

Asynchronous Omega-Regular Games with Partial Information^{1 2}

Bernd Puchala

Mathematische Grundlagen der Informatik, RWTH Aachen University
puchala@logic.rwth-aachen.de

Abstract. We address the strategy problem for ω -regular two-player games with partial information, played on finite game graphs. We consider two different kinds of observability on a general model, a standard synchronous and an asynchronous one. In the asynchronous setting, moves which have no visible effect for a player are hidden completely from that player.

We generalize the usual powerset construction for eliminating partial information to arbitrary, not necessarily observation based, winning conditions, both in the synchronous and in the asynchronous case, and we show that this generalized construction effectively preserves ω -regular winning conditions. From this we infer decidability of the strategy problem for arbitrary ω -regular winning conditions, in both cases. We also show that our ω -regular framework is sufficient for reasoning about synchronous and asynchronous knowledge by proving that any formula of the epistemic temporal specification formalism ETL can be effectively translated into an S1S-formula defining the same specification.

1 Introduction

In a two-player graph game the players, called 0 and 1, move a token along the edges of a labeled nonterminating graph by choosing appropriate edge labels, also called actions. This results in an infinite sequence of positions and actions called play, and each such play is either won by player 0 or by player 1 and lost by the other player. We consider games played on finite graphs where the set of plays won by player 0 is ω -regular, i.e., recognizable by a nondeterministic Büchi automaton. These specifications play a key role in modern computer science. They generalize parity objectives and capture fundamental properties of nonterminating reactive systems, cf. [14].

Such a system can be modeled as a two-player game where changes of the system state correspond to changes of the game position. Situations where the change of the system can be controlled correspond to positions of player 0, uncontrollable situations correspond to positions of player 1. A *winning strategy*

¹ Full version of [11]. Notice that the proof of Theorem 12 in the published version contains a flaw. In this version, a revised proof can be found.

² Completion of this work was supported by the ESF EUROCORES project LogICCC.

for player 0 then yields a controller that forces the system into satisfying an ω -regular specification.

The problem to determine, for a given game G and a position v , whether player 0 has a winning strategy for G from v is called the strategy problem. An important special case of ω -regular games are parity games. Parity games with full information have received much attention during the past years, cf. [7]. The key property of parity games is memoryless determinacy which proves that the strategy problem is in $\text{NP} \cap \text{co-NP}$. In general, ω -regular games with full information are determined with finite memory and winning strategies can be synthesized effectively.

However, assuming that both players have full information about the history of events in an ω -regular game is not always realistic. For example, if the information about the system state is acquired by imprecise sensors or the system encapsulates private states which cannot be read from outside, then a controller for this system must rely on the information about the state and the change of the system to which it has access. I.e., in the game model, player 0 has uncertainties about the positions and actions in the game, so we have to add partial information to games in order to model this kind of problems.

Solving the strategy problem for such games is much harder than solving ω -regular games with full information since we have to keep track of the knowledge of player 0 during the course of events. Such a knowledge tracking is inherently unavoidable and leads to an exponential lower bound for the time complexity of the strategy problem for reachability games with partial information [12] and a super-polynomial lower bound for the memory needed to implement winning strategies in reachability games [1,10]. To keep track of the knowledge of player 0 during a play we compute, for any finite history, the set of positions that player 0 considers possible in this situation. This method, called *powerset construction*, has originally been suggested by John H. Reif in [12] to solve the strategy problem for reachability games with partial information. We note that this is not the only possibility for tracking knowledge. Tree-automata techniques have been used to solve synthesis problems with partial information from linear [17] and branching [8] temporal specifications in input-output frameworks. A tree-automaton processes the tree-representation of a strategy, so the information on which the strategy can rely is always given implicitly by the history.

We consider two different kinds of observability on a general model, a synchronous and an asynchronous one. In the standard synchronous case it is assumed that both players know how many moves have been performed. Intuitively, the players share a common clock. If a player moves from position u to v and the opponent cannot distinguish u from v , then this move has no visible effect for the opponent. However, the common clock tells him that a move has been performed. Now, if the system under consideration is not adapted to synchronization, that means, it cannot be reasonably timed by a global signal, then this assumption is not adequate. The asynchronous case, where we do not require the existence of such a common clock but hide a move completely from a player if it has no visible effect for him, is suited for such settings.

Several other cases of asynchronous synthesis have already been studied. For example in [9], asynchronism is imposed by requiring a program to be robust against all possible schedules and observability is given by the points at which the system is scheduled to read. In [17], an action-based model with blind components is considered, where all that a player knows is the past history of his own actions. Asynchronism is given by the possibility of the components to remain idle, at any point of a run for an arbitrary number of steps. In our setting, which comprehends the model of [17], uncertainties about positions and actions can be arbitrary and asynchronism is defined via observability. Moreover, the behavior of both players is a priori constrained by an arbitrary finite graph. The same notion of asynchronism has also been used in [2] where stutter closed winning conditions (cf. Section 2) have been considered. The main novelty there is a powerset construction for timed games and an efficient symbolic implementation.

Here, we develop a unified solution for arbitrary winning conditions on a general model and we apply this to the powerful specification formalism of ω -regular languages. We also show that the expressiveness of ω -regular languages captures all ETL-specifications.

In Section 3 we adapt the powerset construction to our model and we generalize it to arbitrary, not necessarily observation based, winning conditions. We prove that ω -regular winning conditions are effectively preserved by this generalized construction. From this we infer that the synchronous strategy problem for ω -regular specifications is decidable on our model and finite memory strategies can be synthesized which has, in the synchronous case, also been obtained for various related models in [17,8,16] using tree automata. It has to be strongly emphasized that ω -regular games with partial information are *not determined*, even for reachability conditions. So this result does not imply that from each position one of the players actually has a winning strategy. It merely says that *if* one of the players has a winning strategy, then he has a finite memory winning strategy which can be constructed effectively. In Section 3.1 we modify the construction in order to obtain the same results for the asynchronous case.

In Section 4 we consider the epistemic temporal specification formalism ETL which is a valuable tool in the presence of partial information, since it allows to refer explicitly to the knowledge of the players when specifying the intended behavior of a system. We investigate a version of ETL which contains knowledge operators for both synchronous and asynchronous knowledge and we give some examples for the usefulness of ETL. Our main result is that any ETL-formula can be effectively translated into an SIS-formula, so our ω -regular framework is sufficient for reasoning about synchronous and asynchronous knowledge. In particular, this adapts the decidability result of [15] to the asynchronous case. In Section 4.1 we conclude with some remarks on multiplayer games with partial information. In particular, we discuss an underlying assumption on the evaluation of knowledge operators which has interesting consequences on the synthesis of distributed systems from ETL-specifications, cf. [16].

2 Preliminaries

Games and Strategies. A *deterministic turn based two-player win-loss game* has the form $G = (V, V_0, (f_a)_{a \in A}, W_0)$, where V is the set of positions, A is the set of actions and for $a \in A$, $f_a : \text{dom}(f_a) \subseteq V \rightarrow V$ is a function. Furthermore, $V_0 \subseteq V$ are the positions of player 0 and $W_0 \subseteq V(AV)^\omega$ is the winning condition of player 0. Let $V_1 = V \setminus V_0$ and $W_1 = V(AV)^\omega \setminus W_0$. For each position $v \in V$ the set $\text{act}(v) = \{a \in A \mid v \in \text{dom}(f_a)\}$ of actions available at v has to be nonempty. For $i = 0, 1$ let $A_i = \bigcup \{\text{act}(v) \mid v \in V_i\}$. The directed labeled graph $(V, (E_a)_{a \in A})$ with $E_a = \{(u, v) \in V \times V \mid u \in \text{dom}(f_a) \text{ and } f_a(u) = v\}$ is called the *game graph* of G . We consider games played on *finite* graphs, i.e., V and A are finite.

A *play* in G from v_0 is an infinite sequence $\pi = v_0 a_1 v_1 \dots \in V(AV)^\omega$ such that $v_i \in \text{dom}(f_{a_{i+1}})$ and $f_{a_{i+1}}(v_i) = v_{i+1}$ for each $i < \omega$. A prefix $\rho \preceq \pi$ of a play is called a *history* and \mathcal{H}_{fin} denotes the set of all finite histories $\pi \in V(AV)^*$ of plays in G . If $\pi \in V(AV)^\alpha$, we denote $\alpha \leq \omega$ by $l(\pi)$ and $\pi(\leq j)$ denotes the unique finite prefix of π with $\pi(\leq j) \in V(AV)^j$. Moreover, if $l(\pi) < \omega$, by $\text{last}(\pi)$ we denote the last position in π . A *strategy* for player i for G is a function $g : \{\pi \in \mathcal{H}_{\text{fin}} \mid \text{last}(\pi) \in V_i\} \rightarrow A$ such that $g(\pi) \in \text{act}(\text{last}(\pi))$ for all $\pi \in \text{dom}(g)$. A history $\pi = v_0 a_1 v_1 \dots$ is *compatible* with g if for all $j < l(\pi)$ such that $v_j \in V_i$ we have $a_{j+1} = g(\pi(\leq j))$. A strategy g for player i is *winning* from v_0 if each play π in G from v_0 that is compatible with g is won by player i , that means, $\pi \in W_i$. A *memory structure* for G has the form $M = (S, \delta_0, \delta)$ where S is a set of states, $\delta_0 : V \rightarrow S$ is the initializing function and $\delta : S \times (A \times V) \rightarrow S$ is the update function. A *strategy for player i with memory M* is a function $g : S \times V_i \rightarrow A$ such that $g(s, v) \in \text{act}(v)$ for all $(s, v) \in S \times V_i$. A history $\pi = v_0 a_1 v_1 \dots$ is called *compatible* with g if for all $j < l(\pi)$ such that $v_j \in V_i$ we have $a_j = g(\delta^*(\pi(\leq j)), v_j)$, where the function $\delta^* : \mathcal{H}_{\text{fin}} \rightarrow S$ is inductively defined by $\delta^*(v) = \delta_0(v)$ for $v \in V$ and $\delta^*(\pi a_j v_j) = \delta(\delta^*(\pi), (a_j, v_j))$ for $\pi \in \mathcal{H}_{\text{fin}}$, $a_j \in \text{act}(\text{last}(\pi))$ and $f_{a_j}(\text{last}(\pi)) = v_j$.

Partial Information. The *knowledge* of player i is given by an equivalence relation \sim_i on \mathcal{H}_{fin} . Equivalent histories are indistinguishable for player i , i.e., if $\pi \sim_i \rho$ then after π has been played and after ρ has been played, player i has exactly the same information, so he cannot distinguish one situation from the other. We refer to [4] for an extensive treatment of knowledge in multi-agent systems. A strategy $g : \{\pi \in \mathcal{H}_{\text{fin}} \mid \text{last}(\pi) \in V_i\} \rightarrow A$ for player i for G is called a *partial information strategy* with respect to \sim_i (\sim_i -strategy, for short) if $g(\pi) = g(\pi')$ for all $\pi, \pi' \in \mathcal{H}_{\text{fin}}$ with $\pi \sim_i \pi'$. Notice that a \sim_i -strategy g for player i is winning from all positions in a set $U \subseteq V$ if and only if it is winning from a simulated initial position v_0 which belongs to player $1-i$ and from which he can secretly choose any position $v \in U$. Moreover, any \sim_i -strategy g which is only defined on histories from some initial position v_0 can be extended to a \sim_i -strategy g' with $\text{dom}(g) = \{\pi \in \mathcal{H}_{\text{fin}} \mid \text{last}(\pi) \in V_i\}$ by giving g' appropriate value on histories from some initial position $v'_0 \neq v_0$. So in our *antagonistic* two-player setting, it suffices to consider strategies which are winning from single initial positions v_0 and only defined on histories from v_0 . If we are given a game

G , a position v_0 in G and some equivalence relation \sim_i on \mathcal{H}_{fin} , the question whether player i has a *winning \sim_i -strategy* for G from v_0 is independent of the partial information of player $1 - i$. Therefore, when solving the strategy problem we consider games with partial information only for player 0. However, the logic ETL explicitly refers to the knowledge of player 1 via the knowledge operator K_1 so we introduce the general model with partial information for both players.

We consider games played on finite graphs where the players have uncertainties about the positions and actions in the game, modeled by equivalence relations. The relation \sim_i is then obtained by extending these equivalence relations to an equivalence relation on \mathcal{H}_{fin} . We consider two different types of extensions, resulting in the synchronous and the asynchronous case respectively. A *two-player game with partial information* has the form $\mathcal{G} = (G, (\sim_i^V)_{i=0,1}, (\sim_i^A)_{i=0,1})$ where $G = (V, V_0, (f_a)_{a \in A}, W_0)$ is a two-player game and for $i = 0, 1$, $\sim_i^V \subseteq V \times V$ and $\sim_i^A \subseteq A \times A$ are equivalence relations such that the following conditions hold.

- (1) If $u, v \in V$ with $u \sim_i^V v$ then $u, v \in V_i$ or $u, v \notin V_i$.
- (2) If $a, b \in A_i$ with $a \neq b$ then $a \not\sim_i^A b$.
- (3) If $u, v \in V_i$ with $u \sim_i^V v$, then $\text{act}(u) = \text{act}(v)$.

Condition (1) says that player i always knows when it is his turn and condition (2) says that player i can distinguish all the actions that are available to him at some position of the game. Condition (3) ensures that player i always knows which actions are available to him when it is his turn.

Now we consider two ways to extend the equivalence relations on positions and actions to equivalence relations on finite histories. For positions $u, v \in V$ we say that $u \rightarrow v$ is a *private move of player i* , if $u \in V_i$ and $u \sim_{1-i}^V v$. For $\pi = v_0 a_1 v_1 \dots a_n v_n$, $\rho = w_0 b_1 w_1 \dots b_m w_m \in V(AV)^*$ let

$$\begin{aligned} - \pi \sim_i^* \rho & : \iff n = m \text{ and } v_j \sim_i^V w_j \text{ and } a_j \sim_i^A b_j \text{ for all } j \\ - \pi \prec_i^* \rho & : \iff \overleftarrow{\pi} \sim_i^* \overleftarrow{\rho} \end{aligned}$$

where $\overleftarrow{\pi}$ is obtained from π by contracting each maximal sequence $v_r a_{r+1} v_{r+1} \dots a_s v_s$ of private moves of player $1 - i$ in π to v_r and analogously for $\overleftarrow{\rho}$.

Games where indistinguishability of histories is given by \sim_i^* model synchronous systems. Intuitively, the players share a common clock which tells them how many moves have been performed, even if some of those moves did not have any effect which they could observe. On the other hand, \prec_i^* defines an asynchronous case where we hide such private moves of the opponent completely from a player. If there is no common clock, this is intrinsic to the system.

Remark. Consider the interaction between components of a system where the behavior of each component is prescribed by a controller which has to rely on the information available to this component. In such settings it might seem more appropriate to ask for a \sim_0^* -strategy for player 0 which is winning against all \sim_1^* -strategies of player 1 rather than a winning \sim_0^* -strategy for player 0. However, it is easy to see that in our perfect recall setting, this is equivalent. Therefore, we use the reduced form $\mathcal{G} = (G, \sim^V, \sim^A)$ of our model with partial information only for player 0. The relations \sim_0^* and \prec_0^* are accordingly denoted \sim^* and \prec^* .

The winning region $\text{Win}_0^{\mathcal{G}}$ of player 0 in \mathcal{G} is the set of all positions $v \in V$ such that player 0 has a winning \sim^* -strategy for \mathcal{G} from v . The winning region $\text{Win}_0^{\mathcal{G},h}$ of player 0 in \mathcal{G} if private moves are hidden is the set of all positions $v \in V$ such that player 0 has a winning \prec^* -strategy for \mathcal{G} from v . Notice that we are really considering two different notions of strategies. Of course, $\pi \sim^* \pi'$ implies $\pi \prec^* \pi'$ for all finite histories π, π' . Therefore, a \prec^* -strategy is a \sim^* -strategy. In particular, $\text{Win}_0^{\mathcal{G},h} \subseteq \text{Win}_0^{\mathcal{G}}$. However, as one can easily see the converse inclusion does not hold in general, even for reachability games.

A \sim^* -memory structure for \mathcal{G} is a memory structure $M = (S, \delta_0, \delta)$ for G such that $\delta(s, (a, v)) = \delta(s, (b, w))$ for all $s \in S$ and all $(a, v), (b, w) \in A \times V$ with $v \sim^V w$ and $a \sim^A b$. M is called a \prec^* -memory structure for \mathcal{G} if additionally, for all $\pi, \pi av \in \mathcal{H}_{\text{fin}}$ such that $\text{last}(\pi) \rightarrow v$ is a private move of player 1, we have $\delta^*(\pi av) = \delta^*(\pi)$. A strategy for player 0 for \mathcal{G} with memory M is a strategy $g : S \times V_0 \rightarrow A$ for player 0 with memory M such that for all $s \in S$ and all $u, v \in V$ with $u \sim^V v$ we have $g(s, u) = g(s, v)$.

Notation. For a position $v \in V$ let $[v] := \{w \in V \mid w \sim^V v\}$ be the equivalence class of v and for $a \in A$ let $[a] := \{b \in A \mid b \sim^A a\}$ be the equivalence class of a . Moreover, for a set $U \subseteq V$ of positions and a set $B \subseteq A$ of actions we define $\text{Post}_B(U) = \{v \in V \mid \exists u \in U, \exists b \in B : b \in \text{act}(u) \wedge f_b(u) = v\}$.

Winning Conditions. The winning condition W_0 is called observation based if there is a coloring $\text{col} : V \rightarrow C$ for a set $C \subseteq \mathbb{N}$ of colors with $\text{col}(u) = \text{col}(v)$ for all $u \sim^V v$ and a set $W \subseteq C^\omega$ such that $\pi = v_0 a_1 v_1 \dots \in W_0$ if and only if $\text{col}(v_0) \text{col}(v_1) \dots \in W$. W_0 is called stutter closed if, additionally, W is invariant under extending and contracting finite sequences of identical colors, i.e., stutter closed winning conditions are asynchronously observation based. In this case, the private moves of player 1 do not really matter. He just makes those moves to reach some position, but what happens during the moves does not affect the winner of the play. So we can let player 0 observe the private moves of player 1 and in return give player 1 the possibility to skip these moves: For any $v \in V_1$ we insert new edges with appropriate labels to all positions, which are reachable from v by a sequence of private moves, followed by one non-private move. W_0 remains unchanged. This yields a polynomial time reduction of the asynchronous strategy problem for stutter closed specifications to the synchronous strategy problem for observation based specifications. However, assuming that the specification is completely based on the observations of player 0 is often not realistic. So we do not make such assumptions but allow arbitrary winning conditions.

Nondeterministic Games. The powerset construction which we present in Section 3 yields a nondeterministic game in general. A nondeterministic two-player game has the form $G = (V, V_0, (E_a)_{a \in A}, W_0)$ with edge relations $E_a \subseteq V \times V$ for $a \in A$ and all the other components are as before. Plays, strategies, memory strategies and winning strategies are defined as before. Nondeterministic games are not determined in general, even for reachability conditions, and hence not equivalent to deterministic games. However, for each nondeterministic game G and each player $i \in \{0, 1\}$, we can construct a deterministic game

G^i such that winning strategies for player i are preserved. We simply resolve the nondeterminism by giving player $1 - i$ control of nondeterministic choices. Technically, for any $v \in V$ and any $a \in \text{act}(v)$ we add a unique a -successor of v to the game graph which belongs to player $1 - i$ and from which he can choose any a -successor of v in the original game graph. The winner of a play in G^i is the winner of the corresponding play in G where we delete all positions and actions which do not belong to G . It is easy to see that this construction preserves ω -regular winning conditions and all special cases like parity conditions.

3 Powerset Construction

First, we consider the standard synchronous strategy problem. We present a powerset construction which turns a game with partial information into a game with full information such that the existence of winning strategies for player 0 is preserved. The idea of this construction has originally been suggested by John H. Reif in [12] where he considered reachability objectives on a somewhat restricted game model with finitely branching game graphs. The construction can also easily be applied to observation based parity conditions on finitely branching graphs and for this case, improved methods for solving the resulting games with full information have been developed, cf. [3,1]. We apply the construction to our model and we generalize the definition of the winning condition of the resulting game, so that it works for arbitrary, not necessarily observation based, winning conditions. Notice that, although we consider only finite game graphs here, this generalized construction can be applied to arbitrary graphs.

The construction yields a game in which both players always know the recent position, so if winning strategies for player 0 shall be preserved, any position of this game must precisely capture the uncertainties about the recent position that player 0 actually has after some finite history $\pi \in \mathcal{H}_{\text{fin}}$. So the positions of the new game are of the form $\bar{v}(\pi) = \{\text{last}(\pi') \mid \pi' \sim^* \pi\}$ for $\pi \in \mathcal{H}_{\text{fin}}$. Each such set is a subset of some equivalence class $[v]$ of positions, so as the set of positions of the new game we simply take the set of all such subsets. Moreover, in the new game graph, there is an edge with label a from $\bar{v}(\pi)$ to $\bar{v}(\rho)$ if and only if $\rho = \pi aw$ for some $w \in V$. The following proposition provides an update mechanism which computes, for finite histories π and πaw , the set $\bar{v}(\pi aw)$ from the set $\bar{v}(\pi)$, the action a and the position w , without knowing the history π . In particular, the set of a -successors of $\bar{v}(\pi)$ in the new game graph depends only on the set $\bar{v}(\pi)$ and the action a , but not on the history π .

Proposition 1. $\bar{v}(\pi aw) = \text{Post}_{[a]}(\bar{v}(\pi)) \cap [w]$.

Let $\mathcal{G} = (G, \sim^V, \sim^A)$, $G = (V, V_0, (f_a)_{a \in A}, W_0)$ be a game with partial information. We define the corresponding game $\bar{G} = (\bar{V}, \bar{V}_0, (\bar{E}_a)_{a \in A}, \bar{W}_0)$ with full information as follows. First, $\bar{V} = \{\bar{v} \in 2^V \mid \exists v \in V : \bar{v} \subseteq [v]\}$ and $\bar{V}_0 = \bar{V} \cap 2^{V_0}$. Moreover, for $a \in A$ we have $(\bar{v}, \bar{w}) \in \bar{E}_a$ if and only if there is some $w \in \text{Post}_a(\bar{v})$ such that $\bar{w} = \text{Post}_{[a]}(\bar{v}) \cap [w]$. Finally,

$\bar{\pi} = \bar{v}_0 a_1 \bar{v}_1 a_2 \bar{v}_2 \dots \in \bar{W}_0 \quad :\Leftrightarrow$
 for each play $\pi = v_0 a'_1 v_1 a'_2 v_2 \dots$ in \mathcal{G} , the following holds:
 if $v_i \in \bar{v}_i$ and $a'_{i+1} \sim^A a_{i+1}$ for all $i < \omega$
 then $\pi \in W_0$.

In the following proposition which is proved immediately from the definitions, the key property for the correctness of the powerset construction is formulated.

Proposition 2. *For each finite history $\bar{\pi} = \bar{v}_0 a_1 \bar{v}_1 \dots a_n \bar{v}_n$ in \bar{G} and all $v_n \in \bar{v}_n$, there is a finite history $\pi = v_0 a'_1 v_1 \dots a'_n v_n$ in \mathcal{G} such that $v_i \in \bar{v}_i$ and $a'_i \sim^A a_i$ for all i . Moreover, for each such history, $\bar{v}(\pi(\leq i)) = \bar{v}_i$ for all i .*

Proposition 3. *Player 0 has a winning \sim^* -strategy f for \mathcal{G} from v_0 if and only if he has a winning strategy f for \bar{G} from $\bar{v}_0 = \{v_0\}$.*

Proof. If $f : \{\pi \in \mathcal{H}_{\text{fin}} \mid \text{last}(\pi) \in V_0\} \rightarrow A$ is a winning \sim^* -strategy, we define $\bar{f} : \{\bar{\pi} \in \bar{\mathcal{H}}_{\text{fin}} \mid \text{last}(\bar{\pi}) \in \bar{V}_0\} \rightarrow A$ as follows. For each finite history $\bar{\pi} = \bar{v}_0 a_1 \bar{v}_1 \dots a_n \bar{v}_n$ with $\bar{v}_n \in \bar{V}_0$ there is a finite history $\pi = v_0 a'_1 v_1 \dots a'_n v_n$ such that $v_i \in \bar{v}_i$ and $a'_i \sim^A a_i$ for all i . Since $v_n \in \bar{v}_n \in \bar{V}_0$ we have $v_n \in V_0$ and we define $\bar{f}(\bar{\pi}) = f(\pi)$. Since f is a \sim^* -strategy, this definition is independent of the chosen history π .

Now assume that there is a play $\bar{\pi} = \bar{v}_0 a_1 \bar{v}_1 a_2 \bar{v}_2 \dots$ in \bar{G} which is compatible with \bar{f} but not won by player 0. Then, by definition of \bar{W}_0 , there is a play $\pi = v_0 a'_1 v_1 a'_2 v_2 \dots$ in \mathcal{G} with $a'_i \sim^A a_i$ and $v_i \in \bar{v}_i$ for all $i < \omega$ such that $\pi \notin W_0$. This play is compatible with f , since if $i < \omega$ such that $v_i \in V_0$, then $\bar{v}_i \in \bar{V}_0$ and so $a'_{i+1} = a_{i+1} = f(\bar{\pi}(\leq i)) = f(\pi(\leq i))$ by definition of f . But f is a winning strategy for player 0 which is a contradiction.

If, conversely, $\bar{f} : \{\bar{\pi} \in \bar{\mathcal{H}}_{\text{fin}} \mid \text{last}(\bar{\pi}) \in \bar{V}_0\} \rightarrow A$ is a winning strategy we first fix a subgraph \bar{G}' of \bar{G} such that for each edge $(\bar{u}, \bar{v}) \in \bar{E}_a$ there is no $b \sim^A a$ with $(\bar{u}, \bar{v}) \in \bar{E}_b$ and we define $f : \{\pi \in \mathcal{H}_{\text{fin}} \mid \text{last}(\pi) \in V_0\} \rightarrow A$ as follows. For a finite history $\pi = v_0 a_1 v_1 \dots a_n v_n$ with $v_n \in V_0$ let $\bar{v}_i := \bar{v}(\pi(\leq i))$ for $i \leq n$. Then there are uniquely determined actions $a'_1, \dots, a'_n \in A$ with $a'_i \sim^A a_i$ for all i such that $\bar{\pi} := \bar{v}_0 a'_1 \bar{v}_1 \dots a'_n \bar{v}_n$ is a history of a play in \bar{G}' and we define $f(\pi) := \bar{f}(\bar{\pi})$. To see that f is a \sim^* -strategy for player 0 for \mathcal{G} consider finite histories $\pi = v_0 a_1 v_1 \dots a_n v_n$ and $\rho = w_0 b_1 w_1 \dots b_n w_n$ with $\pi \sim^* \rho$ and $v_n, w_n \in V_0$. Then $\bar{v}_i = \bar{w}_i$ for all i and since $a_i \sim^A b_i$ for all i we also have $a'_i = b'_i$ for all i . Therefore, $f(\pi) = \bar{f}(\bar{v}_0 a'_1 \bar{v}_1 \dots a'_n \bar{v}_n) = \bar{f}(\bar{w}_0 b'_1 \bar{w}_1 \dots b'_n \bar{w}_n) = f(\rho)$.

Now assume there is a play $\pi = v_0 a_1 v_1 a_2 v_2 \dots \notin W_0$ which is compatible with f and let a'_i for $i < \omega$ be the uniquely determined actions with $a'_i \sim^A a_i$ such that $\bar{\pi} = \bar{v}_0 a'_1 \bar{v}_1 a'_2 \bar{v}_2 \dots$ is a play in \bar{G}' . Then, if $i < \omega$ with $\bar{v}_i \in \bar{V}_0$, we have $v_i \in V_0$ and since π is compatible with f , the definition of f yields $a_{i+1} = f(\pi(\leq i)) = \bar{f}(\bar{\pi}(\leq i))$. Moreover, $a'_{i+1} \in \text{act}(\bar{v}_i) \subseteq A_0$ and since $a_{i+1} \sim^A a'_{i+1}$ we have $a'_{i+1} = a_{i+1}$. So $\bar{\pi}$ is compatible with \bar{f} , so $\bar{\pi} \in \bar{W}_0$. But since $\pi \notin W_0$ with $a_i \sim^A a'_i$ and $v_i \in \bar{v}_i$ for all $i < \omega$ we have $\bar{\pi} \notin \bar{W}_0$, a contradiction. \square

Omega-Regular Winning Conditions. Now we prove that ω -regular winning conditions are effectively preserved by this powerset construction, i.e., if the win-

ning condition of the given game \mathcal{G} with partial information is recognized by a Büchi-automaton, then the winning condition of the corresponding game with full information is again recognized by a Büchi-automaton which can be constructed effectively. From this we infer decidability of the synchronous strategy problem for arbitrary ω -regular winning conditions as formulated in Theorem 5.

We use the standard concept of ω -automata $\mathcal{A} = (\Sigma, Q, q_0, \Delta, \text{acc})$, i.e., Σ is an alphabet, Q is the finite set of states, $q_0 \in Q$ is the initial state, $\Delta \subseteq Q \times \Sigma \times Q$ is the transition relation and $\text{acc} \subseteq Q^\omega$. A run of \mathcal{A} on an ω -word $\alpha \in \Sigma^\omega$ is a function $\rho \in Q^\omega$ such that $\rho(0) = q_0$ and $(\rho(i), \alpha(i), \rho(i+1)) \in \Delta$ for all $i < \omega$. A run ρ is accepting if $\rho \in \text{acc}$ and \mathcal{A} accepts α if there is an accepting run of \mathcal{A} on α . We denote $L(\mathcal{A}) = \{\alpha \in \Sigma^\omega \mid \mathcal{A} \text{ accepts } \alpha\}$. \mathcal{A} is called deterministic, if Δ is a function $\delta : Q \times \Sigma \rightarrow Q$. \mathcal{A} is called Büchi-automaton if $\text{acc} = \{\rho \mid \rho(i) \in F \text{ for infinitely many } i\}$ for some $F \subseteq Q$. An ω -language $L \subseteq \Sigma^\omega$ is called regular, if there is a Büchi automaton \mathcal{A} over Σ such that $L = L(\mathcal{A})$. A finite game is called ω -regular, if the winning condition $W_0 \subseteq (\text{VA})^\omega$ is regular.

For the following proof we use the fact that Büchi automata can be complemented effectively. That means, from a given Büchi automaton \mathcal{B} , we can effectively construct a Büchi automaton \mathcal{B}^c such that $L(\mathcal{B}^c)$ is the complement of $L(\mathcal{B})$, see for example [5]. Therefore, it suffices to construct, from a Büchi automaton $\mathcal{A} = (\text{VA}, Q, q_0, \Delta, F)$ recognizing W_1 , a Büchi automaton $\overline{\mathcal{A}}$ recognizing $\overline{W_1}$. By definition of $\overline{W_0}$, for each play $\overline{\pi} = \overline{v_0}a_1\overline{v_1}a_2\overline{v_2}$ in \overline{G} from $\overline{v_0} = \{v_0\}$ we have $\overline{\pi} \in \overline{W_1}$ if and only if there is some play $\pi = v_0a'_1v_1a'_2v_2\dots$ in G from v_0 with $a'_i \sim^A a_i$ and $v_i \in \overline{v_i}$ for all $i < \omega$ such that $\pi \in W_1$. The automaton $\overline{\mathcal{A}}$ that recognizes $\overline{W_1}$, guesses such a play π which is a potential witness for $\overline{\pi} \in \overline{W_1}$ and at the same time simulates \mathcal{A} on π . Formally, we define $\overline{\mathcal{A}} = (\overline{\text{VA}}, \overline{Q}, (q_0, v_0), \overline{\Delta}, \overline{F})$ as follows. First $\overline{Q} = Q \times V$ and $\overline{F} = F \times V$. Moreover,

$$((p, v), \overline{v}a, (q, w)) \in \overline{\Delta} \quad :\iff \quad v \in \overline{v} \text{ and there is some action } b \sim^A a \text{ such} \\ \text{that } b \in \text{act}(v), f_b(v) = w \text{ and } (p, vb, q) \in \Delta.$$

Proposition 4. *For each play $\overline{\pi} = \overline{v_0}a_1\overline{v_1}a_2\overline{v_2}\dots$ in \overline{G} from $\overline{v_0} = \{v_0\}$ we have $\overline{\pi} \in \overline{W_1}$ if and only if $\overline{\pi} \in L(\overline{\mathcal{A}})$.*

Proof. If $\overline{\pi} \in \overline{W_1}$, then there is a play $\pi = v_0a'_1v_1a'_2v_2\dots$ in G with $a'_i \sim^A a_i$ and $v_i \in \overline{v_i}$ for all $i < \omega$ such that $\pi \in W_1$. Since $W_1 = L(\mathcal{A})$, there is an accepting run $\rho : \omega \rightarrow Q$ of \mathcal{A} on π . Consider $\overline{\rho} : \omega \rightarrow \overline{Q}$ with $\overline{\rho}(i) := (\rho(i), v_i)$ for $i < \omega$. By definition of $\overline{\mathcal{A}}$, $\overline{\rho}$ is a run of $\overline{\mathcal{A}}$ on $\overline{\pi}$ and since ρ is accepting, so is $\overline{\rho}$.

Now let conversely $\overline{\pi} \in L(\overline{\mathcal{A}})$ and let $\overline{\rho} : \omega \rightarrow \overline{Q}$ be an accepting run of $\overline{\mathcal{A}}$ on $\overline{\pi}$. We define $\rho : \omega \rightarrow Q$ by $\rho(i) := \text{pr}_1(\overline{\rho}(i))$ for $0 < i < \omega$. Since $\overline{\rho} \in \overline{\text{acc}}$ we have $\rho \in \text{acc}$. Furthermore, we define a sequence $\pi^0 \preceq \pi^1 \preceq \pi^2 \preceq \dots$ of histories $\pi^i = v_0a'_1v_1\dots a'_i v_i \in \mathcal{H}_{\text{fin}}$ with $v_i = \text{pr}_2(\overline{\rho}(i))$ and $a'_i \sim^A a_i$ for $0 < i < \omega$, such that $\rho(\leq i)$ is a finite prefix of a run of \mathcal{A} on some extension of π^i to a play in G for all $i < \omega$. First we define $\pi^0 := v_0$. Now let $0 < i < \omega$. Then $(\overline{\rho}(i), \overline{v_i}a_{i+1}, \overline{\rho}(i+1)) \in \overline{\Delta}$, that means, $v \in \overline{v_i}$ and there exists $b \sim^A a_{i+1}$ such that $b \in \text{act}(v)$, $f_b(v) = w$ and $(p, vb, q) \in \Delta$, where $\overline{\rho}(i) = (p, v)$ and $\overline{\rho}(i+1) = (q, w)$. So we define $\pi^i := \pi^{i-1}bw$. Notice that $\rho(i) = q$, $\text{last}(\pi^{i-1}) = v$ and

$\rho(i+1) = q$. Now the sequence $\pi \in V(AV)^\omega$ with $\text{first}(\pi) = v_0$ and $\pi(i) = \pi^i(i)$ for all $0 < i < \omega$ is a play in G from v_0 and ρ is an accepting run of \mathcal{A} on π , so $\pi \in W_1$. Furthermore, $a'_i \sim^A a_i$ and $v_i \in \bar{v}_i$ for all $i < \omega$, so $\bar{\pi} \in \bar{W}_1$. \square

To infer from this the decidability of the synchronous strategy problem for arbitrary ω -regular winning conditions, we need the fact that any Büchi automaton \mathcal{B} , can be effectively transformed into a deterministic parity automaton \mathcal{B}' with $L(\mathcal{B}) = L(\mathcal{B}')$, see e.g. [5]. So we can solve a full information ω -regular game G by taking the product of the game graph and a deterministic parity automaton recognizing the winning condition of G . This yields a parity game which can be effectively solved. As this is a standard construction, we do not prove its correctness here. Moreover, parity games are positionally determined, from which we infer using this construction, that finite memory strategies suffice to win the original games with partial information. Notice that, since for a game with partial information, the corresponding game with full information may be nondeterministic, we use in fact, that nondeterministic parity games are positionally determined. This, however, is a consequence of the result for deterministic games and our straightforward translation of nondeterministic games into deterministic games, cf. Section 2.

Theorem 5. *Given an ω -regular game \mathcal{G} with partial information and some position v_0 , we can decide whether $v_0 \in \text{Win}_0^{\mathcal{G}}$. If $v_0 \in \text{Win}_0^{\mathcal{G}}$ we can effectively construct a finite memory winning \sim^* -strategy for player 0 for \mathcal{G} from v_0 .*

Proof. Let $\mathcal{A} = (\bar{V}A, Q, q_0, \gamma, \text{col})$ be a deterministic parity automaton with $L(\mathcal{A}) = \bar{W}_0$. We define the game $\bar{\mathcal{G}}' = (\bar{V} \times Q, \bar{V}_0 \times Q, (\bar{E}'_a)_{a \in A}, \text{col})$ as follows. For $(\bar{v}, q) \in \bar{V} \times Q$ and $a \in A$, let $(\bar{v}, q)\bar{E}'_a = \{(\bar{w}, \gamma(q, \bar{v}a)) \mid (\bar{v}, \bar{w}) \in \bar{E}_a\}$ and $\text{col}((\bar{v}, q)) = \text{col}(q)$. Then, $v_0 \in \text{Win}_0^{\mathcal{G}}$ if, and only if, player 0 has a winning strategy for $\bar{\mathcal{G}}'$ from $(\bar{v}(v_0), q_0)$ which is decidable since $\bar{\mathcal{G}}'$ is a finite parity game and can be constructed effectively. Now let $v_0 \in \text{Win}_0^{\mathcal{G}}$. Since $\bar{\mathcal{G}}'$ is a parity game, player 0 has a positional winning strategy $g : \bar{V}_0 \times Q \rightarrow A$ for $\bar{\mathcal{G}}'$ from (\bar{v}, q_0) . Now we define the \sim^* -memory structure $M = (\bar{V} \times Q, \delta_0 : \{v_0\} \rightarrow \{(\{v_0\}, q_0)\}, \delta)$ for player 0 for \mathcal{G} by $\delta((\bar{v}, q), (a, v)) := (\text{Post}_{[a]}(\bar{v}) \cap [v], \gamma(q, \bar{v}a))$ for $(\bar{v}, q) \in \bar{V} \times Q$ and $(a, v) \in A \times V$. Furthermore we define $f : (\bar{V} \times Q) \times V_0 \rightarrow A$ as follows. Let $((\bar{v}, q), v) \in (\bar{V} \times Q) \times V_0$. If $v \sim^V u$ for some $u \in \bar{v}$ then we define $f((\bar{v}, q), v) := g((\bar{v}, q))$. Otherwise we let $f((\bar{v}, q), v)$ be an arbitrary action from $\text{act}(v)$ while regarding the condition that f has to be constant over equivalence classes of positions. A very similar reasoning as in the proof of Proposition 3 shows that f is a winning strategy for player 0 for \mathcal{G} from v_0 . \square

3.1 The Asynchronous Case

To solve the asynchronous strategy we use a modified version of the powerset construction. The idea of the construction is the same as in the synchronous case, i.e., any position of the new game captures the uncertainties about the recent

position that player 0 has after some finite history $\pi \in \mathcal{H}_{\text{fin}}$. These uncertainties are now given by the equivalence relation \prec^* , so the positions of the new game are of the form $\tilde{v}(\pi) = \{\text{last}(\pi') \mid \pi' \prec^* \pi\}$ for $\pi \in \mathcal{H}_{\text{fin}}$. Again, each such set is a subset of some equivalence class $[v]$, so as the set of positions of the new game we take the set of all such subsets. And again, in the new game graph, there is an edge with label a from $\tilde{v}(\pi)$ to $\tilde{v}(\rho)$ if and only if $\rho = \pi aw$ for some $w \in V$. The corresponding update mechanism is, however, more involved than in the synchronous case. For a set $U \subseteq V$ of positions, let $\text{Reach}_p(U) \supseteq U$ be the set of all positions which are reachable from some position in U by a (possibly empty) sequence of private moves of player 1. Moreover, for a set $U \subseteq V$, an action a and a position $w \in \text{Post}_a(U)$, we define $U[aw]$ to be U , if $U \subseteq V_1$ and $w \sim^V v$ for some $v \in U$ and we define $U[aw]$ to be $\text{Reach}_p(\text{Post}_{[a]}(U) \cap [w])$, otherwise.

Proposition 6. $\tilde{v}(\pi aw) = \tilde{v}(\pi)[aw]$.

Let $\mathcal{G} = (G, \sim^V, \sim^A)$, $G = (V, V_0, (f_a)_{a \in A}, W_0)$ be a game with partial information. We define the corresponding game $\tilde{G} = (\tilde{V}, \tilde{V}_0, (\tilde{E}_a)_{a \in A}, \tilde{W}_0)$ with full information as follows. First, $\tilde{V} = \{\tilde{v} \in 2^V \mid \exists v \in V : \tilde{v} \subseteq [v]\}$ and $\tilde{V}_0 = \tilde{V} \cap 2^{V_0}$. Moreover, for $a \in A$ we have $(\tilde{v}, \tilde{w}) \in \tilde{E}_a$ if and only if there is some $w \in \text{Post}_a(\tilde{v})$ such that $\tilde{w} = \tilde{v}[aw]$. Finally,

$$\tilde{\pi} = \tilde{v}_0 a_1 \tilde{v}_1 a_2 \tilde{v}_2 \dots \in \tilde{W}_0 \quad :\iff$$

for each play $\pi = v_0 a'_1 v_1 a'_2 v_2 \dots$ in \mathcal{G} , the following holds:

if there are numbers $0 = k_0 < k_1 < \dots$ with $v_{k_i}, \dots, v_{k_{i+1}-1} \in \tilde{v}_i$

and $a'_{k_{i+1}} \sim^A a_{i+1}$ for all i and $k_{i+1} - k_i = 1$, if $\tilde{v}_i \in \tilde{V}_0$

then $\pi \in W_0$.

To prove the correctness of the powerset construction for the asynchronous case, first we state the analog of Proposition 2.

Proposition 7. *For each finite history $\tilde{\pi} = \tilde{v}_0 a_1 \tilde{v}_1 \dots a_n \tilde{v}_n$ in \tilde{G} with $\tilde{v}_i \neq \tilde{v}_{i+1}$ if $\tilde{v}_i \in \tilde{V}_1$ and all $v_m \in \tilde{v}_n$ there is a finite history $\pi = v_0 a'_1 v_1 \dots a'_m v_m$ in \mathcal{G} such that there are numbers $0 = k_0 < \dots < k_{n+1} = m + 1$ with $v_{k_i}, \dots, v_{k_{i+1}-1} \in \tilde{v}_i$ and $a'_{k_i} \sim^A a_i$ for all i and $k_{i+1} - k_i = 1$, if $\tilde{v}_i \in \tilde{V}_0$. Moreover, for each such history, $\tilde{v}(\pi(\leq k_{i+1} - 1)) = \tilde{v}_i$ for all i .*

Proposition 8. *Player 0 has a winning \prec^* -strategy f for \mathcal{G} from v_0 if and only if he has a winning strategy \tilde{f} for \tilde{G} from $\tilde{v}_0 = \tilde{v}(v_0)$.*

Proof. If $f : \{\pi \in \mathcal{H}_{\text{fin}} \mid \text{last}(\pi) \in V_0\} \rightarrow A$ is a winning \prec^* -strategy we define $\tilde{f} : \{\tilde{\pi} \in \tilde{\mathcal{H}}_{\text{fin}} \mid \text{last}(\tilde{\pi}) \in \tilde{V}_0\} \rightarrow A$ as follows. Let $\tilde{\pi} = \tilde{v}_0 a_1 \tilde{v}_1 \dots a_n \tilde{v}_n$ be a finite history with $\tilde{v}_n \in \tilde{V}_0$ and $\tilde{v}_i \neq \tilde{v}_{i+1}$ if $\tilde{v}_i \in \tilde{V}_1$. (If this is not the case, we let the value of \tilde{f} on $\tilde{\pi}$ be the value of \tilde{f} on the history which is obtained from $\tilde{\pi}$ by deleting all self-loops of player 1.) Then there is some finite history $\pi = v_0 a'_1 v_1 \dots a'_m v_m$ such that there are numbers $0 = k_0 < \dots < k_{n+1} = m + 1$ with $v_{k_i}, \dots, v_{k_{i+1}-1} \in \tilde{v}_i$ and $a'_{k_i} \sim^A a_i$ for all i and $k_{i+1} - k_i = 1$, if $\tilde{v}_i \in \tilde{V}_0$. We define $\tilde{f}(\tilde{\pi}) := f(\pi)$. Since f is a \prec^* -strategy, this definition is independent of the chosen history π .

Now assume that there is a play $\tilde{\pi} = \tilde{v}_0 a_1 \tilde{v}_1 a_1 \tilde{v}_2 \dots$ in \tilde{G} which is compatible with \tilde{f} and not won by player 0. Then there is some play $\pi = v_0 a'_1 v_1 a'_1 v_2 \dots \notin W_0$ such that there are numbers $0 = k_0 < k_1 < k_2 < \dots$ with $v_{k_i}, \dots, v_{k_{i+1}-1} \in \tilde{v}_i$ and $a'_{k_i} \sim^A a_i$ for all $i < \omega$ and $k_{i+1} - k_i = 1$, if $\tilde{v}_i \in V_0$. If $j < \omega$ such that $v_j \in V_0$ and $i < \omega$ such that $k_i \leq j < k_{i+1}$ then $v_j \in \tilde{v}_i$ which yields $\tilde{v}_i \in \tilde{V}_0$ and thus $k_{i+1} - k_i$. Therefore, $a'_{k_{i+1}}, a_{i+1} \in A_0$ and since $a_{i+1} \sim^A a'_{k_{i+1}} = a'_{k_i+1} = a'_{j+1}$ we obtain $a'_{j+1} = a_{i+1} = \tilde{f}(\tilde{\pi}(\leq i)) = f(\pi(\leq k_i)) = f(\pi(\leq j))$ by definition of \tilde{f} . Thus, π is compatible with f in contradiction to f being a winning strategy.

If, conversely, $\tilde{f} : \{\tilde{\pi} \in \tilde{\mathcal{H}}_{\text{fin}} \mid \text{last}(\tilde{\pi}) \in \tilde{V}_0\} \rightarrow A$ is a winning strategy, we first fix a subgraph \tilde{G}' of \tilde{G} such that for each edge $(\tilde{u}, \tilde{v}) \in \tilde{E}_a$ there is no action $b \sim^A a$ with $(\tilde{u}, \tilde{v}) \in \tilde{E}_b$ and we define $f : \{\pi \in \mathcal{H}_{\text{fin}} \mid \text{last}(\pi) \in V_0\} \rightarrow A$ as follows. For a finite history $\pi = v_0 a_1 v_1 \dots a_m v_m$ with $v_m \in V_0$ let $0 = k_0 < \dots < k_{n+1}$ be numbers such that, for any i , $v_{k_i} \rightarrow \dots \rightarrow v_{k_{i+1}-1}$ is a maximal sequence of private moves of player 1 in π and let $\tilde{v}_i := \tilde{v}(\pi(\leq k_{i+1} - 1))$ for $i = 1, \dots, n$. Then there are uniquely determined actions $a'_1, \dots, a'_n \in A$ with $a'_i \sim a_{k_i}$ for all i such that $\tilde{\pi} := \tilde{v}_0 a'_1 \tilde{v}_1 \dots a'_n \tilde{v}_n$ is a history of a play in \tilde{G}' and we define $f(\pi) := \tilde{f}(\tilde{\pi})$. To see that f is a \prec^* -strategy for player 0 for \mathcal{G} from v_0 consider finite histories $\pi = v_0 a_1 v_1 \dots a_m v_m$ and $\rho = w_0 b_1 w_1 \dots a_r w_r$ with $\pi \prec^* \rho$ and $v_m, w_r \in V_0$ and let $0 = k_0 < \dots < k_{n+1}$ be numbers such that $v_{k_i} \rightarrow \dots \rightarrow v_{k_{i+1}-1}$ is a maximal sequence of private moves of player 1 in π for all i . Let analog $0 = l_0 < \dots < l_{p+1}$ be such numbers for ρ . Since $\pi \prec^* \rho$ we have $n = p$ and $\tilde{v}_i = \tilde{w}_i$ and $a_{k_i} \sim^A b_{l_i}$ for all i and hence also $a'_i = b'_i$ for all i . Therefore $f(\pi) = \tilde{f}(\tilde{v}_0 a'_1 \tilde{v}_1 \dots a'_n \tilde{v}_n) = \tilde{f}(\tilde{w}_0 b'_1 \tilde{w}_1 \dots b'_n \tilde{w}_n) = f(\rho)$.

Now assume that there is a play $\pi = v_0 a_1 v_1 a_2 v_2 \dots \notin W_0$ which is compatible with f . Let $0 = k_0 < k_1 < \dots$ be numbers such that for $i < \omega$, $v_{k_i} \rightarrow \dots \rightarrow v_{k_{i+1}-1}$ is a maximal sequence of private moves of player 1 in π . If there is some $i < \omega$ such that $v_j \rightarrow v_{j+1}$ is a private move of player 1 for all $j \geq i$, then we consider the minimal such i , and we let $k_{j+1} = k_j + 1$ for all $j \geq i$. Then there are uniquely determined actions a'_i for $1 \leq i < \omega$ such that $\tilde{\pi} = \tilde{v}_0 a'_1 \tilde{v}_1 a'_2 \tilde{v}_2 \dots$ is a play in \tilde{G}' . Now if $i < \omega$ with $\tilde{v}_i \in \tilde{V}_0$, then $v_{k_{i+1}-1} \in V_0$ and since π is compatible with f , the definition of f yields $a_{k_{i+1}} = f(\pi(\leq k_{i+1}-1)) = \tilde{f}(\tilde{\pi}(\leq i))$. Moreover, $a'_{i+1} \in \text{act}(\tilde{v}_i) \subseteq A_0$ and since $a'_{i+1} \sim^A a_{k_{i+1}}$ we have $a'_{i+1} = a_{k_{i+1}}$. So $\tilde{\pi}$ is compatible with \tilde{f} and thus, $\tilde{\pi}$ is won by player 0. But since $\pi \notin W_0$ we have $\tilde{\pi} \notin \tilde{W}_0$, which is a contradiction. \square

Omega-Regular Winning Conditions. Now we prove that ω -regular winning conditions are effectively preserved by the powerset construction in the asynchronous case. Again, we use complementation. Let $\mathcal{A} = (\text{VA}, Q, q_0, \Delta, F)$ be a Büchi automaton with $L(\mathcal{A}) = W_1$. The idea for the automaton $\tilde{\mathcal{A}}$ recognizing \tilde{W}_1 is as in the synchronous case, that means, for a given play $\tilde{\pi}$ in \tilde{G} it guesses a play π in \mathcal{G} which is a potential witness for $\tilde{\pi} \in W_1$ and at the same time checks whether \mathcal{A} accepts this play. Such potential witnesses are more complicated objects than in the synchronous case, so the construction of $\tilde{\mathcal{A}}$ is more involved. Moreover, it is not obvious that the construction is effective, which we

shall prove separately. We define $\tilde{\mathcal{A}} = (\tilde{V}\mathcal{A}, \tilde{Q}, (q_0, v_0, 0), \tilde{\Delta}, \tilde{F})$ as follows. First, $\tilde{Q} = Q \times V \times \{0, 1\}$ and $\tilde{F} = \{(q, v, i) \mid q \in F\} \cup \{(q, v, 1) \mid q \in Q\}$. Moreover,

$$((p, v, i), \tilde{v}a, (q, w, j)) \in \tilde{\Delta} \quad :\iff$$

there is a finite history $v_1 a_2 v_2 \dots a_n v_n$ in G such that:

1. $n = 1$, if $\tilde{v} \in \tilde{V}_0$
2. $v_1 = v$ and $v_l \in \tilde{v}$ for all $1 \leq l \leq n$
3. there is some $b \in \text{act}(v_n)$ with $b \sim^A a$ and $f_b(v_n) = w$
4. there are $q_1, \dots, q_{n-1} \in Q$ such that
 - 4.1 $(p, v_1 a_2, q_1), \dots, (q_{n-2}, v_{n-1} a_n, q_{n-1}), (q_{n-1}, v_n b, q) \in \Delta$
 - 4.2 $q_l \in F$ for some $0 < l < n$ if $j = 1$.

Proposition 9. *For each play $\tilde{\pi} = \tilde{v}_0 a_1 \tilde{v}_1 a_2 \tilde{v}_2 \dots$ in \tilde{G} from $\tilde{v}_0 = \tilde{v}(v_0)$ we have $\tilde{\pi} \in \tilde{W}_1$ if and only if $\tilde{\pi} \in L(\tilde{\mathcal{A}})$.*

Proof. Let first $\tilde{\pi} \in \tilde{W}_1$, i.e., there is a play $\pi = v_0 a'_1 v_1 a'_2 v_2 \dots \in W_1$ such that there are numbers $0 = k_0 < k_1 < \dots$ with $v_{k_i}, \dots, v_{k_{i+1}-1} \in \tilde{v}_i$ and $a'_{k_{i+1}} \sim^A a_{i+1}$ for all $i < \omega$ and $k_{i+1} - k_i = 1$, if $\tilde{v}_i \in \tilde{V}_0$. Since $\pi \in W_1$ we have $\pi \in L(\mathcal{A})$, that means, there is an accepting run $\rho : \omega \rightarrow Q$ of \mathcal{A} on π . Now let $\tilde{\rho} : \omega \rightarrow \tilde{Q}$ be defined by $\tilde{\rho}(i) := (\rho(k_i), v_{k_i}, 0)$, if $\rho(j) \notin F$ for all $k_{i-1} < j < k_i$ and $\tilde{\rho}(i) := (\rho(k_i), v_{k_i}, 1)$, if $\rho(j) \in F$ for some $k_{i-1} < j < k_i$, where $k_{-1} := 0$. According to the definition of $\tilde{\mathcal{A}}$, $\tilde{\rho}$ is a run of $\tilde{\mathcal{A}}$ on $\tilde{\pi}$ and since ρ is accepting, there are infinitely many $i < \omega$ such that $\rho(i) \in F$. So there must be infinitely many $i < \omega$ such that $\text{pr}_1(\tilde{\rho}(i)) \in F$ or $\text{pr}_3(\tilde{\rho}(i)) = 1$. Thus, $\tilde{\rho}$ is accepting.

Now let conversely $\tilde{\pi} \in L(\tilde{\mathcal{A}})$ and let $\tilde{\rho} : \omega \rightarrow \tilde{Q}$ be an accepting run of $\tilde{\mathcal{A}}$ on $\tilde{\pi}$. We define sequences $\rho^0 \preceq \rho^1 \preceq \dots$ and $\pi^0 \preceq \pi^1 \preceq \dots$ of finite prefixes $\rho^i = q_0 \dots q_{k_i} \in Q^*$ of runs of \mathcal{A} and of finite prefixes $\pi^i = v_0 a'_1 v_1 \dots a'_{k_i} v_{k_i} \in \mathcal{H}_{\text{fin}}$ of plays in G from v_0 , where $k_i < \omega$ for $i < \omega$ and $k_0 = 0$, such that for all $i < \omega$ the following holds. First, $q_{k_i} = \text{pr}_1(\tilde{\rho}(i))$ and $v_{k_i} = \text{pr}_2(\tilde{\rho}(i))$. Furthermore, $v_{k_i}, \dots, v_{k_{i+1}-1} \in \tilde{v}_i$, $a'_{k_{i+1}} \sim^A a_{i+1}$ and $k_{i+1} - k_i = 1$, if $\tilde{v}_i \in \tilde{V}_0$. Finally, ρ^i is a prefix of a run of \mathcal{A} on some extension of π^i to a play in G such that $\rho^i(j) \in F$ for some $k_{i-1} < j < k_i$, if $\text{pr}_3(\tilde{\rho}(i)) = 1$, where $k_{-1} = 0$.

First we define $\pi^0 = v_0$. Now let $0 < i < \omega$. Then $(\tilde{\rho}(i), \tilde{v}_i a_{i+1}, \tilde{\rho}(i+1)) \in \tilde{\Delta}$, so there is a finite history $v_1 a_2 v_2 \dots a_n v_n$ in G such that conditions 1. - 4. hold. That means, $n = 1$ if $\tilde{v}_i \in \tilde{V}_0$, $v_1 = v$, $v_l \in \tilde{v}_l$ for $1 \leq l \leq n$, there is some $b \sim^A a_{i+1}$ with $b \in \text{act}(v_n)$ and $f_b(v_n) = w$ and there are $q_1, \dots, q_{n-1} \in Q$ with $(p, v_1 a_2, q_1), \dots, (q_{n-2}, v_{n-1} a_n, q_{n-1}), (q_{n-1}, v_n b, q) \in \Delta$ and $q_l \in F$ for some $0 < l < n$ if $j = 1$, where $\tilde{\rho}(i) = (p, v, i)$ and $\tilde{\rho}(i+1) = (q, w, j)$. So we define $\pi^i = \pi^{i-1} a_2 v_2 \dots a_n v_n b w$ and $\rho^i = \rho^{i-1} q_1 \dots q_{n-1} q$.

Now if $\pi \in V(AV)^\omega$ and $\rho \in Q^\omega$ are the limit sequences of the finite sequences π^i and ρ^i for $i < \omega$, then π is a play in G from v_0 and ρ is a run of \mathcal{A} on π . Since $\tilde{\rho}$ is an accepting run of $\tilde{\mathcal{A}}$ on $\tilde{\pi}$ we have $\text{pr}_1(\tilde{\rho}(i)) \in F$ for infinitely many $i < \omega$ or $\text{pr}_3(\tilde{\rho}(i)) = 1$ for infinitely many $i < \omega$. By construction of ρ , this yields $\rho(i) \in F$ for infinitely many $i < \omega$. That means, ρ is an accepting run of \mathcal{A} on π and so $\pi \in L(\mathcal{A}) = W_1$. Furthermore, by construction of π we have

$v_{k_i}, \dots, v_{k_{i+1}-1} \in \tilde{v}_i$ and $a'_{k_{i+1}-1} \sim^A a_i$ for all $i < \omega$ and $k_{i+1} - k_i = 1$, if $\tilde{v}_i \in \tilde{V}_0$. Thus we can conclude that $\tilde{\pi} \in \tilde{W}_1$. \square

Proposition 10. *The following problem is decidable.*

Given a transition $t = ((p, v, i), \tilde{v}a, (q, w, j))$, is $t \in \tilde{\Delta}$?

Proof. The idea for the proof is as follows. If $\tilde{v} \in \tilde{V}_0$, $t \in \tilde{\Delta}$ is clearly decidable, so consider $\tilde{v} \in \tilde{V}_1$. We construct an NFA \mathcal{B}_{fin} over VA accepting precisely those words which are finite histories in G such that conditions 2. and 3. hold. If \mathcal{C}_{fin} is then the product automaton of \mathcal{B}_{fin} and the NFA \mathcal{A}_{fin} is obtained from \mathcal{A} , by defining all states to be accepting and augmenting \mathcal{A} by a state to remember whether some state in F has been seen if $j = 1$, then \mathcal{C}_{fin} accepts precisely those words which are finite histories in G such that conditions 2. - 4. hold. So $t \in \tilde{\Delta}$ if, and only if, $L(\mathcal{C}_{\text{fin}}) \neq \emptyset$.

Now we present the technical details of this proof. First let $j = 0$. If $\tilde{v} \in \tilde{V}_0$, then $t \in \tilde{\Delta}$ if and only if $v \in \tilde{v}$ and there is some $b \sim^A a$ with $b \in \text{act}(v)$ and $f_b(v) = w$ such that $(p, vb, q) \in \Delta$. Clearly this is decidable. If $\tilde{v} \in \tilde{V}_1$, then $t \in \tilde{\Delta}$ if and only if $v \in \tilde{v}$ and there is a finite history $v_1 a_2 v_2 \dots a_n v_n$ in G such that $v_1 = v$, $v_j \in \tilde{v}$ for $j = 1, \dots, n$, there is some $b \sim^A a$ with $b \in \text{act}(v_n)$ and $f_b(v_n) = w$ and there are $q_1, \dots, q_{n-1} \in Q$ with $(p, v_1 a_2, q_1), \dots, (q_{n-1}, v_n b, q) \in \Delta$. Consider the NFA $\mathcal{C}_0 = (\tilde{v}A, Q \times \tilde{v} \cup \{q\}, (p, v), \Delta_{\mathcal{C}_0}, \{q\})$:

- $((p_1, v_1), v' a', (p_2, v_2)) \in \Delta_{\mathcal{C}_0} \iff$
 $v' = v_1, a' \in \text{act}(v_1), f_{a'}(v_1) = v_2$ and $(p_1, v' a', p_2) \in \Delta$.
- $((p_1, v_1), v' a', q) \in \Delta_{\mathcal{C}_0} \iff$
 $v' = v_1, a' \in \text{act}(v_1), f_{a'}(v_1) = w, a' \sim^A a$ and $(p_1, v' a', q) \in \Delta$.

It is easy to see that $t \in \tilde{\Delta}$ if and only if $L(\mathcal{C}_0) \neq \emptyset$. Since the latter is decidable and the construction of \mathcal{C}_0 is effective, the former is decidable as well.

Now let $j = 1$. If $\tilde{v} \in \tilde{V}_0$ then $t \notin \tilde{\Delta}$, so let $\tilde{v} \in \tilde{V}_1$. Then $t \in \tilde{\Delta}$ if and only if $v \in \tilde{v}$ and there is a finite history $v_1 a_2 v_2 \dots a_n v_n$ in G such that $v_1 = v$, $v_j \in \tilde{v}$ for $j = 1, \dots, n$, there is some $b \sim^A a$ with $b \in \text{act}(v_n)$ and $f_b(v_n) = w$ and there are $q_1, \dots, q_{n-1} \in Q$ with $(p, v_1 a_2, q_1), \dots, (q_{n-1}, v_n b, q) \in \Delta$ and $q_l \in F$ for some $0 < l < n$. Consider the NFA $\mathcal{C}_1 = (\tilde{v}A, Q \times \tilde{v} \times \{0, 1\} \cup \{q\}, (p, v, 0), \Delta_{\mathcal{C}_1}, \{q\})$:

- $((p_1, v_1, i), v' a', (p_2, v_2, i)) \in \Delta_{\mathcal{C}_0} \iff$
 $v' = v_1, a' \in \text{act}(v_1), f_{a'}(v_1) = v_2$ and $(p_1, v' a', p_2) \in \Delta$.
- $((p_1, v_1, 0), v' a', (p_2, v_2, 1)) \in \Delta_{\mathcal{C}_0} \iff$
 $v' = v_1, a' \in \text{act}(v_1), f_{a'}(v_1) = v_2, p_2 \in F$ and $(p_1, v' a', p_2) \in \Delta$.
- $((p_1, v_1, 1), v' a', q) \in \Delta_{\mathcal{C}_0} \iff$
 $v' = v_1, a' \in \text{act}(v_1), f_{a'}(v_1) = w, a' \sim^A a$ and $(p_1, v' a', q) \in \Delta$.

It is easy to see that $t \in \tilde{\Delta}$ if and only if $L(\mathcal{C}_1) \neq \emptyset$. Since the latter is decidable and the construction of \mathcal{C}_1 is effective, the former is decidable as well. \square

Theorem 11. *Given an ω -regular game \mathcal{G} with partial information and some position v_0 , we can decide whether $v_0 \in \text{Win}_0^{\mathcal{G}, h}$. If $v_0 \in \text{Win}_0^{\mathcal{G}, h}$ we can effectively construct a finite memory winning \prec^* -strategy for player 0 for \mathcal{G} from v_0 .*

Proof. Let $\mathcal{A} = (\tilde{V}A, Q, q_0, \gamma, \text{col})$ be a deterministic parity automaton with $L(\mathcal{A}) = \tilde{W}_1$. We define the game $\tilde{G}' = (\tilde{V} \times Q, \tilde{V}_0 \times Q, (\tilde{E}'_a)_{a \in A}, \text{col})$ as follows. For $(\tilde{v}, q) \in \tilde{V} \times Q$ and $a \in A$ we define $(\tilde{v}, q)\tilde{E}'_a := \{(\tilde{w}, \gamma(q, \tilde{v}a)) \mid (\tilde{v}, \tilde{w}) \in \tilde{E}_a\}$ and $\text{col}((\tilde{v}, q)) := \text{col}(q)$. Then, $v_0 \in \text{Win}_0^{\mathcal{G}, h}$ if, and only if, player 0 has a winning strategy for \tilde{G}' from $(\tilde{v}(v_0), q_0)$. This is decidable since \tilde{G}' is a finite parity game and can be constructed effectively. Now let $v_0 \in \text{Win}_0^{\mathcal{G}, h}$. Since \tilde{G}' is a parity game, player 0 has a positional winning strategy $g : \tilde{V} \times Q \rightarrow A$ for \tilde{G}' from $(\tilde{v}(v_0), q_0)$. We define the \prec^* -memory structure $M = (\tilde{V} \times Q, \delta : \{v_0\} \rightarrow \{(\{v_0\}, q_0)\}, \delta)$ for \mathcal{G} as follows. Let $(a, v) \in A \times V$ and let $(\tilde{v}, q) \in \tilde{V} \times Q$. If $\tilde{v} \subseteq V_0$ and $v \sim^V u$ for $u \in \tilde{v}$, then $\delta((\tilde{v}, q), (a, v)) = (\tilde{v}, q)$. If $\tilde{v} \subseteq V_0$ or $v \not\sim^V u$ for $u \in \tilde{v}$, then $\delta((\tilde{v}, q), (a, v)) = (\text{Reach}_p(\text{Post}_{[a]}(\tilde{v}) \cap [v]), \gamma(q, \tilde{v}a))$. Now we define $f : (\tilde{V} \times Q) \times V_0 \rightarrow A$ as follows. Let $((\tilde{v}, q), v) \in (\tilde{V} \times Q) \times V_0$. If $v \sim^V u$ for some $u \in \tilde{v}$ then we define $f((\tilde{v}, q), v) := g((\tilde{v}, q))$. Otherwise we let $f((\tilde{v}, q), v)$ be an arbitrary action from $\text{act}(v)$ while regarding the condition that f has to be constant over equivalence classes of positions. Similar reasoning as in the proof of Proposition 8 shows that f is a winning strategy for player 0 for \mathcal{G} from v_0 . \square

4 Epistemic Temporal Logic

In this section we consider games $\mathcal{G} = (G, (\sim_i^V)_{i=0,1}, (\sim_i^A)_{i=0,1})$ with partial information for both players. Basically, epistemic temporal logic ETL is obtained from LTL (linear temporal logic) by adding the epistemic operators K_i where $K_i\varphi$ for some formula φ means that agent i in the system knows φ . Many incarnations of this concept have been investigated, for an overview and discussion we refer to [6]. Of course the semantics of the knowledge operators has a different character than the semantics of LTL operators, since we cannot evaluate these formulas by considering a single play but we have to take into account all plays where the history up to the current point is indistinguishable for player i from the actual one. Formally, the syntax of ETL is defined by the grammar

$$\varphi ::= P \mid \varphi \wedge \varphi \mid \neg\varphi \mid X\varphi \mid \varphi U \varphi \mid K_i\varphi \mid \overline{K}_i\varphi$$

where $i \in \{0, 1\}$ and P ranges over a set Prop of atomic propositions. Notice that since we consider two kinds of observability, we also have two knowledge operators for each agent i , K_i for the synchronous and \overline{K}_i for the asynchronous case. Incorporating both operators into the same logic enables us to express properties of systems where, e.g., player 0 has a synchronous view and player 1 has an asynchronous view. If we consider completely synchronous or asynchronous systems, we simply ignore one of the knowledge operators.

We evaluate ETL-formulas in interpreted game models, that means, games \mathcal{G} with partial information together with an interpretation $P^{\mathcal{G}} \subseteq V$ of each of the atomic propositions $P \in \text{Prop}$. We assume that these sets $P^{\mathcal{G}}$ are represented explicitly as subsets of V . Now an ETL-formula φ talks about points during plays in G , so for any play $\pi = v_0 a_1 v_1 \dots \in (\text{VA})^\omega$ and any point in time $t < \omega$ we have to define the truth of $\mathcal{G}, \pi, t \models \varphi$. We usually denote this model relationship simply as $\pi, t \models \varphi$ and assume some given interpreted game model

\mathcal{G} . First, $\pi, t \models P$ if $v_t \in P^{\mathcal{G}}$. The semantics of the boolean connectives is as usual. Furthermore, $\pi, t \models X\varphi$ if $\pi, t+1 \models \varphi$ and $\pi, t \models \varphi_1 U \varphi_2$ if there is some $s \geq t$ such that $\pi, s \models \varphi_2$ and $\pi, u \models \varphi_1$ for all $t \leq u < s$. Finally $\pi, t \models K_i \varphi$ if and only if for all plays ρ and all $s < \omega$ with $\pi(\leq t) \sim_i^* \rho(\leq s)$ we have $\pi, s \models \varphi$ and $\pi, t \models \overleftarrow{K}_i \varphi$ if and only if for all plays ρ and all $s < \omega$ with $\pi(\leq t) \overleftarrow{\sim}_i^* \rho(\leq s)$ we have $\rho, s \models \varphi$. Notice that $\pi(\leq t) \sim_i^* \rho(\leq s)$ implies $s = t$ which is not necessarily the case for $\overleftarrow{\sim}_i^*$.

Now given an interpreted game model \mathcal{G} , a formula $\varphi \in \text{ETL}$ defines a set $L_{\mathcal{G}}(\varphi) = \{\pi \in (\text{VA})^\omega \mid \pi, 0 \models \varphi\}$. In [15], it is shown that the synchronous strategy problem for ETL-specifications with only the knowledge operator K_0 is decidable. In this section we prove that $L_{\mathcal{G}}(\varphi) \subseteq (\text{VA})^\omega$ is ω -regular.

This result is of general interest since it highlights the view of $L_{\mathcal{G}}(\varphi)$ as an ω -language (set of plays = winning condition) and shows that regular ω -languages capture the ability to reason about knowledge. It also makes the powerset construction applicable to ETL-specifications and so our techniques can be used to transform a game with partial information and with ETL winning condition into an ω -regular game with full information. This method can be applied to synthesize finite memory winning strategies from ETL specifications, both in the synchronous and in the asynchronous case, hence lifting the result of [15] to asynchronous games and to ETL-specifications with knowledge operators for both players. Finally, and most notably, the result reveals an interesting consequence on the decidability of multiplayer games with ETL winning conditions which we discuss at the end of this section.

Notice that we should clearly assume player 0 to know his own strategy, so if player 0 plays according to some strategy f then in particular he always considers only those histories possible, which are consistent with f . Thus, when given an ETL-formula φ , we are actually interested in a strategy f for player 0, such that $\pi, 0 \models \varphi$ for each play which is consistent with f , where the evaluation of the knowledge operator K_0 is *relative to histories which are consistent with f* . However, in our two-player setting this makes no difference since if $\pi \sim_0 \rho$ for $\sim_0 \in \{\sim_0^*, \overleftarrow{\sim}_0^*\}$, then π is consistent with f if, and only if, ρ is consistent with f . This is due to the fact that player 0 can distinguish any two of his own actions.

On the other hand, for the knowledge operator K_1 to make sense, we have to assume that player 1 does not know the strategy of player 0, since otherwise player 1 has full information about the history, so $K_1 \varphi$ is equivalent to φ . Whether this is appropriate depends highly on the given application. If player 0 models a controller of an environment which is not actually antagonistic but merely unpredictable, then given any strategy f for player 0, the joint system is constrained by f , so the knowledge of both players should be relative to f . On the contrary, if player 0 models a network server which might interact with a user, then the protocol of the server may be off-limits to the user, so player 1 does not know f . One requirement for the protocol might be that the user is never able to learn the value of some internal variables of the server. This can be expressed using K_1 . Moreover, using K_1 we can talk about higher-level knowledge, i.e., knowledge about knowledge about knowledge ...

Consider, for example, a game \mathcal{G} with atomic proposition P_1, \dots, P_r which are interpreted in such a way, that $P_1^{\mathcal{G}}, \dots, P_r^{\mathcal{G}}$ form a partition of V . We use the usual abbreviations G and F defining *globally* and *finally*. Then the LTL-formula

$$\text{parity} := \bigvee_{c \text{ even}} (\text{GF}P_c \wedge \text{FG} \bigwedge_{c' < c} \neg P_{c'})$$

defines the parity objective for player 0 over the coloring $\text{col} : V \rightarrow \{1, \dots, r\}$ with $\text{col}(v) = c$ if $v \in P_c^{\mathcal{G}}$. Moreover, the ETL-formula

$$K_0^{\text{col}} := \bigvee_{c \in C} K_0 P_c$$

says that player 0 knows the color of the recent position. So the formula

$$\varphi = \text{parity} \wedge G(\neg K_1 K_0^{\text{col}} \wedge \neg K_1 \neg K_0^{\text{col}})$$

additionally requires that player 1 never knows whether player 0 knows the recent color.

To prove that ETL winning conditions are ω -regular, we construct, from a given ETL-formula an equivalent S1S-formula. S1S, the *monadic second order logic over words* is defined as follows. For an alphabet Σ we consider the signature $\mathcal{T}_{\Sigma} = \{S, \min, <, (P_a)_{a \in \Sigma}\}$ and for an ω -word $\alpha \in \Sigma^{\omega}$ we consider the \mathcal{T}_{Σ} -structure $\underline{\alpha} = (\omega, S, \min, <, (P_a^{\alpha})_{a \in \Sigma})$ where S is the successor function on ω , \min and $<$ are as usual and $P_a^{\alpha} = \{i < \omega \mid \alpha(i) = a\}$ for all $a \in \Sigma$. Now an MSO(\mathcal{T}_{Σ})-sentence defines an ω -language $L(\varphi) = \{\alpha \in \Sigma^{\omega} \mid \underline{\alpha} \models \varphi\}$. Büchi has shown in 1962, that an ω -language is regular if and only if it is S1S-definable. Moreover, the translations from formulas into automata and vice versa are effective. For a proof, see for example [13].

Theorem 12. *For any interpreted game model \mathcal{G} and any formula $\varphi \in \text{ETL}$ over the same atomic propositions, we can effectively construct an S1S(\mathcal{T}_{VA})-sentence $\hat{\varphi}$ such that $L(\hat{\varphi}) = L_{\mathcal{G}}(\varphi)$.*

Proof. We construct, inductively over the structure of φ , an S1S-formula $\hat{\varphi}(x)$ over the signature \mathcal{T}_{VA} such that for all plays π and all $t < \omega$ we have $\pi, t \models \varphi$ if and only if $\underline{\pi} \models \hat{\varphi}(t)$. First, if $\varphi = P_{va}$ for some $va \in VA$ we define $\hat{\varphi}(x) = P_{va}x$. As usual, $\hat{\varphi}(x) = \hat{\varphi}_1(x) \wedge \hat{\varphi}_2(x)$ if $\varphi = \varphi_1 \wedge \varphi_2$ and $\hat{\varphi}(x) = \neg \hat{\psi}(x)$ if $\varphi = \neg \psi$. Furthermore, $\hat{\varphi}(x) = \hat{\psi}(x/Sx)$ for $\varphi = X\psi$ and $\hat{\varphi}(x) = \exists y(x \leq y \wedge \hat{\varphi}_2(x/y) \wedge \forall z(x \leq z < y \rightarrow \hat{\varphi}_1(x/z)))$ for $\varphi = \varphi_1 U \varphi_2$. Where, for a formula $\vartheta(x) \in \text{S1S}$ and a term t , the formula $\vartheta(x/t)$ is obtained from $\vartheta(x)$ by replacing (simultaneously) each free occurrence of x with t .

It remains to construct $\hat{\varphi}(x)$ for $\varphi = Z_i \psi$ for $i \in \{0, 1\}$ and $Z \in \{K, \overleftarrow{K}\}$. We only consider the case $i = 0$, the construction for the case $i = 1$ is of course exactly the same. Hence, in the following we denote the knowledge operator Z_0 by Z and use the reduced form (G, \sim^V, \sim^A) of \mathcal{G} with partial information only for player 0.

First, consider the case $Z = K$ and let $\hat{\psi}(x)$ be the corresponding formula for ψ according to the induction hypothesis. The idea is as follows. We have $\pi, t \models \neg K\psi$ if and only if there is some play ρ such that $\rho(\leq t) \sim^* \pi(\leq t)$ and $\rho, t \models \neg\psi$. (We take the negation of ψ as the existential quantification over plays is somehow easier to describe.) This can be expressed equivalently by saying that there is a partition of ω into sets X_{va} , where va ranges over VA , such that for the resulting ω -word ρ we have $\pi(\leq t) \sim^* \rho(\leq t)$. As for checking the last condition we only have to compare labels at corresponding positions within the ω -words, this can be expressed in S1S: We define $\hat{\varphi}(x) \in \text{S1S}(\mathcal{T}_{\text{VA}})$ for φ as follows, where all pairs va, wb range over VA .

$$\neg\exists X_{va} \left[\begin{array}{l} \forall y (\bigvee_{va} (X_{va}y \wedge \bigwedge_{wb \neq va} \neg X_{wb}y)) \\ \wedge \quad \forall y \forall z (Sy = z \rightarrow \bigvee_{(v,w) \in E_a, b \in A} X_{va}y \wedge X_{wb}z) \\ \wedge \quad \forall (y \leq x) (\bigwedge_{va} (P_{va}y \rightarrow \bigvee_{wb \sim va} X_{wb}y)) \\ \wedge \quad \neg \hat{\psi}(P_{va}/X_{va}) \end{array} \right].$$

It is easy to see that for all plays π and all $t < \omega$ we have $\pi, t \models \varphi$ if and only if $\pi \models \hat{\varphi}(t)$. However, in the asynchronous case, that means $Z = \overline{K}$, we have $\pi, t \models \neg K\psi$ if and only if there is some play ρ and some s such that $\rho, s \prec^* \pi, t$ and $\rho, s \not\models \psi$ where $s \neq t$ is possible: between each two observations of player 0 there may be a priori unboundedly long sequences of positions and actions which are not observed by player 0. Due to this, it is not clear how to apply the above approach for defining the formula $\hat{\varphi}$ to the asynchronous case. Instead, to deal with these arbitrarily long sequences of moves, we use an automata approach.

Let $\hat{\psi}(x)$ be the corresponding formula for ψ according to the induction hypothesis. Using the usual translation of S1S-formulas into automata, we obtain a Büchi-automaton $\mathcal{A} = (\text{VA} \times \{0, 1\}, Q^{\mathcal{A}}, q_0^{\mathcal{A}}, \Delta^{\mathcal{A}}, F^{\mathcal{A}})$ over the alphabet $\text{VA} \times \{0, 1\}$ such that $L(\mathcal{A}) = \{\pi^\wedge \alpha_t \in (\text{VA} \times \{0, 1\})^\omega \mid \pi \models \neg \hat{\psi}(t)\} = \{\pi^\wedge \alpha_t \in (\text{VA} \times \{0, 1\})^\omega \mid \pi, t \models \neg \psi\}$, where $\pi^\wedge \alpha_t = (v_0 a_1, \alpha_t(0))(v_1 a_2, \alpha_t(1)) \dots$ and for $s < \omega$, $\alpha_t(s) = 1$, if $s = t$ and $\alpha_t(s) = 0$, if $s \neq t$.

First, we construct a Büchi automaton \mathcal{B} which, while reading $\pi^\wedge \alpha_t$, guesses a play ρ and some s such that $\pi(\leq t) \prec^* \rho(\leq s)$ and simulates \mathcal{A} von $\rho^\wedge \alpha_s$. To handle the fact that $s \neq t$ is possible, reading $\pi^\wedge \alpha_t$ and guessing $\rho^\wedge \alpha_s$ must be asynchronous for the first t steps. More precisely, while not reading a letter $(va, 1)$, the automaton \mathcal{B} skips all the private moves that occur in π : it does not change the current state but only keeps track of the current equivalence class of positions and of the last action that it has read. As soon as \mathcal{B} reads some non-private move in π , it extends ρ by a sequence of private moves (of possibly different length than the sequence which has occurred in π) and inputs the previous non-private move together with this sequence of private moves into \mathcal{A} . Finally, when \mathcal{B} reads some letter $(va, 1)$, then it again extends ρ by sequences of private moves (before the position corresponding to v and after that position) and then inputs a corresponding letter $(wb, 1)$ into \mathcal{A} . Afterwards, \mathcal{B} does not have to take care of equivalence of prefixes of π and ρ anymore. Formally,

$\mathcal{B} = (\mathbf{VA} \times \{0, 1\}, Q, q_0, \Delta, F)$ is defined by $Q = \{q_0\} \cup Q^A \times V \times A \cup V \times A \times Q^A$ and $F = V \times A \times F^A$ and

- $(q_0, (va, 0), (q_0^A, w, a)) \in \Delta \quad :\iff w \sim^V v$
- $(q_0, (va, 1), (w, b, q^A)) \in \Delta \quad :\iff$
there exists a finite history $v_1 a_2 v_2 \dots a_n v_n$ in G such that:
 1. $n = 1$, if $v \in V_0$
 2. $v_n = w$ and $v_l \sim^V v$ for all $1 \leq l \leq n$
 3. there are $q_1^A, \dots, q_{n-1}^A \in Q^A$ such that
 $(q_0^A, (v_1 a_2, 0), q_1^A), \dots, (q_{n-2}^A, (v_{n-1} a_n, 0), q_{n-1}^A), (q_{n-1}^A, (v_n b, 1), q^A) \in \Delta^A$
- **if** $w \in V_1$ and $v \sim^V w$: $((q^A, w, b), (va, 0), (q^A, w, a)) \in \Delta$
- **if** $w \in V_0$ or $v \not\sim^V w$: $((q^A, w, b), (va, 0), (p^A, w', a)) \in \Delta \quad :\iff$
 $w' \sim^V v$ and
there exist $b' \sim^A b$ and some finite history $v_1 a_2 v_2 \dots a_n v_n$ in G such that:
 1. $n = 1$, if $w \in V_0$
 2. $v_1 = w$ and $v_l \sim^V v_1$ for all $1 \leq l \leq n$
 3. $f_{b'}(v_n) = w'$
 4. there are $p_1^A, \dots, p_{n-1}^A \in Q^A$ with
 $(q^A, (v_1 a_2, 0), p_1^A), \dots, (p_{n-2}^A, (v_{n-1} a_n, 0), p_{n-1}^A), (p_{n-1}^A, (v_n b', 0), p^A) \in \Delta^A$
- **if** $w \in V_1$ and $v \sim^V w$: $((q^A, w, b), (va, 1), (w', b', p^A)) \in \Delta \quad :\iff$
there exists a finite history $v_1 a_2 v_2 \dots a_n v_n$ in G such that:
 1. $v_1 = w$, $v_n = w'$ and $v_l \sim^V v_1$ for all $1 \leq l \leq n$
 2. there are $p_1^A, \dots, p_{n-1}^A \in Q^A$ with
 $(q^A, (v_1 a_2, 0), p_1^A), \dots, (p_{n-2}^A, (v_{n-1} a_n, 0), p_{n-1}^A), (p_{n-1}^A, (v_n b', 1), p^A) \in \Delta^A$
- **if** $w \in V_0$ or $v \not\sim^V w$: $((q^A, w, b), (va, 1), (w', b', p^A)) \in \Delta \quad :\iff$
there exist $w'' \sim^V v$ and $r^A \in Q^A$ such that:

(I) there exists $b'' \sim^A b$ and a finite history $v_1 a_2 v_2 \dots a_n v_n$ in G such that:

 1. $n = 1$, if $w \in V_0$
 2. $v_1 = w$ and $v_l \sim^V v_1$ for all $1 \leq l \leq n$
 3. $f_{b''}(v_n) = w''$
 4. there are $p_1^A, \dots, p_{n-1}^A \in Q^A$ with
 $(q^A, (v_1 a_2, 0), p_1^A), \dots, (p_{n-2}^A, (v_{n-1} a_n, 0), p_{n-1}^A), (p_{n-1}^A, (v_n b'', 0), r^A) \in \Delta^A$

(II) there exists a finite history $v_1 a_2 v_2 \dots a_n v_n$ in G such that:

 1. $n = 1$, if $v \in V_0$
 2. $v_1 = w''$, $v_n = w'$ and $v_l \sim^V v$ for all $1 \leq l \leq n$
 3. there are $p_1^A, \dots, p_{n-1}^A \in Q^A$ such that
 $(r^A, (v_1 a_2, 0), p_1^A), \dots, (p_{n-2}^A, (v_{n-1} a_n, 0), p_{n-1}^A), (p_{n-1}^A, (v_n b', 1), p^A) \in \Delta^A$
- $((w, b, q^A), (va, 0), (w', b', p^A)) \in \Delta \quad :\iff$
 $f_b(w) = v$ and $(q^A, (w' b', 0), p^A) \in \Delta^A$

It is not hard to see, that $L(\mathcal{B}) = \{\pi^{\wedge}\alpha_t \mid \pi, t \models \neg K\psi\}$. (Notice that by construction, \mathcal{B} accepts only words of the form $\pi^{\wedge}\alpha_t$ for some $\pi \in \text{VA}^\omega$ and some $t < \omega$.) Moreover, a similar reasoning as in the proof of Proposition 10 shows, that \mathcal{B} can be constructed effectively from \mathcal{A} and \mathcal{G} : All conditions determining the membership of a transition to Δ are regular properties of finite histories in G and can thus be transformed into a finite automaton. The existence of such a finite history can then be checked using the nonemptiness test for such automata.

Finally, using the usual translation of Büchi-automata into formulas, we obtain an S1S-formula $\vartheta(x)$ such that for all plays $\pi \in (\text{VA})^\omega$ and all $t < \omega$ we have $\underline{\pi} \models \vartheta(t)$ if, and only if, $\pi^{\wedge}\alpha_t \in L(\mathcal{B})$, that means, $\underline{\pi} \models \neg\vartheta(t)$ if, and only if, $\pi, t \models K\psi = \varphi$, giving us the desired formula $\hat{\varphi}(x) = \neg\vartheta(x)$. \square

Uniform definability. Notice that this result does *not* yield *uniform* S1S-definability of ETL-specifications: If φ is an ETL-formula over unary predicates $(P_a)_{a \in \Sigma}$ for some finite set Σ , then Theorem 12 does not give us a monadic second order sentence $\hat{\varphi}$ over the signature $\mathcal{T}_\Sigma = \{S, \min, <, (P_a)_{a \in \Sigma}\}$, such that for *any* interpreted game model \mathcal{G} we have $L_{\mathcal{G}}(\varphi) = L(\hat{\varphi})$. Even more, it is obvious, that such a formula $\hat{\varphi}$ does not exist in general, since the semantics of S1S is completely determined by the word models $\underline{\pi}$ and does not additionally access the underlying interpreted game model \mathcal{G} . So if the alphabet Σ over which the word models are defined must be independent of \mathcal{G} , there is no way to define the knowledge operator K :

Consider the ETL-formula $\varphi = \text{KFP}$ over $\text{Prop} = \{P\}$ which expresses that player 0 knows that at some point in the future (including now), P will hold and consider the interpreted game model \mathcal{G} depicted in Figure 1. Moreover, for $i = 1, 2, 3$ let $\pi_i = v_i(aw_i)^\omega$, Obviously, $\mathcal{G}, \pi_1, 0 \models \varphi$ but $\mathcal{G}, \pi_2, 0 \not\models \varphi$, since $\pi_3(\leq 0) \sim^* \pi_2(\leq 0)$ and $\mathcal{G}, \pi_3, 0 \not\models \text{FP}$. On the other hand, for any S1S-sentence φ over the signature $\tau = \{S, \min, <, P\}$ we have $\underline{\pi}_1 \models \varphi$ if, and only if $\underline{\pi}_2 \models \varphi$: since $\underline{\pi}_1$ and $\underline{\pi}_2$ are isomorphic τ -structures, they cannot be distinguished by any S1S(τ)-sentence.

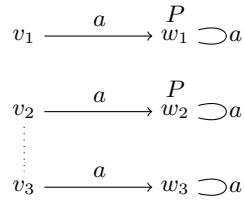


Fig. 1. An interpreted game model with two “isomorphic” plays

4.1 Remarks on Multiplayer Games

The cooperation problem for multiplayer games with partial information is the following decision problem:

Given an n -player game $\mathcal{G} = (G, (\sim_i^V), (\sim_i^A))$, $G = (V, (V_i), (f_a)_{a \in A}, W)$ with partial information, is there, for any $i = 0, \dots, n-1$ a \sim_i^* -strategy f_i for player i such that any play which is consistent with all f_i is in W ?

We call a tuple $f = (f_i)_{i=0, \dots, n-1}$ of \sim_i^* -strategies f_i a joint strategy for the coalition. Notice that all the information about the environment (player n) is implicit in the game model and that we assume the environment to have full information. In [16] it is shown, that this problem is undecidable for three player games where player 1 has *full* information (i.e., only player 0 has actually partial information) and the winning condition is given by an ETL-formula.

However, the question that [16] actually ask is the following: Given a game \mathcal{G} with partial information and some ETL-specification φ , is there a joint strategy f for the coalition such that $\mathcal{R}(\mathcal{G}, f) \models \varphi$, where $\mathcal{R}(\mathcal{G}, f)$ is the unravelling of \mathcal{G} with respect to f . There, evaluation of the knowledge operators is relative to plays which are consistent with f .

The proof in [16] for the case of two players, 0 and 1, makes heavy use of the underlying assumption that player 0 *knows* the strategy f_1 which player 1 uses, i.e., evaluation of K_0 is relative to histories which are consistent with *both* f_0 and f_1 . But f_1 is yet to be synthesized and histories which are consistent with f_1 and those which are not may very well be indistinguishable for player 0, which makes the problem undecidable.

If we assume that player 0 is completely ignorant about the strategy of player 1, then evaluation of K_0 is relative only to plays which are consistent with f_0 . So, by Theorem 12, the given ETL-specification is ω -regular. This generalizes to the case of n cooperating players and ETL-formulas with knowledge operators K_i for any $i \in \{0, \dots, n-1\}$, where each player i is completely ignorant about the strategy of any other player $j \neq i$. (The proof of Theorem 12 works for ETL-formulas with knowledge operators K_0, \dots, K_{n-1} without any modifications). Hence, in this case ETL-specifications can be transformed into nondeterministic Büchi-automata which can be transformed into deterministic parity automata (cf. [5]) which finally allows to transform the given game into a parity game. Moreover, parity conditions are LTL-definable and for LTL-specifications [16] proves that the cooperation problem is decidable for *synchronous hierarchical* games.

Thereby, a game $\mathcal{G} = (G, (\sim_i^V), (\sim_i^A))$ is called hierarchical, if $\sim_0^V \subseteq \sim_1^V \subseteq \dots \subseteq \sim_{n-1}^V$ and $\sim_0^A \subseteq \sim_1^A \subseteq \dots \subseteq \sim_{n-1}^A$. Notice that for the reduction of the given game to a parity game, the game graph has to be changed in general. Nevertheless, if the original game has hierarchical knowledge among the cooperating players, this hierarchy of knowledge is preserved by the reduction. So Theorem 12 shows that modeling the cooperation problem for ETL-specifications under the assumption that cooperating players are ignorant about each others strategies, turns the problem decidable for synchronous hierarchical games.

References

1. D. Berwanger et al. Strategy Construction for Parity Games with Imperfect Information. In *CONCUR '08*, pages 325–339. Springer, 2008.
2. F. Cassez et al. Timed Control with Observation Based and Stuttering Invariant Strategies. In *ATVA '07*, pages 192–206. Springer, 2007.
3. K. Chatterjee et al. Algorithms for Omega-Regular Games with Imperfect Information. *Logical Methods in Computer Science*, 3(3), 2007.
4. R. Fagin et al. *Reasoning About Knowledge*. The MIT Press, 2003.
5. E. Grädel, W. Thomas, and T. Wilke, editors. *Automata, Logics, and Infinite Games*, volume 2500 of *Lecture Notes in Computer Science*. Springer Verlag, 2002.
6. J. Y. Halpern, R. van der Meyden, and M. Y. Vardi. Complete Axiomatizations for Reasoning about Knowledge and Time. *SIAM J. on Comp.*, 33:674–703, 2004.
7. M. Jurdiński. *Games for Verification: Algorithmic Issues*. PhD thesis, University of Aarhus, 2000.
8. O. Kupferman and M. Y. Vardi. Church’s Problem Revisited. *The Bulletin of Symbolic Logic*, 5(2), 1999.
9. A. Pnueli and R. Rosner. On the Synthesis of an Asynchronous Reactive Module. In *ICALP '89*, pages 652–671. Springer-Verlag New York, Inc., 1989.
10. B. Puchala. Infinite Two Player Games with Partial Information: Logic and Algorithms. Diploma Thesis, RWTH Aachen, 2008.
11. B. Puchala. Asynchronous Omega-Regular Games with Partial Information. In *Proceedings of the 35th International Symposium on Mathematical Foundations of Computer Science 2010, MFCS '10*, volume 6281 of *LNCS*, pages 592–603. Springer, 2010.
12. J. H. Reif. The Complexity of Two-player Games of Incomplete Information. *Journal of Computer and System Sciences*, 29:274–301, 1984.
13. W. Thomas. Automata on Infinite Objects. In *Handbook of Theoretical Computer Science, volume B*, pages 133–192. Elsevier Science Publishers, 1990.
14. W. Thomas. On the Synthesis of Strategies in Infinite Games. In *STACS '95*, pages 1–13. Springer, 1995.
15. R. van der Meyden and M. Y. Vardi. Synthesis from Knowledge-Based Specifications (Extended Abstract). In *CONCUR '98*, pages 34–49. Springer, 1998.
16. R. van der Meyden and T. Wilke. Synthesis of Distributed Systems from Knowledge-based Specifications. In *CONCUR '05*, pages 562–576. Springer, 2005.
17. H. Wong-Toi and D. L. Dill. Synthesizing Processes and Schedulers from Temporal Specifications. In *CAV '91*, pages 272–281. Springer, 1991.