Moreover, taking the limit does not suffice to obtain the deterministic value since the equality between the deterministic value and a limit robust value does not hold for two reasons: some configurations may be winning under the exact semantics for the reachability problem and losing for the limit robust reachability problem under a class of robust semantics (as illustrated in Example 8.6); and even if Min wins limit robust reachability problem, the cycle iteration maybe not guaranteed as illustrated in Example III.1, thus Min can not cumulate enough weight. In particular, looking for a characterisation when the deterministic value and a given robust value are equal remains an interesting open question.

[5] https://www.imitator.fr/index.html

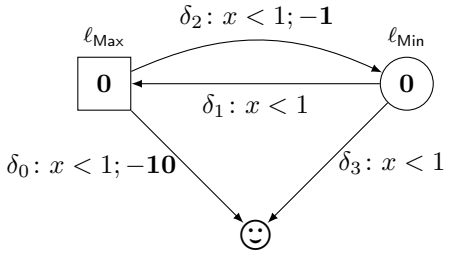


Figure III.1 – A WTG in which Min can not iterate ten times the cyclic path under a robust semantics.

Example III.1. We consider the WTG depicted in Figure III.1 under a robust semantics $\mathsf{rob}_{\mathfrak{p}}$ where $\Delta_{\mathfrak{p}} = [-a, 2\,\mathfrak{p} - a]$. If $a > 0$, Min has no interest to go in ℓ_{Max}. Indeed, from all configurations $(\ell_{\mathsf{Max}}, \nu)$ with $\nu < 1$, Max can always reach ℓ_{Min} in a valuation of at least $1 - a/2$ (by choosing the decision (δ_2, t) with $t = \max(0, 1 - a/2 - \nu)$). In this case, Min needs to play from $(\ell_{\mathsf{Min}}, 1 - a/2)$, and no robust decision exists from this configuration since Min must play at least a as delay to avoid negative time elapsing. Thus, a fixed-parameter robust value for a fixed $\mathfrak{p}$ is always 0.

Otherwise, Max can reach $(\ell_{\mathsf{Min}}, 1 - \mathfrak{p})$. In this case, either Min has no robust decision (as under a conservative semantics), or has no interest to return in ℓ_{Max}. Indeed, Max can guarantee that ℓ_{Max} is reached the second time in a valuation greater than 1, and in this case no feasible decision exist. ⌟

Other notions of robustness in WTGs

In the game setting, defining the robust semantics by adding the possibility for Max to perturb the delay chosen by Min is natural. However, this semantics raises some issues and we want to define other types of robustness (like for the model checking case). For instance, we can not treat the case of Zeno plays: the perturbation of Max may induce a Zeno play even if Min plays delays that do not induce a Zeno play under the exact semantics. Moreover, robust semantics increases the power of Max: he can push Min into a deadlock, even if Min takes care to play far enough of the border of guards. Finally, giving to Max the choice of the perturbation induces that the imprecision of the system is always the worst case for Min. We now give some other ways to define robustness in WTGs.

Pseudo-robustness Under robust semantics, Min can be blocked in a configuration from which a feasible decision exists. To lighten the robustness and deal with this situation, we can suppose that robust semantics allows Min to play a feasible

(non-robust) decision when no robust decision exist. Intuitively, such a definition will simplify the reachability problem under robust semantics, and the capability of Min to iterate some cyclic paths (i.e. robust values in this case may be closer to the deterministic one).

Blame Zeno play Under robust semantics, Max can impose null delays even if Min plays with positive delays. In particular, robust semantics will generate a well-known problem in concurrent WTGs: in some cases, Min could lose the game only because Max was able to produce a Zeno play[6]. From a modelling point of view, this situation is bad and we want to avoid it. To deal with it, we can use the same methods from concurrent WTGs where authors generally proposed to blame the player that is the origin of the Zeno behaviour (see [Alf+03] for details). In this case, Min can win if Max produces a Zeno play. However, the definition of the blame should be discussed: what is the responsibility of Min in the production of a Zeno play if it is thanks to the perturbations that Max can build it?

Robustness and stochastic strategies Finally, robust semantics induce that the chosen perturbation is always the worst for Min. However, from a modelling point of view, perturbations model the imprecision of the system, i.e. they are not always the worst for Min. In particular, since these perturbations may model errors of measure, we can use probabilities to model a random distribution on these errors. A way to do it is the usage of stochastic strategies for Min: we can enforce Min to play with a continuous (on a interval of minimal length) distributions over delays (that is not the case in Part II). In this case, the perturbation will be an expectation over possible perturbations (and not the worst possible). This question is studied in a more general context with the notion of *good-enough synthesis* [AK20]. In this setting, the environment is not supposed completely antagonistic to the controller: it blames him when his behaviour is not reasonable. For instance, if we want to synthesise a coffee machine, we want to consider the obtained machine correct when the user never adds water to the machine (even if in this case, the machine can not correctly work).

[6] We recall that a Zeno play is an infinite play in which the total time spent (the sum of delay) is finite.

Conclusion

In this thesis, we presented new results for several problems in WTGs and, more precisely, on features of strategies. We summarize them while mentioning possible future research directions and open issues developed in the chapter "Discussion" at the end of each part.

Deterministic value In Part I, we have focused on deterministic strategies where memory is permitted. We have solved the deterministic value problem in one-clock WTGs with arbitrary weights, an open problem for several years. Our algorithm consists of an acyclic unfolding that relies on technical ingredients such as the closure of a one-clock WTG, switching strategies for Min, and the fixpoint characterisation of the value function. Moreover, in a prototype, we have implemented the unfolding algorithm and a value iteration algorithm to compute the deterministic value of all one-clock WTGs.

Our prototype is not able to treat too large models: in reality, it is quickly limited by the size of WTGs. To deal with its limitations, an idea is to modify our algorithm to define the unfolding from the leaves instead of the initial location. Moreover, by improving the data structures used in the implementation of the WTGs, we will use a more efficient algorithm in acyclic WTGs that computes the deterministic value with only one computation by location.

Another research direction is the complexity of the deterministic value problem: a gap remains between our exponential algorithm and the known PSPACE lower bound [FIS20]. A line of research consists of a better understanding of cutpoints in a deterministic value function. Two ideas may be considered: the number of cutpoints is exponentially upper bounded, but we do know if the bound is reachable; the clock's valuations are rational numbers, but we do not know if they respect a minimal granularity.

Finally, we plan to explore other classes of WTGs with our results. In particular, we want to look after new classes defined with resets such that the deterministic value problem is decidable. Moreover, by extending our results about the deterministic value function in the case of two-clock WTGs, we hope to solve the deterministic value problem for two-clock WTGs with only non-negative weights that is still open.

Stochastic values In Part II, we studied the trade-off between memoryless using randomness and deterministic strategies with memory. In particular, we showed that Min guarantees the same value when restricted to these two kinds of strategies, or when allowed to play with both memory and randomisation. To achieve this goal, we introduced a new class of strategies, the stochastic strategies, from which we deduced many new values. In particular, we developed good hypotheses on stochastic strategies to obtain mathematical coherence for these strategies.

One of our first motivations to introduce stochastic strategies in shortest-path games is the development of new techniques to solve these games in polynomial time. In particular, by using probabilities in games, we hope that we can obtain a better algorithm using strategy iteration. Hence, by links with mean-payoff games, such an algorithm may be used to solve in polynomial time mean-payoff games that remain a difficult open question in game theory.

In continuation of a series of recent works, a research direction is studying the minimal memory needed by each player to win under a given payoff when probabilities are permitted. A first step would be the characterisation of the existence of memoryless stochastic optimal strategies when memoryless deterministic strategies do not exist: we look after an extension of the characterisation of [GZ05].

With our work, we better understand the probabilistic distribution in timed automata and the deterministic value function in WTGs (by our works in Part I). In particular, we now have more tools to try to solve open problems in models that merge time and probabilities. In particular, we plan to study the shortest-path problem in priced probabilistic timed automata under the hypothesis of divergence, as well as in stochastic timed automata by adding weights.

Robust values In Part III, we studied the tradeoff between mathematical idealization and a more realistic system by focusing on strategies for Min that tolerate small errors. In particular, we focused on the value guaranteed under different robust semantics that model imprecision induced by the environment by developing a new data structure for a robust version of the value iteration algorithm (developed by Alur et al. in the deterministic case [ABM04]). Hence, we solve acyclic WTGs under conservative semantics and the translated excessive semantics where the value of its parameter is arbitrarily small.

Robust values have been little studied. In particular, we do not know if the robustness preserves the decidability of value problems in divergent WTGs or one-clock WTGs. Even if, in our work, we have closed some issues, the main question remains the case of divergent WTGs under excessive semantics. Since the divergence is not preserved under excessive semantics, the fixed-parameter robust value problem may be undecidable in divergent WTGs under excessive semantics, whereas it is decidable under the exact semantics or the conservative semantics.

Another research direction is the computation of the maximal permissive parameter of robust semantics. In Chapter 10, when we study the limit values in acyclic WTGs, we maintain an upper bound over the parameter of strategies that guarantee that value under semantics with a small parameter is finite. However, we do not ensure that it is the maximal one. A natural question is the computation of the maximal upper bound that guarantees the finiteness of robust values: this problem is called maximal permissiveness in the literature [BFM15].

Finally, like robustness in timed automata, we can define many notions of robustness. By considering stochastic strategies with only continuous distribution over delays, introduced in Part II, we have a natural way to define robust semantics from strategies. Hence, we obtain new values to study.

Bibliography

[Aks+16] S. Akshay, Patricia Bouyer, Shankara Narayanan Krishna, Lakshmi Manasa, and Ashutosh Trivedi. "Stochastic Timed Games Revisited". In: *41st International Symposium on Mathematical Foundations of Computer Science, MFCS 2016, August 22-26, 2016 - Kraków, Poland*. Ed. by Piotr Faliszewski, Anca Muscholl, and Rolf Niedermeier. Vol. 58. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016, 8:1–8:14. DOI: 10.4230/LIPICS.MFCS.2016.8. URL: https://doi.org/10.4230/LIPIcs.MFCS.2016.8.

[Alf+03] Luca de Alfaro, Marco Faella, Thomas A. Henzinger, Rupak Majumdar, and Mariëlle Stoelinga. "The Element of Surprise in Timed Games". In: *CONCUR 2003 - Concurrency Theory, 14th International Conference, Marseille, France, September 3-5, 2003, Proceedings*. Ed. by Roberto M. Amadio and Denis Lugiez. Vol. 2761. Lecture Notes in Computer Science. Springer, 2003, pp. 142–156. DOI: 10.1007/978-3-540-45187-7_9. URL: https://doi.org/10.1007/978-3-540-45187-7%5C_9.

[AHK98] Luca de Alfaro, Thomas H. Henzinger, and Orna Kupferman. "Concurrent Reachability Games". In: *39th Annual Symposium on Foundations of Computer Science, FOCS '98, November 8-11, 1998, Palo Alto, California, USA*. IEEE Computer Society, 1998, pp. 564–575. DOI: 10.1109/SFCS.1998.743507. URL: https://doi.org/10.1109/SFCS.1998.743507.

[ABK22] Shaull Almagor, Udi Boker, and Orna Kupferman. "What's decidable about weighted automata?" In: *Inf. Comput.* 282 (2022), p. 104651. DOI: 10.1016/j.ic.2020.104651. URL: https://doi.org/10.1016/j.ic.2020.104651.

[AK20] Shaull Almagor and Orna Kupferman. "Good-Enough Synthesis". In: *Computer Aided Verification - 32nd International Conference, CAV 2020, Los Angeles, CA, USA, July 21-24, 2020, Proceedings, Part II*. Ed. by Shuvendu K. Lahiri and Chao Wang. Vol. 12225. Lecture Notes in Computer Science. Springer, 2020, pp. 541–563. DOI: 10.1007/

978-3-030-53291-8_28. URL: https://doi.org/10.1007/978-3-030-53291-8%5C_28.

[ABM04] Rajeev Alur, Mikhail Bernadsky, and P. Madhusudan. "Optimal Reachability for Weighted Timed Games". In: *Automata, Languages and Programming: 31st International Colloquium, ICALP 2004, Turku, Finland, July 12-16, 2004. Proceedings*. Ed. by Josep Díaz, Juhani Karhumäki, Arto Lepistö, and Donald Sannella. Vol. 3142. LNCS. Springer, 2004, pp. 122–133. DOI: 10.1007/978-3-540-27836-8_13. URL: https://doi.org/10.1007/978-3-540-27836-8%5C_13.

[ACD93] Rajeev Alur, Costas Courcoubetis, and David L. Dill. "Model-Checking in Dense Real-time". In: *Inf. Comput.* 104.1 (1993), pp. 2–34. DOI: 10.1006/INCO.1993.1024. URL: https://doi.org/10.1006/inco.1993.1024.

[AD94] Rajeev Alur and David L. Dill. "A theory of timed automata". In: *Theoretical Computer Science* 126.2 (1994), pp. 183–235. ISSN: 0304-3975. DOI: https://doi.org/10.1016/0304-3975(94)90010-8. URL: https://www.sciencedirect.com/science/article/pii/0304397594900108.

[AFH99] Rajeev Alur, Limor Fix, and Thomas A. Henzinger. "Event-clock automata: a determinizable class of timed automata". In: *Theoretical Computer Science* 211.1 (1999), pp. 253–273. ISSN: 0304-3975. DOI: 10.1016/S0304-3975(97)00173-4. URL: https://doi.org/10.1016/S0304-3975(97)00173-4.

[AH91] Rajeev Alur and Thomas A. Henzinger. "Logics and Models of Real Time: A Survey". In: *Real-Time: Theory in Practice, REX Workshop, Mook, The Netherlands, June 3-7, 1991, Proceedings*. Ed. by J. W. de Bakker, Cornelis Huizing, Willem P. de Roever, and Grzegorz Rozenberg. Vol. 600. Lecture Notes in Computer Science. Springer, 1991, pp. 74–106. DOI: 10.1007/BFB0031988. URL: https://doi.org/10.1007/BFb0031988.

[AH93] Rajeev Alur and Thomas A. Henzinger. "Real-Time Logics: Complexity and Expressiveness". In: *Inf. Comput.* 104.1 (1993), pp. 35–77. DOI: 10.1006/INCO.1993.1025. URL: https://doi.org/10.1006/inco.1993.1025.

[AHV93] Rajeev Alur, Thomas A. Henzinger, and Moshe Y. Vardi. "Parametric Real-Time Reasoning". In: *Twenty-Fifth Annual ACM Symposium on Theory of Computing, May 16-18, 1993, San Diego, CA, USA*. Ed. by

S. Rao Kosaraju, David S. Johnson, and Alok Aggarwal. STOC '93. San Diego, California, USA: Association for Computing Machinery, 1993, pp. 592–601. ISBN: 0897915917. DOI: 10.1145/167088.167242. URL: https://doi.org/10.1145/167088.167242.

[ALP04] Rajeev Alur, Salvatore La Torre, and George J. Pappas. "Optimal paths in weighted timed automata". In: *Theoretical Computer Science* 318.3 (2004), pp. 297–322. ISSN: 0304-3975. DOI: https://doi.org/10.1016/j.tcs.2003.10.038. URL: https://www.sciencedirect.com/science/article/pii/S0304397503005838.

[AM99] Eugene Asarin and Oded Maler. "As Soon as Possible: Time Optimal Control for Timed Automata". In: *Hybrid Systems: Computation and Control, Second International Workshop, HSCC'99, Berg en Dal, The Netherlands, March 29-31, 1999, Proceedings*. Ed. by Frits W. Vaandrager and Jan H. van Schuppen. Vol. 1569. Lecture Notes in Computer Science. Springer, 1999, pp. 19–30. DOI: 10.1007/3-540-48983-5_6. URL: https://doi.org/10.1007/3-540-48983-5%5C_6.

[Bai+07] Christel Baier, Nathalie Bertrand, Patricia Bouyer, Thomas Brihaye, and Marcus Größer. "Probabilistic and Topological Semantics for Timed Automata". In: *FSTTCS 2007: Foundations of Software Technology and Theoretical Computer Science, 27th International Conference, New Delhi, India, December 12-14, 2007, Proceedings*. Ed. by Vikraman Arvind and Sanjiva Prasad. Vol. 4855. Lecture Notes in Computer Science. Springer, 2007, pp. 179–191. DOI: 10.1007/978-3-540-77050-3_15. URL: https://doi.org/10.1007/978-3-540-77050-3%5C_15.

[Bai+08] Christel Baier, Nathalie Bertrand, Patricia Bouyer, Thomas Brihaye, and Marcus Größer. "Almost-Sure Model Checking of Infinite Paths in One-Clock Timed Automata". In: *Twenty-Third Annual IEEE Symposium on Logic in Computer Science, LICS 2008, 24-27 June 2008, Pittsburgh, PA, USA*. IEEE Computer Society, 2008, pp. 217–226. DOI: 10.1109/LICS.2008.25. URL: https://doi.org/10.1109/LICS.2008.25.

[BK08] Christel Baier and Joost-Pieter Katoen. *Principles of model checking*. MIT Press, 2008. ISBN: 978-0-262-02649-9.

[Bea03] Danièle Beauquier. "On probabilistic timed automata". In: *Theoretical Computer Science* 292.1 (2003). Selected Papers in honor of Jean Berstel, pp. 65–84. ISSN: 0304-3975. DOI: 10.1016/S0304-3975(01)

00215-8. URL: https://doi.org/10.1016/S0304-3975(01)00215-8.

[Beh+07] Gerd Behrmann, Agnès Cougnard, Alexandre David, Emmanuel Fleury, Kim Guldstrand Larsen, and Didier Lime. "UPPAAL-Tiga: Time for Playing Games!" In: *Computer Aided Verification, 19th International Conference, CAV 2007, Berlin, Germany, July 3-7, 2007, Proceedings*. Ed. by Werner Damm and Holger Hermanns. Vol. 4590. Lecture Notes in Computer Science. Springer, 2007, pp. 121–125. DOI: 10.1007/978-3-540-73368-3_14. URL: https://doi.org/10.1007/978-3-540-73368-3%5C_14.

[BDL04] Gerd Behrmann, Alexandre David, and Kim G. Larsen. "Formal Methods for the Design of Real-Time Systems: International School on Formal Methods for the Design of Computer, Communication, and Software Systems, Bertinora, Italy, September 13-18, 2004, Revised Lectures". In: ed. by Marco Bernardo and Flavio Corradini. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004. Chap. A Tutorial on Uppaal, pp. 200–236. DOI: 10.1007/978-3-540-30080-9_7.

[Beh+01] Gerd Behrmann, Ansgar Fehnker, Thomas Hune, Kim Guldstrand Larsen, Paul Pettersson, Judi Romijn, and Frits W. Vaandrager. "Minimum-Cost Reachability for Priced Timed Automata". In: *Hybrid Systems: Computation and Control, 4th International Workshop, HSCC 2001, Rome, Italy, March 28-30, 2001, Proceedings*. Ed. by Maria Domenica Di Benedetto and Alberto L. Sangiovanni-Vincentelli. Vol. 2034. Lecture Notes in Computer Science. Springer, 2001, pp. 147–161. DOI: 10.1007/3-540-45351-2_15. URL: https://doi.org/10.1007/3-540-45351-2%5C_15.

[BMN00] Pierfrancesco Bellini, R. Mattonlini, and Paolo Nesi. "Temporal logics for real-time system specification". In: *ACM Comput. Surv.* 32.1 (2000), pp. 12–42. DOI: 10.1145/349194.349197. URL: https://doi.org/10.1145/349194.349197.

[Bér+98] Béatrice Bérard, Antoine Petit, Volker Diekert, and Paul Gastin. "Characterization of the Expressive Power of Silent Transitions in Timed Automata". In: *Fundamenta Informaticae* 36.2,3 (1998), pp. 145–182. ISSN: 0169-2968.

[BCJ09] Jasper Berendsen, Taolue Chen, and David N. Jansen. "Undecidability of cost-bounded reachability in priced probabilistic timed automata". In: *Theory and Applications of Models of Computation, 6th Annual Conference, TAMC 2009, Changsha, China, May 18-22, 2009. Proceedings*. Ed. by Jianer Chen and S. Barry Cooper. Springer.

Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 128–137. ISBN: 978-3-642-02017-9. DOI: 10.1007/978-3-642-02017-9_16. URL: https://doi.org/10.1007/978-3-642-02017-9%5C_16.

[BJK06] Jasper Berendsen, David N. Jansen, and Joost-Pieter Katoen. "Probably on Time and within Budget: On Reachability in Priced Probabilistic Timed Automata". In: *Third International Conference on the Quantitative Evaluation of Systems (QEST 2006), 11-14 September 2006, Riverside, California, USA*. IEEE Computer Society, 2006, pp. 311–322. DOI: 10.1109/QEST.2006.43. URL: https://doi.org/10.1109/QEST.2006.43.

[Ber+18] Nathalie Bertrand, Patricia Bouyer, Thomas Brihaye, and Pierre Carlier. "When are stochastic transition systems tameable?" In: *Journal of Logical and Algebraic Methods in Programming* 99 (2018), pp. 41–96. DOI: 10.1016/J.JLAMP.2018.03.004. URL: https://doi.org/10.1016/j.jlamp.2018.03.004.

[Ber+14] Nathalie Bertrand, Patricia Bouyer, Thomas Brihaye, Quentin Menet, Christel Baier, Marcus Größer, and Marcin Jurdzinski. "Stochastic Timed Automata". In: *Log. Methods Comput. Sci.* 10.4 (2014). DOI: 10.2168/LMCS-10(4:6)2014. URL: https://doi.org/10.2168/LMCS-10(4:6)2014.

[BS12] Nathalie Bertrand and Sven Schewe. "Playing Optimally on Timed Automata with Random Delays". In: *Formal Modeling and Analysis of Timed Systems - 10th International Conference, FORMATS 2012, London, UK, September 18-20, 2012. Proceedings*. Ed. by Marcin Jurdzinski and Dejan Nickovic. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 43–58. DOI: 10.1007/978-3-642-33365-1_5. URL: https://doi.org/10.1007/978-3-642-33365-1%5C_5.

[BT91] Dimitri P. Bertsekas and John N. Tsitsiklis. "An Analysis of Stochastic Shortest Path Problems". In: *Math. Oper. Res.* 16.3 (1991), pp. 580–595. DOI: 10.1287/MOOR.16.3.580. URL: https://doi.org/10.1287/moor.16.3.580.

[BP22] Dirk Beyer and Andreas Podelski. "Principles of Systems Design: Essays Dedicated to Thomas A. Henzinger on the Occasion of His 60th Birthday". In: ed. by Jean-FrançoisRaskin, Krishnendu Chatterjee, Laurent Doyen, and Rupak Majumdar. Springer Nature Switzerland, Dec. 2022. Chap. Software Model Checking: 20 Years and Beyond, pp. 554–582. ISBN: 978-3-031-22336-5. DOI: 10.1007/978-3-031-22337-2_27.

[Bou16] Patricia Bouyer. "Erratum to the FSTTCS'04 paper "Optimal Strategies in Priced Timed Game Automata"". Personal Communication. 2016.

[Bou+11] Patricia Bouyer, Romain Brenguier, Nicolas Markey, and Michael Ummels. "Nash equilibria in concurrent games with Büchi objectives". In: *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2011)*. Ed. by Supratik Chakraborty and Amit Kumar. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2011. DOI: 10.4230/LIPICS.FSTTCS.2011.375. URL: https://doi.org/10.4230/LIPIcs.FSTTCS.2011.375.

[Bou+07] Patricia Bouyer, Thomas Brihaye, Véronique Bruyère, and Jean-François Raskin. "On the Optimal Reachability Problem of Weighted Timed Automata". In: *Formal Methods in System Design* 31.2 (2007), pp. 135–175. DOI: 10.1007/S10703-007-0035-4. URL: https://doi.org/10.1007/s10703-007-0035-4.

[Bou+16] Patricia Bouyer, Thomas Brihaye, Pierre Carlier, and Quentin Menet. "Compositional Design of Stochastic Timed Automata". In: *Computer Science - Theory and Applications - 11th International Computer Science Symposium in Russia, CSR 2016, St. Petersburg, Russia, June 9-13, 2016, Proceedings*. Ed. by Alexander S. Kulikov and Gerhard J. Woeginger. Vol. 9691. Lecture Notes in Computer Science. Springer, 2016, pp. 117–130. DOI: 10.1007/978-3-319-34171-2_9. URL: https://doi.org/10.1007/978-3-319-34171-2%5C_9.

[BBM06] Patricia Bouyer, Thomas Brihaye, and Nicolas Markey. "Improved Undecidability Results on Weighted Timed Automata". In: *Information Processing Letters* 98.5 (2006), pp. 188–194. DOI: 10.1016/J.IPL.2006.01.012. URL: https://doi.org/10.1016/j.ipl.2006.01.012.

[Bou+20] Patricia Bouyer, Thomas Brihaye, Mickael Randour, Cédric Rivière, and Pierre Vandenhove. "Decisiveness of Stochastic Systems and its Application to Hybrid Models". In: *11th International Symposium on Games, Automata, Logics, and Formal Verification, GandALF 2020, Brussels, Belgium, September 21-22, 2020*. Ed. by Davide Bresolin and Jean-François Raskin. Vol. 326. Electronic Proceedings in Theoretical Computer Science. 2020. DOI: 10.4204/EPTCS.326.10. URL: https://doi.org/10.4204/EPTCS.326.10.

[Bou+04a] Patricia Bouyer, Franck Cassez, Emmanuel Fleury, and Kim G. Larsen. "Optimal Strategies in Priced Timed Game Automata". In: *FSTTCS 2004: Foundations of Software Technology and Theoretical Computer*

Science, 24th International Conference, Chennai, India, December 16-18, 2004, Proceedings. Ed. by Kamal Lodaya and Meena Mahajan. Long version at https://doi.org/10.7146/brics.v11i4.21829. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 148–160. DOI: 10.1007/978-3-540-30538-5_13. URL: https://doi.org/10.1007/978-3-540-30538-5%5C_13.

[Bou+04b] Patricia Bouyer, Catherine Dufourd, Emmanuel Fleury, and Antoine Petit. "Updatable timed automata". In: *Theor. Comput. Sci.* 321.2-3 (2004), pp. 291–345. DOI: 10.1016/j.tcs.2004.04.003. URL: https://doi.org/10.1016/j.tcs.2004.04.003.

[BFM15] Patricia Bouyer, Erwin Fang, and Nicolas Markey. "Permissive strategies in timed automata and games". In: *Electron. Commun. Eur. Assoc. Softw. Sci. Technol.* 72 (2015). DOI: 10.14279/tuj.eceasst.72.1015. URL: https://doi.org/10.14279/tuj.eceasst.72.1015.

[BF09] Patricia Bouyer and Vojtěch Forejt. "Reachability in Stochastic Timed Games". In: *Automata, Languages and Programming, 36th Internatilonal Colloquium, ICALP 2009, Rhodes, Greece, July 5-12, 2009, Proceedings, Part II*. Ed. by Susanne Albers, Alberto Marchetti-Spaccamela, Yossi Matias, and Wolfgang Thomas. Vol. 5556. Lecture Notes in Computer Science. Springer, 2009, pp. 103–114. DOI: 10.1007/978-3-642-02930-1_9. URL: https://doi.org/10.1007/978-3-642-02930-1%5C_9.

[BJM15] Patricia Bouyer, Samy Jaziri, and Nicolas Markey. "On the Value Problem in Weighted Timed Games". In: *26th International Conference on Concurrency Theory, CONCUR 2015, Madrid, Spain, September 1.4, 2015*. Ed. by Luca Aceto and David de Frutos-Escrig. Vol. 42. Leibniz International Proceedings in Informatics. Leibniz-Zentrum für Informatik, 2015, pp. 311–324. DOI: 10.4230/LIPICS.CONCUR.2015.311. URL: https://doi.org/10.4230/LIPIcs.CONCUR.2015.311.

[Bou+06] Patricia Bouyer, Kim G. Larsen, Nicolas Markey, and Jacob Illum Rasmussen. "Almost Optimal Strategies in One Clock Priced Timed Games". In: *FSTTCS 2006: Foundations of Software Technology and Theoretical Computer Science, 26th International Conference, Kolkata, India, December 13-15, 2006, Proceedings*. Ed. by S. Arun-Kumar and Naveen Garg. Vol. 4337. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 345–356. DOI: 10.1007/11944836_32. URL: https://doi.org/10.1007/11944836%5C_32.

[BMS13] Patricia Bouyer, Nicolas Markey, and Ocan Sankur. "Robust Weighted Timed Automata and Games". In: *Formal Modeling and Analysis of Timed Systems - 11th International Conference, FORMATS 2013, Buenos Aires, Argentina, August 29-31, 2013. Proceedings*. Ed. by Vıctor A. Braberman and Laurent Fribourg. Vol. 8053. Lecture Notes in Computer Science. Springer, 2013, pp. 31–46. DOI: 10.1007/978-3-642-40229-6_3. URL: https://doi.org/10.1007/978-3-642-40229-6%5C_3.

[BMS15] Patricia Bouyer, Nicolas Markey, and Ocan Sankur. "Robust reachability in timed automata and games: A game-based approach". In: *Theor. Comput. Sci.* 563 (2015), pp. 43–74. DOI: 10.1016/J.TCS.2014.08.014. URL: https://doi.org/10.1016/j.tcs.2014.08.014.

[Bou+21] Patricia Bouyer, Youssouf Oualhadj, Mickael Randour, and Pierre Vandenhove. "Arena-Independent Finite-Memory Determinacy in Stochastic Games". In: *32nd International Conference on Concurrency Theory, CONCUR 2021, August 24-27, 2021, Virtual Conference*. Ed. by Serge Haddad and Daniele Varacca. Vol. 203. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021, 26:1–26:18. DOI: 10.4230/LIPICS.CONCUR.2021.26. URL: https://doi.org/10.4230/LIPIcs.CONCUR.2021.26.

[BRV23] Patricia Bouyer, Mickael Randour, and Pierre Vandenhove. "Characterizing Omega-Regularity through Finite-Memory Determinacy of Games on Infinite Graphs". In: *TheoretiCS* 2 (2023). DOI: 10.46298/THEORETICS.23.1. URL: https://doi.org/10.46298/theoretics.23.1.

[Bou+22] Patricia Bouyer, Stéphane Le Roux, Youssouf Oualhadj, Mickael Randour, and Pierre Vandenhove. "Games Where You Can Play Optimally with Arena-Independent Finite Memory". In: *Log. Methods Comput. Sci.* 18.1 (2022). DOI: 10.46298/LMCS-18(1:11)2022. URL: https://doi.org/10.46298/lmcs-18(1:11)2022.

[Boz+98] Marius Bozga, Conrado Daws, Oded Maler, Alfredo Olivero, Stavros Tripakis, and Sergio Yovine. "Kronos: A Model-Checking Tool for Real-Time Systems". In: *Computer Aided Verification, 10th International Conference, CAV '98, Vancouver, BC, Canada, June 28 - July 2, 1998, Proceedings*. Ed. by Alan J. Hu and Moshe Y. Vardi. Vol. 1427. Lecture Notes in Computer Science. Springer, 1998, pp. 546–550. DOI: 10.1007/BFB0028779. URL: https://doi.org/10.1007/BFb0028779.

[BCR14] Romain Brenguier, Franck Cassez, and Jean-François Raskin. "Energy and mean-payoff timed games". In: *17th International Conference on Hybrid Systems: Computation and Control (part of CPS Week), HSCC'14, Berlin, Germany, April 15-17, 2014*. Ed. by Martin Fränzle and John Lygeros. ACM, 2014, pp. 283–292. DOI: 10.1145/2562059.2562116. URL: https://doi.org/10.1145/2562059.2562116.

[BBR05] Thomas Brihaye, Véronique Bruyère, and Jean-François Raskin. "On Optimal Timed Strategies". In: *Formal Modeling and Analysis of Timed Systems, Third International Conference, FORMATS 2005, Uppsala, Sweden, September 26-28, 2005, Proceedings*. Ed. by Paul Pettersson and Wang Yi. Vol. 3829. Lecture Notes in Computer Science. Springer, 2005, pp. 49–64. DOI: 10.1007/11603009_5. URL: https://doi.org/10.1007/11603009%5C_5.

[Bri+15] Thomas Brihaye, Gilles Geeraerts, Axel Haddad, Engel Lefaucheux, and Benjamin Monmege. "Simple Priced Timed Games are not That Simple". In: *35th IARCS Annual Conference on Foundation of Software Technology and Theoretical Computer Science, FSTTCS 2015, December 16-18, 2015, Bangalore, India*. Ed. by Prahladh Harsha and G. Ramalingam. Vol. 45. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2015, pp. 278–292. DOI: 10.4230/LIPICS.FSTTCS.2015.278. URL: https://doi.org/10.4230/LIPIcs.FSTTCS.2015.278.

[Bri+22] Thomas Brihaye, Gilles Geeraerts, Axel Haddad, Engel Lefaucheux, and Benjamin Monmege. "One-Clock Priced Timed Games with Negative Weights". In: *Log. Methods Comput. Sci.* 18.3 (2022). DOI: 10.46298/lmcs-18(3:17)2022. URL: https://doi.org/10.46298/lmcs-18(3:17)2022.

[Bri+17] Thomas Brihaye, Gilles Geeraerts, Axel Haddad, and Benjamin Monmege. "Pseudopolynomial Iterative Algorithm to Solve Total-Payoff Games and Min-Cost Reachability Games". In: *Acta Informatica* 54.1 (2017), pp. 85–125. DOI: 10.1007/S00236-016-0276-Z. URL: https://doi.org/10.1007/s00236-016-0276-z.

[Bri+14] Thomas Brihaye, Gilles Geeraerts, Shankara Narayanan Krishna, Lakshmi Manasa, Benjamin Monmege, and Ashutosh Trivedi. "Adding Negative Prices to Priced Timed Games". In: *CONCUR 2014 - Concurrency Theory - 25th International Conference, CONCUR 2014, Rome, Italy, September 2-5, 2014. Proceedings*. Ed. by Paolo Baldan and Daniele Gorla. Vol. 8704. Lecture Notes in Computer Science. Springer, 2014, pp. 560–575. DOI: 10.1007/978-3-662-44584-

6_38. URL: https://doi.org/10.1007/978-3-662-44584-6%5C_38.

[Bus19] Damien Busatto-Gaston. "Symbolic controller synthesis for timed systems: robustness and optimality". PhD thesis. Aix-Marseille Université, 2019. URL: https://tel.archives-ouvertes.fr/tel-02436831.

[BMR17] Damien Busatto-Gaston, Benjamin Monmege, and Pierre-Alain Reynier. "Optimal Reachability in Divergent Weighted Timed Games". In: *Foundations of Software Science and Computation Structures - 20th International Conference, FOSSACS 2017, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2017, Uppsala, Sweden, April 22-29, 2017, Proceedings*. Ed. by Javier Esparza and Andrzej S. Murawski. Vol. 10203. Lecture Notes in Computer Science. 2017, pp. 162–178. DOI: 10.1007/978-3-662-54458-7_10. URL: https://doi.org/10.1007/978-3-662-54458-7%5C_10.

[BMR18] Damien Busatto-Gaston, Benjamin Monmege, and Pierre-Alain Reynier. "Symbolic Approximation of Weighted Timed Games". In: *38th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2018, December 11-13, 2018, Ahmedabad, India*. Ed. by Sumit Ganguly and Paritosh Pandya. Vol. 122. Leibniz International Proceedings in Informatics (LIPIcs). Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2018, 28:1–28:16. DOI: 10.4230/LIPICS.FSTTCS.2018.28. URL: https://doi.org/10.4230/LIPIcs.FSTTCS.2018.28.

[BMR23] Damien Busatto-Gaston, Benjamin Monmege, and Pierre-Alain Reynier. "Optimal controller synthesis for timed systems". In: *Log. Methods Comput. Sci.* 19.1 (2023). DOI: 10.46298/lmcs-19(1:20)2023. URL: https://doi.org/10.46298/lmcs-19(1:20)2023.

[Bus+19] Damien Busatto-Gaston, Benjamin Monmege, Pierre-Alain Reynier, and Ocan Sankur. "Robust Controller Synthesis in Timed Büchi Automata: A Symbolic Approach". In: *Computer Aided Verification - 31st International Conference, CAV 2019, New York City, NY, USA, July 15-18, 2019, Proceedings, Part I*. Ed. by Isil Dillig and Serdar Tasiran. Vol. 11561. Lecture Notes in Computer Science. Springer, 2019, pp. 572–590. DOI: 10.1007/978-3-030-25540-4_33. URL: https://doi.org/10.1007/978-3-030-25540-4%5C_33.

[CKS81] Ashok K. Chandra, Dexter C. Kozen, and Larry J. Stockmeyer. "Alternation". In: *J. ACM* 28.1 (1981), pp. 114–133. ISSN: 0004-5411. DOI: 10.1145/322234.322243.

[CAH04] Krishnendu Chatterjee, Luca de Alfaro, and Thomas A. Henzinger. "Trading Memory for Randomness". In: *1st International Conference on Quantitative Evaluation of Systems (QEST 2004), 27-30 September 2004, Enschede, The Netherlands*. QEST '04. IEEE Computer Society, 2004, pp. 206–217. DOI: 10.1109/QEST.2004.1348035. URL: https://doi.org/10.1109/QEST.2004.1348035.

[CDH10] Krishnendu Chatterjee, Laurent Doyen, and Thomas A. Henzinger. "Quantitative languages". In: *ACM Trans. Comput. Log.* 11.4 (2010), 23:1–23:38. DOI: 10.1145/1805950.1805953. URL: https://doi.org/10.1145/1805950.1805953.

[CHJ05] Krishnendu Chatterjee, Thomas A. Henzinger, and Marcin Jurdzinski. "Mean-Payoff Parity Games". In: *20th IEEE Symposium on Logic in Computer Science (LICS 2005), 26-29 June 2005, Chicago, IL, USA, Proceedings*. IEEE Computer Society, 2005, pp. 178–187. DOI: 10.1109/LICS.2005.26. URL: https://doi.org/10.1109/LICS.2005.26.

[CHP08a] Krishnendu Chatterjee, Thomas A. Henzinger, and Vinayak S Prabhu. "Timed Parity Games: Complexity and Robustness". In: *Formal Modeling and Analysis of Timed Systems, 6th International Conference, FORMATS 2008, Saint Malo, France, September 15-17, 2008. Proceedings*. Ed. by Franck Cassez and Claude Jard. Vol. 5215. Lecture Notes in Computer Science. Springer, 2008, pp. 124–140. DOI: 10.1007/978-3-540-85778-5_10. URL: https://doi.org/10.1007/978-3-540-85778-5%5C_10.

[CHP08b] Krishnendu Chatterjee, Thomas A. Henzinger, and Vinayak S. Prabhu. "Trading Infinite Memory for Uniform Randomness in Timed Games". In: *Hybrid Systems: Computation and Control, 11th International Workshop, HSCC 2008, St. Louis, MO, USA, April 22-24, 2008. Proceedings*. Ed. by Magnus Egerstedt and Bud Mishra. Vol. 4981. Lecture Notes in Computer Science. Springer, 2008, pp. 87–100. DOI: 10.1007/978-3-540-78929-1_7. URL: https://doi.org/10.1007/978-3-540-78929-1%5C_7.

[CHP11] Krishnendu Chatterjee, Thomas A. Henzinger, and Vinayak S. Prabhu. "Timed Parity Games: Complexity and Robustness". In: *Log. Methods Comput. Sci.* 7.4 (2011). DOI: 10.2168/LMCS-7(4:8)2011. URL: https://doi.org/10.2168/LMCS-7(4:8)2011.

[CRR13] Krishnendu Chatterjee, Mickael Randour, and Jean-François Raskin. "Strategy Synthesis for Multi-Dimensional Quantitative Objectives". In: *Acta Informatica* 51.3-4 (2013), pp. 129–163. DOI: 10.1007/s00236-013-0182-6. URL: https://doi.org/10.1016/j.jcss.2012.12.001.

[CRR14] Krishnendu Chatterjee, Mickael Randour, and Jean-François Raskin. "Strategy synthesis for multi-dimensional quantitative objectives". In: *Acta Informatica* 51.3-4 (2014), pp. 129–163. DOI: 10.1007/s00236-013-0182-6. URL: https://doi.org/10.1007/s00236-013-0182-6.

[Chu63] Alonzo Church. "Application of Recursive Arithmetic to the Problem of Circuit Synthesis". In: *Journal of Symbolic Logic* 28.4 (1963), pp. 289–290. DOI: 10.2307/2271310.

[Cim01] Alessandro Cimatti. "Industrial Applications of Model Checking". In: *Modeling and Verification of Parallel Processes: 4th Summer School, MOVEP 2000 Nantes, France, June 19–23, 2000 Revised Tutorial Lectures*. Ed. by Franck Cassez, Claude Jard, Brigitte Rozoy, and Mark Dermot Ryan. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 153–168. DOI: 10.1007/3-540-45510-8_6. URL: https://doi.org/10.1007/3-540-45510-8_6.

[Cla+18] Edmund M. Clarke, Thomas A. Henzinger, Helmut Veith, and Roderick Bloem, eds. *Handbook of Model Checking*. Springer, 2018. ISBN: 978-3-319-10574-1. DOI: 10.1007/978-3-319-10575-8. URL: https://doi.org/10.1007/978-3-319-10575-8.

[Cle+20] Emily Clement, Thierry Jéron, Nicolas Markey, and David Mentré. "Computing Maximally-Permissive Strategies in Acyclic Timed Automata". In: *Formal Modeling and Analysis of Timed Systems - 18th International Conference, FORMATS 2020, Vienna, Austria, September 1-3, 2020, Proceedings*. Ed. by Nathalie Bertrand and Nils Jansen. Cham: Springer International Publishing, 2020, pp. 111–126. DOI: 10.1007/978-3-030-57628-8_7. URL: https://doi.org/10.1007/978-3-030-57628-8%5C_7.

[CC77] Patrick Cousot and Radhia Cousot. "Abstract Interpretation: A Unified Lattice Model for Static Analysis of Programs by Construction or Approximation of Fixpoints". In: *Conference Record of the Fourth ACM Symposium on Principles of Programming Languages, Los Angeles, California, USA, January 1977*. Ed. by Robert M. Graham, Michael A. Harrison, and Ravi Sethi. ACM, 1977, pp. 238–252. DOI:

10.1145/512950.512973. URL: https://doi.org/10.1145/512950.512973.

[Dav+15] Alexandre David, Peter Gjøl Jensen, Kim Guldstrand Larsen, Marius Mikučionis, and Jakob Haahr Taankvist. "Uppaal Stratego". In: *Tools and Algorithms for the Construction and Analysis of Systems - 21st International Conference, TACAS 2015, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2015, London, UK, April 11-18, 2015. Proceedings*. Ed. by Christel Baier and Cesare Tinelli. Vol. 9035. Lecture Notes in Computer Science. Springer, 2015, pp. 206–211. DOI: 10.1007/978-3-662-46681-0_16. URL: https://doi.org/10.1007/978-3-662-46681-0%5C_16.

[Dav+10] Alexandre David, Kim Larsen, Axel Legay, Ulrik Nyman, and Andrzej Wasowski. "Timed I/O automata: a complete specification theory for real-time systems". In: *13th ACM International Conference on Hybrid Systems: Computation and Control, HSCC 2010, Stockholm, Sweden, April 12-15, 2010*. Ed. by Karl Henrik Johansson and Wang Yi. ACM, 2010, pp. 91–100. DOI: 10.1145/1755952.1755967. URL: https://doi.org/10.1145/1755952.1755967.

[dM04] Luca de Alfaro and Rupak Majumdar. "Quantitative solution of omega-regular games". In: *Journal of Computer and System Sciences* 68.2 (2004). Special Issue on STOC 2001, pp. 374–397. ISSN: 0022-0000. DOI: https://doi.org/10.1016/j.jcss.2003.07.009. URL: https://www.sciencedirect.com/science/article/pii/S0022000003001429.

[DG07] Manfred Droste and Paul Gastin. "Weighted automata and weighted logics". In: *Theoretical Computer Science* 380 (2007). Automata, Languages and Programming, pp. 69–86. ISSN: 0304-3975. DOI: 10.1016/J.TCS.2007.02.055.

[EM79] A. Ehrenfeucht and J. Mycielski. "Positional Strategies for Mean Payoff Games". In: *Int. J. Game Theory* 8.2 (1979), pp. 109–113. ISSN: 0020-7276. DOI: 10.1007/BF01768705. URL: https://doi.org/10.1007/BF01768705.

[ÉFE08] and Thomas Chatain Étienne André, Laurent Fribourg, and Emmanuelle Encrenaz. "An Inverse Method for Parametric Timed Automata". In: *Second Workshop on Reachability Problems in Computational Models, RP 2008, Liverpool, UK, September 15-17, 2008*. Ed. by Vesa Halava and Igor Potapov. Vol. 223. Electronic Notes in Theoretical Computer Science. Elsevier, 2008, pp. 29–46. DOI: 10.1016/

J.ENTCS.2008.12.029. URL: https://doi.org/10.1016/j.entcs.2008.12.029.

[FIS20] John Fearnley, Rasmus Ibsen-Jensen, and Rahul Savani. "One-Clock Priced Timed Games are PSPACE-hard". In: *LICS '20: 35th Annual ACM/IEEE Symposium on Logic in Computer Science, Saarbrücken, Germany, July 8-11, 2020*. Ed. by Holger Hermanns, Lijun Zhang, Naoki Kobayashi, and Dale Miller. ACM, 2020, pp. 397–409. DOI: 10.1145/3373718.3394772. URL: https://doi.org/10.1145/3373718.3394772.

[Fij+23] Nathanaël Fijalkow, Nathalie Bertrand, Patricia Bouyer-Decitre, Romain Brenguier, Arnaud Carayol, John Fearnley, Hugo Gimbert, Florian Horn, Rasmus Ibsen-Jensen, Nicolas Markey, Benjamin Monmege, Petr Novotný, Mickael Randour, Ocan Sankur, Sylvain Schmitz, Olivier Serre, and Mateusz Skomra. *Games on Graphs*. 2023. arXiv: 2305.10546.

[For+16] Vojtěch Forejt, Marta Kwiatkowska, Gethin Norman, and Ashutosh Trivedi. "Expected Reachability-Time Games". In: *Theoretical Computer Science* 631 (June 2016), pp. 139–160. DOI: 10.1016/J.TCS.2016.04.021. URL: https://doi.org/10.1016/j.tcs.2016.04.021.

[GS09] Thomas Martin Gawlitza and Helmut Seidl. "Games through Nested Fixpoints". In: *Computer Aided Verification, 21st International Conference, CAV 2009, Grenoble, France, June 26 - July 2, 2009. Proceedings*. Ed. by Ahmed Bouajjani and Oded Maler. Vol. 5643. Lecture Notes in Computer Science. Springer, 2009, pp. 291–305. DOI: 10.1007/978-3-642-02658-4_24. URL: https://doi.org/10.1007/978-3-642-02658-4%5C_24.

[GZ04] Hugo Gimbert and Wieslaw Zielonka. "When Can You Play Positionally?" In: Lecture Notes in Computer Science 3153 (2004). Ed. by Jirı Fiala, Václav Koubek, and Jan Kratochvıl, pp. 686–697. DOI: 10.1007/978-3-540-28629-5_53. URL: https://doi.org/10.1007/978-3-540-28629-5%5C_53.

[GZ05] Hugo Gimbert and Wieslaw Zielonka. "Games Where You Can Play Optimally Without Any Memory". In: *CONCUR 2005 - Concurrency Theory, 16th International Conference, CONCUR 2005, San Francisco, CA, USA, August 23-26, 2005, Proceedings*. Ed. by Martın Abadi and Luca de Alfaro. Vol. 3653. Lecture Notes in Computer Science. Springer, 2005, pp. 428–442. DOI: 10.1007/11539452_33. URL: https://doi.org/10.1007/11539452%5C_33.

[GTW02] Erich Grädel, Wolfgang Thomas, and Thomas Wilke, eds. *Automata, Logics, and Infinite Games: A Guide to Current Research [outcome of a Dagstuhl seminar, February 2001]*. Vol. 2500. Lecture Notes in Computer Science. Springer, 2002. ISBN: 3-540-00388-6. DOI: 10.1007/3-540-36387-4. URL: https://doi.org/10.1007/3-540-36387-4.

[GJ95] H. Gregersen and H. E. Jensen. *Formal Design of Reliable Real Time Systems*. Master's thesis, Department of Mathematics and Computer Science. 1995.

[Guh+15] Shibashis Guha, Shankara Narayanan Krishna, Lakshmi Manasa, and Ashutosh Trivedi. "Revisiting Robustness in Priced Timed Games". In: *35th IARCS Annual Conference on Foundation of Software Technology and Theoretical Computer Science, FSTTCS 2015, December 16-18, 2015, Bangalore, India*. Ed. by Prahladh Harsha and G. Ramalingam. Vol. 45. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2015, pp. 261–277. DOI: 10.4230/LIPICS.FSTTCS.2015.261. URL: https://doi.org/10.4230/LIPIcs.FSTTCS.2015.261.

[GHJ97] Vineet Gupta, Thomas A. Henzinger, and Radha Jagadeesan. "Robust Timed Automata". In: *Hybrid and Real-Time Systems, International Workshop. HART'97, Grenoble, France, March 26-28, 1997, Proceedings*. Ed. by Oded Maler. Vol. 1201. Lecture Notes in Computer Science. Springer, 1997, pp. 331–345. DOI: 10.1007/BFB0014736. URL: https://doi.org/10.1007/BFb0014736.

[HIM13] Thomas Dueholm Hansen, Rasmus Ibsen-Jensen, and Peter Bro Miltersen. "A Faster Algorithm for Solving One-Clock Priced Timed Games". In: *CONCUR 2013 - Concurrency Theory - 24th International Conference, CONCUR 2013, Buenos Aires, Argentina, August 27-30, 2013. Proceedings*. Ed. by Pedro R. D'Argenio and Hernán C. Melgratti. Vol. 8052. Lecture Notes in Computer Science. Springer, 2013, pp. 531–545. DOI: 10.1007/978-3-642-40184-8_37. URL: https://doi.org/10.1007/978-3-642-40184-8%5C_37.

[HMP91] Thomas A. Henzinger, Zohar Manna, and Amir Pnueli. "Timed Transition Systems". In: *Real-Time: Theory in Practice, REX Workshop, Mook, The Netherlands, June 3-7, 1991, Proceedings*. Ed. by J. W. de Bakker, Cornelis Huizing, Willem P. de Roever, and Grzegorz Rozenberg. Vol. 600. Lecture Notes in Computer Science. Springer, 1991, pp. 226–251. DOI: 10.1007/BFB0031995. URL: https://doi.org/10.1007/BFb0031995.

[HS06] Thomas A. Henzinger and Joseph Sifakis. "The Embedded Systems Design Challenge". In: *FM 2006: Formal Methods, 14th International Symposium on Formal Methods, Hamilton, Canada, August 21-27, 2006, Proceedings*. Ed. by Jayadev Misra, Tobias Nipkow, and Emil Sekerinski. Vol. 4085. Lecture Notes in Computer Science. Springer, 2006, pp. 1–15. DOI: 10.1007/11813040_1. URL: https://doi.org/10.1007/11813040%5C_1.

[HSW12] Frédéric Herbreteau, B. Srivathsan, and Igor Walukiewicz. "Efficient emptiness check for timed Büchi automata". In: *Formal Methods Syst. Des.* 40.2 (2012), pp. 122–146. DOI: 10.1007/S10703-011-0133-1. URL: https://doi.org/10.1007/s10703-011-0133-1.

[Hor09] Florian Horn. "Random Fruits on the Zielonka Tree". In: *26th International Symposium on Theoretical Aspects of Computer Science, STACS 2009, February 26-28, 2009, Freiburg, Germany, Proceedings*. Ed. by Susanne Albers and Jean-Yves Marion. Vol. 3. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Germany, 2009, pp. 541–552. DOI: 10.4230/LIPIcs.STACS.2009.1848. URL: https://doi.org/10.4230/LIPIcs.STACS.2009.1848.

[JLR19] Aleksandra Jovanovic, Didier Lime, and Olivier H. Roux. "A game approach to the parametric control of real-time systems". In: *Int. J. Control* 92.9 (2019), pp. 2025–2036. DOI: 10.1080/00207179.2018.1426883. URL: https://doi.org/10.1080/00207179.2018.1426883.

[JSL08] Marcin Jurdzinski, Jeremy Sproston, and François Laroussinie. "Model Checking Probabilistic Timed Automata with One or Two Clocks". In: *Logical Methods in Computer Science* 4.3 (2008). DOI: 10.2168/LMCS-4(3:12)2008. URL: https://doi.org/10.2168/LMCS-4(3:12)2008.

[JT07] Marcin Jurdzinski and Ashutosh Trivedi. "Reachability-Time Games on Timed Automata". In: *Automata, Languages and Programming, 34th International Colloquium, ICALP 2007, Wroclaw, Poland, July 9-13, 2007, Proceedings*. Ed. by Lars Arge, Christian Cachin, Tomasz Jurdzinski, and Andrzej Tarlecki. Vol. 4596. Lecture Notes in Computer Science. Springer, 2007, pp. 838–849. DOI: 10.1007/978-3-540-73420-8_72. URL: https://doi.org/10.1007/978-3-540-73420-8%5C_72.

[Kar+18] Anil Kumar Karna, Yuting Chen, Haibo Yu, Hao Zhong, and Jianjun Zhao. "The role of model checking in software engineering". In: *Frontiers of Computer Science* 12.4 (2018), pp. 642–668. DOI: 10.1007/

S11704-016-6192-0. URL: https://doi.org/10.1007/s11704-016-6192-0.

[KSK76] John G. Kemeny, J. Laurie Snell, and Anthony W. Knapp. *Denumerable Markov Chains*. Graduate Texts in Mathematics. Springer-Verlag, 1976. ISBN: 9780387901770. URL: https://books.google.fr/books?id=porsRg4NDM8C.

[Kha+08] Leonid Khachiyan, Endre Boros, Konrad Borys, Khaled M. Elbassioni, Vladimir Gurvich, Gábor Rudolf, and Jihui Zhao. "On Short Paths Interdiction Problems: Total and Node-Wise Limited Interdiction". In: *Theory Comput. Syst.* 43.2 (2008), pp. 204–233. DOI: 10.1007/S00224-007-9025-6. URL: https://doi.org/10.1007/s00224-007-9025-6.

[Knu77] Donald E. Knuth. "A generalization of Dijkstra's algorithm". In: *Information Processing Letters* 6.1 (1977), pp. 1–5. ISSN: 0020-0190. DOI: https://doi.org/10.1016/0020-0190(77)90002-3. URL: https://www.sciencedirect.com/science/article/pii/0020019077900023.

[Koz22] Alexander Kozachinskiy. "One-To-Two-Player Lifting for Mildly Growing Memory". In: *39th International Symposium on Theoretical Aspects of Computer Science, STACS 2022, March 15-18, 2022, Marseille, France (Virtual Conference)*. Ed. by Petra Berenbrink and Benjamin Monmege. Vol. 219. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022, 43:1–43:21. DOI: 10.4230/LIPICS.STACS.2022.43. URL: https://doi.org/10.4230/LIPIcs.STACS.2022.43.

[KAY10] Pavel Krcal, Parosh Aziz Abdulla, and Wang Yi. "Sampled Semantics of Timed Automata". In: *Logical Methods in Computer Science* Volume 6, Issue 3 (2010). DOI: 10.2168/LMCS-6(3:14)2010. URL: https://lmcs.episciences.org/868.

[KR65] Kazimierz Kuratowski and Czeslaw Ryll-Nardzewski. "A general theorem on selectors". English. In: *Bull. Acad. Pol. Sci., Sér. Sci. Math. Astron. Phys.* 13 (1965), pp. 397–403. ISSN: 0001-4117.

[KNP19] Marta Kwiatkowska, Gethin Norman, and David Parker. "Verification and Control of Turn-Based Probabilistic Real-Time Games". In: *Art of Modelling Computational Systems: A Journey from Logic and Concurrency to Security and Privacy - Essays Dedicated to Catuscia Palamidessi on the Occasion of Her 60th Birthday*. Ed. by Mário S. Alvim, Kostas Chatzikokolakis, Carlos Olarte, and Frank Valencia. Vol. 11760. Lecture Notes in Computer Science. Springer, 2019, pp. 379–396. DOI: 10.1007/978-3-030-31175-9_22.

[Kwi+18] Marta Kwiatkowska, Gethin Norman, David Parker, and Gabriel Santos. "Automated Verification of Concurrent Stochastic Games". In: *Quantitative Evaluation of Systems - 15th International Conference, QEST 2018, Beijing, China, September 4-7, 2018, Proceedings*. Ed. by Annabelle McIver and András Horváth. Vol. 11024. Lecture Notes in Computer Science. Springer, 2018, pp. 223–239. DOI: 10.1007/978-3-319-99154-2_14. URL: https://doi.org/10.1007/978-3-319-99154-2%5C_14.

[Kwi+21] Marta Kwiatkowska, Gethin Norman, David Parker, and Gabriel Santos. "Automatic verification of concurrent stochastic systems". In: *Formal Methods Syst. Des.* 58.1-2 (2021), pp. 188–250. DOI: 10.1007/S10703-020-00356-Y. URL: https://doi.org/10.1007/s10703-020-00356-y.

[Kwi+06] Marta Z. Kwiatkowska, Gethin Norman, David Parker, and Jeremy Sproston. "Performance analysis of probabilistic timed automata using digital clocks". In: *Formal Methods Syst. Des.* 29.1 (2006), pp. 33–78. DOI: 10.1007/S10703-006-0005-2. URL: https://doi.org/10.1007/s10703-006-0005-2.

[Kwi+02] Marta Z. Kwiatkowska, Gethin Norman, Roberto Segala, and Jeremy Sproston. "Automatic verification of real-time systems with discrete probability distributions". In: *Theor. Comput. Sci.* 282.1 (2002), pp. 101–150. DOI: 10.1016/S0304-3975(01)00046-9. URL: https://doi.org/10.1016/S0304-3975(01)00046-9.

[LMS04] François Laroussinie, Nicolas Markey, and Philippe Schnoebelen. "Model Checking Timed Automata with One or Two Clocks". In: *CONCUR 2004 - Concurrency Theory, 15th International Conference, London, UK, August 31 - September 3, 2004, Proceedings*. Ed. by Philippa Gardner and Nobuko Yoshida. Vol. 3170. Lecture Notes in Computer Science. Springer, 2004, pp. 387–401. DOI: 10.1007/978-3-540-28644-8_25. URL: https://doi.org/10.1007/978-3-540-28644-8%5C_25.

[Lar+01] Kim Guldstrand Larsen, Gerd Behrmann, Ed Brinksma, Ansgar Fehnker, Thomas Hune, Paul Pettersson, and Judi Romijn. "As Cheap as Possible: Efficient Cost-Optimal Reachability for Priced Timed Automata". In: *Computer Aided Verification, 13th International Conference, CAV 2001, Paris, France, July 18-22, 2001, Proceedings*. Ed. by Gérard Berry, Hubert Comon, and Alain Finkel. Vol. 2102. Lecture Notes in Computer Science. Springer, 2001, pp. 493–505. DOI: 10.1007/3-

540-44585-4_47. URL: https://doi.org/10.1007/3-540-44585-4%5C_47.

[LSD08] Yang Liu, Jun Sun, and Jin Song Dong. "An analyzer for extended compositional process algebras". In: *30th International Conference on Software Engineering (ICSE 2008), Leipzig, Germany, May 10-18, 2008, Companion Volume*. Ed. by Wilhelm Schäfer, Matthew B. Dwyer, and Volker Gruhn. ACM, 2008, pp. 919–920. DOI: 10.1145/1370175.1370187. URL: https://doi.org/10.1145/1370175.1370187.

[Mar11] Nicolas Markey. "Robustness in real-time systems". In: *Industrial Embedded Systems (SIES), 2011 6th IEEE International Symposium on, SIES 2011. Vasteras, Sweden, June 15-17, 2011*. IEEE, 2011, pp. 28–34. DOI: 10.1109/SIES.2011.5953652. URL: https://doi.org/10.1109/SIES.2011.5953652.

[Min67] Marvin L. Minsky. *Computation: Finite and Infinite Machines*. Prentice-Hall Series in Automatic Computation. Prentice-Hall, 1967.

[MPR20] Benjamin Monmege, Julie Parreaux, and Pierre-Alain Reynier. "Reaching Your Goal Optimally by Playing at Random with No Memory". In: *31st International Conference on Concurrency Theory, CONCUR 2020, September 1-4, 2020, Vienna, Austria (Virtual Conference)*. Ed. by Igor Konnov and Laura Kovács. Vol. 171. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020, 26:1–26:21. DOI: 10.4230/LIPICS.CONCUR.2020.26. URL: https://doi.org/10.4230/LIPIcs.CONCUR.2020.26.

[MPR21] Benjamin Monmege, Julie Parreaux, and Pierre-Alain Reynier. "Playing Stochastically in Weighted Timed Games to Emulate Memory". In: *48th International Colloquium on Automata, Languages, and Programming, ICALP 2021, July 12-16, 2021, Glasgow, Scotland (Virtual Conference)*. Ed. by Nikhil Bansal, Emanuela Merelli, and James Worrell. Vol. 198. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021, 137:1–137:17. DOI: 10.4230/LIPICS.ICALP.2021.137. URL: https://doi.org/10.4230/LIPIcs.ICALP.2021.137.

[MPR22] Benjamin Monmege, Julie Parreaux, and Pierre-Alain Reynier. "Decidability of One-Clock Weighted Timed Games with Arbitrary Weights". In: *33rd International Conference on Concurrency Theory, CONCUR 2022, September 12-16, 2022, Warsaw, Poland*. Ed. by Bartek Klin, Slawomir Lasota, and Anca Muscholl. Vol. 243. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022, 15:1–15:22. DOI: 10.4230/

LIPICS.CONCUR.2022.15. URL: https://doi.org/10.4230/LIPIcs.CONCUR.2022.15.

[MPR23a] Benjamin Monmege, Julie Parreaux, and Pierre-Alain Reynier. "Decidability of One-Clock Weighted Timed Games with Arbitrary Weights". Under submission in LMCS, special issue CONCUR 2022. 2023. arXiv: 2207.01608.

[MPR23b] Benjamin Monmege, Julie Parreaux, and Pierre-Alain Reynier. "Playing Stochastically in Weighted Timed Games to Emulate Memory". Under submission in LMCS. 2023. arXiv: 2105.00984.

[Nas50] John F. Nash. "Equilibrium Points in n-Person Games". In: *Proceedings of the National Academy of Sciences of the United States of America* 36.1 (1950), pp. 48–49.

[NM47] John von Neumann and Oskar Morgenstern. *Theory of games and economic behavior*. Princeton University Press, 1947.

[Ost92] Jonathan S. Ostroff. "Formal methods for the specification and design of real-time safety critical systems". In: *Journal of Systems and Software* 18.1 (1992), pp. 33–60. DOI: 10.1016/0164-1212(92)90045-L. URL: https://doi.org/10.1016/0164-1212(92)90045-L.

[ORS14] Youssouf Oualhadj, Pierre-Alain Reynier, and Ocan Sankur. "Probabilistic Robust Timed Games". In: *CONCUR 2014 - Concurrency Theory - 25th International Conference, CONCUR 2014, Rome, Italy, September 2-5, 2014. Proceedings*. Ed. by Paolo Baldan and Daniele Gorla. Vol. 8704. Lecture Notes in Computer Science. Springer, 2014, pp. 203–217. DOI: 10.1007/978-3-662-44584-6_15. URL: https://doi.org/10.1007/978-3-662-44584-6%5C_15.

[RS11] Markus N. Rabe and Sven Schewe. "Finite optimal control for time-bounded reachability in CTMDPs and continuous-time Markov games". In: *Acta Informatica* 48.5-6 (2011), pp. 291–315. DOI: 10.1007/S00236-011-0140-0. URL: https://doi.org/10.1007/s00236-011-0140-0.

[Rut11] Michal Rutkowski. "Two-Player Reachability-Price Games on Single-Clock Timed Automata". In: *Ninth Workshop on Quantitative Aspects of Programming Languages, QAPL 2011, Saarbrücken, Germany, April 1-3, 2011*. Ed. by Mieke Massink and Gethin Norman. Vol. 57. EPTCS. 2011, pp. 31–46. DOI: 10.4204/EPTCS.57.3. URL: https://doi.org/10.4204/EPTCS.57.3.

[San13a] Ocan Sankur. "Robustness in timed automata : analysis, synthesis, implementation. (Robustesse dans les automates temporisés : analyse, synthèse, implémentation)". PhD thesis. École normale supérieure de Cachan, Paris, France, 2013. URL: https://tel.archives-ouvertes.fr/tel-00910333.

[San13b] Ocan Sankur. "Shrinktech: A Tool for the Robustness Analysis of Timed Automata". In: *Computer Aided Verification - 25th International Conference, CAV 2013, Saint Petersburg, Russia, July 13-19, 2013. Proceedings*. Ed. by Natasha Sharygina and Helmut Veith. Vol. 8044. Lecture Notes in Computer Science. Springer, 2013, pp. 1006–1012. DOI: 10.1007/978-3-642-39799-8_72. URL: https://doi.org/10.1007/978-3-642-39799-8%5C_72.

[SBM11] Ocan Sankur, Patricia Bouyer, and Nicolas Markey. "Shrinking Timed Automata". In: *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2011)*. Ed. by Supratik Chakraborty and Amit Kumar. Vol. 13. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2011, pp. 90–102. DOI: 10.4230/LIPIcs.FSTTCS.2011.90. URL: https://doi.org/10.4230/LIPIcs.FSTTCS.2011.90.

[SBM14] Ocan Sankur, Patricia Bouyer, and Nicolas Markey. "Shrinking timed automata". In: *Inf. Comput.* 234 (2014), pp. 107–132. DOI: 10.1016/J.IC.2014.01.002. URL: https://doi.org/10.1016/j.ic.2014.01.002.

[San+13] Ocan Sankur, Patricia Bouyer, Nicolas Markey, and Pierre-Alain Reynier. "Robust Controller Synthesis in Timed Automata". In: *CONCUR 2013 - Concurrency Theory - 24th International Conference, CONCUR 2013, Buenos Aires, Argentina, August 27-30, 2013. Proceedings*. Ed. by Pedro R. D'Argenio and Hernán C. Melgratti. Vol. 8052. Lecture Notes in Computer Science. Springer, 2013, pp. 546–560. DOI: 10.1007/978-3-642-40184-8_38. URL: https://doi.org/10.1007/978-3-642-40184-8%5C_38.

[Tar55] Alfred Tarski. "A lattice-theoretical fixpoint theorem and its applications." In: *Pacific Journal of Mathematics* 5.2 (1955), pp. 285–309.

[Tho95] Wolfgang Thomas. "On the Synthesis of Strategies in Infinite Games". In: *STACS 95, 12th Annual Symposium on Theoretical Aspects of Computer Science, Munich, Germany, March 2-4, 1995, Proceedings*. Ed. by Ernst W. Mayr and Claude Puech. Vol. 900. Lecture Notes in Computer Science. Springer, 1995, pp. 1–13. DOI: 10.1007/3-540-

59042-0_57. URL: https://doi.org/10.1007/3-540-59042-0%5C_57.

[Win93] Glynn Winskel. *The formal semantics of programming languages - an introduction*. Foundation of computing series. MIT Press, 1993. ISBN: 978-0-262-23169-5.

[WDR05] Martin De Wulf, Laurent Doyen, and Jean-François Raskin. "Almost ASAP semantics: from timed models to timed implementations". In: *Formal Aspects Comput.* 17.3 (2005), pp. 319–341. DOI: 10.1007/S00165-005-0067-8. URL: https://doi.org/10.1007/s00165-005-0067-8.

[Yov97] Sergio Yovine. "KRONOS: A Verification Tool for Real-Time Systems". In: *Int. J. Softw. Tools Technol. Transf.* 1.1-2 (1997), pp. 123–133. DOI: 10.1007/S100090050009. URL: https://doi.org/10.1007/s100090050009.

[Zie98] Wieslaw Zielonka. "Infinite Games on Finitely Coloured Graphs with Applications to Automata on Infinite Trees". In: *Theor. Comput. Sci.* 200.1-2 (1998), pp. 135–183. DOI: 10.1016/S0304-3975(98)00009-7. URL: https://doi.org/10.1016/S0304-3975(98)00009-7.

[ZP96] Uri Zwick and Mike Paterson. "The Complexity of Mean Payoff Games on Graphs". In: *Theoretical Computer Science* 158.1&2 (1996), pp. 343–359. DOI: 10.1016/0304-3975(95)00188-3. URL: https://doi.org/10.1016/0304-3975(95)00188-3.