Minimum Attention Controller Synthesis for Omega-Regular Objectives*

Krishnendu Chatterjee[1] and Rupak Majumdar[2]

[1] IST Austria
[2] MPI-SWS

Abstract. A controller for a discrete game with ω-regular objectives requires *attention* if, intuitively, it requires measuring the state and switching from the current control action. Minimum attention controllers are preferable in modern shared implementations of cyber-physical systems because they produce the least burden on system resources such as processor time or communication bandwidth. We give algorithms to compute minimum attention controllers for ω-regular objectives in imperfect information discrete two-player games. We show a polynomial-time reduction from minimum attention controller synthesis to synthesis of controllers for mean-payoff parity objectives in games of incomplete information. This gives an optimal EXPTIME-complete synthesis algorithm. We show that the minimum attention controller problem is decidable for infinite state systems with finite bisimulation quotients. In particular, the problem is decidable for timed and rectangular automata.

1 Introduction

Automata-theoretic reactive synthesis techniques [10, 5, 31, 15, 32, 29, 37, 21, 23] hold the promise to correct-by-construction design of complex reactive systems, and over the years, have seen impressive technical advances that have brought them within striking distance of practice in the design of cyber-physical systems [17, 19, 20, 26]. Despite the many advances, there is still a gap between the *abstract* two-person game models considered by the theory of synthesis and *implementation issues* associated with controller implementations. Classically, automata-theoretic synthesis considers the size of the memory as the notion of optimality; and any memoryless strategy is considered optimal (a strategy is *memoryless* if it depends only on the current state and not on the history of the play). While roughly adequate for applications of synthesis in hardware circuit design, for more recent applications of synthesis techniques in cyber-physical systems, there are additional implementation costs whose effects can significantly influence the practical applicability of synthesized controllers.

Classical controller synthesis assumes that measurement of the current state and computation of the control action is computed instantaneously. In practice, state measurement and control computations take time and consume other system resources such

* This research was funded in part by the US National Science Foundation grants CCF-0546170, CNS-0702881, DARPA grant HR0011-09-1-0037, Austrian Science Fund (FWF) NFN Grant S11407-N23 (RiSE), and a Microsoft faculty fellowship.

as network bandwidth. In a modern control system application, where a controller shares the platform with other tasks, the controller task must compete for resources with other tasks. In this context, it is important to find a controller that can be implemented without diverting attention from other, possibly more pressing, system tasks, or to enable other tasks which might not be schedulable if the controller hogs resources.

Consider, for example, the simple game in Figure 1 where the objective is to visit the state s. For states in $\{ 1, \ldots, n \}$, any function $\xi : \{ 1, \ldots, n \} \to \{ 0, 1 \}$ is a memoryless winning strategy ensuring a visit to s, and equally good in the view of classical synthesis. However, consider a real-time implementation of the controller where the control task must be scheduled to compute the next control action. The strategy which always plays 0 (or 1) has an advantage over the strategy playing the action $i \mod 2$. For the former strategy, the controller task is scheduled once to set the control action, and never again. For the latter, the control task runs every cycle, looking at the state and changing the control action accordingly, using up communication resources to measure state and processor resources to compute the new control, which could be used for other tasks.

Intuitively, the "simplest" strategy is to play a constant action throughout. Anything else requires *attention* [3]: to measure the state and to switch to a different action if necessary. Measuring the state can involve running code on the platform activating sensors and processing sensed values, and using network bandwidth or bus slots to transmit the sensed values to a central processor. Switching to a different action may require dynamic computation of lower-level control laws implementing these actions, switching modes and tasks, as well as re-scheduling bus or network slots. The more frequently these tasks must be performed, the more attention is required.

Of course, there may not be a winning strategy that plays a constant action throughout. Consider in Figure 1 the objective of visiting t infinitely often. Again, any action is possible from states $\{ 1, \ldots, n \}$, as long as 0 is played at s and 1 at t. A possible strategy can, starting from 1, play 1 for n steps, then 0 for a step, then 1 for $n+1$ steps, etc., or dually, play 0 for $n+1$ steps, 1 for $n+1$ steps, a single 0, etc. Both strategies can be implemented with lower processor requirements than one that alternates between 0 and 1. The precise strategy chosen will depend on the actual costs involved in switching between 0 and 1. In general, the lowest-cost controller must optimize the usage, over the long run, of system resources while ensuring the winning condition. Formally, the controller must optimize costs associated with measuring state and switching controllers while ensuring the winning condition is satisfied.

In this paper, we consider the problem of minimum attention controllers for ω-regular objectives. We introduce a cost for measuring the state, as well as a cost for changing the control action, in the model of two-player games. We then ask for a strategy which ensures the ω-regular objective while minimizing the long-run costs incurred due to measurement and switching actions. Technically, playing a game without measuring the state or changing the control action is similar to playing a game of incomplete information [33, 22, 7]. We formulate the minimum attention control problem as a game of incomplete information, and we show a polynomial-time reduction from the problem of minimum attention control for ω-regular objectives to solving a mean-payoff parity condition [8] on a game of incomplete information. Together with results on incomplete

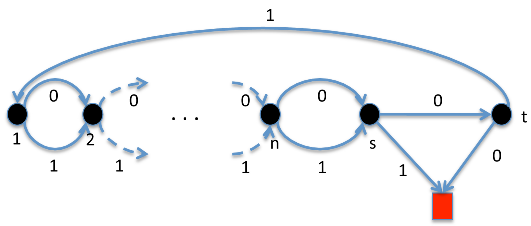

Fig. 1. Simple example

information games [7], this gives an EXPTIME-complete procedure when the winning objective is given as a parity condition on states, and a triply exponential procedure when the objective is given in linear-temporal logic.

We develop the theory both for finite-state, discrete control problems, as well as for infinite state systems for which there is a finite bisimulation quotient. Using known results about stable partitions of timed games [1, 24] and rectangular automata [16], it follows that the minimum attention controller synthesis problem is decidable for timed games and discrete-time control for rectangular automata.

Attention, and minimum attention controllers, were introduced in a seminal paper by Brockett [3]. There, the problem of minimum attention synthesis is formulated for controlled dynamical systems, and set up as the minimization problem for an attention index, a functional involving the partial derivatives $\frac{\partial u}{\partial t}$ and $\frac{\partial u}{\partial x}$ of the control function u w.r.t. time and state, respectively, subject to constraints on u to guarantee a minimum level of system performance. The resulting problem involves minimization of non-linear functions subject to systems of partial differential equations, even in the case of linear control systems. Minimum attention control was applied to solve control problems for vehicular control [4] and for control under network bandwidth constraints [27], but the algorithmic complexity of the methods are not immediate. More recently, [2] studies approximations of the problem using event-driven control.

Solving the minimum attention control problem for discrete systems suggests a computational approach to *approximately* solve the minimum attention synthesis problem for controlled dynamical system. The link between continuous dynamical systems and discrete systems is provided by *approximate abstractions* of continuous models [30, 14]. Approximate abstractions generalize the classical language-theoretic notions of language containment and simulation to the quantitative case; an ε-approximate abstraction of a continuous system is a discrete system such that for any trace of the original system, there is a trace of the abstract system which is at a distance of at most ε, for a design parameter ε. With approximate abstraction relations, the minimum attention control problem for dynamical systems can be approximately solved in two steps: first compute the abstraction, then solve the problem on the discrete abstraction.

Related works. In this work we consider the minimum attention controller synthesis problem and show that the problem can be solved by solving a special class of incomplete-information mean-payoff parity games and is EXPTIME-complete. The general problem of incomplete-information mean-payoff games was studied in [11] and the problem was shown to be undecidable, whereas we show that the minimum attention synthesis problem belongs to a decidable subclass. The problem of mean-payoff parity games was studied in [8] but in the setting of perfect-information games, and we show that for the special class of incomplete-information games we obtain, the problem is decidable using solutions of [8]. Incomplete-information games with Boolean objectives (such as parity objectives) were considered in [7], whereas in this paper the problem we consider reduces to incomplete-information games with mixed quantitative and Boolean objective (combination of mean-payoff and parity objectives). The problem of fault diagnosis with static and dynamic observers has been considered in [6], and it was shown that the static observer problem is NP-complete and the dynamic observer problem can be solved in 2EXPTIME using solution of mean-payoff games. In contrast our problem requires solution of mean-payoff parity games and is EXPTIME-complete. The fault diagnosis problem was also considered in [36] where a dynamic programming approach was used to solve the problem. No complexity bounds are known. In contrast our approach is game theoretic and we establish optimal complexity bounds.

2 Preliminaries

In this section we present the required preliminaries. We first present the mathematical framework of imperfect information games, and then present a reduction of imperfect information games to perfect information games. In the following section we will use the definitions and the results of this section to develop the theory of minimum attention control.

2.1 Imperfect information games

A *game structure* (*of imperfect information*) is a tuple $G = \langle L, l_0, \Sigma, \Delta, \mathcal{O}, \gamma \rangle$, where L is a finite set of states, $l_0 \in L$ is the initial state, Σ is a finite alphabet (of input letters or actions), $\Delta \subseteq L \times \Sigma \times L$ is a set of labeled transitions, $\mathcal{O}$ is a finite set of observations, and $\gamma : \mathcal{O} \to 2^L \backslash \emptyset$ maps each observation to the set of states that it represents. We require the following two properties on G: (i) for all $\ell \in L$ and all $\sigma \in \Sigma$, there exists $\ell' \in L$ such that $(\ell, \sigma, \ell') \in \Delta$; and (ii) the set $\{\gamma(o) \mid o \in \mathcal{O}\}$ partitions L. We say that G is a game structure of *perfect information* if $\mathcal{O} = L$ and $\gamma(\ell) = \{\ell\}$ for all $\ell \in L$. We omit $(\mathcal{O}, \gamma)$ in the description of games of perfect information. For $\sigma \in \Sigma$ and $s \subseteq L$, let $\mathsf{Post}^G_\sigma(s) = \{\ell' \in L \mid \exists \ell \in s : (\ell, \sigma, \ell') \in \Delta\}$.

In a game structure, in each turn, Player 1 (controller) chooses a letter in Σ, and Player 2 (system or plant) resolves nondeterminism by choosing the successor state. A *play* in G is an infinite sequence $\pi = \ell_0 \sigma_0 \ell_1 \dots \sigma_{n-1} \ell_n \sigma_n \dots$ such that (i) $\ell_0 = l_0$, and (ii) for all $i \geq 0$, we have $(\ell_i, \sigma_i, \ell_{i+1}) \in \Delta$. The *prefix up to* ℓ_n of the play π is denoted by $\pi(n)$; its *length* is $|\pi(n)| = n + 1$; and its *last element* is $\mathsf{Last}(\pi(n)) = \ell_n$. The *observation sequence* of π is the unique infinite sequence $\gamma^{-1}(\pi) = o_0 \sigma_0 o_1 \dots \sigma_{n-1} o_n \sigma_n \dots$ such that for all $i \geq 0$, we have $\ell_i \in \gamma(o_i)$. Similarly, the *observation sequence* of $\pi(n)$ is the prefix up to o_n of $\gamma^{-1}(\pi)$. The set of infinite plays in G is denoted $\mathsf{Plays}(G)$, and the set of corresponding finite prefixes is denoted $\mathsf{Prefs}(G)$. A state $\ell \in L$ is *reachable* in G if there exists a prefix $\rho \in \mathsf{Prefs}(G)$ such that $\mathsf{Last}(\rho) = \ell$. The *knowledge* associated with a finite observation sequence $\tau = o_0\sigma_0o_1\sigma_1 \dots \sigma_{n-1}o_n$ is the set $\mathsf{K}(\tau)$ of states in which a play can be after this sequence of observations, that is, $\mathsf{K}(\tau) = \{\mathsf{Last}(\rho) \mid \rho \in \mathsf{Prefs}(G) \text{ and } \gamma^{-1}(\rho) = \tau\}$. The following lemma presents a inductive construction of the knowledge. The proof of the lemma is standard.

Lemma 1. *Let $G = \langle L, l_0, \Sigma, \Delta, \mathcal{O}, \gamma\rangle$ be a game structure. For $\sigma \in \Sigma$, $\ell \in L$, and $\rho, \rho' \in \mathsf{Prefs}(G)$ with $\rho' = \rho \cdot \sigma \cdot \ell$, let $o_\ell \in \mathcal{O}$ be the unique observation such that $\ell \in \gamma(o_\ell)$. Then $\mathsf{K}(\gamma^{-1}(\rho')) = \mathsf{Post}^G_\sigma(\mathsf{K}(\gamma^{-1}(\rho))) \cap \gamma(o_\ell)$.*

Strategies. A *strategy* in G for Player 1 is a function $\alpha : \mathsf{Prefs}(G) \to \Sigma$ that given a finite prefix or history of a play specifies the next input letter or action. A strategy α for Player 1 is *observation-based* if for all prefixes $\rho, \rho' \in \mathsf{Prefs}(G)$, if $\gamma^{-1}(\rho) = \gamma^{-1}(\rho')$, then $\alpha(\rho) = \alpha(\rho')$. In games of imperfect information we are interested in the existence of observation-based strategies for Player 1. A *strategy* in G for Player 2 is a function $\beta : \mathsf{Prefs}(G) \times \Sigma \to L$ such that for all $\rho \in \mathsf{Prefs}(G)$ and all $\sigma \in \Sigma$, we have $(\mathsf{Last}(\rho), \sigma, \beta(\rho, \sigma)) \in \Delta$. We denote by $\mathcal{A}_G$, $\mathcal{A}^O_G$, and $\mathcal{B}_G$ the set of all Player-1 strategies, the set of all observation-based Player-1 strategies, and the set of all Player-2 strategies in G, respectively.

The *outcome* of two strategies α (for Player 1) and β (for Player 2) in G is the play $\pi = \ell_0\sigma_0\ell_1 \dots \sigma_{n-1}\ell_n\sigma_n \dots \in \mathsf{Plays}(G)$ such that for all $i \geq 0$, we have $\sigma_i = \alpha(\pi(i))$ and $\ell_{i+1} = \beta(\pi(i), \sigma_i)$. This play is denoted $\mathsf{outcome}(G, \alpha, \beta)$. The *outcome* of a strategy α for Player 1 in G is the set $\mathsf{Outcome}_1(G, \alpha)$ of plays π such that there exists a strategy β for Player 2 with $\pi = \mathsf{outcome}(G, \alpha, \beta)$. The outcome sets for Player 2 are defined symmetrically.

Qualitative objectives. A *qualitative objective* for G is a set ϕ of infinite sequences of states and input letters, that is, $\phi \subseteq (L \times \Sigma)^\omega$. A play $\pi = \ell_0\sigma_0\ell_1 \dots \sigma_{n-1}\ell_n\sigma_n \dots \in \mathsf{Plays}(G)$ *satisfies* the objective ϕ, denoted $\pi \models \phi$, if $\pi \in \phi$. We assume objectives are Borel measurable, that is, a qualitative objective is a Borel set in the Cantor topology on $(L \times \Sigma)^\omega$ [18].

We specifically consider *parity objectives* [12, 35]. Parity objectives are a canonical form to express all ω-regular objectives [35] and lie in the intersection $\Sigma_3 \cap \Pi_3$ of the third levels of the Borel hierarchy. For a play $\pi = \ell_0\sigma_0\ell_1 \dots$, we write $\mathsf{Inf}(\pi)$ for the set of states that appear infinitely often in π, that is, $\mathsf{Inf}(\pi) = \{\ell \in L \mid \ell_i = \ell \text{ for infinitely many } i\text{'s}\}$. For $d \in \mathbb{N}$, let $p : L \to \{\, 0, 1, \dots, d \,\}$ be a *priority function*, which maps each state to a nonnegative integer priority. The *parity* objective $\mathsf{Parity}(p)$ requires that the minimum priority that appears infinitely often be even. Formally, $\mathsf{Parity}(p) = \{\, \pi \mid \min\{\, p(\ell) \mid \ell \in \mathsf{Inf}(\pi) \,\} \text{ is even} \,\}$. Observe that the objectives are defined on sequence of state and input letters, and not on observation and input letters.

Quantitative objectives. In addition to parity (ω-regular) objectives, our algorithms will require solving games with quantitative objectives. A *quantitative objective* for G is a Borel measurable function f on infinite sequences of observations and input letters to reals, that is, $f : (L \times \Sigma)^\omega \to \mathbb{R} \cup \{\, \infty, -\infty \,\}$. We specifically consider mean-payoff and mean-payoff parity objectives. Let $r : \Sigma \to \mathbb{R}$ be a reward-function that maps every input letter σ to a real-valued reward $r(\sigma)$, and let $p : L \to \{ 0, 1, \ldots, d \}$ be a priority function. We define the mean-payoff and mean-payoff parity objectives as follows.

1. *Mean-payoff objectives.* For a play $\pi = \ell_0\sigma_0\ell_1 \ldots \sigma_{n-1}\ell_n\sigma_n \ldots$ the mean-payoff objective is the long-run average of the rewards of the input letters [38]. Formally, for a reward function $r : \Sigma \to \mathbb{R}$, the mean-payoff objective is a function $\mathsf{M}(r)$ from plays to reals that maps the play $\pi = \ell_0\sigma_0\ell_1 \ldots \sigma_{n-1}\ell_n\sigma_n \ldots$ to $\mathsf{M}(r)(\pi) = \limsup_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} r(\sigma_i)$.
2. *Mean-payoff parity objectives.* For a play $\pi = \ell_0\sigma_0\ell_1 \ldots \sigma_{n-1}\ell_n\sigma_n \ldots$ the mean-payoff parity objective is the long-run average of the rewards of the input letters if the parity objective is satisfied and $-\infty$ otherwise. Formally, for a reward function $r : \Sigma \to \mathbb{R}$ and a priority function p, the mean-payoff parity objective is a function $\mathsf{MP}(p, r)$ defined on plays as follows: for a play $\pi = \ell_0\sigma_0\ell_1 \ldots \sigma_{n-1}\ell_n\sigma_n \ldots$ we have $\mathsf{MP}(p, r)(\pi) = \mathsf{M}(\pi)$ if $\pi \in \mathsf{Parity}(p)$, and $\mathsf{MP}(p, r)(\pi) = -\infty$ otherwise.

Observe that the reward function are on input letters, rather than transition of the game graph. If we consider reward function on transitions, then mean-payoff games with imperfect information is undecidable [11], whereas if the rewards are on input letters, then the problem is EXPTIME-complete (Corollary 1).

Sure winning and optimal winning. A strategy λ_i for Player i in G is *sure winning* for a qualitative objective ϕ if for all $\pi \in \mathsf{Outcome}_i(G, \lambda_i)$, we have $\pi \models \phi$. A strategy λ_i for Player i in G is *optimal* for a quantitative objective f if for all strategies λ for Player i we have $\inf_{\pi\in\mathsf{Outcome}_i(G,\lambda_i)} f(\pi) \geq \inf_{\pi\in\mathsf{Outcome}_i(G,\lambda)} f(\pi)$. The following theorem from Martin [25] states that perfect-information games with (qualitative or quantitative) Borel objectives are *determined*: from each state, either Player 1 or Player 2 wins (for qualitative objectives), or a value can be defined (for quantitative objectives).

Theorem 1 (Determinacy). [25] *(1) For all perfect-information game structures G and all qualitative Borel objectives ϕ, either there exists a sure-winning strategy for Player 1 for the objective ϕ, or there exists a sure-winning strategy for Player 2 for the complementary objective $\mathsf{Plays}(G) \setminus \phi$. (2) For all perfect-information game structures G and all quantitative Borel objectives f, we have $\sup_{\alpha\in\mathcal{A}} \inf_{\pi\in\mathsf{Outcome}(G,\alpha)} f(\pi) = \inf_{\beta\in\mathcal{B}} \sup_{\pi\in\mathsf{Outcome}(G,\beta)} f(\pi)$.*

2.2 From Imperfect-information to Perfect-information

In this subsection we present results related to reduction of imperfect information games to perfect information games by subset construction. First, we use the results of [7] to show that a game structure G of imperfect information can be encoded by a game structure G^{K} of perfect information such that for every qualitative Borel objective ϕ, there is an observation-based sure-winning strategy for Player 1 in G for ϕ if and only if there is a sure-winning strategy for Player 1 in G^{K} for ϕ. The same construction works for quantitative Borel objectives. We obtain G^{K} using a subset construction. Each state in G^{K} is a set of states of G representing the knowledge of Player 1. In the worst case, the size of G^{K} is exponentially larger than the size of G.

Subset construction. Given a game structure of imperfect information $G = \langle L, l_0, \Sigma, \Delta, \mathcal{O}, \gamma\rangle$, we define the *knowledge-based subset construction* of G as the following game structure of perfect information: $G^{\mathsf{K}} = \langle \mathcal{L}, \{l_0\}, \Sigma, \Delta^{\mathsf{K}} \rangle$, where $\mathcal{L} = 2^L \setminus \{\emptyset\}$, and $(s_1, \sigma, s_2) \in \Delta^{\mathsf{K}}$ iff there exists an observation $o \in \mathcal{O}$ such that $s_2 = \mathsf{Post}^G_\sigma(s_1) \cap \gamma(o)$ and $s_2 \neq \emptyset$. Notice that for all $s \in \mathcal{L}$ and all $\sigma \in \Sigma$, there exists a set $s' \in \mathcal{L}$ such that $(s, \sigma, s') \in \Delta^{\mathsf{K}}$. Given a game structure of imperfect information G we refer to the game structure G^{K} as $\mathsf{Pft}(G)$.

Lemma 2 ([7]). *For all sets $s \in \mathcal{L}$ that are reachable in G^{K}, and all observations $o \in \mathcal{O}$, either $s \subseteq \gamma(o)$ or $s \cap \gamma(o) = \emptyset$.*

By an abuse of notation, we define the *observation sequence* of a play $\pi = s_0\sigma_0 s_1 \ldots \sigma_{n-1} s_n \sigma_n \ldots \in \mathsf{Plays}(G^{\mathsf{K}})$ as the infinite sequence $\gamma^{-1}(\pi) = o_0\sigma_0 o_1 \ldots \sigma_{n-1} o_n \sigma_n \ldots$ of observations such that for all $i \geq 0$, we have $s_i \subseteq \gamma(o_i)$. Since the observations partition the states, and by Lemma 2, this sequence is unique. The play π *satisfies* an objective $\phi \subseteq (\mathcal{O} \times \Sigma)^\omega$ if $\gamma^{-1}(\pi) \in \phi$. As above, we say that a play $\pi = s_0\sigma_0 s_1 \ldots \sigma_{n-1} s_n \sigma_n \cdots \in \mathsf{Plays}(G^{\mathsf{K}})$ *satisfies* an objective ϕ iff the sequence of observations $o_0 o_1 \ldots o_n \ldots$ such that for all $i \geq 0$, $\ell_i \in \gamma(o_i)$ belongs to ϕ. The following lemma follows from the results of [7].

Lemma 3 ([7]). *If Player 1 has a sure-winning strategy in G^{K} for an objective ϕ, then Player 1 has an observation-based sure-winning strategy in G for ϕ. If Player 1 does not have a deterministic sure-winning strategy in G^{K} for a Borel objective ϕ, then Player 1 does not have an observation-based sure-winning strategy in G for ϕ.*

Together with Theorem 1, Lemma 3 implies the first part of the following theorem, also used in [7]. The second part of the theorem generalizes the result to quantitative Borel objectives. The proofs can be found in [7, 9].

Theorem 2. *Let G be a game structure, and $G^{\mathsf{K}} = \mathsf{Pft}(G)$. The following assertions hold. (1) Player 1 has an observation-based sure-winning strategy in G for a qualitative Borel objective ϕ if and only if Player 1 has a sure-winning strategy in G^{K} for ϕ. (2) $\sup_{\alpha \in \mathcal{A}^O_G} \inf_{\pi \in \mathsf{Outcome}(G,\alpha)} f(\pi) = \sup_{\alpha \in \mathcal{A}_{G^{\mathsf{K}}}} \inf_{\pi \in \mathsf{Outcome}(G,\alpha)} f(\pi)$.*

Theorem 2 and the results of [8] on perfect information mean-payoff parity games show that imperfect information mean-payoff parity games can be solved in EXPTIME, and an EXPTIME lower bound follows from the lower bound for imperfect information parity games [7]. We have the following corollary.

Corollary 1. *Given an imperfect information game structure G, a priority function $p : L \to \{0, 1, \ldots, d\}$ and a reward function $r : \Sigma \cdot \mathbb{R}$, the decision problem of whether $\sup_{\alpha \in \mathcal{A}^O_G} \inf_{\pi \in \mathsf{Outcome}(G,\alpha)} \mathsf{MP}(p, r)(\pi) \geq \nu$, for a rational threshold ν, is EXPTIME-complete.*

3 Minimum Attention Control

We now consider minimum attention control of imperfect information games. We will present a polynomial reduction of the minimum attention control problem for imperfect information games to the classical imperfect information games presented in the previous section. We will also show that the minimum attention control problem is EXPTIME-complete.

Switching and monitoring costs. We associate two kinds of costs for control: *switching costs* and *monitoring costs*. The switching cost is incurred when the control switches between two input letters, and the monitoring cost is incurred when the controller monitors the state of the plant (i.e., the current observation). Formally, let $\mathsf{cost} : \Sigma \times \Sigma \to \mathbb{R}$ denote the cost of switching between two input letters (or actions), i.e., $\mathsf{cost}(\sigma, \sigma')$ denote the cost of switching from input letter σ to input letter σ'. Let mon denote the cost of monitoring, i.e., monitoring the current observation of the plant. Given a ω-regular specification specified as a parity objective, the goal of the controller is to ensure the parity objective minimizing the long-run average cost of switching and monitoring. We now formally present monitor-action strategies, the notion of cost of a play, then the notion of minimum attention control, and finally the reduction of minimum attention control problem to imperfect information games with mean-payoff parity objective.

Monitor-action strategies. Let $G = \langle L, l_0, \Sigma, \Delta, \mathcal{O}, \gamma \rangle$ be a game structure of imperfect information. Let $\widehat{\Sigma} = \{ 0, 1 \} \times \Sigma$, be the action for controller where the first component denotes monitoring or not (0 denotes no monitoring and 1 denotes monitoring). Let $\widehat{o}$ be a new observation not in $\mathcal{O}$, and let $\widehat{\gamma}$ be a new observation mapping such that $\widehat{\gamma}(o) = \gamma(o)$ for $o \in \mathcal{O}$, and $\widehat{\gamma}(\widehat{o}) = L$. If player 1 chooses not to monitor, then player 1 does not see the current observation (this is equivalent to say that player 1 gets to observe $\widehat{o}$). A *monitor-action* strategy for player 1 is a function $\widehat{\alpha} : (L \times \widehat{\Sigma})^* \times L \to \widehat{\Sigma}$. Given a play $\pi = l_0 \widehat{\sigma}_0 l_1 \widehat{\sigma}_1 l_2 \widehat{\sigma}_2 \ldots$, the observation sequence $\widehat{\gamma}^{-1}\pi = o_0 \widehat{\sigma}_0 o_1 \widehat{\sigma}_1 o_2 \widehat{\sigma}_2 \ldots$, where $o_i = \gamma^{-1}(l_i)$ if $\sigma_i \in \{ 1 \} \times \Sigma$, and $\widehat{o}$ otherwise. A monitor-action strategy $\widehat{\alpha}$ is observation-based, if for all finite prefixes $\widehat{\rho}, \widehat{\rho}' \in (L \times \widehat{\Sigma})^* \times L$ such that $\widehat{\gamma}^{-1}(\widehat{\rho}) = \widehat{\gamma}^{-1}(\widehat{\rho}')$ we have $\widehat{\alpha}^{-1}(\widehat{\rho}) = \widehat{\alpha}^{-1}(\widehat{\rho}')$.

Cost of a play. Given a play $\pi = l_0 \widehat{\sigma}_0 l_1 \widehat{\sigma}_1 l_2 \widehat{\sigma}_2 \ldots$, the monitor-switching cost of π is as follows. For $i > 1$ and $z \in \{ 0, 1 \}$, let $\widehat{c}(\widehat{\sigma}_i) = \mathsf{cost}(\sigma_{i-1}, \sigma_i)$ if $\widehat{\sigma}_i = (0, \sigma_{i-1})$ and $\widehat{\sigma}_{i-1} = (z, \sigma_{i-1})$, and $\widehat{c}(\widehat{\sigma}_i) = \mathsf{cost}(\sigma_{i-1}, \sigma_i) + \mathsf{mon}$ if $\widehat{\sigma}_i = (1, \sigma_{i-1})$ and $\widehat{\sigma}_{i-1} = (z, \sigma_{i-1})$. denote the cost of monitoring and switching in the i-th step. Then the cost of the play is defined as the long-run average of the monitoring and switching cost, i.e., $\widehat{c}(\pi) = \limsup_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} \widehat{c}(\widehat{\sigma}_{i-1}, \widehat{\sigma}_i)$.

Minimum attention control. Given an imperfect information game structure G, and a parity objective ϕ, a monitor-action strategy $\widehat{\alpha}$ is ν-frugal iff the following conditions hold: (a) $\widehat{\alpha}$ is obervation-based; (b) for all π in $\mathsf{Outcome}(G, \widehat{\alpha})$ we have $\pi \in \phi$, i.e., the parity objective is ensured; and (c) $\widehat{c}(\pi) \leq \nu$, i.e., the monitor-switching cost is at most ν. In other words, the strategy $\widehat{\alpha}$ ensures the parity objective without incurring monitor-switching cost more than ν.

Reduction of games with move assignment. We will present a reduction of the minimum attention control problem to imperfect information games with mean-payoff parity objectives. We first consider an extension of imperfect information games where there is an input assignment function $\Gamma_1 : L \to 2^{\Sigma} \setminus \emptyset$, that assigns to every state ℓ the set of available input letters $\Gamma_1(\ell)$, i.e., every input letter may not be available at every state. Imperfect information games with input assignment function can be reduced to imperfect information games with no input assignment function as follows: (1) add an additional absorbing state $\widetilde{\ell}$ that is loosing for player 1; (2) for a state ℓ and an input letter $\sigma \in \Sigma \setminus \Gamma_1(\ell)$ not available at ℓ by the input assignment, we add σ as available input, and add a transition from ℓ to the loosing state $\widetilde{\ell}$ for σ. Thus it is ensured if player 1 chooses an input that is not available, then player 1 loses immediately.

Reduction of minimum attention control. Let $G = \langle L, l_0, \Sigma, \Delta, \mathcal{O}, \gamma \rangle$ be a game structure of imperfect information with a priority function $p : L \to \{ 0, 1, \ldots, d \}$. For minimum attention control we construct a game structure of imperfect information with move assignment as follows: the game structure is $\widetilde{G} = \langle \widetilde{L}, \widetilde{l_0}, \widetilde{\Sigma}, \widetilde{\Delta}, \widetilde{\mathcal{O}}, \widetilde{\gamma}, \widetilde{\Gamma_1} \rangle$ along with cost function $\widetilde{c} : \widetilde{\Sigma} \to \mathbb{R}$. We describe the components below.

1. *State space.* We have $\widetilde{L} = (L \times \{ 0, 1, 2 \} \times \Sigma) \cup (\{ l_0 \} \times \{ 0, 1, 2 \})$. The first component is the state of G, the second component is 0 or 1 depending on whether player 1 decides to monitor or not, and 2 if it is player 1's turn to decide whether to monitor or not. The third component is an input letter (to remember the choice of last letter of player 1). Additionally, there are states of the form (l_0, j) for $j \in \{ 0, 1, 2 \}$. The starting state $\widetilde{l_0}$ is $(l_0, 2)$.
2. *Input letters.* We have $\widetilde{\Sigma} = (\Sigma \times \Sigma) \cup \{ 0, 1 \}$, i.e., the set of input letters is a pair of input letters of the original game (switching between input letters) with the $\{ 0, 1 \}$ to denote the choice of monitoring.
3. *Observation.* We have $\widetilde{\mathcal{O}} = \mathcal{O} \cup \{ \widetilde{o} \}$, where $\widetilde{o}$ is a new observation.
4. *Move assignment.* We have $\widetilde{\Gamma}_1((\ell, 2, \sigma)) = \{ 0, 1 \}$ for $\ell \in L$ and $\sigma \in \Sigma$; and $\widetilde{\Gamma}_1((\ell, j, \sigma)) = \{ (\sigma, \sigma') \mid \sigma' \in \Sigma \}$ for $\ell \in L, j \in \{ 0, 1 \}$ and $\sigma \in \Sigma$. At states where the second component is 2, player 1 can choose between two input letters: 0 to denote no monitoring, and 1 to denote monitoring. At states where the second component is 0 or 1, player 1 can choose input letters matching with the input letter of the state (player 1 specifies the switching from the last letter to a new letter). Similarly, we have $\Gamma_1((l_0, 2)) = \{ 0, 1 \}$ and $\Gamma_1((l_0, j)) = \Sigma$, for $j \in \{ 0, 1 \}$.
5. *Transition function.* We have the following cases: (a) for states $(\ell, 2, \sigma)$ we have $((\ell, 2, \sigma), j, (\ell, j, \sigma)) \in \widetilde{\Delta}$, for $j \in \{ 0, 1 \}$, i.e., given the choice of input letter only the second component of state changes according to the input letter; (b) for states (ℓ, j, σ) with $j \in \{ 0, 1 \}$ we have $((\ell, j, \sigma), (\sigma, \sigma')(\ell', 2, \sigma')) \in \widetilde{\Delta}$ iff $(\ell, \sigma', \ell') \in \Delta$, i.e., the transition of the game structure is mimicked according to the first component, the second component changes to 2, and the last input letter is remembered in the third component; (c) for state $(l_0, 2)$ we have $((l_0, 2), j, (l_0, j)) \in \widetilde{\Delta}$ for $j \in \{ 0, 1 \}$; and (d) for states (l_0, j), with $j \in \{ 0, 1 \}$ we have $((l_0, j), \sigma'(\ell', 2, \sigma')) \in \widetilde{\Delta}$ iff $(l_0, \sigma', \ell') \in \Delta$.
6. *Observation mapping.* We have (a) $\widetilde{\gamma}^{-1}((\ell, j, \sigma)) = \widetilde{\gamma}^{-1}((l_0, j)) = \widetilde{o}$, for $j \in \{ 0, 2 \}$; and (b) $\widetilde{\gamma}^{-1}((\ell, 1, \sigma)) = \gamma(\ell)$ and $\widetilde{\gamma}^{-1}((l_0, 1)) = \gamma(l_0)$; i.e., when the second component is 0 or 2, then player 1 is not monitoring and hence observes nothing, and otherwise if the second component is 1, then player 1 is monitoring and hence observes the observation of the original game.
7. *Cost function.* We have $\widetilde{c}(0) = 0$ (no cost); $\widetilde{c}(1) = -\mathsf{mon}$ (cost of monitoring); and $\widetilde{c}((\sigma, \sigma')) = -\mathsf{cost}(\sigma, \sigma')$ (cost of switching).
8. *Parity function.* The priority function $\widetilde{p} : \widetilde{L} \to \{0, 1, \ldots, d\}$ is obtained as follows: for a state $\widetilde{\ell} \in \widetilde{L}$ the priority $\widetilde{p}(\widetilde{\ell})$ is $p(\ell)$, where ℓ is the first component of $\widetilde{\ell}$.

There is a one-to-one correspondence between monitor-action strategies that are observation-based in G, and observation-based strategies in the game $\widetilde{G}$. The cost incurred in G in every step is incurred in two steps in $\widetilde{G}$ as in $\widetilde{G}$ we mimic the choice of monitoring and choice of action switch of G in two steps. Hence we have the following lemma.

Lemma 4. *Let G be a game structure of imperfect information, and let $p : L \to \{ 0, 1, \ldots, d \}$ be a priority function. There is a monitor-action strategy $\widehat{\alpha}$ in G that is ν-frugal for the objective* $\mathsf{Parity}(p)$ *iff* $\sup_{\alpha \in \mathcal{A}^O_{\widetilde{G}}} \inf_{\pi \in \mathsf{Outcome}(\widetilde{G}, \alpha)} \mathsf{MP}(\widetilde{c}, \widetilde{p})(\pi) \geq -\frac{\nu}{2}$.

We have the following result for minimum attention control: (a) the EXPTIME upper bound follows from Corollary 1 and Lemma 4 and the fact that our reduction from G to $\widetilde{G}$ is polynomial; (b) the lower bound follows from EXPTIME-hardness of imperfect information parity games: with monitoring and switching costs both set to 0, the minimum attention control problem is the same as winning an imperfect information parity game.

Theorem 3. *The minimum attention control problem for imperfect information game structures with parity objectives is EXPTIME-complete.*

We have assumed that the winning objective is given as a parity condition on the state space of the game. If instead, we are given a two-player game structure, and separately, a specification in linear-temporal logic (LTL) [28], then standard automata-theoretic constructions [34] can be used to reduce the problem to our case. That is, from the LTL specification φ, one constructs a deterministic parity automaton whose size is at most doubly exponential in the size of φ and whose number of parities is exponential in the size of φ. A synchronous product of the game structure with this automaton gives an imperfect information game whose size is the product of the size of the original game and the size of the automaton. The solution to the imperfect information game involves a subset construction, adding an extra exponential, and then solving a mean-payoff parity game which is polynomial in the size of the game and exponential in the number of parities. But a triple exponential raised to a single exponential is still triple exponential, so we conclude the following.

Corollary 2. *The minimum attention control problem for imperfect information game structures and linear-temporal logic specifications is in 3EXPTIME.*

We note that the high complexity of our procedure is disappointing, and unlikely to yield an efficient tool. It will be interesting to see if more efficient algorithms can be designed for fragments of LTL.

4 Infinite State Systems

We now apply the theory of minimum attention controller synthesis to the discrete time control problem for *rectangular automata* [16]. We obtain our results using a general decidability result about imperfect-information games on infinite state spaces that have a stable partition with a finite quotient.

R-stable games. In this section we drop the assumption of finite state space of games. Let $G = \langle L, l_0, \Sigma, \Delta, \mathcal{O}, \gamma \rangle$ be a game structure of imperfect-information such that L is infinite. Let $R = \{ r_1, r_2, \ldots, r_l \}$ be a finite partition of L. A set $Q \subseteq L$ is *R-definable* if $Q = \bigcup_{r \in Z} r$, for some $Z \subseteq R$. The game G is *R-stable* if the following conditions hold for all $\sigma \in \Sigma$: (a) the set $\{ l \in L \mid \exists l' \in L.(l, \sigma, l') \in \Delta \}$ is R-definable; (b) for all $r \in R$, the set $\mathsf{Post}_\sigma^G(r)$ is R-definable; (c) for all $r, r' \in R$, if for some $x \in r$ we have $\mathsf{Post}_\sigma^G(\{ x \}) \cap r' \neq \emptyset$, then for all $x' \in r$ we have $\mathsf{Post}_\sigma^G(\{ x' \}) \cap r' \neq \emptyset$; and (d) for all $o \in \mathcal{O}$, the set $\gamma(o)$ is R-definable.

Lemma 5. *The following assertions hold. (1) Let G be a game structure of imperfect information, and let R be a finite partition of the state space of G such that the game G is R-stable. Then the perfect-information game $\mathsf{Pft}(G)$ is 2^R-stable. (2) Let $\overline{G}$ be a perfect-information game structure with a parity objective with d-priorities and a mean-payoff objective with rewards on Σ such that the maximal absolute value of the rewards is W. If $\overline{G}$ is $\overline{R}$-stable, for a given finite partition $\overline{R}$, then value for the mean-payoff parity objective in $\overline{G}$ can be computed in time $O((|\overline{R}| \cdot W)^{d+5})$.*

We present the definition of rectangular automata with imperfect information and then reduce the minimum attention control problem to the problem of game with imperfect information. Using a result of [16] we establish the game of imperfect information is R-stable for a finite set R.

Rectangular constraints. Let $Y = \{ y_1, y_2, \ldots, y_k \}$ be a set of real-valued variables. A *rectangular inequality* over Y is of the form $x_i \sim d$, where d is an integer constant, and $\sim \in \{ \leq, <, \geq, > \}$. A *rectangular predicate* over Y is a conjunction of rectangular inequalities. We denote the set of rectangular predicates over Y as $\mathit{Rect}(Y)$. The rectangular predicate ϕ defines the set of vectors $[\![\phi]\!] = \{ y \in \mathbb{R}^k \mid \phi[Y := y] \text{ is true} \}$. For $1 \leq i \leq k$, let $[\![\phi]\!]_i$ be the projection on variable y_i of the set $[\![\phi]\!]$. A set of the form $[\![\phi]\!]$, where ϕ is a rectangular predicate, is called a rectangle. Given a non-negative integer $m \in \mathbb{N}$, the rectangular predicate ϕ is m-bounded if $|d| \leq m$, for every conjunct $y_i \sim d$ of ϕ. Let us denote by $\mathit{Rect}_m(Y)$ the set of m-bounded rectangular predicates on Y.

Rectangular automata. A *rectangular automaton of imperfect information H* is a tuple $\langle Q, \mathit{Lab}, \mathit{Edg}, Y, \mathit{Init}, \mathit{Inv}, \mathit{Flow}, \mathit{Jump}, \mathcal{O}, \gamma \rangle$ where (a) Q is a finite set of locations; (b) Lab is a finite set of *labels*; (b) $\mathit{Edg} \subseteq Q \times \mathit{Lab} \times Q$ is a finite set of *edges*; (d) $Y = \{ y_1, y_2, \ldots, y_k \}$ is a finite set of *variables*; (e) $\mathit{Init} : Q \to \mathit{Rect}(Y)$ gives the *initial condition* $\mathit{Init}(q)$ of a location q; (f) $\mathit{Inv} : Q \to \mathit{Rect}(Y)$ gives the *invariant condition* $\mathit{Inv}(q)$ of location q (i.e., the automaton can stay in q as long as the values of variables lie in $[\![\mathit{Inv}(v)]\!]$); (g) $\mathit{Flow} : Q \to \mathit{Rect}(\dot{Y})$ governs the evolution of the variables in each location; (h) Jump maps each edge e to a predicate $\mathit{Jump}(e)$ of the form $\phi \wedge \phi' \wedge \bigwedge_{i \not\in \mathit{Update}(e)} (y'_i = y_i)$, where $\phi \in \mathit{Rect}(Y)$, $\phi' \in \mathit{Rect}(Y')$, and $\mathit{Update}(e) \subseteq \{ 1, 2, \ldots, k \}$; (i) $\mathcal{O}$ is a finite set of observations and $\gamma : \mathcal{O} \to 2^Q \setminus \emptyset$ is the observation mapping such that $\{ \gamma(o) \mid o \in \mathcal{O} \}$ is a partition of Q. The variables in Y' refer to the updated values of the variables after the edge has been traversed. Each variable y_i with $i \in \mathit{Update}(e)$ is updated nondeterministically to a new value in $[\![\phi']\!]_i$. A rectangular automaton is m-bounded if all rectangular constraints are m-bounded. A rectangular automaton is called a *timed automaton* if for each variable $y \in Y$ and each state $q \in Q$, we have $1 \leq \mathit{Flow}(q)(\dot{y}) \leq 1$.

Nondecreasing and bounded variables. Let H be a rectangular automaton, and let $i \in \{ 1, 2, \ldots, k \}$. The variable y_i of H is *nondecreasing* if for all $q \in Q$, the invariant interval $[\![Inv(q)]\!]_i$ and the flow interval $[\![Flow(q)]\!]_i$ are subsets of the nonnegative reals. The variable y_i of H is *bounded* if for all $q \in Q$, the invariant interval $[\![Inv(q)]\!]_i$ is a bounded set. The automaton H is bounded (resp. nondecreasing) if all the variables are bounded (resp. nondecreasing). In sequel we consider automata that are bounded or nondecreasing.

Game semantics. The rectangular automaton game with imperfect information is played as follows: the game starts at a location q and values for the continuous variables $y \in [\![Init(q)]\!]$. At each round the controller can choose to observe (monitor) the observation (paying the monitoring cost) or not; and then the controller decides to take one of the enabled edges (if one exists). Then the environment nondeterministically updates the continuous variables according to the flow predicates by letting time pass for 1 time unit. Then the new round of the game starts. We now present a reduction to imperfect-information game, and then show that the game is stable with respect to a finite partition.

Reduction. A rectangular automaton H with imperfect information $\langle Q, Lab, Edg, Y, Init, Inv, Flow, Jump, \mathcal{O}, \gamma \rangle$ reduces to an infinite state imperfect-information game $\overline{H} = \langle L, l_0, \Sigma, \Delta, \mathcal{O}, \overline{\gamma} \rangle$ as follows:

1. *States.* The set of states is $L = Q \times \mathbb{R}^k$; that is the set of states consists of a tuple of location and values of variables.
2. *Input letters.* The set of input letters is $\Sigma = Lab \cup \{ 1 \}$. The set of input letters is the set of labels Lab of H, and unit time 1.
3. *Observation map.* The observation map is as follows: $\overline{\gamma}(o) = \{ (q, y) \in L \mid \gamma(q) = o \}$.
4. *Transition function.* The transition function is as follows: (a) $((q, y), \sigma, (\overline{q}', y')) \in \Delta$, such that there exists $e = (q, \sigma, q') \in Edg$ with $(y, y') \in [\![Jump(e)]\!]$; and (c) $((q, y), 1, (q, y')) \in \Delta$ such that there exists a continuously differentiable function $f : [0, 1] \to Inv(q)$ such that $f(0) = y$, $f(1) = y'$ and for all $t \in (0, 1)$ we have $\dot{f}(t) \in [\![Flow(q)]\!]$.

The set of observation-based strategies of $\overline{H}$ represents the observation-based strategies for the rectangular automaton game defined by H.

Equivalence relation. Let H be a m-bounded rectangular automaton with imperfect information, and let $\overline{H}$ be the game of imperfect information obtained by the reduction. We define the equivalence relation $\equiv_m$ on the state space as follows: $(q, y) \equiv_m (q', y')$ iff (a) $q = q'$; and (b) for all $1 \leq i \leq k$, either $\lfloor y_i \rfloor = \lfloor y_i' \rfloor$ and $\lceil y_i \rceil = \lceil y_i' \rceil$, or both y_i and y_i' are greater than m. We denote by $R_{\equiv_m}$ the set of equivalence classes of $\equiv_m$. It is easy to observe that $R_{\equiv_m}$ is finite (in fact exponential in the size of H). An extension of the result of [16] gives us the following result.

Lemma 6. *Let H be a m-bounded rectangular automaton game with imperfect information. The imperfect-information game $\overline{H}$ is $R_{\equiv_m}$-stable.*

Theorem 4. *Let H be a rectangular automaton with imperfect information and let $p : Q \to \{ 0, 1, \ldots, d \}$ be a priority function. Let $\overline{p} : Q \cdot \mathbb{R}^k \to \{ 0, 1, \ldots, d \}$ be such that $\overline{p}(q, y) = p(q)$, for $q \in Q$ and $y \in \mathbb{R}^k$. The answer to the ν-frugal problem for H for* $\mathsf{Parity}(p)$ *is true iff the answer to the ν-frugal problem is true in $\overline{H}$ for* $\mathsf{Parity}(\overline{p})$.

From Lemma 5, Lemma 6, Theorem 4, and Theorem 3 we obtain the following corollary.

Corollary 3. *Let H be a rectangular automaton with imperfect information and let $p : Q \to \{ 0, 1, \ldots, d \}$ be a priority function. Whether there is a ν-frugal strategy for the controller in H to satisfy the objective* $\mathsf{Parity}(p)$ *can be decided in 2EXPTIME.*

The result for timed automata follows similarly, using the finite bisimilarity relation for timed automata [1].

5 Discussions

We have presented algorithms for minimum attention controller synthesis for ω-regular objectives purely in the discrete setting, or in a setting (rectangular automata) which can be reduced to the discrete setting. A natural next step is to extend the results in the presence of more general continuous dynamics. One potential direction is to combine the optimization problems in [3] with mean-payoff parity games as described here. This seems hard algorithmically, because the optimization problem in [3] is already quite difficult (and even in the case of linear dynamics, it is not obvious if closed form solutions can be obtained).

A different direction is motivated by work in *approximate abstraction* of continuous control models [30, 14, 13]. The results in these papers provide techniques to abstract the continuous state space and dynamics of systems to discrete systems such that, any controller for the discrete system is guaranteed to ensure the property in the continuous system up to an error of ϵ, where ϵ is a parameter of the abstraction. Therefore, the study of minimum attention control for hybrid dynamics can be broken into two parts: first, construct a discrete abstraction of the continuous system, and second, apply the techniques described in this paper to solve the problem of minimum attention control.

References

1. R. Alur and D. Dill. A theory of timed automata. *Theoretical Computer Science*, 126(2):183–236, 1994.
2. A. Anta and P. Tabuada. On the minimum attention and anytime attention problems for nonlinear systems. In *CDC 10*. IEEE, 2010.
3. R.W. Brockett. Minimum attention control. In *CDC 97*. IEEE, 1997.
4. R.W. Brockett. Minimizing attention in a motion control context. In *CDC 03*. IEEE, 2003.
5. J.R. Büchi and L.H. Landweber. Solving sequential conditions by finite-state strategies. *Transactions of the AMS*, 138:295–311, 1969.
6. F. Cassez and S. Tripakis. Fault diagnosis with static and dynamic observers. *Fundam. Inform.*, 88(4):497–540, 2008.
7. K. Chatterjee, L. Doyen, T.A. Henzinger, and J.F. Raskin. Algorithms for omega-regular games with imperfect information. In *CSL'06*, pages 287–302. LNCS 4207, Springer, 2006.
8. K. Chatterjee, T.A. Henzinger, and M. Jurdziński. Mean-payoff parity games. In *LICS'05*, pages 178–187. IEEE, 2005.
9. K. Chatterjee, R. Majumdar, and T. A. Henzinger. Controller synthesis with budget constraints. In *HSCC*, pages 72–86, 2008.

10. A. Church. Logic, arithmetic, and automata. In *Proceedings of the International Congress of Mathematicians*, pages 23–35. Institut Mittag-Leffler, 1962.
11. A. Degorre, L. Doyen, R. Gentilini, J.-F. Raskin, and S. Toruńczyk. Energy and mean-payoff games with imperfect information. In *Proc. of CSL*, LNCS. Springer, 2010.
12. E.A. Emerson and C. Jutla. Tree automata, mu-calculus and determinacy. In *FOCS'91*, pages 368–377. IEEE, 1991.
13. A. Girard. Synthesis using approximately bisimilar abstractions: state-feedback controllers for safety specifications. In *HSCC*, pages 111–120. ACM, 2010.
14. A. Girard, A. A. Julius, and G. J. Pappas. Approximate simulation relations for hybrid systems. *Discrete Event Dynamic Systems*, 18(2):163–179, 2008.
15. Y. Gurevich and L. Harrington. Trees, automata, and games. In *STOC'82*, pages 60–65. ACM, 1982.
16. T.A. Henzinger and P.W. Kopke. Discrete-time control for rectangular hybrid automata. *Theoretical Computer Science*, 221:369–392, 1999.
17. B. Jobstmann. *Applications and Optimizations for LTL Synthesis*. PhD thesis, Graz University of Technology, March 2007.
18. A. Kechris. *Classical Descriptive Set Theory*. Springer, 1995.
19. M. Kloetzer and C. Belta. A fully automated framework for control of linear systems from temporal logic specifications. *IEEE Transactions on Automatic Control*, 53(1):287–297, 2008.
20. H. Kress-Gazit, G. E. Fainekos, and G. J. Pappas. Temporal logic-based reactive mission and motion planning. *IEEE Transactions on Robotics*, 25(6):1370–1381, 2009.
21. O. Kupferman and M.Y. Vardi. μ-calculus synthesis. In *Proc. 25th International Symp. on Mathematical Foundations of Computer Science*, volume 1893 of *Lecture Notes in Computer Science*, pages 497–507. Springer-Verlag, 2000.
22. O. Kupferman and M.Y. Vardi. Synthesis with incomplete informatio. In *Advances in Temporal Logic*, pages 109–127. Kluwer Academic Publishers, January 2000.
23. O. Kupferman and M.Y. Vardi. Safraless decision procedures. In *FOCS 05: Foundations of Computer Science*, pages 531–540, 2005.
24. O. Maler, A. Pnueli, and J. Sifakis. On the synthesis of discrete controllers for timed systems. In *STACS'95*, LNCS 900, pages 229–242. Springer-Verlag, 1995.
25. D.A. Martin. Borel determinacy. *Annals of Mathematics*, 102(2):363–371, 1975.
26. Manuel Mazo, Anna Davitian, and Paulo Tabuada. Pessoa: A tool for embedded controller synthesis. In *CAV*, volume 6174 of *Lecture Notes in Computer Science*, pages 566–569. Springer, 2010.
27. L. Montestruque and P. Antsaklis. On the model-based control of networked systems. *Automatica*, 39(10):1837–1843, 2003.
28. A. Pnueli. The temporal logic of programs. In *FOCS 77*, pages 46–57, 1977.
29. A. Pnueli and R. Rosner. On the synthesis of a reactive module. In *POPL'89*, pages 179–190. ACM, 1989.
30. G. Pola, A. Girard, and P. Tabuada. Approximately bisimilar symbolic models for nonlinear control systems. *Automatica*, 44(10):2508–2516, 2008.
31. M.O. Rabin. *Automata on Infinite Objects and Church's Problem*. Number 13 in Conference Series in Mathematics. American Mathematical Society, 1969.
32. P.J.G. Ramadge and W.M. Wonham. The control of discrete event systems. *IEEE Transactions on Control Theory*, 77:81–98, 1989.
33. J.H. Reif. The complexity of two-player games of incomplete information. *Journal of Computer and System Sciences*, 29:274–301, 1984.
34. S. Safra. On the complexity of ω-automata. In *Proceedings of the 29th Annual Symposium on Foundations of Computer Science*, pages 319–327. IEEE Computer Society Press, 1988.

35. W. Thomas. Languages, automata, and logic. In *Handbook of Formal Languages*, volume 3, Beyond Words, chapter 7, pages 389–455. Springer, 1997.
36. D. Thorsley and D. Teneketzis. Active acquisition of information for diagnosis and supervisory control of discrete event systems. *Discrete Event Dynamic Systems*, 17:531–583, 2007.
37. W. Zielonka. Infinite games on finitely coloured graphs with applications to automata on infinite trees. In *Theoretical Computer Science*, volume 200(1-2), pages 135–183, 1998.
38. U. Zwick and M.S. Paterson. The complexity of mean payoff games on graphs. *Theoretical Computer Science*, 158:343–359, 1996.