Cassting

Deliverable D2.4

Efficient strategy synthesis for complex objectives

Nicolas Markey (CNRS),
Namit Chaturvedi (RWTH), Gilles Geeraerts (ULB),
Jiří Srba (AAU)

Project	Cassting — Collective Adaptive Systems SynThesIs with Non-zero-sum Games		
Project id.	FP7-601148		
Workpackage	WP2	Nature	R (report)
Deliverable	D2.4	Dissemination	PU (public)
Date submitted	September 2014	Date due	September 2014
Version	1.0 (final)		

SEVENTH FRAMEWORK PROGRAMME

Efficient strategy synthesis for complex objectives

Nicolas Markey (CNRS) Namit Chaturvedi (RWTH)
Gilles Geeraerts (ULB) Jiří Srba (AAU)

Abstract

This deliverable surveys the main contributions of the Cassting consortium regarding synthesis of strategies in games played on graphs, with a particular focus on efficient algorithms (and possibly implementations) to synthesize strategies, and on complex objectives.

Contents

Introduction

Developing efficient algorithms and tools for synthesizing strategies is one of the main objectives of the Cassting project. This deliverable summarizes the results we obtained during the first half of the project. Our contributions have been numerous and, as we shall see, applies on a broad range of games and objectives. We decided to arrange our contributions depending on the class of games they focus.

The first contribution thus considers games played on finite graphs: these are the most basic models we will consider, where two players in turn try to move a token along the edges of a graph in order to get an infinite run satisfying their objectives. Such games form a basic block in the setting of games for synthesis, and more complex games are often reduced to such simple games, hence the importance of making progresses in understanding them. The contribution we present here deals with simple safety and reachability objectives, and proposes a way of computing succinct representations of strategies using antichains.

The second area where we contrubuted concerns stochastic games, and more precisely Markov Decision Processes. Again using antichains, we developed a so-called *symblicit* algorithm for synthesizing optimal strategies in MDPs with several kinds of objectives. *Symblicit* algorithms are strategy-iteration algorithms combining symbolic and explicit representations of the (usually large) state space. We show that using antichains as the symbolic data structure can be efficient, in particular for an important subclass of MDPs and for mean-payoff and shortest-path objectives.

We then turn to games on weighted graphs: such games are basic models for representing simple quantitative aspects such as energy consumption. The quantitative aspects gives rise to various optimization objectives (trying to minimize energy consumption while achieving one's objective), as well as safety objectives (maintaining a reasonable level some quantity all along the play). We developed efficient algorithms for the former, as we present in Section 3.1. We also considered the non-zero-sum setting on weighted games, following earlier works on this topic. We develop in Section 3.2 our findings about the existence and computation of secure Nash equilibria in weighted games.

The fourth family of models we considered is timed games. There our contributions are two-fold: first, we studied timed games with assume-guarantee objectives expressed in a fragment of the temporal logic MTL. Using an approximate translation of the specification into *deterministic* timed automata, we get an efficient (but incomplete) method for computing strategies in that setting. Second, we focused on *priced* timed games: on the one hand, we implemented a recent algorithms for computing optimal winning strategies in 1-clock priced timed games; on the other hand, we considered the case of a randomized environment, and developed an algorithm for computing strategies that ensure given time-bounds in the worst case, and at the same time (near-)optimize the expected cost. This is achieved by considering most permissive strategies, and refining it by evaluating the effects of various control choices using statistical model checking.

Finally, our last contribution concerns distributed games: our aim there is to lift the classical approach using (Büchi) word automata for synthesis to traces, which are a suitable model for representing the runs of distributed systems. For this, we introduced new kinds of automata running over traces and over trace-closed word languages, and proved that those automata have similar properties as in the world of words, thus opening the way to controller-synthesis algorithms for distributed systems.

1 Strategy synthesis in games on graphs

Background. Finite, turn-based, games are a very simple, yet relevant, class of games. They are played by two players ($\mathcal{S}$ and $\mathcal{R}$) on a finite graph (called the arena), whose set of vertices is partitioned into Player $\mathcal{S}$ and Player $\mathcal{R}$ vertices. A play is an infinite path in this graph, obtained by letting the players move a token on the vertices. Initially, the token is on a designated initial vertex. At each round of the game, the player who owns the vertex marked by the token decides on which successor node to move it next. A play is winning for $\mathcal{R}$ if the token eventually touches some designated *bad* nodes (the objective for $\mathcal{R}$ is thus a reachability objective), otherwise it is winning for $\mathcal{S}$ (for whom the objective is a safety objective). Such games are a natural model to describe the interaction of a potential controller with a given environment, where the aim of the controller (modeled by Player $\mathcal{S}$) is to avoid the bad states that model system failures. In this framework, computing a winning strategy for the player amounts to synthesising a control policy that guarantees no bad state will be reached, no matter how the environment behaves. Safety games have also been used as a tool to solve other problems such as LTL realisability, real-time scheduler synthesis, or timed automata determinisation.

1.1 Efficient representation and computation of strategies using antichains

It is well-known that memory-less winning strategies (i.e., that depend only on the current state) are sufficient for both players in those games. Memoryless strategies are often regarded as simple and straightforward to implement (remember that the winning strategy is very often the actual control policy that we want to implement in, say, an embedded controller). Yet, this belief falls short in many practical applications such as the three mentioned above because the arena is not given explicitly, and its size is at least exponential in the size of the original problem instance. Hence, the computation of winning strategies might be intractable in practice because it could request to traverse the whole arena. Moreover, a naive implementation of a winning strategy by means of a table mapping each $\mathcal{S}$-state v to its safe successor $\sigma(v)$, is not realistic because this table would have the size of the arena.

Our contributions We considered the problem of computing efficiently winning strategies that can be succinctly represented. To formalise this problem, we introduced $\star$-strategies, which are partial functions defined on a subset of $\mathcal{S}$-states only. A $\star$-strategy can be regarded as an abstract representation of a family of (plain) strategies, which we call concretisations of the $\star$-strategies (they are all the strategies that agree with the $\star$-strategy).

In order to keep the description of winning $\star$-strategies succinct, and to obtain efficient algorithms to compute them, we proposed heuristics inspired from the antichain line of research [DR10]. These heuristics have been developed mainly in the verification setting, to deal with automata-based models. Roughly speaking, they rely on a simulation partial order on the states of the system, which is exploited to prune the state space that the algorithms need to explore, and to obtain efficient data structures to store the set of states that the algorithms need to maintain.

We introduced general antichain-based techniques for solving reachability and safety games, and computing efficiently succinct representations of winning strategies. We proposed a general and elegant theory which is built on top of the notion of turn-based alternating simulation (tba-simulation for

[DR10] Laurent Doyen and Jean-François Raskin. Antichain algorithms for finite automata. In Javier Esparza and Rupak Majumdar, editors, *Proceedings of the 16th International Conference on Tools and Algorithms for Construction and Analysis of Systems (TACAS'10)*, volume 6015 of *Lecture Notes in Computer Science*, pages 2–22. Springer-Verlag, March 2010.

short, a notion adapted from [AHKV98]), instead of simulation. We showed that, in general, it is sufficient to store the strategy on the maximal antichain of the reachable winning states. We then presented an efficient on-the-fly algorithm to compute such succinct $\star$-strategies (adapted from the classical OTFUR algorithm to solve reachability games [CDF$^+$05]). Our algorithm generalises the algorithm of Filiot *et al.* [FJR09], with several improvements:

- it applies to a general class of games whose arena is equipped with a tba-simulation (not only those generated from an instance of the LTL realisability problem);
- it contains an additional heuristic that was not present in [FJR09];
- its proof of correctness is straightforward, and stems directly from the definition of tba-simulation.

Finally, we proved that our approach can be straightforwardly applied to the games one obtains in three applications, namely LTL realisability, real-time feasibility and determinisation of timed automata, which demonstrates the wide applicability of our approach.

Participants:
Gilles Geeraerts (ULB)
Joël Goossens (ULB)
Amélie Stainer (ULB)

References: [GGS14]

[AHKV98] Rajeev Alur, Thomas A. Henzinger, Orna Kupferman, and Moshe Y. Vardi. Alternating refinement relations. In Davide Sangiorgi and Robert de Simone, editors, *Proceedings of the 9th International Conference on Concurrency Theory (CONCUR'98)*, volume 1466 of *Lecture Notes in Computer Science*, pages 163–178. Springer-Verlag, September 1998.

[CDF$^+$05] Franck Cassez, Alexandre David, Emmanuel Fleury, Kim Gulstrand Larsen, and Didier Lime. Efficient on-the-fly algorithms for the analysis of timed games. In Martín Abadi and Luca de Alfaro, editors, *Proceedings of the 16th International Conference on Concurrency Theory (CONCUR'05)*, volume 3653 of *Lecture Notes in Computer Science*, pages 66–80. Springer-Verlag, August 2005.

[FJR09] Emmanuel Filiot, Naiyong Jin, and Jean-François Raskin. An antichain algorithm for LTL realizability. In Ahmed Bouajjani and Oded Maler, editors, *Proceedings of the 21st International Conference on Computer Aided Verification (CAV'09)*, volume 5643 of *Lecture Notes in Computer Science*, pages 263–277. Springer-Verlag, June 2009.

2 Strategy synthesis in stochastic systems

Background. Markov decision processes (MDPs) are rich models that exhibit both nondeterministic choices and stochastic transitions. Model-checking and synthesis algorithms for MDPs exist for various kinds of properties (temporal-logic properties, quantitative properties such as the long-run average reward (mean-payoff) or the stochastic shortest path). Those algorithms have been implemented in several tools (PRISM, MODEST, QUASY, ...)

There are two main families of algorithms for MDPs:

- *value-iteration* algorithms assign values to states of the MDPs and refines locally those values by successive approximations. If a fixpoint is reached, the value at a state s represents a probability or an expectation that can be achieved by an optimal strategy that resolves the choices present in the MDP starting from s. This value can be, for example, the maximal probability to reach a set of goal states.

- *strategy-iteration* algorithms start from an arbitrary strategy and iteratively improve the current strategy by local changes up to the convergence to an optimal strategy.

Both methods have their advantages and disadvantages. Value iteration algorithms usually lead to easy and efficient implementations, but in general the fixpoint is not guaranteed to be reached in a finite number of iterations, and so only approximations are computed. On the other hand, strategy iteration algorithms have better theoretical properties as convergence towards an optimal strategy in a finite number of steps is usually ensured, but they often require to solve systems of linear equations, and so they are more difficult to implement efficiently.

When considering large MDPs, that are obtained from high level descriptions or as the product of several components, explicit methods often exhaust available memory and are thus impractical. This is the manifestation of the well-known state explosion problem. In non-probabilistic systems, symbolic data structures such as binary decision diagrams (BDDs) have been investigated to mitigate this phenomenon. For probabilistic systems, multi-terminal BDDs (MTBDDs) are useful but they are usually limited to systems with around 10^{10} states. Also, as mentioned above, some algorithms for MDPs rely on solving linear systems, and there is no easy use of BDD like structures for implementing such algorithms.

2.1 Symblicit algorithms for optimal strategy synthesis in monotonic Markov decision processes

Recently, Wimmer *et al.* [WBB+10] have proposed a method that mixes symbolic and explicit representations to efficiently implement the Howard and Veinott strategy-iteration algorithm to synthesize optimal strategies for mean-payoff objectives in MDPs. Their solution is as follows. First, the MDP is represented and handled symbolically using MTBDDs. Second, a strategy is fixed symbolically and the MDP is transformed into a Markov chain (MC). To analyze this MC, a linear system needs to be constructed from its state space. As this state space is potentially huge, the MC is first reduced by lumping (bisimulation reduction), and then a (hopefully) compact linear system can be constructed and solved. Solutions to this linear system allow to show that the current strategy is optimal, or to obtain sufficient information to improve it. A new iteration is then started. The main difference between this method and the other methods proposed in the literature is its hybrid nature: it is symbolic for handling the MDP and for computing the lumping, and it is explicit for the analysis of the reduced MC.

Our contributions Our contributions are threefold:

- we showed that the symblicit approach and strategy iteration can also be efficiently applied to the stochastic shortest path problem. We start from an algorithm proposed by Bertsekas and Tsitsiklis [BT96] with a preliminary step of de Alfaro [dA99], and we show how to cast it in the symblicit approach.
- we showed that alternative data structures can be more efficient than BDDs or MTBDDs for implementing a symblicit approach, both for

[WBB+10] Ralf Wimmer, Bettina Braitling, Bernd Becker, Pepijn Crouzen, Holger Hermanns, Abhishek Dhama, and Oliver E. Theel. Symblicit calculation of long-run averages for concurrent probabilistic systems. In *Proceedings of the 7th International Conference on Quantitative Evaluation of Systems (QEST'10)*, pages 27–36. IEEE Comp. Soc. Press, September 2010.

[BT96] Dimitri P. Bertsekas and John N. Tsitsiklis. *Neuro-Dynamic Programming*, volume 3 of *Anthropological Field Studies*. Athena Scientific, 1996.

[dA99] Luca de Alfaro. Computing minimum and maximum reachability times in probabilistic systems. In Jos C. M. Baeten and Sjouke Mauw, editors, *Proceedings of the 10th International Conference on Concurrency Theory (CONCUR'99)*, volume 1664 of *Lecture Notes in Computer Science*, pages 66–81. Springer-Verlag, August 1999.

mean-payoff and stochastic shortest path objectives. In particular, we considered a natural class of MDPs with monotonic properties on which our alternative data structure is more efficient. For such MDPs, as for subset constructions in automata theory, antichain-based data structures usually behave better than BDDs. The application of antichains to monotonic MDPs requires nontrivial extensions: for instance, to handle the lumping step, we need to generalize existing antichain based data structures in order to be closed under negation. To this end, we introduce a new data structure called pseudo-antichain.

- we have implemented our algorithms and we show that they are more efficient than existing solutions on natural examples of monotonic MDPs. We proved that monotonic MDPs naturally arise in probabilistic planning and when optimizing controllers synthesized from LTL specifications with mean-payoff objectives.

Participants:
Aaron Bohy (UMONS)
Véronique Bruyère (UMONS)
Jea-François Raskin (ULB)

References: [BBR14]

3 Strategy synthesis in weighted systems

Background. In order to model quantitative aspects in games, such as energy harvesting and consumption for instance, graphs have been extended with weights. Such weighted games have been extensively studied in the literature, with various kinds of quantitative objectives such as minimization of the accumulated weight (this extends shortest-path problems to games), of the average weight (aka. *mean payoff* objectives), or of the discounted sum.

3.1 Synthesizing optimal strategies with mixed boolean/quantitative objectives

We studied weighted games with objectives combining reachability together with several variants of payoff functions. In that setting, a few results exist already when weights are non-negative [LMO06, KBB+08].

The shortest path problem becomes more complex when edges may have arbitrary weights (positive and negative). In the one-player case, Floyd-Warshall's algorithm computes all shortest paths (or detects negative cycles) in polynomial time. The two-player case is quoted as an open problem in [KBB+08]. Partial results exist: determining whether Player 1 can guarantee to reach the target with an accumulated negative weight is shown in $\mathsf{NP} \cap \mathsf{co} - \mathsf{NP}$.

There are two conventional ways to express the difficulty of games. The former measures the complexity of the winning or optimal strategies. For several classes of qualitative games, including those with reachability winning conditions, it is well-known that memoryless strategies (that depend only on the current vertex) are sufficient. This result carries on to the quantitative games with total-, mean- or discounted-payoff [GZ04]. Another measure is given by the computational complexity of calculating winning strategies (for qualitative games) or the optimal payoffs and strategies (for quantitative games). The existence of memoryless winning/optimal strategies in the aforementioned cases yields an $\mathsf{NP} \cap \mathsf{co} - \mathsf{NP}$ upper-bound. Nevertheless, variants of the iterative backward induction technique are often more efficient, both from theoretical and practical points of view. For instance, the attractor technique is a polynomial time algorithm for reachability games. Also, value iteration algorithms with pseudo-polynomial time complexities (i.e., polynomial if weights of the arena, respectively, the discount factor, is encoded in unary) exist for mean-, respectively, discounted-payoff games.

[LMO06] François Laroussinie, Nicolas Markey, and Ghassan Oreiby. Model-checking timed ATL for durational concurrent game structures. In Eugene Asarin and Patricia Bouyer, editors, *Proceedings of the 4th International Conferences on Formal Modelling and Analysis of Timed Systems, (FORMATS'06)*, volume 4202 of *Lecture Notes in Computer Science*, pages 245–259. Springer-Verlag, September 2006.

[KBB+08] Leonid Khachiyan, Endre Boros, Konrad Borys, Khaled Elbassioni, Vladimir Gurevich, Gabor Rudolf, and Jihui Zhao. On short paths interdiction problems: Total and node-wise limited interdiction. *Theory of Computing Systems*, 43(2):204–233, August 2008.

[GZ04] Hugo Gimbert and Wiesław Zielonka. When can you play positionnaly? In Jiří Fiala, Václav Koubek, and Jan Kratochíl, editors, *Proceedings of the 29th International Symposium on Mathematical Foundations of Computer Science (MFCS'04)*, volume 3153 of *Lecture Notes in Computer Science*, pages 686–697. Springer-Verlag, August 2004.

Our contributions We considered total-, mean- and discounted-payoff games with reachability objectives, and designed efficient (pseudo-polynomial time) algorithms to solve them. For reachability mean- and discounted-payoff games, we proved that they are logarithmic space inter-reducible with classical mean- and discounted-payoff games (i.e., without reachability objectives). Since pseudo-polynomial algorithms exist for the mean- and discounted- payoff games, we obtain pseudo-polynomial time algorithms for mean- and discounted- payoff games with reachability objectives.

Since no efficient algorithm exists for total-payoff games, we introduced a value iteration algorithm for reachability total-payoff games that runs in pseudo-polynomial time. We characterised the optimal strategies of both players: while Player 1 may need memory, Player 2 always has a memoryless optimal counter-strategy. We developed acceleration heuristics to improve the running time of the algorithm in practice. Finally, relying on a pseudo-polynomial time translation from total-payoff games to reachability total-payoff games, we obtained a pseudo-polynomial time algorithm for total payoff-games. Our acceleration techniques still apply in that case, producing a nontrivial class of total-payoff games that we are able to solve in polynomial time.

Participants:

Thomas Brihaye (UMONS)
Gilles Geeraerts (ULB)
Axel Haddad (UMONS)
Benjamin Monmege (ULB)

References: [BGHM14]

3.2 Secure equilibria in weighted games

Weighted games are also a natural model for considering non-zero-sum objectives, in case the systems we want to model are not purely adversarial. The classical notion of rational behavior in this context is commonly formalized as Nash equilibria [Nas50]. Nash equilibria capture rational behaviors when the players only care about their own payoff (internal criteria), and they are indifferent to the payoff of the other player (external criteria). In the setting of synthesis, the more appropriate notion is the adversarial external criteria, where the players are as harmful as possible to the other players without

[Nas50] John F. Nash, Jr. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences*, 36(1):48–49, January 1950.

sabotaging their own objectives. This has inspired the study of refinements of Nash equilibria, like the notion of secure equilibria, which captures the adversarial external criteria and is at the basis of compositional synthesis algorithms [CHJ06]. In secure equilibria, lexicographic objectives are considered: each player first tries to maximize his own payoff, and then tries to minimize the opponent's payoff.

Our contributions We extended the notion of secure equilibria from the Boolean setting with ω-regular objectives to a quantitative setting where objectives are non necessarily ω-regular. More precisely, we consider two-player non zero-sum turn-based games played on weighted graphs, called *weighted games*, with objectives defined by classical measures considered in the literature for infinite plays: sup, inf, lim sup, lim inf, mean-payoff, and discounted sum. In our setting, the edges of the weighted graph are labelled with pairs of rational values that are used to assign two values to each infinite play: one value models the reward of Player 1, and the other the reward of Player 2.

Our contributions are threefold:

- we proved that all weighted games with the classical measures have secure equilibria. We also establish that there exist simple profiles of strategies that witness such equilibria: finite-memory strategies with a linear memory size are sufficient for most of the measures (polynomial size for inf and sup measures).
- we designed polynomial or pseudo-polynomial (depending on the measures) algorithms for the automatic synthesis of such strategy profiles.
- we proved that one can decide the existence of a secure equilibrium whose outcome satisfies some constraints, for all measures except the discounted sum. In the latter case, we show that this problem is connected to a challenging open problem.

Our solutions rely on the analysis of two-player zero-sum games with lexicographic objectives (for all the measures) for which we provide worst-case optimal algorithms.

Participants:

[CHJ06] Krishnendu Chatterjee, Thomas A. Henzinger, and Marcin Jurdziński. Games with secure equilibria. *Theoretical Computer Science*, 365(1-2):67–82, November 2006.

Véronique Bruyère (UMONS)
Noémie Meunier (UMONS)
Jea-François Raskin (ULB)

References: [BMR14]

4 Strategy synthesis in timed games

Background. Timed automata [AD94] were introduced 20 years ago as a convenient extension of finite-state automata with *clocks*, hence opening the way to formal verification of real-time systems. Timed automata have been extensively studied since then, and this has led to the development of several efficient tools such as KRONOS or the UPPAAL tool suite. Timed automata have also been extended to timed games, in order to model multi-agent systems subject to real-time constraints. Again, this leads to decidable problems, and UPPAAL is able to decide the existence of winning strategies (and to compute them) in timed games with reachability or safety objectives (and combinations thereof).

4.1 Strategy synthesis for a fragment of MTL

The synthesis problem is computationally harder for linear time logics than the satisfiability and model-checking problems, and has for this reason been considered to be intractable for a long time. The main problem was that the first synthesis approaches involve the determinization of Büchi automata, which is a computationally hard problem. However, the synthesis problem has recently gained in practical performance with the development of the so-called Safraless synthesis algorithms that avoid the Büchi determinization phase. This approach has been strengthened by using the bounded synthesis, and the antichain-based algorithms.

Our contributions We studied the synthesis problem in a real-time setting, focusing our attention to the fragment of the Metric Interval Temporal Logic (MITL) [AFH91] that consists of assume-guarantee properties, ie, for-

[AD94] Rajeev Alur and David L. Dill. A theory of timed automata. *Theoretical Computer Science*, 126(2):183–235, April 1994.

[AFH91] Rajeev Alur, Tómas Feder, and Thomas A. Henzinger. The benefits of relaxing punctuality. In *Proceedings of the 10th ACM Symposium on Principles of Distributed Computing (PODC'91)*, pages 139–152. ACM Press, August 1991.

mulas of the form $\psi = (\phi_1 \wedge ... \wedge \phi_n) \Rightarrow (\psi_1 \wedge ... \wedge \psi_m)$ where we require that every ϕ_i and ψ_j belong to some fragment $\mathsf{safety\text{-}MTL}_{0,\infty}$ which we do not detail here. This formalism is powerful enough to express e.g. bounded-response properties such as $\mathbf{G}(a \Rightarrow \mathbf{F}_{\leq 1} b)$, which requires that every a should be followed by b within 1 time unit.

Our approach translates the overall specification ψ to a network of timed game automata—one automaton for every requirement ψ_j and one automaton for the negation of every assumption ϕ_i. A central contribution is the extension of the translation presented in [BDL+12], with the advantage that the resulting automata contain no Büchi acceptance conditions and are deterministic by construction. Hence, there is no need for a determinization phase. However, it is not always possible to translate $\mathsf{safety\text{-}MTL}_{0,\infty}$-properties into language-equivalent deterministic automata, and in this case we produce a deterministic under-approximation. Thus our approach is sound but not complete, i.e., it may fail to generate a strategy even if the specification is realizable. However, such theoretical incompleteness did not show up in our case studies.

It then remained to use Uppaal-Tiga to synthesize a control strategy for the resulting timed game fulfilling all the requirements $\{\psi_j\}$ or violating at least one of the assumptions $\{\phi_i\}$. Additionally, the synthesized strategy should avoid generating Zeno behaviour (playing infinite number of actions in a finite amount of time). We do this by either forcing Uppaal-Tiga to generate non-Zeno strategy using Büchi winning condition; or by proving that no Zeno strategy will be winning for the controller.

We applied this approach to several simple case studies in job-shop scheduling, dining philosophers, and buffer controller, showing promising results of our solution.

Participants:

Peter Bulychev (AAU)
Alexandre David (AAU)
Kim G. Larsen (AAU)

References: [BDLL14]

[BDL+12] Peter Bulychev, Alexandre David, Kim Gulstrand Larsen, Axel Legay, Guangyuan Li, Danny Bøgsted Poulsen, and Amélie Stainer. Monitor-based statistical model checking for weighted metric temporal logic. In Nikolaj Bjørner and Andrei Voronkov, editors, *Proceedings of the 18th International Conference Logic Programming and Automated Reasoning (LPAR'12)*, volume 7180 of *Lecture Notes in Computer Science*, pages 168–182. Springer-Verlag, March 2012.

4.2 Synthesizing optimal strategies in priced timed games

Priced timed games mix two quantitative extensions we already developed above. What is intersting and powerful here is that the weights may now depend on the time elapsed in each state, as opposed to plain weighted games where only transitions would carry a weight. Unfortunately, priced timed games have been shown undecidable in general [BBM06]. There are two common ways to circumvent such negative results: either restrict the model, or consider incomplete approaches.

Our contributions Our contributions here are twofold:

- on the one hand, we considered the restriction of weighted timed games to one-clock models: this severely limits the expressive power of automata, but still preserves intricate behaviors. Weighted timed games were proved decidable (more precisely, it is possible to compute the optimal weight that can be achieved from a given configuration) in [BLMR06, HIM13]. We implemented the algorithm of [HIM13], and merged part of their approach with the algorithm of [BLMR06] as it outperformed the former on intermediary models. Our implementation tends to confirm our intuition that our exponential-time upper bound on the computation time of the algorithm might not be tight, and we are currently trying to prove that the problem can actually be solved in polynomial time.
- on the other hand, we applied statistical model checking to tackle cost minimization that would still enforce worst-case time bounds. When

[BBM06] Patricia Bouyer, Thomas Brihaye, and Nicolas Markey. Improved undecidability results on weighted timed automata. *Information Processing Letters*, 98(5):188–194, June 2006.

[BLMR06] Patricia Bouyer, Kim Gulstrand Larsen, Nicolas Markey, and Jacob Illum Rasmussen. Almost optimal strategies in one-clock priced timed automata. In S. Arun-Kumar and Naveen Garg, editors, *Proceedings of the 26th Conferentce on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'06)*, volume 4337 of *Lecture Notes in Computer Science*, pages 345–356. Springer-Verlag, December 2006.

[HIM13] Thomas Dueholm Hansen, Rasmus Ibsen-Jensen, and Peter Bro Miltersen. A faster algorithm for solving one-clock priced timed games. In Pedro R. D'Argenio and Hernán C. Melgratt, editors, *Proceedings of the 24th International Conference on Concurrency Theory (CONCUR'13)*, volume 8052 of *Lecture Notes in Computer Science*, pages 531–545. Springer-Verlag, August 2013.

considering priced timed games, the classical approaches try to optimize the accumulated price to achieve some goal, with no constraints on the total time it may take to reach that goal. Our approach combines optimizing the total payoff in a Markov reward decision process while still guaranteeing a given worst-case time bound. For this, starting from a uniformized version of the most permissive strategy guaranteeing the given time bound, the learning technique iteratively improves the strategy—by observing the eect of control-choices in sampled runs—until a strategy with satisfactory expected reachability-cost is found. Crucial to the efficiency of our simulation-based synthesis method is the effective and space-efficient representation and manipulation of strategies. Besides the symbolic (zone-based) representation used for timed games, we consider a number of techniques well-known from Machine Intelligence (covariance matrices, logistic regression) as well as a new splitting data-structure of ours. The resulting method is implemented in a new version of Uppaal-Tiga that supports the statistical model-checking techniques of Uppaal. The experimental evaluation has been performed on a large collection of job-shop-like problems, demonstrating several order or magnitude improvements with respect to previous exact synthesis methods [KBM13].

Participants:

Samuel Dehouck (ULB)
Gilles Geeraerts (ULB)
Nicolas Markey (CNRS)
Benjamin Monmege (ULB)
Alexandre David (AAU)
Peter G. Jensen (AAU)
Kim G. Larsen (AAU)
Mathias G. Sørensen (AAU)
Jakob H. Taankvist (AAU)

References: [BGK+14, DJL+14]

[KBM13] Jean-François Kempf, Marius Bozga, and Oded Maler. As soon as probable: Optimal scheduling under stochastic uncertainty. In Nir Piterman and Scott A. Smolka, editors, *Proceedings of the 19th International Conference on Tools and Algorithms for Construction and Analysis of Systems (TACAS'13)*, volume 7795 of *Lecture Notes in Computer Science*, pages 385–400. Springer-Verlag, March 2013.

5 Strategy synthesis in distributed systems

Background. Church's synthesis problem consists in building a program realizing a given specification. While this problem can in general not be automatized, there are classes of specifications that admit (efficient) algorithms for synthesizing such programs. Regular specifications are a celebrated case where such synthesis is possible, as was proved by Büchi and Landweber in the late 1960's. These solutions rely on fundamental links between regular languages (of words) and automata. These links include the characterization of limit of regular languages in terms of deterministic Büchi automata.

5.1 Classification of trace languages with automata

When trying to lift this approach to distributed systems, we move from words to Mazurkiewicz traces, which are a convenient model for representing concurrent behaviors of finitely-many interacting processes. Traces are labeled directed acyclic graphs, where branching is used to represent the concurrent occurrence of independent events. A linearization of a trace is a word obtained by ordering (in an arbitrary way) the occurrences of independent events. The concept of "ω-regular trace language" can be introduced in close correspondence to the case of ω-regular word languages, for example, in terms of finite partially-commutative monoids, asynchronous automata, concurrent regular expressions, or MSO logic [DR95]. However, there does not exist a definition of Büchi automaton over traces that would allow for similar characterizations as we have in the setting of word languages. Such automata would open the way to efficient algorithms for controller synthesis for distributed systems.

Our contributions. We introduce a new concept of asynchronous automata, which we coined *synchronization-aware asynchronous automata*. These automata run over traces. By equiping them with Büchi or Muller acceptance conditions, we establish similar characterizations for trace languages as the ones that exist for word languages.

We also investigated traces in terms of their corresponding trace-equivalent words (which are essentially their linearizations): we introduced classes of automata over trace-closed word languages, again extending to traces several important characterizations that have been known for long for words.

[DR95] Volker Diekert and Grzegorz Rozenberg. *The Book of Traces.* World Scientific, 1995.

While these results do not explicitly offer efficient synthesis algorithms yet, they are a big step in that direction, and they open interesting avenues for further progress on the difficult problem of distributed-system synthesis.

Participants:
Namit Chaturvedi (AACHEN)
Marcus Gelderie (AACHEN)

References: [Cha14, CG14]

Conclusions

This deliverable contains a wealth of important algorithms and results for synthesizing strategies in a wide variety of games. All these results were obtained during the first half of the Cassting project. Additionally, we currently have several ongoing works extending the results presented above, which we hope will give many more synthesis algorithms during the second part of the project.

6 List of publications

[BBR14] Aaron Bohy, Véronique Bruyère, and Jean-François Raskin. Symblicit algorithms for mean-payoff and shortest path in monotonic markov decision processes. In Krishnendu Chatterjee, Rüdiger Ehlers, and Susmit Jha, editors, *Proceedings of the 3rd Workshop on Synthesis (SYNT'14)*, volume 157 of *Electronic Proceedings in Theoretical Computer Science*, pages 51–67, Vienna, Austria, July 2014.

[BDLL14] Peter Bulychev, Alexandre David, Kim G. Larsen, and Guangyuan Li. Efficient controller synthesis for a fragment of $\mathrm{MTL}_{0,\infty}$. *Acta Informatica*, 51(3-4):165–192, June 2014.

[BGHM14] Thomas Brihaye, Gilles Geeraerts, Axel Haddad, and Benjamin Monmege. To reach or not to reach? Efficient algorithms for total-payoff games. Research report 1407.5030, arXiv, 2014.

[BGK$^+$14] Thomas Brihaye, Gilles Geeraerts, Shankara Narayanan Krishna, Lakshmi Manasa, Benjamin Monmege, and Ashutosh Trivedi. Adding negative prices to priced timed games. In Paolo

Baldan and Daniele Gorla, editors, *Proceedings of the 25th International Conference on Concurrency Theory (CONCUR'14)*, volume 8704 of *Lecture Notes in Computer Science*, pages 560–575, Rome, Italy, September 2014. Springer.

[BMR14] Véronique Bruyère, Noëmie Meunier, and Jean-François Raskin. Secure equilibria in weighted games. In *Proceedings of the Joint Meeting of the 23rd EACSL Annual Conference on Computer Science Logic and the 29th Annual ACM/IEEE Symposium on Logic In Computer Science (CSL/LICS'14)*, Vienna, Austria, July 2014. ACM Press.

[CG14] Namit Chaturvedi and Marcus Gelderie. Classifying recognizable infinitary trace languages using word automata. In Erzsébet Csuhaj-Varjú, Martin Dietzfelbinger, and Zoltán Ésik, editors, *Proceedings of the 39th International Symposium on Mathematical Foundations of Computer Science (MFCS'14)*, volume 8634 of *Lecture Notes in Computer Science*, pages 171–182, Budapest, Hungary, August 2014. Springer.

[Cha14] Namit Chaturvedi. Toward a structure theory of regular infinitary trace languages. In Javier Esparza, Pierre Fraigniaud, and Elias Koutsoupias, editors, *Proceedings of the 41st International Colloquium on Automata, Languages and Programming (ICALP'14) – Part II*, volume 8573 of *Lecture Notes in Computer Science*, pages 134–145, Copenhagen, Denmark, July 2014. Springer.

[DJL+14] Alexandre David, Peter G. Jensen, Kim G. Larsen, Axel Legay, Didier Lime, Mathias G. Sørensen, and Jakob H. Taankvist. On time with minimal expected cost! In Franck Cassez and Jean-François Raskin, editors, *Proceedings of the 12th International Symposium on Automated Technology for Verification and Analysis (ATVA'14)*, volume 8837 of *Lecture Notes in Computer Science*, Sydnay, Australia, November 2014. Springer. To appear.

[GGS14] Gilles Geeraerts, Joël Goossens, and Amélie Stainer. Synthesising succinct strategies in safety games. In Joël Ouaknine, Igor Potapov, and James Worrell, editors, *Proceedings of the 8th Workshop on Reachability Problems in Computational Models (RP'14)*, volume 8762 of *Lecture Notes in Computer Science*, pages 98–111, Uppsala, Sweden, September 2014. Springer.