# A Decidable Class of Problems for Control under Partial Observation

Sophie Pinchinat [a] and Stéphane Riedweg [b]

[a] *IRISA, Rennes, France,* pinchina@irisa.fr

[b] *LSV ENS-Cachan, Cachan, France,* riedweg@lsv.ens-cachan.fr

*Key words:* Supervision, Discrete-events systems, Optimal control, Temporal logic, Specification Language, Formal Methods.

---

The commonly accepted control theory for discrete event systems, due to Ramadge and Wonham [13], followed by several other [17,4], has been more recently extended to temporal logic specifications [8,2,14]. Control consists in supervising a plant to guarantee some desired behavior, called *control objectives*; the objectives are standard properties such as *non-blocking, safety*, temporal logic definable behaviors, etc. Concerning the nature of the supervision, it is natural and standard to suppose a partial observation of the plant, as information on it moves and states is incomplete; we then talk about *control under partial observation* (see [11]).
In this paper, we adapt the logical approach of [14] to specify control under partial observation. This approach is based on quantification over atomic propositions of the mu-calculus of [7], called the quantified mu-calculus. We prove the decidability of controller synthesis when the specification is a *nested observational* formula: the construction of controllers relies on the generalizations of the automata quotient of [2] and the automata projection of [14]. An immediate important corollary is the synthesis of maximally permissive controllers under partial observation for mu-calculus definable control objectives. To our knowledge, maximal permissiveness of controllers has never been properly answered before: permissiveness is manageable in the regular languages framework [10], but becomes intricate when branching-time objectives are considered. The few results of the literature are [14] which concern control problems with full observation, and [8,2] which do not take maximal permissiveness into account.
The paper is organized as follows: Sec.1 introduces the quantified mu-calculus and show its adequacy to control specification ; Sec.2 is dedicated to the control synthesis for the decidable fragment of the logic.

## 1 Quantified Mu-Calculus for Control Specifications

We assume given a finite set of events $\mathtt{Ev} = \{\alpha, \sigma, \sigma', \dots\}$ and a finite set of atomic propositions $AP = \{c, p, \dots\}$. Models of systems are operational: they are standard state machines called here *processes*.

**Definition 1 Processes.** *Given two sets $\Sigma \subseteq \mathtt{Ev}$ and $\Gamma \subseteq AP$, a* process on *$(\Sigma, \Gamma)$ is a tuple $\mathcal{S} = \langle S, s^0, t, L\rangle$, where $S$ is the set of states, $s^0 \in S$ is the initial state, $t : S \times \Sigma \to S$ is the* transition function *- it is partial -, and $L : S \to 2^\Gamma$ is a labeling function, labeling states with propositions - it is total. The set $\Sigma$ is called* the type of $\mathcal{S}$. *The process $\mathcal{S}$ is* complete *if $t(s, \sigma)$ is defined for all $s \in S$ and $\sigma \in \Sigma$, and it is* finite *if $S$ is finite. We will use $\mathcal{S}, \mathcal{P}, \mathcal{E}, \mathcal{E}', \dots$ as typical elements for processes.*
*The* (weak) synchronous product *of processes $\mathcal{S}_1 = \langle S_1, s_1^0, t_1, L_1\rangle$ of type $\Sigma_1$ and $\mathcal{S}_2 = \langle S_2, s_2^0, t_2, L_2\rangle$ of type $\Sigma_2$ is the process $\mathcal{S}_1 \otimes \mathcal{S}_2 = \langle S_1 \times S_2, (s_1^0, s_2^0), t, L\rangle$ on $(\Sigma_1 \cup \Sigma_2, \Gamma_1 \cup \Gamma_2)$, hence of type $\Sigma_1 \cup \Sigma_2$, where $t((s_1, s_2), \sigma) = (s_1', s_2')$ whenever $(\sigma \in \Sigma_1 \cap \Sigma_2)$ and $(s_1 \xrightarrow{\sigma} s_1')$ and $(s_2 \xrightarrow{\sigma} s_2')$ ; or $(\sigma \in \Sigma_1 \setminus \Sigma_2)$ and $(s_1 \xrightarrow{\sigma} s_1')$ and $(s_2' = s_2)$ ; or $(\sigma \in \Sigma_2 \setminus \Sigma_1)$ and $(s_1' = s_1)$ and $(s_2 \xrightarrow{\sigma} s_2')$. Moreover, $L(s_1, s_2) = L_1(s_1) \cup L_2(s_2)$.*

The weak synchronous product gives the formal meaning to the notion of controller under partial observation : given a process $\mathcal{P}$ - for "plant" - of type $\Sigma$, a subset $O \subseteq \Sigma$ of *observable* events, and a property $\Psi$ on processes - which will be made clear further - *a controller of $\mathcal{P}$ for $\Psi$ under observation $O$* is some non-empty process $\mathcal{C}$ on $(O, \emptyset)$ with additional properties we explain now. The controlled plant is the process $\mathcal{P} \otimes \mathcal{C}$ and it satisfies $\Psi$. Note that $\mathcal{C}$ cannot prevent non-observable events from occurring, and neither can it take the occurrence of a non-observable event into account. Moreover, it is standard that disallowing transitions is subject to additional constraints : the observable events set $O$ splits into the *controllable* events set $\Sigma_{co}$ and the *uncontrollable* events set $\Sigma_{uco}$. Hence, realistic controllers should disallow only *controllable* transitions (labeled by controllable events), the controllers are then called *admissible*. In the following, an admissible controller under observation $O$ of $\mathcal{P}$ for $\Psi$ is called a controller for $(\mathcal{P}, \Sigma_{co}, O, \Psi)$. The property $\Psi$ is any formula of the mu-calculus logic [7,1], which subsumes classical temporal logics as $CTL^\star$ [5] . We take the convention that given $\mathcal{P}$ on $(\Sigma, \Gamma)$ with $\Sigma_{co} \subseteq O \subseteq \Sigma$, and a mu-calculus formula $\Psi$, the *Basic Control Under Partial Observation Problem* is: "*Does there exist a controller for $(\mathcal{P}, \Sigma_{co}, O, \Psi)$? If so, compute one*".
We recall what the mu-calculus (written $L_\mu$) is.

**Definition 2** *Assume given a set of variables $Var = \{X, Y, \dots\}$, the set of formula of $L_\mu$ is defined, for any $\sigma \in \mathtt{Ev}$, $p \in AP$ and $X \in Var$, by:*

$$\Phi ::= \top \mid p \mid X \mid \neg\Phi \mid \Phi \vee \Phi \mid \langle\sigma\rangle\Phi \mid \mu X.\Phi(X)$$

*Fixed-point formulas $\mu X.\Phi(X)$ can be properly interpreted whenever each occurrence of $X$ in $\Phi(X)$ is under the scope of an even number of negation symbols $\neg$ (see [1]).* Sentences *are the formulas for which each occurrence of a variable $X$ is beneath some $\mu X$. We will use $\bot$, $\Phi \wedge \Phi'$, $[\sigma]\Phi$, $\nu X.\Phi(X)$ respectively for $\neg\top$, $\neg(\neg\Phi \vee \neg\Phi')$, $\neg\langle\sigma\rangle\neg\Phi$, $\neg\mu X.\neg\Phi(\neg X)$, as well as $\xrightarrow{\sigma}$, $[\ ]\Phi$ and $\Phi \Rightarrow \Phi'$ respectively for $\langle\sigma\rangle\top$, $\bigwedge_{\sigma\in\mathtt{Ev}}[\sigma]\Phi$ and $\neg\Phi \vee \Phi'$. For $\Phi \in L_\mu$, $\mathbf{AG}\Phi$ is a notation for $\nu X.[\ ]X \wedge \Phi$: $\mathbf{AG}\Phi$ expresses the* invariance *of $\Phi$, namely "from now on, the property $\Phi$ always holds".*

*Given a process $\mathcal{S} = \langle S, s^0, t, L\rangle$ and a valuation $val : Var \to 2^S$, the interpretation $[\![\,\Phi\,]\!]_{\mathcal{S}}^{[val]}$ of an $L_\mu$-formula $\Phi$ is a subset of $S$ defined inductively by: $[\![\,p\,]\!]_{\mathcal{S}}^{[val]} = \{s \in S \,|\, p \in L(s)\}$, $[\![\,\neg\Phi\,]\!]_{\mathcal{S}}^{[val]} = S \setminus [\![\,\Phi\,]\!]_{\mathcal{S}}^{[val]}$, $[\![\,\top\,]\!]_{\mathcal{S}}^{[val]} = S$, $[\![\Phi\vee\Phi']\!]_{\mathcal{S}}^{[val]} = [\![\Phi]\!]_{\mathcal{S}}^{[val]} \cup [\![\Phi']\!]_{\mathcal{S}}^{[val]}$, $[\![\langle\sigma\rangle\Phi]\!]_{\mathcal{S}}^{[val]} = \{s\in S | \exists s' \in [\![\Phi]\!]_{\mathcal{S}}^{[val]}, t(s,\sigma) = s'\}$, $[\![\,\mu X.\Phi(X)\,]\!]_{\mathcal{S}}^{[val]} = \bigcap\{V \subseteq S \,|\, [\![\,\Phi\,]\!]_{\mathcal{S}}^{[val(V/X)]} \subseteq V\}$, $[\![\,X\,]\!]_{\mathcal{S}}^{[val]} = val(X)$. Since for sentences $[\![\,\Phi\,]\!]_{\mathcal{S}}^{[val]}$ is independent of val, we simply write $[\![\,\Phi\,]\!]_{\mathcal{S}}$, and write $\mathcal{S} \models \Phi$, read "$\mathcal{S}$* satisfies *$\Phi$," whenever $s^0 \in [\![\,\Phi\,]\!]_{\mathcal{S}}$.*

We consider now the logic $\mathrm{Q}L_\mu$ (the *quantified mu-calculus*) introduced in [14] propositions. We and we extend it to specify controllers under partial observation . To define the logic, we consider particular classes of processes : we let $Lab_p^O$ be the set of complete processes on $(O, \{p\})$ (called labelling processes in [14]).

**Definition 3** *The set of formulas of the Quantified-Mu-Calculus (written $\mathrm{Q}L_\mu$) is defined by:*

$$\varphi ::= \exists p(O).\varphi \ |\neg\varphi|\varphi \vee \varphi|\Phi,$$

*where $p \in AP$, $O \subseteq \mathtt{Ev}$, and $\Phi$ is a sentence of $L_\mu$. We write $\forall p(O).\varphi$ for $\neg\exists p(O).\neg\varphi$. The interpretation of the formulas in $\mathrm{Q}L_\mu$ is relative to a process $\mathcal{S} = \langle S, s^0, t, L\rangle$ on $(\Sigma, \Gamma)$. It is $[\![\,\varphi\,]\!]_{\mathcal{S}} \subseteq S$, defined inductively as follows: the case where $\varphi \in L_\mu$ is given by Def.2; $[\![\,\exists p(O).\varphi\,]\!]_{\mathcal{S}}$ is the set of states $s \in S$ s.t. there exists a (complete) process $\mathcal{E} = \langle E, \varepsilon^0, t', L'\rangle \in Lab_p^{O\cap\Sigma}$ with $(s, \varepsilon^0) \in [\![\,\varphi\,]\!]_{\mathcal{S}\times\mathcal{E}}$; the remaining cases for $\neg\varphi$ and $\varphi \vee \varphi'$ are obvious.*

The quantification-free fragment of $\mathrm{Q}L_\mu$ is simply $L_\mu$. Clearly, bisimilar [1] processes satisfy the same $\mathrm{Q}L_\mu$ formulas since processes are deterministic.

Controllers under observation $O$ will be represented in their extended form: processes of $Lab_c^O$, with the implicite assumption that $O \subseteq \Sigma$ (the events set of the plant). Informally, a fresh atomic proposition $c$ is chosen which labels original states of the controller, whereas new states, not labeled by $c$, are added in order to make the result a complete process of type $O$. Given a controller $\mathcal{C}$, we will write $\mathcal{E}_\mathcal{C}$ the process obtained by this completion procedure. The relationship between $\mathcal{C}$ and $\mathcal{E}_\mathcal{C}$, is formalized by the notion of *pruning*.

[1] We mean strong bisimulation, which takes events into acount.

**Definition 4 Pruning.** *Given* $\mathcal{S} = \langle S, s^0, t, L\rangle$ *on* $(\Sigma, \Gamma)$ *and* $c \in AP$*, the* $c$-pruning *of* $\mathcal{S}$ *is* $\mathcal{S}_{\to c} = \langle S, s^0, t', L'\rangle$ *on* $(\Sigma, \Gamma \setminus \{c\})$ *where: (1) for all* $s \in S$ *and* $\sigma \in \Sigma$, $t'(s,\sigma) = t(s,\sigma)$ *if* $c \in L(t(s,\sigma))$*, undefined otherwise, and (2)* $L'(s) = L(s) \setminus \{c\}$.

Now, for a controller $\mathcal{C}$, we have $\mathcal{E}_{\mathcal{C}\to c} = \mathcal{C}$. The relationship between the properties of $\mathcal{P} \otimes \mathcal{C}$ and $\mathcal{P} \otimes \mathcal{E}_{\mathcal{C}}$ relies on the notion of *adjustment*.

**Definition 5 Adjustment.** *For all* $\Phi \in L_\mu$ *and* $c \in AP$*, the* $c$-adjustment *of* $\Phi$ *is* $\Phi_{\to c} \in L_\mu$*, inductively defined by:* $(\Phi \vee \Phi')_{\to c} = \Phi_{\to c} \vee \Phi'_{\to c}$ , $\top_{\to c} = \top$, $(\langle\sigma\rangle\Phi)_{\to c} = \langle\sigma\rangle(c \wedge \Phi_{\to c})$, $X_{\to c} = X$, $(\mu X.\Phi)_{\to c} = \mu X.\Phi_{\to c}$, $(\neg\Phi)_{\to c} = \neg\Phi_{\to c}$, *and finally,* $p_{\to c} = p$ *whenever* $p \neq c$*, and* $p_{\to c} = \perp$ *otherwise.*

**Proposition 6** *Given a process* $\mathcal{S}$ *on* $(\Sigma, \Gamma)$, $\mathcal{E} \in Lab_c^O$ *with* $c \notin \Gamma$*, and* $\Phi \in L_\mu$*, we have:* $\mathcal{S} \otimes (\mathcal{E})_{\to c} \models \Phi$ *iff* $\mathcal{S} \otimes \mathcal{E} \models \Phi_{\to c}$.

**PROOF.** $\mathcal{S} \otimes (\mathcal{E})_{\to c}$ and $(\mathcal{S} \otimes \mathcal{E})_{\to c}$ being isomorphic, $[\![\Phi]\!]_{\mathcal{S}\otimes(\mathcal{E})_{\to c}} = [\![\Phi]\!]_{(\mathcal{S}\otimes\mathcal{E})_{\to c}}$; an induction on $\Phi$ gives $[\![\ \Phi\ ]\!]_{(\mathcal{S}\otimes\mathcal{E})_{\to c}} = [\![\ \Phi_{\to c}\ ]\!]_{\mathcal{S}\otimes\mathcal{E}}$ to conclude.

**Proposition 7** *Let* $\mathtt{Adm}(c, \Sigma_{co})$ *be the formula* $\nu Y.[\ ](c \Rightarrow Y) \wedge (\bigwedge_{u\notin\Sigma_{co}}[u]c)$. *Assume given a process* $\mathcal{S}$ *on* $(\Sigma, \Gamma)$ *and two sets* $\Sigma_{co}$ *and* $O$ *s.t.* $\Sigma_{co} \subseteq O \subseteq \Sigma$. *For any* $\mathcal{E} \in Lab_c^O$ *we have:*

$\mathcal{S} \otimes \mathcal{E} \models \mathtt{Adm}(c, \Sigma_{co}) \wedge \Psi_{\to c}$ *iff* $\mathcal{E}_{\to c}$ *is a controller for* $(\mathcal{S}, \Sigma_{co}, O, \Psi)$

**PROOF.** Assume $\mathcal{E}_{\to c}$ is a controller for $(\mathcal{P}, \Sigma_{co}, O, \Psi)$; hence $\mathcal{P} \otimes \mathcal{E}_{\to c} \models \Psi$, which entails $\mathcal{P} \otimes \mathcal{E} \models \Psi_{\to c}$, by Prop.6. It remains to show that $\mathcal{P} \otimes \mathcal{E} \models \mathtt{Adm}(c, \Sigma_{co})$ : assume given a proposition $p \in AP$ s.t. $[\![\ p\ ]\!]_{\mathcal{P}\otimes\mathcal{E}}$ is the set $[\![\bigwedge_{u\notin\Sigma_{co}}[u]c]\!]_{\mathcal{P}\otimes\mathcal{E}}$. Hence, $[\![p]\!]_{\mathcal{P}\otimes\mathcal{E}}$ is the the set of states $(s, e)$ in $\mathcal{P} \otimes \mathcal{E}_{\to c}$ s.t. for any $u \notin \Sigma_{co}$, if a $u$-transition is firable from $s$ then it is not disallowed by $\mathcal{E}_{\to c}$. Since $\mathcal{E}_{\to c}$ is admissible, $\mathcal{P} \otimes \mathcal{E}_{(\to c)} \models \mathbf{AG}(p)$ and then $\mathcal{P} \otimes \mathcal{E}$ satisfies $\mathbf{AG}(p)_{\to c}$ which is $\mathtt{Adm}(c, \Sigma_{co})$. Reciprocally, let $\mathcal{E} \in Lab_c^O$ s.t. $\mathcal{P} \otimes \mathcal{E} \models \mathtt{Adm}(c, \Sigma_{co}) \wedge \Psi_{\to c}$. By assumption, according to Prop.6, $\mathcal{P} \otimes \mathcal{E}_{\to c} \models \Psi$ and because $\mathcal{P} \otimes \mathcal{E}$ satisfies $\mathtt{Adm}(c, \Sigma_{co})$, we show that $\mathcal{E}_{\to c}$ is admissible, using similar arguments as above.

We can now state the following result, as immediate consequence of Prop.7.

**Theorem 8 Basic Control Problem.** *For any* $\Psi \in L_\mu$*, there exists a controller for* $(\mathcal{P}, \Sigma_{co}, O, \Psi)$ *if and only if*

$$\mathcal{P} \models \exists c(O).\mathtt{Adm}(c, \Sigma_{co}) \wedge \Psi_{\to c} \qquad \textbf{(1)}$$

We illustrate now the use of $\mathrm{Q}L_\mu$ to specify various control requirements. The formula of Th.8 is enriched to integrate new control rules ; we focus on permissiveness issue and decentralized control. A controller $\mathcal{C}$ for $(\mathcal{P}, \Sigma_{co}, O, \Psi)$ is maximally permissive if no other controller for $(\mathcal{P}, \Sigma_{co}, O, \Psi)$ can disallow strictly fewer transitions than $\mathcal{C}$. Let us write $c \sqsubset c'$ for $c \sqsubseteq c' \wedge \neg(c' \sqsubseteq c)$ where $c \sqsubseteq c'$ is a notation for the invariant property $([\,]\mathbf{AG}(c'))_{\to c}$. According to [15] generalized to the weak synchronous product:

**Theorem 9 Maximally permissive controllers.** *For any $\Psi \in L_\mu$, there exists a maximally permissive controller for $(\mathcal{P}, \Sigma_{co}, O, \Psi)$ if and only if*

$$\mathcal{P} \models \exists c(O).[\mathtt{Adm}(c, \Sigma_{co}) \wedge \Psi_{\to c}] \wedge \forall c'(O).c \sqsubset c' \Rightarrow \neg[\mathtt{Adm}(c', \Sigma_{co}) \wedge \Psi_{\to c'}] \quad \mathbf{(2)}$$

**PROOF.** By Eq.(2), there exists $\mathcal{E} \in Lab_c^O$ s.t. (Prop.7) $\mathcal{E}_{\to c}$ is a controller for $(\mathcal{P}, \Sigma_{co}, O, \Psi)$ and $\mathcal{P} \otimes \mathcal{E} \otimes \mathcal{E}'$ satisfies $c \sqsubset c' \Rightarrow \neg[\mathtt{Adm}(c', \Sigma_{co}) \wedge \Psi_{\to c'}]$, for any $\mathcal{E}' \in Lab_{c'}^O$. By [15], $\mathcal{E}'_{\to c'}$ cannot be a controller for $(\mathcal{P}, \Sigma_{co}, O, \Psi)$ and strictly more permissive than $\mathcal{E}_{\to c}$. The converse follows the same reasoning backwards for a maximally permissive controller $\mathcal{E}_{\to c}$.

**Theorem 10 Decentralized controllers.** *For any $\Psi \in L_\mu$, there exist two controllers: $\mathcal{C}_1$ of type $O_1$ admissible for $\Sigma_{co}^1 \subseteq O_1$ and $\mathcal{C}_2$ of type $O_2$ admissible for $\Sigma_{co}^2 \subseteq O_2$ s.t. $\mathcal{P} \otimes \mathcal{C}_1 \otimes \mathcal{C}_2 \models \Psi$ if and only if*

$$\mathcal{P} \models \exists c_1(O_1).\mathtt{Adm}(c_1, \Sigma_{co}^1) \wedge \exists c_2(O_2).\mathtt{Adm}(c_2, \Sigma_{co}^2) \wedge (\Psi_{\to c_1})_{\to c_2} \quad \mathbf{(3)}$$

## 2 A Decidable Class of Control Problems

According to [2], there is no hope to decide the Model Checking Problem for the full logic $\mathrm{Q}L_\mu$ since the decentralized control problem is undecidable. Nevertheless, we focus on a decidable fragment of the logic which can be used to require maximally permissive controllers.

**Definition 11** *A formula $\varphi$ of $\mathrm{Q}L_\mu$ is* nested observational (n.o. for short *whenever it is written $Q_1c_1(O_1).Q_2c_2(O_2)\ldots Q_nc_n(O_n).\Phi$ where (1) the $Q_i$'s are quantifiers, (2) $\Phi$ is a sentence of $L_\mu$, and (3) $O_i \subseteq O_{i+1}$ for $1 \leq i < n$.*

W.l.o.g. we only consider n.o. formulas which outermost quantifier is existential. In Th.16, we prove that the Model Checking of such formulas is decidable, with a complexity bound of $nEXPTIME$ in the size of the problem, where $n$ is the number of quantifiers. Automata-theoretic approaches provide the model theory of mu-calculus, and they offer decision algorithms for the satisfiability and the model-checking problems [6,16,9,1]; we consider *alternating parity automata*, or $APA$ in short [6,1].

**Definition 12** *An APA on* $(\Sigma, \Gamma)$ *(with* $\Sigma \subseteq \texttt{Ev}$ *and* $\Gamma \subseteq AP$*) is a tuple* $\mathcal{A} = \langle Q, Q^{\exists}, Q^{\forall}, q^0, \delta, r\rangle$ *where* $Q$ *is a finite set of states partitioned into two subsets* $Q^{\exists}$ *and* $Q^{\forall}$ *of* existential *and* universal *states,* $q^0 \in Q$ *is the initial state,* the transition function $\delta$ *which assigns to each state* $q$ *and each subset of* $\Gamma$ *and a set of pairs in* $Moves = ((\Sigma \cup \{\varepsilon\}) \times Q) \cup (\Sigma \times \{\rightarrow, \not\rightarrow\})$*. Formally,* $\delta : Q \times 2^{\Gamma} \rightarrow 2^{((\Sigma \cup \{\varepsilon\}) \times Q) \cup (\Sigma \times \{\rightarrow, \not\rightarrow\})}$*. Finally,* $r : Q \rightarrow I\!N$ *is the* parity condition.

Parity games provide the semantics for APA . A *parity game* is a graph with an initial vertex $v^0$, with a partition $(V_I, V_{II})$ of the vertices, and with a partial mapping $r$ from the vertices to a given finite set of integers. A *play from some vertex* $v$ proceeds as follows: if $v \in V_I$, then player I chooses a successor vertex $v'$, else player II chooses a successor vertex $v'$, and so on ad infinitum unless one player cannot make any move. The *play is winning for player I* if it is finite and ends in a vertex of $V_{II}$, or if it is infinite and the upper bound of the set of ranks $r(v)$ of vertices $v$ that are encountered infinitely often is even. A *strategy for player I* is a function $f$ assigning a successor vertex to every sequence of vertices $\vec{v}$, ending in a vertex of $V_I$. A *strategy* $f$ *is memoryless* if $f(\vec{v}) = f(\vec{w})$ whenever the sequences $\vec{v}$ and $\vec{w}$ end in the same vertex. A *strategy for player I is winning* if all play following the strategy from the initial vertex are winning for player I. Winning strategies for player II are defined similarly. The fundamental result of parity games is the memoryless determinacy Theorem, established in [6,1] which states that for any parity game, one of the two players has a (memoryless) winning strategy.

**Definition 13** *Given an APA* $\mathcal{A} = \langle Q, Q^{\exists}, Q^{\forall}, q^0, \delta, r\rangle$ *and a process* $\mathcal{S} = \langle S, s^0, t, L\rangle$*, we define* the parity game $G(\mathcal{A}, \mathcal{S})$*; where the vertices of player I are in* $(Q^{\exists} \cup \{\bot\}) \times S$ *and the vertices of player II are in* $(Q^{\forall} \cup \{\top\}) \times S$*; the initial vertex* $v^0$ *is* $(q^0, s^0)$*, the other vertices and transitions are defined inductively as follows. Vertices in* $\{\top\} \times S$ *and vertices in* $\{\bot\} \times S$ *have no successor. For any vertex* $(q, s)$*, there is an edge from* $(q, s)$ *to* $(q', s')$ *if* $s' = s$ *and* $(\epsilon, q') \in \delta(q, L(s))$*, or* $(\sigma, q') \in \delta(q, L(s))$ *and* $t(s, \sigma) = s'$*; and to* $(\top, s)$ *if* $(\sigma, \rightarrow) \in \delta(q, L(s))$ *and* $t(s, \sigma)$ *is defined, or* $(\sigma, \not\rightarrow) \in \delta(q, L(s))$ *and* $t(s, \sigma)$ *is undefined; and to* $(\bot, s)$ *if* $(\sigma, \rightarrow) \in \delta(q, L(s))$ *and* $t(s, \sigma)$ *is undefined, or* $(\sigma, \not\rightarrow) \in \delta(q, L(s))$ *and* $t(s, \sigma)$ *is defined. The* APA $\mathcal{A}$ accepts the process $\mathcal{S}$ *(noted* $\mathcal{S} \models \mathcal{A}$*) if there is a winning strategy for player I in* $G(\mathcal{A}, \mathcal{S})$*.*

For the Model-Checking of n.o. formulas, we consider two automata constructions: the *quotient of automata* and the *projection of automata*, respectively adapted from [2] and [14] for the weak synchronous product case.

**Theorem 14 Quotient of APA .** *Given a process* $\mathcal{S}$ *on* $(\Sigma, \Gamma)$ *and an APA* $\mathcal{A}$ *on* $(\Sigma \cup \Sigma', \Gamma \uplus \Gamma')$*, we can construct an APA* $\mathcal{A}/\mathcal{S}$ *on* $(\Sigma', \Gamma')$ *s.t. for any complete process* $\mathcal{E}$ *on* $(\Sigma', \Gamma')$*,* $\mathcal{S} \otimes \mathcal{E} \models \mathcal{A}$ *iff* $\mathcal{E} \models \mathcal{A}/\mathcal{S}$

**PROOF.** The existential states of $\mathcal{A}/\mathcal{S}$ are $(Q^{\exists} \cup \{\bot\}) \times S$ and its universal states are $(Q^{\forall} \cup \{\top\}) \times S$ ; its initial state is $(q^0, s^0)$, its parity condition $r/$ verifies $r/(q,s) = r(q)$, for all $(q,s) \in Q \times S$, and its transition function $\delta/$ is defined by: for all $\Lambda \subseteq \Gamma'$ and $\sigma \in \Sigma'$, $\delta/((q,s),\Lambda)$ is the least set containing:

- $(\epsilon, (\top, s))$ in cases *(i)* $(\alpha, \rightarrow) \in \delta(q, L(s) \uplus \Lambda)$ and either $\alpha \in \Sigma' \setminus \Sigma$, or $\alpha \in \Sigma$ and $t(s,\alpha)$ is defined; or *(ii)* $(\alpha, \not\rightarrow) \in \delta(q, L(s) \uplus \Lambda)$ and $\alpha \in \Sigma$ and $t(s,\alpha)$ is undefined;
- $(\epsilon, (\bot, s))$ in cases *(i)* $(\alpha, \not\rightarrow) \in \delta(q, L(s) \uplus \Lambda)$ and either $\alpha \in \Sigma' \setminus \Sigma$ or $\alpha \in \Sigma$ and $t(s,\alpha)$ is defined; or *(ii)* $(\alpha, \rightarrow) \in \delta(q, L(s) \uplus \Lambda)$ and $\alpha \in \Sigma$ and $t(s,\alpha)$ is undefined;
- $(\epsilon, (q', s))$ whenever $(\epsilon, q') \in \delta(q, L(s) \uplus \Lambda)$;
- $(\epsilon, (q', s'))$ whenever $(\alpha, q') \in \delta(q, L(s) \uplus \Lambda)$ and $t(s,\alpha) = s'$ and $\alpha \in \Sigma \setminus \Sigma'$;
- $(\sigma, q', s)$ whenever $(\sigma, q') \in \delta(q, L(s) \uplus \Lambda)$ and $\sigma \in \Sigma' \setminus \Sigma$;
- $(\sigma, q', s')$ whenever $(\sigma, q') \in \delta(q, L(s) \uplus \Lambda)$ and $\sigma \in \Sigma$ and $t(s,\sigma) = s'$.

By construction, for any complete process $\mathcal{E}$ on $(\Sigma', \Gamma')$, the games $G(\mathcal{A}, \mathcal{S} \otimes \mathcal{E})$ and $G(\mathcal{A}/\mathcal{S}, \mathcal{E})$ are isomorphic: the isomorphism relates positions $(q,(s,e))$ and $((q,s),e)$, positions $((\top,s),e)$ and $(\top,(s,e))$, and positions $((\bot,s),e)$ and $(\bot,(s,e))$ in $G/$. Hence, the existence of winning strategy holds for both games exactly at the same time

**Theorem 15 Projection of APA .** *Given an APA $\mathcal{A}$ on $(\Sigma', \Gamma')$, sets $\Sigma \subseteq \Sigma'$ and $\Gamma \subseteq \Gamma'$, we can construct an APA $\mathcal{A}{\downarrow_{(\Sigma,\Gamma)}}$ on $(\Sigma, \Gamma)$ s.t. for any complete process $\mathcal{E}$ on $(\Sigma,\Gamma)$, $\mathcal{E} \models \mathcal{A}{\downarrow_{(\Sigma,\Gamma)}}$ if and only if there exists a complete process $\mathcal{E}'$ on $(\Sigma', \Gamma' \setminus \Gamma)$ s.t. $\mathcal{E} \otimes \mathcal{E}' \models \mathcal{A}$.*

**PROOF.** We construct $\mathcal{A}{\downarrow_{(\Sigma,\Gamma)}}$ in two steps: first, according to [14], from $\mathcal{A}$, we can construct an APA $\mathcal{B}$ on $(\Sigma', \Gamma)$ s.t. for any process $\mathcal{S}$ on $(\Sigma', \Gamma)$, $\mathcal{S} \models \mathcal{B}$ if and only if there exists a complete process $\mathcal{E}'$ on $(\Sigma', \Gamma' \setminus \Gamma)$ - hence of type $\Sigma'$ - s.t. $\mathcal{S} \otimes \mathcal{E}' \models \mathcal{A}$. Secondly, we derive from $\mathcal{B}$ another APA $\mathcal{B}'$ on $(\Sigma, \Gamma)$ s.t. for any complete process $\mathcal{E}$ on $(\Sigma, \Gamma)$, $\mathcal{E} \models \mathcal{B}'$ if and only if *loop*$(\mathcal{E}) \models \mathcal{B}$, where *loop*$(\mathcal{E})$ is the complete process on $(\Sigma', \Gamma)$ obtained from $\mathcal{E}$ by adding self-loop $\alpha$-transitions, for each $\alpha \in \Sigma' \setminus \Sigma$, as explained in the next paragraph. Taking $\mathcal{A}{\downarrow_{(\Sigma,\Gamma)}}$ as $\mathcal{B}'$ is adequate: for any complete process $\mathcal{E}$ on $(\Sigma, \Gamma)$, we have $\mathcal{E} \models \mathcal{A}{\downarrow_{(\Sigma,\Gamma)}}$ if and only if *loop*$(\mathcal{E}) \models \mathcal{B}$ which is equivalent to the existence of a complete process $\mathcal{E}'$ on $(\Sigma', \Gamma' \setminus \Gamma)$ s.t. *loop*$(\mathcal{E}) \otimes \mathcal{E}' \models \mathcal{A}$. Now, because *loop*$(\mathcal{E}) \otimes \mathcal{E}'$ and $\mathcal{E} \otimes \mathcal{E}'$ are the same, we conclude.
We now explain the construction of $\mathcal{B}'$: for $\mathcal{B}$ an APA $\langle Q, Q^{\exists}, Q^{\forall}, q^0, \delta, r\rangle$ on $(\Sigma', \Gamma)$, we define $\mathcal{B}'$ on $(\Sigma, \Gamma)$ by $\mathcal{B}' = \langle Q \cup \{\top, \bot\}, Q^{\exists} \cup \{\bot\}, Q^{\forall} \cup \{\top\}, q^0, \delta', r\rangle$ where for any $q \in Q$, and $\Lambda \subseteq \Gamma$ and $\sigma \in \Sigma$, $\delta'(q, \Lambda)$ is the least set containing: $(\epsilon, q')$ if $(\epsilon, q') \in \delta(q, \Lambda)$ or $(\alpha, q') \in \delta(q,\Lambda)$ with $\alpha \in \Sigma' \setminus \Sigma$; and $(\epsilon, \top)$ if $(\alpha, \rightarrow) \in \delta(q, \Lambda)$; and $(\epsilon, \bot)$ if $(\alpha, \not\rightarrow) \in \delta(q, \Lambda)$ with $\alpha \in \Sigma' \setminus \Sigma$; and $(\sigma, q')$ if $(\sigma, q') \in \delta(q, \Lambda)$; and $(\sigma, \rightarrow)$ if $(\sigma, \rightarrow) \in \delta(q, \Lambda)$; and $(\sigma, \not\rightarrow)$ if $(\sigma, \not\rightarrow) \in \delta(q, \Lambda)$.

It can shown that the games $G(\mathcal{B}', \mathcal{S})$ and $G(\mathcal{B}, loop(\mathcal{S}))$ are isomorphic for any $\mathcal{S}$ on $(\Sigma, \Gamma)$, since $\mathcal{S}$ and $loop(\mathcal{S})$ only differ by self-loop transitions labeled over $\Sigma' \setminus \Sigma$.

**Theorem 16 Automata for n.o. formulas.** *Given a finite process $\mathcal{S}$ and a n.o. formula $\varphi$, there exists an APA $\mathcal{A}_{(\varphi,\mathcal{S})}$ s.t. $\mathcal{S} \models \varphi$ if and only if $\mathcal{A}_{(\varphi,\mathcal{S})}$ has a model.*

**PROOF.** Consider $\varphi$ a n.o. formula of the form $Q_1c_1(O_1). \dots Q_nc_n(O_n).\Phi$ and a process $\mathcal{S}$ on $(\Sigma, \Gamma)$. For any APA $\mathcal{A}$, $\neg A$ is the APA obtained by complementation ([12]). First, we construct the APA $\mathcal{A}_\Phi$ on $(\Sigma \cup O_n, \Gamma \uplus \{c_1, \dots c_n\})$ equivalent to the mu-calculus sentence $\Phi$: the construction is standard (see [1]). According to Th.14, we construct $\mathcal{A}_\Phi/\mathcal{S}$ on $(O_n, \{c_1, \dots c_n\})$ s.t. for any complete process $\mathcal{E}$ on $(O_n, \{c_1, \dots c_n\})$, we have: $\mathcal{E} \models \mathcal{A}_\Phi/\mathcal{S}$ iff $\mathcal{S} \otimes \mathcal{E} \models \mathcal{A}_\Phi$. Lastly, we construct define $\mathcal{A}_i$ on $(O_i, \{c_1, \dots c_i\})$ by $\mathcal{A}_\Phi/\mathcal{S}$ if $i = n$, otherwise $\mathcal{A}_i$ is $[\mathcal{A}_{i+1}] \downarrow_{(O_i,\{c_1,\dots,c_i\})}$ if $Q_{i+1} = \exists$, otherwise ($Q_{i+1} = \forall$) it is $\neg[(\neg\mathcal{A}_{i+1})] \downarrow_{(O_i,\{c_1,\dots,c_i\})}$. By construction, for any $\mathcal{E}_1 \in Lab_{c_1}^{O_1}$, $\mathcal{E}_1 \models \mathcal{A}_1$ iff $\mathcal{S} \otimes \mathcal{E}_1 \models Q_2c_2(O_2) \dots Q_nc_n(O_n).\Phi$. Take $\mathcal{A}_{(\varphi,\mathcal{S})} = \mathcal{A}_1$.

Th.16 can be exploited for control synthesis: given a finite plant $\mathcal{P}$ and $\varphi$ of the form $\exists c(O).\varphi'$, the APA $\mathcal{A}_{(\varphi,\mathcal{P})}$ specifies a possibly empty family of controllers which answer $(\mathcal{P}, \Sigma_{co}, O, \Psi)$; assume the family is non-empty: according to a classic result, still based on the memoryless determinacy Theorem, we can compute some regular (i.e. finite) process $\mathcal{E} \in Lab_c^O$ model of $\mathcal{A}_{(\varphi',\mathcal{P})}$, hence a controller $\mathcal{E}_{\to c}$. In turn, if $\varphi'$ states the existence of other controllers (as in the decentralized case), they can be synthesized similarly by considering the models of $\mathcal{A}_{(\varphi',\mathcal{P}\otimes\mathcal{E})}$. By [14], the size of $\mathcal{A}_{(\varphi,\mathcal{P})}$ is bounded by $(n-1)EXP(|\mathcal{P}| \times |\varphi|)$, where $n$ is the number of quantifiers in $\varphi$ ($|\mathcal{P}|$ the cardinal of $P$ and $|\varphi|$ the number of sub-formulas in $\varphi$). Now, its non-emptiness can be checked in $nEXPTIME(|\mathcal{P}| \times |\varphi|)$ and a model can be synthesized with the same complexity (see [6], for example).

**Conclusion** The present work proposes a logical framework for the specification of control problems under partial observation and the controller synthesis of n.o. formulas is proved decidable with a model synthesis procedure. The Model-Checking problem of the complementary set of the n.o. formulas is undecidable, since both the Post Correspondence Problem (like in [2]) and the Tiling Problem [3] can be encoded. The fragment of n.o. formulas is expressive enough to deal with maximally permissive controllers under partial observation for any mu-calculus definable control objective. Up to our knowledge, this is the first time that existence of maximally permissive controllers under partial observation is proved, e.g. [8,2] could not consider this aspect, hence this is a strong argument in favor of our approach.

## References


[1] A. Arnold and D. Niwinski. *Rudiments of mu-calculus.* North-Holland, 2001.

[2] A. Arnold, A. Vincent, and I. Walukiewicz. Games for synthesis of controllers with partial observation. *TCS*, 1:7–34, 2003.

[3] R. Berger. The undecidability of the domino problem. *Memoirs American Mathematical Society*, 66, 1966.

[4] C. G. Cassandras and S. Lafortune. *Introduction to Discrete Event Systems.* Kluwer Academic, 1999.

[5] E. A. Emerson and J. Y. Halpern. On branching versus linear time temporal logic. *Journal of the ACM*, 33(1):151–178, 1986.

[6] E.A. Emerson and C.S. Jutla. Tree automata, mu-calculus and determinacy. In *Proc. IEEE FOCS'91*, pages 368–377. IEEE Computer Society Press, 1991.

[7] D. Kozen. Results on the propositional $\mu$-calculus. *TCS*, 27(3):333–354, 1983.

[8] O. Kupferman, P. Madhusudan, P.S. Thiagarajan, and M.Y. Vardi. Open systems in reactive environments: Control and synthesis. *Proc. CONCUR'00, LNCS 1877*, pages 92–107, 2000.

[9] O. Kupferman, M.Y. Vardi, and P. Wolper. An automata-theoretic approach to branching-time model checking. *Journal of the ACM*, 47(2):312–360, 2000.

[10] Y. Li, F. Lin, and Z.H. Lin. A generalized framework for supervisory control of discrete event systems. *International Journal of Intelligent Control and Systems*, 2(1):139–159, 1998.

[11] F. Lin and W.M. Wonham. On observability of discrete-event systems. *Information Sciences*, 44(3):173–198, 1988.

[12] D.E Muller and P.E. Schupp. Simulating alternating tree automata by nondeterministic automata: new results and new proofs of the theorems of rabin, mcnaughton and safra. *TCS*, 141:69–107, 1995.

[13] P.J. Ramadge and W.M. Wonham. The control of discrete event systems. *Proceedings of the IEEE, Special issue on Dynamics of Discrete Event Systems*, 77(1):81–98, 1989.

[14] S. Riedweg and P. Pinchinat. Quantified mu-calculus for control synthesis. In *MFCS'03*, volume 2747 of *LNCS*, pages 642–651, 2003.

[15] S. Riedweg and S. Pinchinat. Maximally permissive controllers in all contexts. In *WODES'04*, pages 283–288, Reims, France, 2004.

[16] R.S. Streett and E.A. Emerson. An automata theoretic decision procedure for the propositional *mu*-calculus. *Information and computation*, 81:249–264, 1989.

[17] J.G. Thistle and W.M. Wonham. Supervision of infinite behavior of discrete-event systems. *SIAM Journal on Control and Optimization*, 32:1098–1113, 1994.