ARTICLE IN PRESS
YINCO:4365

Information and Computation ••• (••••) •••–•••

Contents lists available at ScienceDirect

Information and Computation

www.elsevier.com/locate/yinco

Imperfect information in Reactive Modules games

Julian Gutierrez *, Giuseppe Perelli, Michael Wooldridge

Department of Computer Science, University of Oxford, United Kingdom

ARTICLE INFO

Article history:
Received 3 January 2017
Available online xxxx

Keywords:
Games
Temporal logic
Multi-agent systems
Formal verification

ABSTRACT

Reactive Modules is a high-level modelling language for concurrent, distributed, and multi-agent systems, which is used in a number of practical model checking tools. Reactive Modules Games are a game-theoretic extension of Reactive Modules, in which system components are assumed to act strategically in an attempt to satisfy a temporal logic formula representing their individual goal. Reactive Modules Games with perfect information have been extensively studied, and the complexity of game theoretic decision problems relating to such games (such as the existence of Nash equilibria) have been comprehensively classified. In this article, we study Reactive Modules Games in which agents have only partial visibility of their environment.

1. Introduction

A common technique in the formal analysis computer systems is to model a system as a game in which two players—sometimes called “System” and “Environment” or “Player” and “Opponent”—interact with each other, possibly over infinitely many rounds. In these games, it is typically assumed that the system has a goal given in a logical form, *e.g.*, expressed as a temporal logic formula φ, which the system wishes to satisfy. Such a goal may represent either the behaviour of the computer system one wants to synthesize (an automated design problem [1]) or a particular system property which one wants to check (an automated verification problem [2]). In this framework, it is assumed that the system plays against an adversarial environment, that is, that the goal of the environment is to prevent the system from achieving its goal. In game-theoretic terms, this means that the problem is modelled as a zero-sum game, and hence that its solution is given by the computation of a winning strategy for either the system or the environment. From a logical viewpoint, this assumption amounts to letting the goal of the environment be $\neg\varphi$, whenever the goal of the system is given by the temporal logic formula φ. A great deal of work has been done based on this idea—see, *e.g.*, [3,4] and the references therein for surveys containing several results on this topic.

Although this approach has been found to be appropriate in a wide range of settings, the zero-sum assumption is often either too restrictive or else simply inappropriate. For example, when dealing with concurrent systems one may have several system components, each with their own temporal goal, which are not necessarily in conflict. The appropriate model here is a non-zero-sum n-player game, rather than a two-player zero-sum game. In the non-zero-sum n-player setting it is no longer the computation of a winning strategy that provides a solution to the problem under consideration, but rather, the computation of a strategy profile (a set of strategies, one for each player in the game) which can be regarded as in equilibrium in the game-theoretic sense [5]: a situation where no player wishes to deviate from the strategy it is currently using. While the use of zero-sum games in the analysis and design of computer systems is well-established, the approach

* Corresponding author.
E-mail address: Julian.Gutierrez@cs.ox.ac.uk (J. Gutierrez).

https://doi.org/10.1016/j.ic.2018.02.023
0890-5401/© 2018 Published by Elsevier Inc.

Please cite this article in press as: J. Gutierrez et al., Imperfect information in Reactive Modules games, Inf. Comput. (2018), https://doi.org/10.1016/j.ic.2018.02.023

of modelling computer systems as non-zero-sum games is much less so. Nevertheless, over the past decade, an increasing number of authors have begun to investigate this approach—see, for example, [6–8] for references.

In this article, we study non-zero-sum *n*-player games in which the choices available to players are defined using the Simple Reactive Modules Language (SRML), a subset of Reactive Modules [9], a popular and expressive system modelling language that is used in several practical model checking systems (e.g., MOCHA [10] and Prism [11]). Reactive Modules supports *succinct* and *high-level* modelling of concurrent and multi-agent systems. In the games we study, the preferences of system components are specified by associating with each player in the game a Linear Temporal Logic (LTL) formula, representing a goal that the player desires to be satisfied. Reactive Modules Games with perfect information (where each player can see the entire system state) have been extensively studied [12], but in this paper we focus on *imperfect information* cases. We interpret imperfect information to mean that system components must make choices based on a partial view of the system state—more precisely, that there are some variables in the system whose value is hidden from them. Agents may have different views of the system: thus, one agent might have only a very minimal view of the system state while another is able to perceive the values of all variables in the system.

We study the decidability and complexity of checking the existence of Nash equilibria in Reactive Modules games with imperfect information. Using our framework, one can analyse the behaviour of open systems modelled as multi-player games using a modelling language that is widely used, and which already has a number of tool implementations. However, our results go beyond simply the specific language of SRML itself as, more generally, we provide complexity results that apply to a wide range of imperfect information games with succinct representations.

There are a number of good reasons to study imperfect information in this setting. Firstly, from a modelling point of view, it may not be realistic to expect that a simple module, which is supposed to represent some local, reactive, and independent behaviour, must be aware of the entirety of the state space of the systems it is part of. A more realistic situation is, for instance, that the module is aware only of the values of the Boolean variables associated with its linear temporal logic goal. Secondly, from a practical point of view, it is desirable to have specifications that are as small as possible so that the synthesis task may be simpler. Take, for instance, a system composed of thousands of Boolean variables. In a perfect information setting, every module will have to have a strategy that considers all possible valuations for those variables. Instead, if a particular module is allowed to have imperfect information, then a specification for such a module could be given so that the set of strategies associated with such a module will only have to take into account the values of a small subset of some of those Boolean variables, leading to potentially better results in practice. Thirdly, it may also be the case, as illustrated by Example 1 presented later on, that it is simply not the case that a setting with perfect information faithfully represents the situation we want to model. Finally, from a game-theoretic point of view, because the set of Nash equilibria of a system is sensitive to the visibility of players in a game, we can use imperfect information rather easily in the context of Reactive Modules Games to modify the set of Nash equilibria of a given game. This issue is discussed in more detail in Section 9.

The main results contained in the paper are as follows. We show that Reactive Modules Games with imperfect information are undecidable if three or more players are allowed. In contrast, if restricted to two players, the games are decidable and their solution (computing a Nash equilibrium if one exists) can be obtained in 2EXPTIME. For the latter decidability result, we provide a conceptually simple decision procedure based on synthesis techniques for CTL* under imperfect information. We also explore a number of variants of the general imperfect-information framework. For instance, we study variants of these games with respect to the class of strategies under consideration, *e.g.*, memoryless, myopic, polynomially bounded, and show that such games can be solved, respectively, in NEXPTIME, EXPSPACE, and PSPACE; we also explore the use of a solution concept where coordinated behaviour is allowed—in whose case strong Nash equilibrium is considered instead—and show that going from Nash to strong Nash equilibria can be done without paying a (worst-case) complexity cost. We then study in more detail the connection between imperfect information and the existence of Nash equilibria. Specifically, we provide conditions under which the set of Nash equilibria of an imperfect-information game can be preserved (or refined) with respect to the amount of information that players in such a Reactive Modules Game have.

Technically, the undecidability results in the paper rely on the undecidability of the *uniform synthesis problem* for LTL formulae [13]. In Section 4 we show how such a problem can be described as a Reactive Modules Game with three players in which one is interested in the existence of a Nash equilibrium in the game. On the positive side, our decidability results rely on the solution of different problems, showing the diversity of techniques that one could use to solve the particular instances at hand. For instance, for two-player games with imperfect information we propose a technique, to the best of our knowledge only previously explored in [14], where checking the existence of a Nash equilibrium is reduced to a number of CTL* synthesis problems. On the other hand, for the positive results given for memoryless and polynomially bounded strategies a simple nondeterministic algorithm can be used. The lower bounds, in NEXPTIME and PSPACE, respectively, are obtained via reductions from the satisfiability problem for Dependency Quantified Boolean Formulae (DQBF [15, pp. 86–87]) and the LTL model checking problem for compressed words [16]. Finally for the case considering myopic strategies we use the satisfiability problem for Quantified LTL (QPTL [17]) formulae.

Structure of the paper. Section 2 gives some background information and Section 3 introduces the model of games we consider here, namely Reactive Modules Games, and illustrate with an example the difference between perfect and imperfect information games. In Section 4 we show that checking the existence of Nash equilibria in imperfect information games with more than two players is an undecidable problem, even for the simple case of *iterated Boolean games* [18]. Section 5,