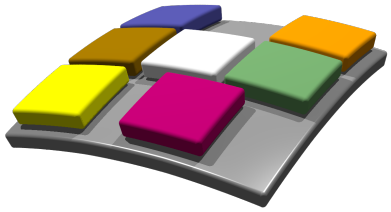

Cassting

Deliverable D3.1

Robustness of collective adaptive systems

Nicolas Markey (CNRS),
Thomas Brihaye (UMONS), Kim G. Larsen (AAU)

Project	Cassting — Collective Adaptive Systems SynThesIs with Non-zero-sum Games		
Project id.	FP7-601148		
Workpackage	WP3	**Nature**	R (report)
Deliverable	D3.1	**Dissemination**	PU (public)
Date submitted	March 2014	**Date due**	March 2014
Version	1.0		

SEVENTH FRAMEWORK PROGRAMME

Robustness of collective adaptive systems

Nicolas Markey, Thomas Brihaye, Kim G. Larsen

Abstract

This deliverable reports about our recent advances about robust verification of complex systems. The general aim of *robust* verification is to take into account unexpected behaviours that emerge because of imprecisions or perturbations in the global system: indeed, formal models usually assume perfect, zero-delay communications between components, and absolute precision of timing measurements, which does not perfectly fit with physical systems. Robust verification is thus an attempt to bridge this gap and make our models and techniques more faithful.

This deliverable summarizes the contributions achieved by the Casstting consortium. The list of publications on page 14 lists the corresponding papers, which contain the full details.

Contents

Introduction

Formal, model-based methods for verification (in particular model checking [CES09]) have been successfully applied to numerous cases over the last 30 years: they consist in representing a computerized system as a mathematical model (usually, a finite-state automaton or an extension thereof), and applying algorithms for checking properties on the whole set of executions of this model. The success of those techniques is mostly based on the fact that it is automatic and exhaustive: they are able to find corner-case behaviours that the system designers often overlook.

Because they are based on mathematical reasoning, those techniques are very rigid: for instance, in the setting of a distributed system, they will usually assume perfect communications and synchrony between the components (unless imprefection is explicitly included in the model). In the case of real-time systems, they will assume perfect measurement of time. In practice, these are too strong assumptions, especially in when modelling complex systems (such as collective adaptive systems, which are usually made of many distributed components, or embedded systems [HS06]). Robust verification can be seen as a relaxation of these hypotheses: it consists in taking imperfections of the physical systems into account, without adding them explcitly. In other terms, it amounts to changing the semantics of the models, or of the properties we evaluate on them.

During its first year of existence, the Cassting project already studied various aspects of this problem, as we report in this deliverable:

- first in the setting of real-time systems: this is perhaps the most blatant setting where robustness has to be taken into account. Indeed, timed automata [AD94], which are the most used model for reasonning about real-time systems, have been known for long to include unrealistic behaviours, with time-convergence phenomena. Zeno runs are the most famous example: they are runs along which infinitely many tran-

[CES09] Edmund M. Clarke, E. Allen Emerson, and Joseph Sifakis. Model checking: algorithmic verification and debugging. *Communications of the ACM*, 52(11):74–84, November 2009.

[HS06] Thomas A. Henzinger and Joseph Sifakis. The embedded systems design challenge. In Jayadev Misra, Tobias Nipkow, and Emil Sekerinski, editors, *Proceedings of the 14th International Symposium on Formal Methods (FM'06)*, volume 4085 of *Lecture Notes in Computer Science*, pages 1–15. Springer-Verlag, August 2006.

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

sitions are taken in finite time. Several approaches have already been proposed to make timed models more realistic. The most prominent one consists in enlarging all timing conditions by some parameter δ, and look for a positive value of this parameter under which the resulting model satisfies the expected propertes. We have developed this technique further, as we explain in Section 1.

- using a slightly different point of view, robustenss of a model can also be expressed in terms of distances: on the one hand, a *syntactic* distance can be defined as the value by which some constants in a quantitative model are modified (this would be the value of δ in our guard-enlargement approach above); on the other hand, several *semantic* distances have been defined over the last few years, in order to measure how different the behaviours from two states can be. Then a model is said robust when slight changes in terms of the syntactic distance involve only slight changes in terms of the semantic distance.

  In order to pursue this novel approach, we first need to get better knowledge about semantic distances. We present our recent advances on computing bisimulation distances between stochastic models in Section 2.

- finally, another way to make verification robust is to make it quantitative (as opposed to the current, purely boolean framework). This way, when a property is *almost* true, the model checker would return a value close to 1, instead of returning 0 (or `false`) as current model checkers would do. What *almost* means is of course crucial, and should reflect what we intend. We recently defined a quantitative semantics for the temporal logic LTL, involving average values of properties. We present our results in Section 3.

1 Robustness for real-time systems

1.1 Shrinking timed automata.

Background. Timed automata are widely used for verifying (qualitative or quantitative) properties of real-time systems. They extend finite-state automata with real-valued variables, named *clocks*, for measuring time: those clocks can be reset along transitions, and their values can be checked against

an integer in order to allow or forbid some transitions. Because clocks are real-valued, the resulting system has infinitely many states, but efficient symbolic techniques exist for effectively checking properties on those models.

Timed automata are usually not robust: it can be the case that a transition is only allowed as long as clock x is strictly less than c, for instance. however, such a constraint is way too precise for physical systems, because of the latency between the time at which the value of the clock is queried and the time at which the action is effective. In order to take such imprecisions into account, *guard enlargement* has been proposed: constraints of the form $x < c$ (and $x \leq c$) are relaxed into $x \leq c + \delta$, where δ is a parameter. A few years ago, we proved that the existence of a positive value for δ under which a given property holds is decidable.

Our contribution. In the approach to robustness depicted above (guard enlargement), the model unser study is the model that will be implemented: guard enlargement can be proved to precisely corresponds to the imprecisions of the hardware on which the model is implemented [DDR05].

Considering the problem from another point of view, we may want to see the model under study as the enveloppe of the behaviours we want the final model to exhibit. In other terms, we would be looking for another timed automaton whose behaviours, once implemented, are contained in the behaviours of the original model. Of course, we also want to preserve most of the behaviours of our original model.

A natural candidate for the automaton to be implemented is the automaton obtained by *shrinking* the timing constraints: shrinking is just the opposite of enlarging, consisting in reinforcing constraints of the form $x \leq c$ into $x \leq c - \delta$. This however may introduce deadlocks, or remove important sets of behaviours. We say that a timed automaton is *shrinkable* if there exists a value for δ for which the shrunk automaton has no deadlocks, and/or time-abstract simulates the original automaton. We proved that this property is decidable, and implemented a tool, named `shrinktech`, for deciding this property.

[DDR05] Martin De Wulf, Laurent Doyen, and Jean-François Raskin. Almost ASAP semantics: From timed models to timed implementations. *Formal Aspects of Computing*, 17(3):319–341, 2005.

Participants: Patricia Bouyer (CNRS)
Nicolas Markey (CNRS)
Ocan Sankur (ULB)

References: [SBM14]
`http://www.lsv.ens-cachan.fr/Software/shrinktech/`.

1.2 Robust controller synthesis

Background. The setting of controller synthesis is similar to that of model checking. However, the actions are now partitionned into *controllable* and *uncontrollable* ones; the question is then whether there is a strategy to trigger controllable actions in such a way that the resulting behaviours are correct.

This problem was defined long ago, and even in the timed setting, the problem is decidable (for reasonnable correctness requirements, such as reachability and safety). But obviously, robustness questions arise also in this setting, and the problem then is to synthesize *robust* strategies, which enforce the required property even in a perturbed setting.

Our contributions. Our first contribution in this domain is an extension of the *enlargement* approach: we considered a controller-synthesis problem for the case where all the actions are controllable, except the guard enlargement: in other terms, the controller proposes to take actions at given dates, and those dates can be modified by a small perturbation up to δ. In [SBMR13], we proved that with Büchi winning conditions (*i.e.*, if the correctness property can be expressed as a Büchi condition, which can encode many important properties such as safety), the existence of a positive value for δ under which there is a winning strategy is decidable.

Another approach we developed is the construction of *permissive* strategies. Indeed, the very definition of a strategy for timed models does not fit with the robustness framework: strategies, even in a timed setting, are functions prescribing to play a given action *at one precise date*. Following previous works, we extended *permissive strategies* to the timed setting: the aim is now to have strategies return an *interval* of possible dates at which to play the given action. We proved that this problem is decidable for one-clock timed games. The algorithm is under implementation, and we are currently trying to see whether this decidability result extends to general

timed automata. Our results are presented in [Fan14].

Specification theories for real-time systems allow reasoning about interfaces and their implementation models, using a set of operators that includes satisfaction, refinement, logical and parallel composition. They use input/output timed automata in order to describe components, and a game-based semantics to define the operations of the theory, including satisfaction (whether a specification can be implemented), refinement (how two specifications compare), logical composition (superposition of two specifications), structural composition (combining smaller components into larger ones), and quotient (synthesizing a component in a large design).

To make such theories applicable throughout the entire design process from an abstract specification to an implementation, we need to reason about the possibility to effectively implement the theoretical specifications on physical systems, despite their limited precision. In the literature, this implementation problem has been linked to the robustness problem that analyzes the consequences of introducing small perturbations into formal models.

We addressed the problem of robust implementations in timed specification theories. For a fixed perturbation, we studied the robustness of timed specifications with respect to the operators of the theory. To this end we synthesized robust strategies in timed games. We then considered the parametric robustness problem, and proposed a counter-example refinement heuristic for computing safe perturbation values.

Participants: Patricia Bouyer (CNRS)
Erwin Fang (CNRS)
Kim G. Larsen (AAU)
Axel Legay
Nicolas Markey (CNRS)
Pierre-alain Reynier (CNRS)
Ocan Sankur (ULB)
Louis-Marie Traonouez
Andrzej Wąsowski

References: [SBMR13, Fan14, LLTW14]

1.3 Robust weighted timed automata

Background. Weighted timed automata (aka. priced timed automata) [ALP01, BFH+01] are timed automata equiped with additional quantitative variables that can be used to measure some values along the runs, such as energy consumption. Those extra variables cannot be used in the constraints used for defining the availability of transitions: this would make reachability undecidable in those models, while *optimal reachability* is decidable for priced timed automata, and *reachability under energy constraints* is decidable for one-clock weighted timed automata. However, optimal reachability is undecidable in priced timed games, the extension of weighted timed automata with controllable and uncontrollable actions.

Our contribution. Whether these decidability and undecidability results extend to the robust setting was an interesting open question: on the one hand, extending decidability results to the robust world is a natural continuation of the robust framework for timed automata; on the other hand, most of the undecidability proofs involving timed automata consist in encoding Turing machines or 2-counter machines using clocks, and rely on the arbitrary precision of the models in order to encode arbitrary long tapes or counter values. Whether undecidability extends to the robust setting is not so obvious.

We proved in [BMS13] that, in the game-based approach to robustness (*i.e.*, with an opponent adding small imprecisions to the delays proposed by the protagonist), undecidability still holds, and some decidable problems in the classical setting become undecidable in the robust setting.

[ALP01] Rajeev Alur, Salvatore La Torre, and George J. Pappas. Optimal paths in weighted timed automata. In Maria Domenica Di Benedetto and Alberto L. Sangiovani-Vincentelli, editors, *Proceedings of the 4th International Workshop on Hybrid Systems: Computation and Control (HSCC'01)*, volume 2034 of *Lecture Notes in Computer Science*, pages 49–62. Springer-Verlag, March 2001.

[BFH+01] Gerd Behrmann, Ansgar Fehnker, Thomas Hune, Kim Gulstrand Larsen, Paul Pettersson, Judi Romijn, and Frits Vaandrager. Minimum-cost reachability for priced timed automata. In Maria Domenica Di Benedetto and Alberto L. Sangiovani-Vincentelli, editors, *Proceedings of the 4th International Workshop on Hybrid Systems: Computation and Control (HSCC'01)*, volume 2034 of *Lecture Notes in Computer Science*, pages 147–161. Springer-Verlag, March 2001.

Participants: Patricia Bouyer (CNRS)
Nicolas Markey (CNRS)
Ocan Sankur (ULB)
References: [BMS13]

2 Robustness for probabilistic systems

2.1 Bisimulation distances for stochastic models

Background. In classical model checking the goal is to algorithmically settle whether a given logical property (e.g. expressed in LTL or CTL) is satisfied by a given behavioural model (e.g. a finite state machine, a timed automaton or a Markov chain). In order to gain efficiency in settling the model checking problem (compositional) reduction of the model prior to the actual model checking have proved useful in several settings. In the setting of quantitative models (e.g. timed automata or Markov chains) it has been strongly argued (e.g. by Tom Henzinger and Joseph Sifakis) that model checking problems should result in quantitative rather than boolean verdicts. Thus rather that deciding on whether a model satisfies a properties or not we want to measure the degree by which a quantitative model satisfies a quantitative property (typically a real number in the interval $[0, 1]$). Strongly related to robustness, an ungoing research challenge is to identify behavioural metrics on models, where a small behavioural distance between two models will ensure a small deviation wrt the degree by which logical properties are satisfied.

Our contribution. In a series of papers we are considering this problem in the setting of (discrete time) Markov chains and LTL. In particular we have been developing an efficient on-the-fly algorithm for exact computation of bisimilarity distances between discrete-time Markov chains introduced by Desharnais *et al.* Our work is inspired by the theoretical results presented by Chen *et al.* at FoSSaCS'12, proving that these distances can be computed in polynomial time using the ellipsoid method. Despite its theoretical importance, the ellipsoid method is known to be inefficient in practice. To overcome this problem, we propose an efficient on-the-fly algorithm which, unlike other existing solutions, computes exactly the distances between given states and avoids the exhaustive state space exploration. It is parametric in

a given set of states for which we want to compute the distances. Our technique successively refines over-approximations of the target distances using a greedy strategy which ensures that the state space is further explored only when the current approximations are improved. Tests performed on a consistent set of (pseudo-)randomly generated Markov chains shows that our algorithm improves, on average, the efficiency of the corresponding iterative algorithms with orders of magnitude.

Also, we propose a general definition of composition operator on Markov Decision Processes with rewards (MDPs) and identify a well behaved class of operators, called safe, that are guaranteed to be non-extensive w.r.t. the bisimilarity distance between MDPs. For MDPs built using safe/non-extensive operators, we present the first method that exploits the structure of the system for (exactly) computing the bisimilarity distance on MDPs. Experimental results show significant improvements upon the non-compositional technique.

Finally, we offer a Mathematica package library for exactly computing the bisimilarity distance between Markov chains and between Markov decision processes.

Participants: Giorgio Bacci (AAU)
Giovanni Bacci (AAU)
Kim G. Larsen (AAU)
Radu Mardare (AAU)
References: [BBLM13b, BBLM13a]
`http://people.cs.aau.dk/~giovbacci/tools/bisimdist.zip`

3 Robust verification of finite-state systems

3.1 A quantitative semantics for LTL

Background. LTL (Linera-time Temporal Logic) is a convenient formalism for expressing properties of models: LTL formulas are based on atomic propositions, and are built using boolean operators and *temporal modalities*, in particular the *until-modality*. As an example, $\phi \mathsf{U} \psi$ (where U reads "until") states that property ψ has to hold true at some later time, and property ϕ must be true in all intermediary states. Using this modality,

LTL can express properties about the order in which events occur along the executions of the system. LTL was defined long ago [Pnu77], and comes with efficient algorithms which have been implemented and used in various case studies [Hol97].

As can be noted, this mathematical semantics of the "until" modality is very strict: this semantics, and the corresponding model-checking algorithms, cannot tell whether a formula is *almost* true, *i.e.*, whether the left-hand-side formula failed to hold at only one state out of several hundreds. Of course, there are cases where such a strictness is important, but there are less critical settings where a relaxed version would be of interest (and in any case, it may always be interesting to know whether a formula is *almost* true or if there is a deep problem).

Our contribution. We defined a quantitative semantics for *until*, in which the value of $\phi \mathsf{U} \psi$ along an (infinite) run π is defined as

$$\max_{p \in \mathbb{N}} \Big[\min \Big(\mathsf{average}([\![\phi, \pi_{<p}]\!]), [\![\psi, \pi_{=p}]\!] \Big) \Big]$$

In other terms, when ϕ is always true along some prefix up to a position where ψ holds, then the value will be 1, as for the classical semantics. However, the fact that ϕ fails to hold at only a few positions along the prefix will be smoothed out. In the end, the value of a formula gives an approximation on how far a formula is to hold in a model.

Our preliminary results about this semantics, unfortunately, are rather negative for the moment: we proved that the problems of computing and approximating the value of a formula along a run or in a given Kripke structure are undecidable [BMM14].

Participants: Patricia Bouyer (CNRS)
Nicolas Markey (CNRS)
Raj Mohan M. (CNRS)

References: [BMM14]

[Pnu77] Amir Pnueli. The temporal logic of programs. In *Proceedings of the 18th Annual Symposium on Foundations of Computer Science (FOCS'77)*, pages 46–57. IEEE Comp. Soc. Press, October-November 1977.

[Hol97] Gerard J. Holzmann. The model checker SPIN. *IEEE Transaction on Software Engineering*, 23(5):279–295, May 1997.

3.2 Fairly correct systems and Banach-Mazur games

Background. In the same spirit as in the previous section, *fairly correct systems* are systems in which the set of incorrect behaviours are very seldom. What *seldom* means here is not precise: it could be defined in terms of probabilities (assuming a uniform probability distribution over transitions from each state, does the set of correct runs have probability one?), or in terms of topology (is the set of correct runs *large*, *ie.*, can its complement be defined as a countable union of nowhere-dense sets?).

As proved in [VV06], both notions are different, but when restricting to ω-regular properties, they do coincide. On the other hand, large sets have been characterized by *Banach-Mazur games*: Banach-Mazur games are two-player path-forming games played on finite graphs, where the players successively append finite paths to the previously formed path. A set of runs is large if, and only if, there is a winning strategy for the protagonist in the corresponding Banach-Mazur game.

Our contribution. We extended the results above in several respects, by considering various kind of simple strategies in Banach-Mazur games: in particular, we prove that the existence of *bounded* winning strategies in those games implies that the winning set has probability 1. However, the converse does not hold. In order to obtain a characterization, we define *generalized* Banach-Mazur games by letting each player choose the path to append in a set proposed by the opponent. Using such games. Using such generalized games, we obtain a characterization of winning sets of probability one. Those results were published in [BM13].

Participants: Thomas Brihaye (UMONS)
Quentin Menet (UMONS)

References: [BM13]

[VV06] Daniele Varacca and Hagen Völzer. Temporal logics and model checking for fairly correct systems. In *Proceedings of the 21st Annual Symposium on Logic in Computer Science (LICS'06)*, pages 389–398. IEEE Comp. Soc. Press, July 2006.

4 Conclusions and perspectives

Robustness issues in model checking and synthesis are an important, multi-faceted problem, and are especially important in the context of the Cassting project, which focuses on complex systems made of numerous components.

We studied several aspects of robustness: robustness in timed systems, where timing imprecisions and perturbations can give rise to emergent behaviours; robustness in stochastic systems, with the aim os measuring how the probability of some behaviour is linked to the values appearing in the model; and robustness in finite-state systems, where the objective is to have a semantics that tolerates infrequent failures.

Our previous work opens numerous avenues for future work on robustness in collective adaptive systems. We will keep on developing all the approaches presented in this deliverable, and especially the following:

- the study of permissive strategies in timed games with several clocks is our most immediate research direction: however, while our intuition was that the optimal counter-strategy would always play close to an integer date or to a the bounds ot the proposed interval, it seems that the situation is harder, and we now conjecture that the general problem will be undecidable.
- while our first results on robustness in weighted timed automata are negative, we still have some hope of finding settings where robustness comes with decidability.
- an alternative approach to study robustness of timed systems is to use probabilities to alleviate the disadvantages of mathematical models. In order to do so, we propose a probabilistic semantics for timed automata which assigns probabilities both on delays and on discrete choices, leading the concept of stochastic timed automata (STA) which has already been well studied by several members of the project. More precisely, we studied the almost-sure model-checking problem on STA, which, given a STA and a property, asks whether the STA satisfies the property with probability one. Decidability results of the almost-sure model-checking problem have been obtained for one-clock STA and reactive STA.

  We are currently studying the compositionality of stochastic timed

automata. Indeed, some restrictions are necessary to ensure that the product of two STA is still a STA. Our goal is to identify a large class of STA which is compositional and for which the almost-sure model-checking problem is decidable.

- developing the *distance-based* approach to robustness is a difficult but very relevant approach: our tool for evaluating bisimulation distances in stochastic systems is a first step towards such a framework. Extending this to other quantitative approaches is also part of our plans.

5 List of publications

[BBLM13a] Giorgio Bacci, Giovani Bacci, Kim Gulstrand Larsen, and Radu Mardare. The bisimdist library: Efficient computation of bisimilarity distances for markovian models. In *Proceedings of the 10th International Conference on Quantitative Evaluation of Systems (QEST'13)*, pages 278–281. IEEE Comp. Soc. Press, August 2013.

[BBLM13b] Giorgio Bacci, Giovani Bacci, Kim Gulstrand Larsen, and Radu Mardare. Computing behavioral distances, compositionally. In Krishnendu Chatterjee and Jiří Sgall, editors, *Proceedings of the 38th International Symposium on Mathematical Foundations of Computer Science (MFCS'13)*, volume 8087 of *Lecture Notes in Computer Science*, pages 74–85. Springer-Verlag, August 2013.

[BM13] Thomas Brihaye and Quentin Menet. Simple strategies for Banach-Mazur games and fairly correct systems. In Gabriele Puppis and Tiziano Villa, editors, *Proceedings of the 4th International Symposium on Games, Automata, Logics, and Formal Verification (GandALF'13)*, volume 119 of *Electronic Proceedings in Theoretical Computer Science*, pages 21–34, Borca di Cadore, Italy, August 2013.

[BMM14] Patricia Bouyer, Nicolas Markey, and Raj Mohan Matteplackel. Averaging in LTL. Technical Report LSV-14-02, Laboratoire Spécification et Vérification, ENS Cachan, France, February 2014. 28 pages.

[BMS13] Patricia Bouyer, Nicolas Markey, and Ocan Sankur. Robust weighted timed automata and games. In Víctor Braberman and Laurent Fribourg, editors, *Proceedings of the 11th International Conference on Formal Modelling and Analysis of Timed Systems (FORMATS'13)*, volume 8053 of *Lecture Notes in Computer Science*, pages 31–36, Buenos Aires, Argentina, August 2013. Springer.

[Fan14] Erwin Fang. Permissive multi-strategies in timed games. Research Report LSV-14-04, Laboratoire Spécification et Vérification, ENS Cachan, France, March 2014. 36 pages.

[LLTW14] Kim G. Larsen, Axel Legay, Louis-Marie Traonouez, and Andrzej Wąsowski. Robust synthesis for real-time systems. *Theoretical Computer Science*, 515:96–122, January 2014.

[SBM14] Ocan Sankur, Patricia Bouyer, and Nicolas Markey. Shrinking timed automata. *Information and Computation*, 234:107–132, February 2014.

[SBMR13] Ocan Sankur, Patricia Bouyer, Nicolas Markey, and Pierre-Alain Reynier. Robust controller synthesis in timed automata. In Pedro R. D'Argenio and Hernán) Melgratti, editors, *Proceedings of the 24th International Conference on Concurrency Theory (CONCUR'13)*, volume 8052 of *Lecture Notes in Computer Science*, pages 546–560, Buenos Aires, Argentina, August 2013. Springer.