$2^{1/2}$ -PLAYER GENERALIZED REACTIVITY (1) GAMES*

Natalia Rodríguez¹, Victor Braberman¹, Nicolás D'Ippolito¹ and Sebastián Uchitel^{1,2}

Abstract—We introduce a new class of $2^{1/2}$ -player games, the $2^{1/2}$ -player GR(1) games, that allows for solving problems of stochastic nature by adding a probabilistic component to simple 2-player GR(1) games. Further, we present an efficient approach for solving qualitative $2^{1/2}$ -player GR(1) games with polynomial-time complexity. Our approach is based on a reduction from $2^{1/2}$ -player GR(1) games to 2-player GR(1) games that allows for solving the game and constructing, from a sure winning strategy for player $\square$ (resp. $\diamond$) in a 2-player GR(1) game, an almost-sure (resp. positively) winning strategy for its corresponding $2^{1/2}$ -player GR(1) game. Key to the effectiveness of the proposed approach is the fact that the reduction generates a 2-player game that is linearly larger than the original $2^{1/2}$ -player game, more precisely, it is linear with respect to the number of probabilistic states in the $2^{1/2}$ -player GR(1) game.

I. INTRODUCTION

Given an environment model and a goal specification, the reactive synthesis problem is to automatically generate a behaviour description, i.e. a controller, such that the controlled environment satisfies the specified goal. The problem of automatically generating controllers that guarantee the goals under assumptions about the behavior of the environment is tackled by extracting the controller from a strategy in a 2-player game modeling the situation and assuming that the controller and the environment are adversaries.

2-player games are turn-based zero-sum perfect information games played by two players (namely $\square$ and $\diamond$) in a finite directed game graph (S, E) . The game starts in a distinguished state of the graph. Then $\square$ and $\diamond$ make movements along the edges, constructing an infinite sequence of states: a play. We restrict our attention to games with objectives (or winning conditions) expressed by arbitrary linear-time temporal logic formulas (or ω -regular conditions) [37].

Although 2-player games have shown to be applicable in many domains, in some cases there is a stochastic component in the problem in which case $2^{1/2}$ -player games are a natural fit. $2^{1/2}$ -player games are 2-player games with an extra player that behaves probabilistically.

A general way of specifying objectives of players in non-terminating zero-sum games is by providing the set of ω -regular winning plays $\mathcal{W}$ for player $\square$. In 2-player games we consider the problem of finding a *sure* winning strategy

from the initial state for $\square$ such that any play formed by applying such a strategy and any possible strategy for $\diamond$ belong to $\mathcal{W}$. In contrast, in $2^{1/2}$ -player games, we restrict our attention to the problem of deciding whether player $\square$ has an *almost-sure* winning strategy to beat her opponent (i.e. whether $\square$ can ensure that against all strategies of the opponent a play consistent with her decisions belongs to $\mathcal{W}$ with probability 1 [8]). Such a decision problem is known as the *qualitative problem*.

Even though $2^{1/2}$ -player games with ω -regular winning conditions allow for very expressive problem specifications, they lead to complex algorithms compromising their practical application. The problem of finding a qualitative solution in $2^{1/2}$ -player games with winning conditions like Rabin, Streett, Muller or Parity are known to be P-complete, coNP-complete, PSPACE-complete and $\text{NP} \cap \text{coNP}$ respectively [7]. There exists some ω -regular subsets, like safety and reachability, known to allow for polynomial-time qualitative solutions in $2^{1/2}$ -player games, however such winning conditions are still very limiting in practical terms. We propose something in between: A winning condition (subset of ω -regular conditions) with efficient qualitative solutions in $2^{1/2}$ -player games but more expressive than safety and reachability allowing to specify realistic systems.

In 2-player games conditions that allow for polynomial-time solutions but with higher expressiveness than safety or reachability conditions have been studied. One of such cases is that of Generalized Reactivity (1) (GR(1)) conditions [3] for which a qualitative solution can be computed in PTIME. An outstanding characteristic of GR(1) is that it allows not only describing liveness goals that the controller is to achieve but also describing liveness assumptions on the environment that the controller can rely on to achieve its goals. It is well understood (e.g. [20]) that assumptions play an important role in software specifications.

GR(1) games have been applied in numerous settings such as hardware design [4], high-level strategies for self-adaptive systems [15], [5], robot-controller planning [11], [21], [24], [25], [40] and user programming [26], [27]. In addition, many tools for solving GR(1) games have been developed [22], [38], [2], [16] showing that there is real interest in the application of games with GR(1) winning conditions. However, to the best of our knowledge, in $2^{1/2}$ -player games the research for this kind of subsets (i.e. particular cases that allow for efficient solutions but expressive enough from a practical point of view) has been scarcely explored. Furthermore, to the best of our knowledge, there is no work relating explicitly GR(1) conditions with $2^{1/2}$ -player games.

In this work, we introduce a new class of $2^{1/2}$ -player

*This work was partially supported by grants ANPCYT PICT 2012-0724, ANPCYT PICT 2013- 2341, ANPCYT PICT 2014-1656, ANPCYT PICT2015-0638, ANPCYT PICT2015-1718, UBACYT 2014 N°0384BA, UBACYT2014 N°0036BA, CONICET PIP2012N°0596CO, CONICET PIP2014N°0688CO, CONICET PIP2015N°0931CO.

¹Departamento de Computación, Facultad de Ciencias Exactas y Naturales, Universidad de Buenos Aires, Argentina ² Department of Computing Imperial College, London, UK nrodriguez@dc.uba.ar vbraber@dc.uba.ar ndippolito@dc.uba.ar suchitel@dc.uba.ar

games, the $2^{1/2}$ -player GR(1) games, that allows solving problems of stochastic nature by adding a probabilistic component to simple 2-player GR(1) games. Further, we present an efficient approach for solving $2^{1/2}$ -player GR(1) games with polynomial-time complexity. Our approach is based on a reduction from $2^{1/2}$ -player GR(1) games to 2-player GR(1) games that allows solving the game and constructing, from a sure winning strategy for $\square$ (resp. $\diamond$) in a 2-player GR(1) game, an almost-sure (resp. positively) winning strategy for its corresponding in $2^{1/2}$ -player GR(1) game.

The paper is organized as follows. Section II includes the necessary background. Section III presents a formal definition of $2^{1/2}$ -player GR(1) games. Section IV includes a proposal of an efficient (i.e. polynomial-time) solution for solving qualitatively $2^{1/2}$ -player GR(1) games. This section presents the formal definition of a reduction from $2^{1/2}$ -player GR(1) games to 2-player GR(1) games and the derived algorithm for constructing strategies as well as a proof of correctness of the approach and its complexity. Finally, we report on the corresponding related work and conclusions.

II. PRELIMINARIES

We consider *simple $2^{1/2}$ -player games*. That is non-terminating turn-based zero-sum perfect-information finite state games played over a graph (S, E) between three players, $\square$, $\diamond$ and $\circ$, who move a token from state to state along edges into a the graph so that an infinite path is formed. A zero-sum game is one in which, every situation that makes one participant win is equivalent to making the others lose. In perfect-information games any participant is supposed to have all the required information to force a decision to make the game move.

Definition II.1 [$2^{1/2}$ -player game graph] A $2^{1/2}$ -player game graph is a tuple $G = ((S, E), (S_\square, S_\diamond, S_\circ))$ where (S, E) is a finite directed graph with state set S , edge set E and, $(S_\square, S_\diamond, S_\circ)$ is a partition of S .

The partition of states in S is used to define the *turns* of each player in the game. $\square$ moves along the edges of G from states in $S_\square$ and $\diamond$ from those in $S_\diamond$. Both players have full freedom of movement when it is their turn. Player $\circ$, on the other hand, moves by choosing a successor state from states in $S_\circ$ accordingly to a probabilistic distribution. For simplicity and w.l.o.g, we consider this distribution to be uniformly random. Furthermore, although player $\circ$ is part of the game, he does not look forward to winning or losing. Due to its limited freedom of choice and lack of winning condition we refer to player $\circ$ as a “half player”.

Definition II.2 [$2^{1/2}$ -player game] A $2^{1/2}$ -player game G is a non-terminating turn-based zero-sum perfect-information game played on a game graph $G = ((S, E), (S_\square, S_\diamond, S_\circ))$ by three players: $\square$, $\diamond$ and $\circ$. In G , $\square$ and $\diamond$ chooses the successor of states in $S_\square$ and $S_\diamond$ respectively; states in $S_\circ$ are $\circ$ ’s states and it always moves by choosing a successor uniformly at random.

The special case where $S_\circ = \emptyset$ corresponds to 2-player game graphs.

Definition II.3 [*2-player game*] A 2-player game is a non-terminating turn-based zero-sum perfect-information game G played on the game arena $G = ((S, E), (S_\square, S_\diamond))$.

Plays. A play $\rho = s_0, s_1, s_2, \dots$ over a game graph G is an infinite sequence of states in S such that, for all $i \geq 0$, $(s_i, s_{i+1}) \in E$. We write Ω for the set of all plays, Ω_s for the set of all plays starting from the state s , and $\Delta_G(s)$ the set of all states s' such that $(s, s') \in E$. We denote $\rho[i]$ the i -th state in a play ρ . We say that $s_i \leadsto_G s_j$ if there is a play ρ over the game G such that, for some $i \leq j$, $\rho[i] = s_i$ and $\rho[j] = s_j$. And, if $j = i + 1$ we write $s_i \rightarrow_G s_j$.

Strategies. The choices of players are formalized in the form of *strategies*. In other words, a strategy is the policy that each player applies to move along the game graph. In general, the strategy that a player *uses* or *follows* in the game may depend or not on the history of a play.

A *finite-memory* strategy σ for $\square$ can be encoded by a deterministic transducer $\langle \mathcal{M}, m_0, \sigma_u, \sigma_t \rangle$ where *i*) $\mathcal{M}$ is a finite set representing the memory of the strategy, *ii*) $m_0 \in \mathcal{M}$ is the initial memory value, *iii*) $\sigma_t : S_\square \times \mathcal{M} \rightarrow S$ defines a partial transition function, and *iv*) $\sigma_u : S \times \mathcal{M} \rightarrow \mathcal{M}$ represents how the memory is updated by σ . Player $\square$ follows (or uses) a strategy σ if in each move, given a current value of memory m at $s \in S_\square$ she chooses as successor the state $\sigma_t(s, m)$ and the memory is updated according to $\sigma_u(s, m)$ for every $s \in S$. A play $\rho = s_0, s_1, \dots$ is consistent with a strategy σ if $\sigma_t(s_i, m_i) = s_{i+1}$, where $m_{i+1} = \sigma_u(s_{i+1}, m_i)$. Strategies for $\diamond$ are defined analogously. We write Σ and Π for the set of all strategies for $\square$ and $\diamond$.

W.l.o.g we consider memory values tracking goals for $\square$ and assumptions for $\diamond$ in lexicographical order.

Once strategies $\sigma \in \Sigma$ and $\pi \in \Pi$ are fixed, a play starting in state $s \in S$ is determined by a random path (denoted $\rho_s^{\sigma, \pi}$), consistent with σ and π , for which the probabilities of events are uniquely defined, where an *event* $\mathcal{E} \subseteq \Omega$ is a measurable set of paths. We define $G_{\sigma, \pi}$ as the sub-graph obtained by pruning G according to the decisions made by strategies σ and π . We write $Pr_s^{\sigma, \pi}[\mathcal{E}]$ for the probability that a path starting on state s in $G_{\sigma, \pi}$ belongs to the event $\mathcal{E}$.

Once strategies σ or π are fixed a Markov Decision Process (MDP) is obtained. A set $\mathcal{U}$ is an *end component* in such MDP if the sub-graph induced by $\mathcal{U}$ is strongly connected, and for each probabilistic states in $\mathcal{U}$ every outgoing edges end up in $\mathcal{U}$ (i.e., $\mathcal{U}$ is closed for probabilistic states). The key property about MDPs that is used in our proofs is a result established by [12], [13]: given an MDP, for every strategy, with probability 1 the set of states visited infinitely often is an end component.

Winning conditions. Winning conditions of players in non-terminating $2^{1/2}$ -player games are specified by a set of *winning plays* $\mathcal{W} \subseteq \Omega$ for a player. As we consider zero-sum games if the winning condition of a player is the set $\mathcal{W}$ then the winning condition for the other player is $\Omega \setminus \mathcal{W}$. Given a

game graph G and a winning condition $\mathcal{W}$, we write $G(\mathcal{W})$ for the game played on G with winning condition $\mathcal{W}$ for $\square$.

We focus in Generalized Reactivity (1) (GR(1)) conditions [3]. Let ρ be a play in G and $g_1, \dots, g_n$ and $a_1, \dots, a_m$ be subsets of S . Let $\text{inf}(\rho)$ denote the set of states that occur infinitely often in ρ . A play ρ is in the GR(1) condition $gr((g_1, \dots, g_n), (a_1, \dots, a_m))$ if either for some i we have $\text{inf}(\rho) \cap a_i = \emptyset$ or for every j we have $\text{inf}(\rho) \cap g_j \neq \emptyset$.

A practical interpretation of GR(1) conditions is letting a_1 through a_m be the assumptions on the environment and g_1 through g_n be the guarantees that the system has to fulfill. If the environment ($\diamond$) behaves as expected by satisfying always its progress assumptions then goals must be guaranteed by $\square$'s strategy to win (i.e., the underlying controller will always eventually progress towards its goals as long as the environment is live).

A 2-player game with a Generalized Reactivity (1) condition is called 2-player GR(1) game. Formally:

Definition II.4 [2-player GR(1) game] Let $G = ((S, E), (S_\square, S_\diamond))$ be a game graph, and $\mathcal{W}$ be a winning condition. We say that $G(\mathcal{W})$ is a GR(1) game iff $\mathcal{W} = gr((g_1, \dots, g_n), (a_1, \dots, a_m))$, with $g_1, \dots, g_n$ and $a_1, \dots, a_m$ subsets of S .

A winning strategy for $\square$ in a 2-player GR(1) games can be computed in $O(mnS^2)$ [35] where m and n stand for number of assumptions and goals respectively and, S is the set of states in the game graph.

Qualitatively winning modes. [23] In this work, we focus on the *qualitative* decision problem. The *qualitative* solution of a game amounts to computing the set of states from which a player has the possibility to win.

Let $G(\mathcal{W})$ be a $2^{1/2}$ -player game, we say that a strategy $\sigma \in \Sigma$ is:

- *sure winning* for $\square$ in the game $G(\mathcal{W})$ if, for all $\pi \in \Pi$ we have $\rho^{\sigma, \pi} \in \mathcal{W}$,
- *almost-sure winning* for $\square$ in the game $G(\mathcal{W})$ if, for all $\pi \in \Pi$ we have $Pr^{\sigma, \pi}[\mathcal{W}] = 1$,
- *positive-probability winning* for $\square$ in the game $G(\mathcal{W})$ if, for all $\pi \in \Pi$ we have $Pr^{\sigma, \pi}[\mathcal{W}] > 0$.

Note that in 2-player games players can only win surely.

Determinacy. Let $\mu \in \{s, a, p\}$ be a winning mode, where “s” stands for sure win, “a” for almost-sure win and “p” for positive-probability win. We say that a game $G(\mathcal{W})$ is μ -determined if for all state in S , either $\square$ can μ -win or $\diamond$ can $\tilde{\mu}$ -win from such state, where $\tilde{s} = s$, $\tilde{a} = p$ and $\tilde{p} = a$.

In particular, every 2-player game with GR(1) winning condition for player $\square$ (2-player GR(1) game) is s-determined (or simply determined) [3]. Thus, the set of states in a 2-player GR(1) game graph is partitioned in those from which $\square$ has a sure winning strategy (i.e. $\overline{W}_\square$) are those from which $\diamond$ has a sure winning strategy ($\overline{W}_\diamond$).

A proof that $2^{1/2}$ -player GR(1) games are a-determined can be derived from [32].

III. $2^{1/2}$ -PLAYER GAMES

In this section we introduce a new class of $2^{1/2}$ -player games.

Definition III.1 [$2^{1/2}$ -player GR(1) game] Let $G(\mathcal{W})$ be a $2^{1/2}$ -player game with winning condition $\mathcal{W}$ for $\square$. We say that G is a $2^{1/2}$ -player GR(1) game iff $\mathcal{W}$ is a Generalized Reactivity (1) winning condition.

More specifically, a $2^{1/2}$ -player GR(1) game $G(\mathcal{W})$ is a turn-based zero-sum perfect-information game played by three players: $\square$, $\diamond$ and $\circ$ over a game graph $G = ((S, E), (S_\square, S_\diamond, S_\circ))$ with a winning condition $\mathcal{W} = gr((g_1, \dots, g_n), (a_1, \dots, a_m))$ for $\square$ where $g_1, \dots, g_n$ and $a_1, \dots, a_m$ denote subsets of S , and $\mathcal{W} = gr((g_1, \dots, g_n), (a_1, \dots, a_m))$ denotes the set of infinite sequences ρ such that either for some i we have $\text{inf}(\rho) \cap a_i = \emptyset$ or for all j we have $\text{inf}(\rho) \cap g_j \neq \emptyset$.

For simplicity and w.l.o.g we assume that each state in the graph has at least one and at most two outgoing edges and that player $\circ$ moves from states in $S_\circ$ by choosing a successor uniformly at random.

IV. SOLVING QUALITATIVELY $2^{1/2}$ -PLAYER GR(1) GAMES

We present an efficient qualitative solution for $2^{1/2}$ -player GR(1) games that allows low polynomial-time complexity.

A. Reduction

The notion of reduction between games is an important aspect in the study of games as it allows to understand which classes of games are subsumed by others. The main result of this section is a proposal for solving $2^{1/2}$ -player GR(1) games via an efficient reduction to 2-player GR(1) games. We translate a $2^{1/2}$ -player game to a 2-player GR(1) game where we know the solution has low polynomial-time complexity. *Key to the effectiveness of the proposed approach is the fact that the reduction generates a 2-player game that is linearly larger than the original $2^{1/2}$ -player game, more precisely, it is linear with respect to the number of random states.* The reduction abstracts away probabilistic choices by allowing the other players to make the choice while balancing the power of each player.

Definition IV.1 [$2^{1/2}$ -player GR(1) to 2-player GR(1) games reduction] Given a $2^{1/2}$ -player GR(1) game $G = ((S, E), (S_\square, S_\diamond, S_\circ))$ with winning condition $\mathcal{W} = gr(\mathcal{G}, \mathcal{A})$ where $\mathcal{G} = (g_1, \dots, g_n)$ and $\mathcal{A} = (a_1, \dots, a_m)$ with $g_1, \dots, g_n$ and $a_1, \dots, a_m$ subsets of S , we define the 2-player GR(1) game $\overline{G} = ((\overline{S}, \overline{E}), (\overline{S}_\square, \overline{S}_\diamond))$ with winning condition $\overline{\mathcal{W}}$ as follows:

$$\begin{aligned} \overline{S}_\square &= S_\square \cup \{s_\square^{1,1}, s_\square^{2,1}, s_\square^{2,2} \mid s \in S_\circ\} \\ \overline{S}_\diamond &= S_\diamond \cup \{s, s_\diamond^{1,2}, s_\diamond^{2,3} \mid s \in S_\circ\} \\ \overline{S} &= \overline{S}_\square \cup \overline{S}_\diamond \\ \overline{E} &= \begin{cases} \{(s, s') \mid s \in S_\square \cup S_\diamond, (s, s') \in E\} \cup \\ \{(s, s_\square^{1,1}), (s_\square^{1,1}, s_\diamond^{1,2}) \mid s \in S_\circ\} \cup \\ \{(s, s_\square^{2,1}), (s_\square^{2,1}, s_\diamond^{2,2}), (s_\square^{2,1}, s_\diamond^{2,3}) \mid s \in S_\circ\} \cup \\ \{(s_\diamond^{1,2}, s'), (s_\diamond^{2,2}, s'), (s_\diamond^{2,3}, s') \mid s \in S_\circ, s' \in \Delta_G(s)\} \end{cases} \\ \overline{\mathcal{W}} &= gr(\overline{\mathcal{A}}, \overline{\mathcal{G}}) \text{ where} \end{aligned}$$

Fig. 1. 3-step game construction for a random state in G .

- $\bar{\mathcal{A}} = \{\bar{a}_1, \dots, \bar{a}_n \mid \bar{a}_i = a_i \cup \{s_{\square}^{2,2} \mid s \in S_{\diamond}\}\}$
- $\bar{\mathcal{G}} = \{\bar{g}_1, \dots, \bar{g}_n \mid \bar{g}_j = g_j \cup \{s_{\diamond}^{1,2} \mid s \in S_{\diamond}\}\}$

For every state $s \in S_{\square} \cup S_{\diamond}$ there is a state $s \in \bar{S}_{\square} \cup \bar{S}_{\diamond}$ with “the same” outgoing edges in G and $\bar{G}$. Random states $s \in S_{\diamond}$ together with their transition relation in G are substituted in $\bar{G}$ by the construction exemplified in Figure 1. Such a construction can be interpreted as a 3-step game where in each step players have the following options:

Step 1: $\diamond$'s turn. At state s in $\bar{G}$, $\diamond$ chooses a successor between 2 possibilities $s_{\square}^{1,1}$ or $s_{\square}^{2,1}$ yielding the control to $\square$.
Step 2: $\square$'s turn. At state $s_{\square}^{1,1}$ $\square$ can only choose $s_{\diamond}^{1,2}$ as successor. Then, from $s_{\diamond}^{1,2}$ $\diamond$ gets to choose the successor. Note that since $s_{\diamond}^{1,2}$ is added to all sets in $\bar{\mathcal{G}}$, all goals are satisfied in $s_{\diamond}^{1,2}$.

At state $s_{\square}^{2,1}$ $\square$ plays, and can make one of two choices:

- Choose $s_{\diamond}^{2,2}$ as successor. In such a case it is her turn again but since $s_{\diamond}^{2,2}$ is added to all sets in $\bar{\mathcal{A}}$, all assumptions are satisfied in $s_{\diamond}^{2,2}$.
- Choose $s_{\diamond}^{2,3}$ as successor. In this case is $\diamond$'s turn, and the winning condition $\bar{\mathcal{W}}$ is not affected by $s_{\diamond}^{2,2}$.

Step 3: mixed. In states $s_{\diamond}^{1,2}$, $s_{\diamond}^{2,2}$ and $s_{\diamond}^{2,3}$ the possible successors are the s' such that $(s, s') \in E$.

The 3-step game proposed abstracts away the probabilities while balancing the power of each player in the game $\bar{G}$.

The token belongs first to $\diamond$ who decides the first successor of s in $\bar{G}$. If he wants to be sure that he will get to choose in $\bar{G}$ between s' and s'' , then he must choose $s_{\square}^{1,1}$, which lets $\diamond$ choose his successor but leads to $s_{\diamond}^{1,2}$ that satisfies the goal. Here $\diamond$ pays the price of choosing the successor by adding a state that satisfies every $\bar{g}_j$. $\diamond$'s decision to visit the state can be winning for him if for instance, at some point, from s' he has a strategy to either avoid visiting s or stop choosing $s_{\diamond}^{1,2}$ (possibly depending on some memory value).

On the other hand, if $\diamond$ chooses $s_{\square}^{2,1}$, then $\square$ gets to play. Now $\square$ has two options: $s_{\diamond}^{2,2}$ and $s_{\diamond}^{2,3}$. Choosing $s_{\diamond}^{2,2}$ gives $\square$ the power to choose between s' and s'' but paying the price of choosing a state that satisfies every $\bar{a}_i$. She won't decide to visit infinitely often $s_{\diamond}^{2,2}$ if, for instance, she wins by avoiding some of the assumptions. $\square$ must move the token to $s_{\diamond}^{2,3}$, from which $\diamond$ decides the outcome of the game.

B. Correctness

The reduction from $2^{1/2}$ -player GR(1) games to 2-player GR(1) games presented in Definition IV.1 allows for quali-

tatively solving a $2^{1/2}$ -player game efficiently by computing a solution to a 2-player one. More specifically, based on a sure winning strategy for $\square$ (resp. $\diamond$) found in the 2-player GR(1) game, an almost-sure (resp. positive-probability) winning strategy for its corresponding in the $2^{1/2}$ -player GR(1) game can be constructed. In this section, we define such a construction and we present a proof of its correctness.

Definition IV.2 [*2-player GR(1) to $2^{1/2}$ -player GR(1) games solution*] Let $\bar{\sigma}$ (resp. $\bar{\pi}$) be a sure winning strategy for $\square$ (resp. $\diamond$) in $\bar{G}$ then a strategy σ (resp. π) for $\square$ (resp. $\diamond$) in G can be constructed as follows:

- For every pair (s, m) :
 $\sigma_t(s, m) = \bar{\sigma}_t(s, m)$, for every $s \in S_{\square}$ and,
 $\sigma_u(s, m) = \begin{cases} \bar{\sigma}_u(s, m) & s \in S_{\square} \cup S_{\diamond} \\ m & s \in S_{\diamond} \end{cases}$
- Analogously, for every pair (s, m) :
 $\pi_t(s, m) = \bar{\pi}_t(s, m)$, for every $s \in S_{\diamond}$ and,
 $\pi_u(s, m) = \begin{cases} \bar{\pi}_u(s, m) & s \in S_{\square} \cup S_{\diamond} \\ m & s \in S_{\diamond} \end{cases}$

W.l.o.g our approach relies on the existence of a sure winning strategy that propagates the memory through the reduction. The existence of such a strategy is warranted with the existence of a generic winning one by considering memory values tracking visited goals (resp. assumptions) for $\square$ (resp. $\diamond$) in lexicographical order. Even more, such assumption does not compromise the correctness of the proposed technique but it may implies a slight adaptation of the algorithm applied to solve the 2-player game constructed.

We now present a proof sketch of correctness for the proposed solution mapping from Definition IV.2.

Theorem 1 [*Correctness*] Given a $2^{1/2}$ -player GR(1) game G and a 2-player GR(1) game $\bar{G}$. Let $\bar{\sigma}$ (resp. $\bar{\pi}$) be a sure winning strategy for $\square$ (resp. $\diamond$) in $\bar{G}$ and σ (resp. π) the corresponding ones in G constructed as in Definition IV.2. If $\bar{G}$ is the game obtained by applying the reduction in Definition IV.1 to G , the following holds:

For every state s in a $2^{1/2}$ -player GR(1) game G , σ (resp. π) is an almost-sure (resp. positive-probability) winning strategy from s in G , if and only if $\bar{\sigma}$ (resp. $\bar{\pi}$) is a sure winning strategy from state s in $\bar{G}$.

Proof: [Sketch] ($\Leftarrow$) For $\square$. The proof proceeds by contradiction assuming that there exists a positive-probability winning strategy $\pi \in \Pi$ for $\diamond$. We argue that if such a π exists then, it is possible to define a $\bar{\pi} \in \bar{\Pi}$ adequately that makes a “mimic” of π decisions in G so that a path leading to “the same” loosing end component is formed in $\bar{G}$ (which is a contradiction since $\bar{\sigma}$ was supposed to be a sure winning strategy in $\bar{G}$ from s).

($\Rightarrow$) For $\diamond$. The proof for $\diamond$ is very similar but, now assuming that exists an almost-sure strategy $\sigma \in \Sigma$ for $\square$ and defining adequately a $\bar{\sigma} \in \bar{\Sigma}$ leading to the contradiction. ■

Theorem 1 states that for every state s in a $2^{1/2}$ -player GR(1) game G , $\square$ (resp. $\diamond$) has an almost-sure (resp.

positive-probability) winning strategy from s in G , **then** if $\square$ (resp. $\diamond$) has a sure winning strategy from state s in $\bar{G}$. Proof of $\Rightarrow$ for both players follows from the fact that both games are determined.

Correctness states that the problem of finding a μ -winning strategy for a player in a GR(1) game G is equivalent to the problem of finding a sure winning for the same player in $\bar{G}$, constructed following Definition IV.1.

C. Algorithm

A simple algorithm can be derived directly from the definitions above by following the steps described below:

- 1) *Pre-processing input*: Given the $2^{1/2}$ -player GR(1) game G , build $\bar{G}$ by applying the reduction in Definition IV.1.
- 2) *Solving the 2-player GR(1) game*: Find a sure (memory preserving) winning strategy $\bar{\sigma}$ (resp. $\bar{\pi}$) for $\square$ (resp. $\diamond$) in $\bar{G}$ using some existing algorithm.
- 3) *Post-processing output*: Build an almost-sure σ (resp. positive-probability π) winning strategy for $\square$ (resp. $\diamond$) in G by applying the mapping in Definition IV.2.

The growth in the number of states of the reduction is linear (with constant 5) in the number of random states in G since for each state $s \in S_o$ we add five additional states in the construction of $\bar{G}$. Since $\bar{G}$ state space is at most five times larger than G and 2-player GR(1) games have polynomial-time complexity we show that qualitatively solving $2^{1/2}$ -player GR(1) games is also polynomial.

More precisely, the qualitative decision problem to a $2^{1/2}$ -player GR(1) game with the proposed solution is in $O(mnS^2)$ with m, n the number of assumptions and goals respectively, and S is the set of states in the game graph.

V. RELATED WORK

$2^{1/2}$ -player games have recently gained attention, mainly due to their possible applications and open problems. The complexity of solving $2^{1/2}$ -player games has been the center of attention in the literature, e.g. [10], [6], [9], [8], [7], [19], [39], [17]. Based on the known complexities for the standard ω -regular conditions [7], central to the applicability of $2^{1/2}$ -player games is to find efficient, yet expressive, settings that allow modeling interesting problems in practice.

Since the complexity of the synthesis problem with linear-time temporal logics [37] specifications was established as 2EXPTIME-complete [36], a number of games with restricted winning conditions have been explored [1], [30]. In [3], games with GR(1) conditions are introduced. GR(1) games allow for polynomial-time solutions and have been widely applied in practice [4], [26], [40]. By extending 2-player GR(1) games with a stochastic component introducing $2^{1/2}$ -player GR(1) games, our approach expands the applicability of 2-player GR(1) games in domains such as robotics and autonomous systems where stochastic models are a more natural modeling approach.

Central to our approach is the definition of a reduction. We are not the first in obtaining results in game theory by means of a reduction from one game to another. Examples of this approach are [8], [7], [23], [18]. In particular, reductions such

as those used for Parity [8] Rabin [7] and Muller [33] games may be adapted to efficiently solve qualitatively $2^{1/2}$ -player GR(1) games. One possible approach can be to combine the split tree [41] of the GR(1) condition with a reduction of a $2^{1/2}$ -player GR(1) game to a $2^{1/2}$ -player Parity of three priorities that are known to allow for polynomial-time solution [8]. Combining such results suggests a polynomial-time complexity result in the same order as ours. However, no previous work was found dealing *explicitly* with $2^{1/2}$ -player games. Furthermore, a naive combination of these results leads to a concrete complexity that is beyond the linear complexity (with constant 5) of the algorithm provided in this paper. In addition, the alternative reduction ends up in a Parity game while our proposal reduces to GR(1) games which are well studied and with several tools enabling applications to concrete problems in practice.

Another naive solution for trying to reduce the complexity of $2^{1/2}$ -player games can be to replace the stochastic component by non-determinism in 2-player games. However, this approach may result in oversimplified representations of the environment leading to synthesis problems where the environment is simply too powerful. For instance, in a reactive systems scenario where some of the system's components may fail, considering failures as purely adversarial (i.e. controlled by the environment) may lead to unrealizable problems even when solutions to such problems may exist. The setting presented in [14] solves this problem by considering a very restricted notion of failure and imposing fairness condition on failures. Our approach allows for a generalization of [14] permitting to naturally model failures with probabilistic behaviour but requires a slight adaptation of the standard GR(1) algorithm (see Section IV.C).

Supervisory control community has also shown interest in defining frameworks for dealing with plants and properties based on probabilistic languages (e.g. [29], [28], [34]) or stochastic process-theoretic approaches (e.g. [31]). Those approaches aim at building a (randomized) controller that try to match the behavior of a controlled plant with a given probabilistic specification language as a way to relax hard constraints imposed by the classical supervisory frameworks or optimize quality attributes. Note that our goal -of qualitative nature- is different: define a strategy that almost-sure satisfies progress specifications in the presence of, or resorting to, random behavior in the plant (the half player).

VI. CONCLUSIONS

2-player as well as $2^{1/2}$ -player games have several applications. We introduced an expressive family of $2^{1/2}$ -player games and presented a specific solution for qualitative decision problem, proven to be correct and efficient since it conforms to the theoretical complexity of the problem (PTIME). Our approach is based on a reduction from $2^{1/2}$ -player GR(1) games to 2-player GR(1) games that are known to have polynomial-time solution. We showed that the reduction preserves the number of non-random states and adds only five states for each random state, and that the exact complexity of solving qualitatively $2^{1/2}$ -player GR(1) games

with the proposed technique is $O(mnS^2)$ where m and n stand for number of assumptions and goals respectively, and S is the set of states in the game graph. In addition, we presented how to construct an strategy in the $2^{1/2}$ -player game from the corresponding one in the reduction.

REFERENCES

- [1] Alur, R., La Torre, S.: Deterministic generators and games for LTL fragments. *ACM Trans. Comput. Logic* 5(1), 1–25 (Jan 2004)
- [2] Bloem, R., Cimatti, A., Greimel, K., Hofferek, G., Könighofer, R., Roveri, M., Schuppan, V., Seeber, R.: Ratsy – a new requirements analysis tool with synthesis. In: *Procs, 22Nd Int. Conf. on Comp. Aided Verif.* pp. 425–429. CAV’10, Springer-Verlag (2010)
- [3] Bloem, R., Jobstmann, B., Piterman, N., Pnueli, A., Saar, Y.: Synthesis of reactive(1) designs. *J. Comput. Syst. Sci.* 78(3), 911–938 (2012)
- [4] Bloem, R.P., Galler, S., Jobstmann, B., Piterman, N., Pnueli, A., Weighofer, M.: Interactive presentation: Automatic hardware synthesis from specifications: a case study. In: *2007 Design, Automation and Test in Europe Conf. and Exposition April 16-20, Nice, France* (2007)
- [5] Braberman, V., D’Ippolito, N., Piterman, N., Sykes, D., Uchitel, S.: Controller synthesis: From modelling to enactment. In: *Procs, of the 2013 Int. Conf. on Soft. Eng.* pp. 1347–1350. ICSE ’13, IEEE Press, Piscataway, NJ, USA (2013)
- [6] Chatterjee, K., de Alfaro, L., Henzinger, T.A.: The Complexity of Stochastic Rabin and Street Games, pp. 878–890. Springer (2005)
- [7] Chatterjee, K., Henzinger, T.A.: A survey of stochastic ω -regular games. *J. Comput. Syst. Sci.* 78(2), 394–413 (Mar 2012)
- [8] Chatterjee, K., Jurdziński, M., Henzinger, T.A.: Simple Stochastic Parity Games, pp. 100–113. Springer Berlin Heidelberg (2003)
- [9] Chatterjee, K., Jurdziński, M., Henzinger, T.A.: Quantitative stochastic parity games. In: *Procs, of the Fifteenth Annual ACM-SIAM Symp. on Discrete Algorithms*. pp. 121–130. SODA ’04, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA (2004)
- [10] Condon, A.: The complexity of stochastic games. *Inf. Comput.* 96(2), 203–224 (Feb 1992)
- [11] Conner, D.C., Kress-Gazit, H., Choset, H., Rizzi, A.A., Pappas, G.J.: Valet parking without a valet. In: *2007 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, October 29 - November 2, 2007, San Diego, USA. pp. 572–577. IEEE (2007)
- [12] Courcoubetis, C., Yannakakis, M.: The complexity of probabilistic verification. *J. ACM* 42(4), 857–907 (Jul 1995)
- [13] De Alfaro, L.: Formal Verification of Probabilistic Systems. Ph.D. thesis, Stanford, CA, USA (1998), aAI9837082
- [14] D’Ippolito, N., Braberman, V., Piterman, N., Uchitel, S.: Synthesis of live behaviour models for fallible domains. In: *Procs, of the 33rd Int. Conf. on Soft. Eng.* pp. 211–220. ICSE ’11, ACM, USA (2011)
- [15] D’Ippolito, N., Braberman, V., Piterman, N., Uchitel, S.: Synthesizing nonanomalous event-based controllers for liveness goals. *ACM Trans. Softw. Eng. Methodol.* 22(1), 9:1–9:36 (Mar 2013)
- [16] D’Ippolito, N., Fischbein, D., Chechik, M., Uchitel, S.: MTSA: The modal transition system analyser. In: *Procs, of the 2008 23rd IEEE/ACM Int. Conf. on Automated Soft. Eng. ASE ’08*, IEEE Comp. Society, Washington, DC, USA (2008)
- [17] Etessami, K., Yannakakis, M.: Efficient qualitative analysis of classes of recursive markov decision processes and simple stochastic games. In: *Procs, of the 23rd Annual Conf. on Theoret. Aspects of Comp. Sci.* pp. 634–645. STACS’06, Springer-Verlag, Berlin, Heidelberg (2006)
- [18] Etessami, K., Yannakakis, M.: Recursive concurrent stochastic games. In: *Procs, 33rd Int. Conf. on Automata, Lang. and Program. - Vol. Part II*. pp. 324–335. ICALP’06, Springer-Verlag (2006)
- [19] Gimbert, H., Horn, F.: Simple stochastic games with few random vertices are easy to solve. In: *Procs, 11th Int. Conf. on Foundations of Soft. Sci. and Computational Structures*. pp. 5–19. FOS-SACS’08/ETAPS’08, Springer-Verlag, Berlin, Heidelberg (2008)
- [20] Jackson, M.: The world and the machine. In: *Procs, of the 17th Int. Conf. on Soft. Eng.* pp. 283–292. ICSE ’95, ACM, USA (1995)
- [21] Jing, G., Finucane, C., Raman, V., Kress-Gazit, H.: Correct high-level robot control from structured english. In: *Robotics and Automation (ICRA)*, 2012 IEEE Int. Conf. on. pp. 3543–3544 (May 2012)
- [22] Jobstmann, B., Galler, S., Weighofer, M., Bloem, R.: Anzu: A tool for property synthesis. In: *Procs, of the 19th Int. Conf. on Comp. Aided Verification*. pp. 258–262. CAV’07, Springer-Verlag, Berlin, Heidelberg (2007)
- [23] Jurdzinski, M., Kupferman, O., Henzinger, T.A.: Trading probability for fairness. In: *Procs, of the 16th Int. Workshop and 11th Annual Conf. of the EACSL on Comp. Sci. Logic*. pp. 292–305. CSL ’02, Springer-Verlag, London, UK, UK (2002)
- [24] Kress-Gazit, H., Fainekos, G.E., Pappas, G.J.: From structured english to robot motion. In: *2007 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. pp. 2717–2722 (Oct 2007)
- [25] Kress-Gazit, H., Fainekos, G.E., Pappas, G.J.: Where’s waldo? sensor-based temporal logic motion planning. In: *2007 IEEE Int. Conf. on Robotics and Automation, ICRA 2007*, 10-14 April 2007, Roma, Italy. pp. 3116–3121. IEEE (2007)
- [26] Kugler, H., Plock, C., Pnueli, A.: Controller synthesis from LSC requirements. In: *Procs, of the 12th Int. Conf. on Fundamental Approaches to Soft. Eng.: Held As Part of the Joint European Conferences on Theory and Practice of Soft., ETAPS 2009*. pp. 79–93. FASE ’09, Springer-Verlag, Berlin, Heidelberg (2009)
- [27] Kugler, H., Segall, I.: Compositional synthesis of reactive systems from live sequence chart specifications. In: *Procs, 15th Int. Conf. on Tools and Algorithms for the Const. and Anal. of Sys., ETAPS 2009*. pp. 77–91. TACAS ’09, Springer-Verlag, Berlin, Heidelberg (2009)
- [28] Kumar, R., Garg, V.: Control of stochastic discrete event systems modeled by probabilistic languages. *IEEE Transactions on Automatic Control* 46(4), 593–606 (4 2001)
- [29] Li, Y., Lin, F., Lin, Z.H.: Supervisory control of probabilistic discrete-event systems with recovery. *IEEE Transactions on Automatic Control* 44(10), 1971–1975 (Oct 1999)
- [30] Maler, O., Pnueli, A., Sifakis, J.: On the synthesis of discrete controllers for timed systems, pp. 229–242. Springer Berlin (1995)
- [31] Markovskii, J., Su, R.: Towards optimal supervisory controller synthesis of stochastic nondeterministic discrete-event systems. In: *Procs, of the 52nd IEEE Conf. on Decision and Control, CDC 2013*, December 10-13, 2013, Firenze, Italy. pp. 7615–7620. IEEE (2013)
- [32] Martin, D.A.: The determinacy of blackwell games. *The Journal of Symbolic Logic* 63(4), 1565–1581 (1998)
- [33] Neider, D., Rabinovich, R., Zimmermann, M.: Down the borel hierarchy. *Theor. Comput. Sci.* 560(P3), 219–234 (Dec 2014)
- [34] Pantelice, V., Lawford, M., Postma, S.M.: A framework for supervisory control of probabilistic discrete event systems. In: *Lesage, J., Faure, J., Cury, J.E.R., Lennartson, B. (eds.) 12th Int. Workshop on Discrete Event Systems, WODES 2014, Cachan, France, May 14-16, 2014*. pp. 477–484. Int. Federation of Automatic Control (2014)
- [35] Piterman, N., Pnueli, A., Sa’ar, Y.: Synthesis of reactive(1) designs. In: *Procs, of the 7th Int. Conf. on Verif., Model Checking, and Abstract Interpretation*. pp. 364–380. VMCAI’06, Springer-Verlag (2006)
- [36] Pnueli, A., Rosner, R.: On the synthesis of a reactive module. In: *Procs, of the 16th ACM SIGPLAN-SIGACT Symp. on Principles of Programming Languages*. pp. 179–190. POPL ’89, ACM, USA (1989)
- [37] Pnueli, A.: The temporal logic of programs. In: *Procs, of the 18th Annual Symp. on Foundations of Comp. Sci.* pp. 46–57. SFCS ’77, IEEE Comp. Society, Washington, DC, USA (1977)
- [38] Pnueli, A., Sa’ar, Y., Zuck, L.D.: Jtlv: A framework for developing verification algorithms. In: *Procs, of the 22Nd Int. Conf. on Comp. Aided Verification. CAV’10*, Springer-Verlag, Berlin, Heidelberg (2010)
- [39] Tripathi, R., Valkanova, E., Kumar, V.S.A.: On strategy improvement algorithms for simple stochastic games. In: *Procs, of the 7th Int. Conf. on Algorithms and Complexity. CIAC’10*, Springer-Verlag (2010)
- [40] Wongpiromsarn, T., Topcu, U., Murray, R.M.: Automatic synthesis of robust embedded control soft. In: *Embedded Reasoning, Papers from the 2010 AAAI Spring Symp.*, Technical Report SS-10-04, Stanford, California, USA, March 22-24, 2010. AAAI (2010)
- [41] Zielonka, W.: Infinite games on finitely coloured graphs with applications to automata on infinite trees. *Theoret. Comp. Sci.* 200(1), 135 – 183 (1998)