Symbolic Algorithms for Omega-Regular Games under Strong Transition Fairness

Technical Report
MPI-SWS-2020-007

Rupak Majumdar
MPI-SWS, Germany.
rupak@mpi-sws.org

Kaushik Mallik
MPI-SWS, Germany.
kmallik@mpi-sws.org

Anne-Kathrin Schmuck
MPI-SWS, Germany.
akschmuck@mpi-sws.org

Sadegh Soudjani
Newcastle University, UK.
sadegh.soudjani@ncl.ac.uk

ABSTRACT
We consider fixpoint algorithms for two-player games on graphs with Rabin winning conditions, where the environment is constrained by a strong transition fairness assumption. Strong transition fairness is a widely occurring special case of strong fairness, which requires that any execution is strongly fair with respect to a specified set of live edges: whenever the source vertex of a live edge is visited infinitely often along a play, the edge itself is traversed infinitely often along the play as well. We show that, surprisingly, strong transition fairness retains the alternation depth of a fixpoint algorithm for Rabin games—the new algorithm can be obtained simply by replacing certain occurrences of the controllable predecessor by a new almost sure predecessor operator! The complexity of the algorithm is $O(n^{k+1}k!)$, with $k$ being the number of Rabin pairs, which is independent of the number of live edges in the strong transition fairness assumption. In contrast, strong fairness necessarily requires increasing the alternation depth. As a byproduct of our result, we get symbolic algorithms for parity and GR(1) objectives under strong transition fairness assumptions as well as the first direct symbolic algorithm for qualitative winning in stochastic generalized Rabin games. Previous approaches for the problem would either increase the alternation depth to the sum of the number of Rabin pairs and the number of live edges in the fairness assumption (for Rabin conditions) or require an up-front automata-theoretic construction from the specification to a Rabin automaton (for GR(1) specifications). In particular, we show that GR(1) specifications with strong transition fairness assumption can be solved in $O(n^3)$ time.

1 INTRODUCTION
Symbolic algorithms for two-player graph games are at the heart of many problems in the automatic synthesis of correct-by-construction hardware, software, and cyber-physical systems from logical specifications. The problem has a rich pedigree, going back to Church [6] and a sequence of seminal results [4, 10, 12, 14, 16, 26, 28, 30]. A chain of reductions can be used to reduce the synthesis problem for $\omega$-regular specifications to finding winning strategies in two-player games on graphs (see, e.g., [12, 25, 30]).

In practice, it is often the case that no solution exists to a given synthesis problem, but for “uninteresting” reasons. For example, consider synthesizing a mutual exclusion protocol from a specification that requires that at most one of two processes can be in the critical section at the same time and that a process wishing to enter the critical section is eventually allowed to do so. As stated, there may not be a feasible solution to the problem because a process within the critical section may decide to stay there forever. Similarly, in a synthesis problem involving concurrent threads, no solution may exist simply because the scheduler may never decide to pick a particular thread. Fairness assumptions rule out such uninteresting conditions by constraining the possible behaviors of the environment. The winning condition under fairness is of the form

$$\text{Fairness Assumption } \Rightarrow \omega\text{-regular Specification.} \quad (1)$$

For example, a fairness constraint can state that whenever a process is in its critical section, it must eventually leave it or that, if a thread is enabled infinitely often, then it is picked by the scheduler infinitely often. These two examples actually fall into a particular subclass of fairness assumptions, called strong transition fairness [1, 13, 27]. A strong transition fairness assumption can be modeled by a set of live environment transitions in the underlying two-player game graph such
that whenever the source vertex of a live transition is visited infinitely often, the edge will be taken infinitely often by the environment.

In this paper, we show a surprisingly simple syntactic trick that modifies well-known symbolic fixpoint algorithms for solving two-player games on graphs, such that the winning condition (1) is solved instead whenever the given fairness assumption can be specified as strong transition fairness. To appreciate the simplicity of our modification, let us consider the well-known symbolic fixpoint algorithm for parity games [12] given by the $\mu$-calculus formula

$$\mu X_1. vY_2. \mu X_3 \ldots vY_{2k}. (C_1 \cap \text{Cpre}(X_1)) \cup (C_2 \cap \text{Cpre}(Y_2)) \cup \ldots \cup (C_{2k} \cap \text{Cpre}(Y_{2k})),$$

where Cpre(X) denotes the controllable predecessor operator. In the presence of strong transition fairness, the new algorithm is

$$vY_{0}. \mu X_1. vY_2. \mu X_3 \ldots vY_{2k}. (C_1 \cap \text{Apre}(Y_0, X_1)) \cup (C_2 \cap \text{Cpre}(Y_2)) \cup \ldots \cup (C_{2k} \cap \text{Cpre}(Y_{2k})).$$

The only syntactic change we make is to substitute every variable $X_i$ by an almost sure predecessor variable $\text{Apre}(Y_{i-1}, X_i)$ incorporating also the previous $v$ variable $Y_{i-1}$. That’s it!

In a nutshell, our results show that strong transition fairness retains the algorithmic characteristics (alternation depth) of known symbolic fixpoint algorithms, while allowing expressive fairness constraints on environment behaviors. More concretely, we show the correctness of our syntactic fixpoint manipulation for Rabin games [22, 28] and generalized Rabin games and then show its correctness also for Reachability, Safety, (generalized) Büchi, (generalized) co-Büchi, Rabin-chain, parity [12, 19] and GR(1) games [23] as special cases.

These considered symbolic algorithms solve two-player games by finding the set of states of the underlying game graph from which the game can be won. They do so by manipulating sets of states and computing fixed points of monotone operators. The benefit of symbolic approaches is that they allow efficient implementations based on manipulations of formulas (often represented using data structures such as BDDs). Such implementations can scale to very large finite state spaces or to infinite, but symbolically representable, state spaces. Indeed, these fixpoint expressions are the cornerstone of many reactive synthesis tools [3, 9, 20].

A symbolic fixpoint algorithm for Rabin games is given by Piterman and Pnueli in [22]. A Rabin game is played between two players Player 0 and Player 1, which move a token along the edges of a directed graph whose vertices are partitioned between them. If the token is in a vertex owned by Player 0, she moves the token along some outgoing edge. If, on the other hand, the vertex is owned by Player 1, he decides the edge. Whether the resulting infinite play is winning for Player 0 is decided by the Rabin winning condition which is defined using a set of pairs of subsets of the the graph vertices, $\{(G_t, R_t), \ldots, (G_k, R_k)\}$. Player 0 wins the Rabin game if there is some $i \in \{1, \ldots, k\}$ such that the set of infinitely visited vertices intersects $G_i$ and does not intersect $R_i$. The fixpoint algorithm of Piterman and Pnueli [22] has alternation depth $2k + 1$ for a Rabin condition with $k$ pairs and runs in time $O(n^{k+1}k)$.

Rabin conditions are a canonical acceptance condition for all $\omega$-regular objectives, thus, solving a game with any $\omega$-regular objective can be reduced to solving a Rabin game after a product construction with a suitable deterministic automaton. Therefore, our new fixpoint algorithm for Rabin games under strong transition fairness solves the winning condition (1) whenever the environment assumption can be expressed by live edges, that are edges of the graph originating in Player 1 vertices s.t. whenever the source vertex of a live edge is visited infinitely often, the edge will be taken infinitely often by Player 1.

A Rabin game under strong transition fairness is a special case of a Rabin game under a strong fairness (compassion) assumption [2, p.364]. A compassion assumption is described by a Streett winning condition, which is the dual of a Rabin winning condition. A Streett condition is also specified by a set of pairs of subsets of vertices $\{(G_t, R_t), \ldots, (G_k, R_k)\}$. It is satisfied by an infinite play if for each $i \in \{1, \ldots, l\}$, whenever the set of vertices visited infinitely often intersects $G_i$ it also intersects $R_i$. Since the dual of a Streett condition is a Rabin condition, a Rabin game (with $k$ Rabin pairs) under a compassion assumption (with $l$ Streett pairs) is equivalent to a Rabin game (without environment assumptions) with $k + l$ Rabin pairs. Hence, it can be solved by the Piterman and Pnueli algorithm [22] with alternation depth $2(k + l) + 1$ that runs in time $O(n^{k+l+1}(k + l))$. In a well-defined sense, one cannot expect a general fixpoint solution of lower alternation depth. In contrast, our algorithm for strong transition fairness has alternation depth $2(k + 1)$ and runs in time $O(n^{k+1}k!)$ – independent of the number of transitions in the strong transition fairness assumption! In many practical cases, including the example of synthesizing mutual exclusion protocols, finding schedulers for concurrent threads, and many other applications, strong transition fairness is however sufficient to express many interesting environment assumptions.

By applying the same syntactic trick as outlined for Parity games above, we generalizes the 3-nested fixpoint algorithm for GR(1) objectives to a new 3-nested fixpoint algorithm for GR(1) objectives with additional strong transition fairness constraints! Recall that the GR(1) fragment is designed
explicitly to rule out strong fairness constraints because of
the absence of suitable low-depth fixpoint algorithms. Our
result shows that, in contrast to full strong fairness, strong
transition fairness retains algorithmic efficiency while en-
abling many expressive fairness constraints that go beyond
the ones expressible in OR(1).

A byproduct of our algorithm is a fully symbolic algo-
rithm for qualitative winning for stochastic generalized Ra-
bin games. Stochastic two-player games generalize two-player
graph games with an additional category of “random” ver-
tices: whenever the game reaches a random vertex, a ran-
don process picks one of the outgoing edges (uniformly at random, w.l.o.g.). The qualitative winning problem asks
whether a vertex of the game is almost surely winning for
Player 0. Stochastic Rabin games were studied by Chatterjee,
De Alfaro, and Henzinger [5], who showed that the problem
remains NP-complete and that winning strategies can be
restricted to be memoryless. Moreover, they showed reduc-
tion from qualitative winning in a n-vertex k-pair stochastic
Rabin game to an O((nk)(k + 1)-pair (deterministic)
Rabin game, resulting in an O((nk)k+2(k + 1)!) algorithm. In
contrast, we get a direct O(nk+1k!) symbolic algorithm for
the problem.

Our result yields a symbolic algorithm in the following
way. We replace the probabilistic transitions with transitions
of the environment constrained by extreme fairness [24]. Ex-
treme fairness is a special case of strong transition fairness,
and is specified via a set of Player 0 vertices. A run is ex-
tremely fair if it is strongly transition fair for every outgoing
edge from these vertices. We show that, to solve a qualita-
tive stochastic generalized Rabin game, we can equivalently
solve the generalized Rabin game under extreme fairness.
Thus, our algorithm gives a direct symbolic algorithm for
this problem.

In conclusion, the contributions of the paper are as follows.

(1) We provide a direct symbolic fixpoint algorithm for
Rabin games under a strong transition fairness assumption
on the environment. The alternation depth of the
fixpoint expression depends only on the number of
Rabin pairs and not on the number of transition fair-
ness constraints. This is in contrast to strong fairness
(Streett) assumptions on the environment.

(2) As special cases, we show that our fixpoint formula
generalizes fixpoint algorithms for well-known sub-
cases: parity objectives and GR(1) objectives, both un-
der strong transition fairness. In all cases, the recipe for
the new algorithm is surprisingly simple: it replaces
some controllable predecessor operators in the “usual”
fixpoint algorithms with an almost sure predecessor operator.

(3) Since extreme fairness is a special case, we obtain a
direct symbolic algorithm for qualitative generalized
Rabin conditions for stochastic two-player games.

2 PRELIMINARIES

Notation: We use the notation N0 to denote the set of natural
numbers including “0”. Given a, b ∈ N0, we use the notation
[a; b] to denote the set {n ∈ N0 | a ≤ n ≤ b}. Observe that,
by definition, [a; b] is an empty set if a > b.

For any set A ⊆ U defined on the universe U, we use the
notation A over the complement of A.

Let A and B be two sets and R ⊆ A × B be a relation. We use
the notation dom(R) to denote the domain of R, which is the
set {a ∈ A | ∃b ∈ B . (a, b) ∈ R}. For any element a ∈ A, we
use the notation R(a) to denote the set {b ∈ B | (a, b) ∈ R},
and for any element b ∈ B, we use the notation R−1(b) to
denote the set {a ∈ A | (a, b) ∈ R}. We generalize R(·) to
operate on sets in the following way: for any A′ ⊆ A, R(A′) :=
∪a∈A′R(a), and for any B′ ⊆ B, R−1(B′) := ∪b∈B′R−1(b).

Given an alphabet A, we use the notation A and Aω to
denote respectively the set of all finite words and the set of
all infinite words formed using the letters of the alphabet
A. We use Aω to denote the set A ∪ Aω. Given two words
a ∈ A and b ∈ Aω, we use a · b to denote their concatenation.

2.1 Two-Player Games

Game Graphs: We define a two-player game graph as a tuple
G = (V, V0, V1, E), where (i) V is a finite set of vertices that
is partitioned into the sets V0 and V1; (ii) E ⊆ (V × V) is a
relation denoting the set of edges;

Strategies: The two players are called Player 0 and Player 1,
who control the vertices V0 and V1 respectively. A strategy
of Player 0 is a function ρ0 : V̄ · V0 → V with the constraint
ρ0(w · v) ∈ E(v) for every w · v ∈ V̄ × V0. Likewise, a
strategy of Player 1 is a function ρ1 : V̄ · V1 → V with the
constraint ρ1(v) ∈ E(v) for every w · v ∈ V̄ × V1. Of special
interest is the class of memoryless strategies: The strategy ρ0
of Player 0 is memoryless if for every w1 · v, w2 · v ∈ V̄ × V0,
ρ0(w1 · v) = ρ0(w2 · v).

Plays: Consider an infinite sequence of vertices π = v0 · v1 · v2
... ∈ Vω. The sequence π is called a play over G starting
at the vertex v0 if for every i ∈ N0, we have vi ∈ V and
(vi, vi+1) ∈ E. In our convention for denoting vertices, su-
perscripts, ranging in N0, will denote the position of a vertex
within a given play, whereas subscripts, either 0 or 1, will
denote the membership of a vertex in the sets V0 or V1 respec-
tively. Let ρ0 and ρ1 be a given pair of strategies of Player 0
and Player 1, and v0 be a given initial vertex v0. The play
compliant with ρ0 and ρ1 is the play π = v0 · v1 · v2 ... for which
for every i ∈ N0, if vi ∈ V0 then vi+1 = ρ0(v0 ... vi), and if
vi ∈ V1 then vi+1 = ρ1(v0 ... vi).
Winning Conditions: A winning condition \( \varphi \) is a set of plays over \( G \), i.e., \( \varphi \subseteq V^\omega \). We adopt Linear Temporal Logic (LTL) notation for describing winning conditions. The atomic propositions for the LTL formulae are sets of vertices, i.e., elements of the set \( 2^V \). We use the standard symbols for the Boolean and the temporal operators: “\( \neg \)” for negation, “\( \land \)” for conjunction, “\( \lor \)” for disjunction, “\( \rightarrow \)” for implication, “\( \Box \)” for until \((A \cup U B \text{ means “the play remains inside the set } A \text{ until it moves to the set } B)\), “\( 
abla \)” for next \((\Box A \text{ means “the next vertex is in the set } A\))”, “\( \diamond \)” for eventually \((\nabla A \text{ means “the play will eventually visit a vertex from the set } A)\), and “\( \bowtie \)” for always \((\Box A \text{ means “the play will only visit vertices from the set } A)\). The syntax and semantics of LTL can be found in standard textbooks [2]. By slightly abusing notation, we will use \( \varphi \) interchangeably to denote both the LTL formula and the set of plays satisfying \( \varphi \). Hence, we write \( \pi \in \varphi \) (instead of \( \pi \models \varphi \)) to denote the satisfaction of the formula \( \varphi \) by the play \( \pi \).

Winning Regions: Player 0 wins a two-player game over the game graph \( G \) with respect to a winning condition \( \varphi \) from a vertex \( v^0 \in V \) if there is a Player 0 strategy \( \rho_0 \) such that for all Player 1 strategies \( \rho_1 \), the play \( \pi \) from \( v^0 \) compliant with \( \rho_0 \) and \( \rho_1 \) satisfies \( \varphi \), i.e., \( \pi \in \varphi \). The winning region \( \mathcal{W} \subseteq V \) for Player 0 is the set of vertices from which Player 0 wins the game.

2.2 Fair Adversarial Games

Let \( G \) be a two-player game graph and let \( E^f \subseteq (V \times V) \cap E \) be a given set of live edges. We denote by \( V_f \) the set of Player 1 edges in the domain of \( E^f \), i.e., \( V_f := \text{dom}(E^f) \). Intuitively, the edges in \( E^f \) represent fairness assumptions on Player 1: for each edge \((v, v') \in E^f \), if \( v \) is visited infinitely often along a play, we expect that the edge \((v, v') \) is picked infinitely often by Player 1.

We write \( G^f = (G, E^f) \) to denote a game graph with live edges, and extend notions such as plays, strategies, winning conditions, winning region, etc., from game graphs to those with live edges. A play \( \pi \) over \( G^f \) is strongly transition fair if it satisfies the LTL formula:

\[
\varphi := \bigwedge_{(v, v') \in E^f} \Box \Diamond v \rightarrow \Diamond (v \land \Box v').
\]  

(3)

Given \( G^f \) and a winning condition \( \varphi \), we say Player 0 wins the fair adversarial game over \( G^f \) with respect to the winning condition \( \varphi \) from a vertex \( v^0 \in V \) if Player 0 wins the game over \( G^f \) with respect to the winning condition \( \alpha \rightarrow \varphi \) from \( v^0 \).

2.3 Symbolic Computations over Game Graphs

Set Transformers: Our goal is to give symbolic fixpoint algorithms to characterize the winning region of a fair adversarial game over a game graph with live edges. As a first step, given \( G^f \), we define the required transformers of sets of states. We define the existential, universal, and controllable predecessor operators as follows. For \( S \subseteq V \), we have

\[
\text{Pre}_0^3(S) := \{ v \in V_0 \mid E(v) \cap S \neq \emptyset \}, \quad \text{Pre}_1^3(S) := \{ v \in V_1 \mid E(v) \subseteq S \}, \quad \text{and} \quad \text{Cpre}(S) := \text{Pre}_0^3(S) \cup \text{Pre}_1^3(S).
\]  

(4a)

(4b)

(4c)

Intuitively, the controllable predecessor operator \( \text{Cpre}(S) \) computes the set of all states that can be controlled by Player 0 to stay in \( S \) regardless of the strategy of Player 1. Additionally, we define two operators which take advantage of the fairness assumption on the live edges. Given two sets \( S, T \subseteq V \), we define the live-existential and almost-sure predecessor operators:

\[
\text{Pre}_f(S, T) := \text{Cpre}(T) \cup \left( \text{Pre}_0^3(T) \cap \text{Pre}_1^3(S) \right).
\]  

(5a)

(5b)

Intuitively, the almost-sure predecessor operator \( \text{Apre}(S, T) \) computes the set of all states that can be controlled by Player 0 to stay in \( T \) (via \( \text{Cpre}(T) \)) as well as all Player 1 states in \( V_f \) which will eventually make progress towards \( T \) if Player 1 obeys its fairness-assumptions encoded in \( \alpha \) (via \( \text{Pre}_1^3(T) \)) and never leave \( S \) in the “meantime” (via \( \text{Pre}_0^3(S) \)). We immediately see that all these set transformers are monotonic with respect to set inclusion, \( \text{Cpre}(T) \subseteq \text{Apre}(S, T) \). Moreover, we have \( \text{Apre}(S, T) \subseteq \text{Cpre}(S) \) if \( T \subseteq S \) (see Lem. A.1 for a proof).

Fixpoint Algorithms in the \( \mu \)-calculus: We use the \( \mu \)-calculus as a convenient logical notation used to define a symbolic algorithm (i.e., an algorithm that manipulates sets of states rather than individual states) for computing a set of states with a particular property over a given game graph \( G \). The formulas of the \( \mu \)-calculus, interpreted over a two-player game graph \( G \), are given by the grammar

\[
\varphi ::= p \mid X \mid \varphi \lor \varphi \mid \varphi \land \varphi \mid \text{pre}(\varphi) \mid \mu X.\varphi \mid \nu X.\varphi
\]

where \( p \) ranges over subsets of \( V \), \( X \) ranges over a set of formal variables, \( \text{pre} \in \{ \text{Pre}_0^3, \text{Pre}_1^3, \text{Cpre}, \text{Apre} \} \) ranges over set transformers, and \( \mu \) and \( \nu \) denote, respectively, the least and the greatest fixed point of the functional defined as \( X \mapsto \varphi(X) \). Since the operations \( \cup, \cap, \) and the set transformers \( \text{pre} \) are all monotonic, the fixed points are guaranteed to exist. A \( \mu \)-calculus formula evaluates to a set of states over
\( G \), and the set can be computed by induction over the structure of the formula, where the fixed points are evaluated by iteration. We omit the (standard) semantics of formulas [15].

3 FAIR ADVERSARIAL RABIN GAMES

This section presents the main result of this paper, which is a symbolic fixpoint algorithm that computes the winning region of Player 0 in the fair adversarial game over \( G^f \) with respect to any \( \omega \)-regular property formalized as a Rabin winning condition.

Our new fixpoint algorithm has multiple unique features. (I) It works directly over \( G^f \), without requiring any preprocessing step to reduce \( G^f \) to a “normal” two-player game. This feature allows us to obtain a direct symbolic algorithm for stochastic games as a by-product (see Sec. 5).

(II) Conceptually our symbolic algorithm is not more complex than the known algorithm solving Rabin games over “normal” two-player game graphs [22] (see Sec. 3.3).

(III) Our new fixpoint algorithm is obtained from the known algorithm in [22] by a simple syntactic change (as previewed in (2)). We simply replace all controllable predecessor operators over least fixpoint variables by the almost-sure predecessor operator invoking the preceding maximal fixpoint variable. This makes the proof of our new fixpoint algorithm conceptually simple (see Sec. 3.2).

On a higher level, this section shows that our syntactic change is a very simple yet efficient trick to incorporate environment assumptions expressible by live edges into reactive synthesis while retaining computational efficiency. Most remarkably, this trick also works directly for fixpoint algorithms solving Rabin-Chain, Parity and (generalized) Co-Büchi games, as these can be formalized as particular instances of a Rabin game (see Sec. 3.4). Moreover, it also works for GR(1) winning conditions. However, as GR(1) is a particular instance of a generalized Rabin game, we prove this special case separately in Sec. 4 after formally introducing generalized Rabin games.

3.1 The Symbolic Algorithm

Fair adversarial Rabin Games: A Rabin winning condition is defined by the set \( \mathcal{R} = \{ (G_1, R_1), \ldots, (G_k, R_k) \} \), where each \( G_i, R_i \subseteq V \). We say that \( \mathcal{R} \) has index set \( P = [1:k] \). A play \( \pi \) satisfies the Rabin condition \( \mathcal{R} \) if \( \pi \) satisfies the LTL formula

\[
\phi := \bigvee_{i \in P} (\square \Diamond R_i \land \square \Diamond G_i). \tag{6}
\]

We now present our new symbolic fixpoint algorithm that computes the winning region of Player 0 in the fair adversarial game over \( G^f \) with respect to a Rabin winning condition \( \mathcal{R} \) which constitutes the main result of this paper.

Theorem 3.1. Let \( G^f = (G, E^f) \) be a game graph with live edges and \( \mathcal{R} \) be a Rabin condition over \( G \) with index set \( P = [1:k] \). Further, let

\[
Z^* := \nu Y_{p_0}. \mu X_{p_0}. \bigcup_{p_i \in P} \nu Y_{p_i}. \mu X_{p_i}. \bigcup_{p_i \in P \setminus \{p_1\}} \nu Y_{p_i}. \mu X_{p_i}. \bigcup_{p_i \in P \setminus \{p_1, \ldots, p_{k-1}\}} \nu Y_{p_i}. \mu X_{p_i}. \left[ \bigcup_{j=0}^k C_{p_j} \right],
\]

where

\[
C_{p_j} := \bigcap_{i=0}^j \overline{R}_{p_i} \cap \left( (G_{p_j} \cap \text{Cpre}(Y_{p_j})) \cup (\text{Apres}(Y_{p_j}, X_{p_j})) \right),
\]

with \( p_0 = 0, G_{p_0} := \emptyset \) and \( R_{p_0} := \emptyset \). Then \( Z^* \) is equivalent to the winning region \( W \) of Player 0 in the fair adversarial game over \( G^f \) for the winning condition \( \phi \) in (6). Moreover, a memoryless winning strategy for Player 0 can be extracted from this fixpoint algorithm.

Intuition: The way the fixpoint variables are organized in (7) makes the fixpoint algorithm explore all possible permutation sequences of indices from \( P \). Hence, it always starts with the (artificially) introduced index \( p_0 = 0 \) and then chooses one index from the set \( P \) and calls it \( p_1 \) (note that \( p_1 \in [1:k] \) and not necessarily \( p_1 = 1 \)). After fixing this choice, it chooses another index \( p_2 \in P \setminus \{p_1\} \) and so forth until only one index \( p_k \in P \setminus \{p_1, \ldots, p_{k-1}\} \) is left to be chosen. So, at the end we have a permutation sequence \( \delta = p_0 p_1 \ldots p_k \) where each index from \( P \) occurs exactly once.

Based on this sequence, we compute \( C_\delta := \bigcup_{j=0}^k C_{p_j} \), with \( C_{p_j} \) as in (7b). That is, for each permutation sequence \( \delta \), we start with \( p_0 \) and compute \( C_{p_j} \) where the leading term \( Q_{p_j} := \bigcap_{i=0}^j \overline{R}_{p_i} \) is simply \( \overline{R}_{p_0} = Q \). Then we compute \( C_{p_j} \) which takes also \( p_0 \) into account when computing \( Q_{p_j} \) and so forth. Finally, \( C_{p_k} \) contains the intersection of all sets \( \overline{R}_{p_i} \) for any \( p_i \in P \) in \( Q_{p_j} \).

During the computation of \( C_\delta \) the algorithm updates the fixpoint variables \( X_{p_i} \) by disjoining the current values of all fixpoint variables \( Y_{p_i} \) with \( p_i \setminus P \subset \{p_1, \ldots, p_i\} \). To see how this update works, recall that \( p_k \in P \setminus \{p_1, \ldots, p_{k-1}\} \) is a fixed, unique index choice. Going backward in the permutation sequence, we see that for \( p_{k-1} \in P \setminus \{p_1, \ldots, p_{k-2}\} \) we have two choices left, say either \( p_{k-1} = a \) or \( p_{k-1} = b \). Whatever choice we make for \( p_{k-1} \) determines the (unique) choice for \( p_k \). I.e., given a common permutation sequence prefix \( \gamma := p_0 \ldots p_{k-1} \) we either get \( \delta = \gamma ab \) or \( \delta' = \gamma ba \) as full permutation sequences. Now the algorithm computes \( C_\delta \) and \( C_{\delta'} \) using (a) fixed values for all fixpoint variables

\[1\) The Rabin pair \((G_{p_0}, R_{p_0}) = (\emptyset, \emptyset) \) in (7) is artificially introduced to make the fixpoint representation more compact. It is not part of \( \mathcal{R} \).
with winning condition

ward throughout all index choices in (7a), we successively
the term

C

Safe Büchi Games

game

G

is known to define exactly the states of a "normal" two player
for two sets

Q

are not ordered by inclusion, we need to look at all possible
updates (the unique) variable

X

= 0

disjoin all disjuncted -terms for all possible permutations.

fixpoint algorithm by considering only two pairs. For every Rabin
sider every possible permutation sequence in the fixpoint
eral structure of (7), let us understand why we need to con-
Necessity for Permutation: After understanding the gen-
eral structure of (7), let us understand why we need to con-
side every possible permutation sequence in the fixpoint
algorithm by considering only two pairs. For every Rabin
pair

(Gi, Rj)

a winning play must eventually stay in

Ri

and always eventually re-visit
Gi.
Now it could be the case that
player 1 decides which of the local specifications can be ful-
filled. Hence, player 0 might be able to force that the play
stays in

Ri

but can’t decide whether
Gi

or
G2

is visited in-
fi nitely often inside

Ri.
Now if
G2

instead of
G1
is visited infinitely often along a play, at some point the play must also
stay in

R2

not only
R1
(this corresponds to the permutation
sequence “12”). However, as in general

Ri

and

Rj
for
i ≠ j
are not ordered by inclusion, we need to look at all possible
orderings of indices to consider all possible ways of winning.

3.2 Proof Outline

Given a Rabin winning condition over a “normal” two-player
game, Piterman and Pnueli [22] have provided a symbolic
fixpoint algorithm which computes the winning region for
Player 0. The fixpoint algorithm in their paper is almost
identical to our fixpoint algorithm in (7); it only differs in
the last term of the constructed -terms in (7b). I.e., in [22]
the term

Cp,i
is defined as

\[
\bigcap_{j=0}^{i} \overline{R}_{p,j} \cap \left( \left( G_{p,i} \cap C_{\text{pre}}(Y_{p,i}) \right) \cup \left( C_{\text{pre}}(X_{p,i}) \right) \right).
\]

As discussed before, a single term

Cp,i
essentially computes the set of states that always remain within

Q_{p,i} := \bigcap_{j=0}^{i} \overline{R}_{p,j}
while always re-visiting
G_{p,i}.
I.e., given the simpler (local) winning condition

\[
\psi := \Box Q \land \Box \Box G
\]
for two sets

Q, G \subseteq V,
the set

\[
vY. \mu X. Q \cap \left( \left( G \cap C_{\text{pre}}(Y) \right) \cup \left( C_{\text{pre}}(X) \right) \right)
\]
(9)
is known to define exactly the states of a “normal” two player
game

G
from which Player 0 has a strategy to win the game
with winning condition

\[
\psi [19].
\]
Such games are typically called Safe Büchi Games.

Therefore, the key insight in the proof of Thm. 3.1 is to
show that the new definition of -terms in (7b) via the new
almost-sure predecessor operator

A_{\text{pre}}
actually computes the winning state sets of fair adversarial safe Büchi games.

Fair Adversarial Safe Büchi Games: A fair adversarial
safe Büchi game is formalized in the following theorem.

Theorem 3.2. Let

G^f = \langle G, E^f \rangle
be a game graph with live edges and

Q, G \subseteq V
be two state sets over

G.
Further, let

\[
Z^* := vY. \mu X. Q \cap \left[ \left( G \cap C_{\text{pre}}(Y) \right) \cup \left( A_{\text{pre}}(Y, X) \right) \right].
\]
(10)

Then

Z^*

is equivalent to the winning region of Player 0 in the
fair adversarial game over

G^f
for the winning condition

\psi
in (8). Moreover, a memoryless winning strategy for Player 0 can be
extracted from this fixpoint algorithm.

Intuitively, the fixed points in (9) and (10) consist of two
parts. (a) A minimal fixed point over
X
which computes (for
any fixed value of
Y
the set of states that can reach the
“target state set”

T := Q \cap G \cap C_{\text{pre}}(Y)
while staying inside

Q
, and
(b) a maximal fixed point over
Y
which ensures that only states are considered in the target

T
allow to re-visit a state in
T
while staying in
Q.
By comparing (9) and (10) we see that our syntactic trick
only changes part (a). Hence, in order to prove Thm. 3.2 it
essentially remains to show that this trick works for simple
safe reachability games.

Fair Adversarial Safe Reachability Games: A safe reach-
ability condition is a tuple

\langle T, Q \rangle
with

T, Q \subseteq V
and a play

\pi
satisfies the safe reachability condition

\langle T, Q \rangle
if

\psi := QUT.
(11)
A safe reachability game is often called a reach-while-avoid
game, where the safe sets are specified by an unsafe set

R := \overline{Q}
that needs to be avoided. Their fair adversarial version is
formalized in the following theorem, proven in App. B.1.

Theorem 3.3. Let

G^f = \langle G, E^f \rangle
be a game graph with live edges and

\langle T, Q \rangle
a safe reachability winning condition. Further let

\[
Z^* := vY. \mu X. T \cup \left( Q \cap A_{\text{pre}}(Y, X) \right).
\]
(12)

Then

Z^*

is equivalent to the winning region of Player 0 in the
fair adversarial game over

G^f
for the winning condition

\psi
in (11). Moreover, a memoryless winning strategy for Player 0 can be
extracted from this fixpoint algorithm.

With this insight, Thm. 3.2 becomes a simple consequence of
Thm. 3.3 (see Sec. B.2 for a full proof). Further, with
Thm. 3.2 in place, the proof of Thm. 3.1 is essentially equivalent to
the proof of Piterman and Pnueli in [22] while utilizing
Thms. 3.3 and Thm. 3.2 at all suitable places. For completeness,
we give the full proof of Thm. 3.1, including the memoryless
strategy construction, in App. C.
3.3 Complexity

Complexity Analysis of (7): Piterman and Pnueli show that the fixpoint algorithm for the “normal” two-player Rabin games runs in time $O(n^2k!)$, where $n$ is the number of vertices of the underlying game graph, $d$ is the alternation depth of the fixpoint algorithm and $k$ is the number of Rabin pairs [22]. As the alternation depth of the Piterman and Pnueli algorithm is $d = 2k + 1$, their algorithm runs in $O(n^{k+1}k!)$. This result is obtained by showing that the accelerated fixpoint computation technique in [17] is applicable to the particular Rabin fixpoint

As we do not change the structure of the fixpoint algorithm and only substitute one monotone operator by another one, the accelerated computation of the fixpoint in [17] is equivalently applicable to our new algorithm. Further, we do not modify the original game graph and only slightly increase the alternation depth of the fixpoint algorithm to $d = 2(k + 1)$. We therefore obtain the same time complexity $O(n^{k+1}k!)$. This means our algorithm is as efficient as the original algorithm for Rabin games without environment assumptions—indeed for the number of considered strong transition fairness assumptions!

Comparison with a Naïve Solution: We show a naïve reduction from the fair adversarial Rabin games to the usual Rabin games, which can be solved using the fixpoint algorithm of Piterman and Pnueli, albeit by spending much more computational effort than our direct fixpoint algorithm in Thm. 3.1. Suppose $G^f = \langle G, E^f \rangle$ is a game graph with live edges, $R = \{ \langle G_1, R_1 \rangle, \ldots, \langle G_k, R_k \rangle \}$ is a Rabin winning condition defined over $G^f$, and $\varphi$ is the corresponding LTL specification as defined in (6). Let $\tilde{G} = \langle \tilde{V}, \tilde{V}_0, \tilde{V}_1, \tilde{E} \rangle$ be a game graph obtained by just replacing every live edge of $G^f$ with a gadget shown in Fig. 1 and explained next. For every live edge $(v, v') \in E^f$ we introduce a new intermediate vertex named $vv'$ in $\tilde{V}$, and without loss of generality we assume that $vv' \in \tilde{V}_1$. (We could have equivalently used the convention that $vv' \in \tilde{V}_0$. Then we replace the edge $(v, v')$ with a pair of new edges $(v, vv') \in \tilde{E}$ and $(uvv', vv') \in \tilde{E}$; the rest remains the same as in $G$. Assuming that $|E^f| = l$ and $|V| = n$, the number of vertices of $\tilde{G}$ is $n + l$.

Intuitively, the event of the newly introduced vertices being reached in $\tilde{G}$ simulates the event of the corresponding live edge being taken in $G^f$, and vice versa. We are now ready to transfer the specification $\alpha \rightarrow \varphi$ to a new Rabin winning condition $\tilde{R}$ for $\tilde{G}$. First observe that $\alpha \rightarrow \varphi$ is equivalent to $\neg \alpha \lor \varphi$, and $\neg \alpha$ can be expressed in LTL as

$$\bigvee_{(v, v') \in E^f} \Box \{v\} \land \Box \{vv'\}.$$  \hspace{1cm} (13)

Eq. (13) can be alternatively expressed as the Rabin winning condition $R^f := \{ \{v\}, \{vv'\} \mid (v, v') \in E^f \}$. Since the Rabin winning conditions are closed under union, hence we can define the new Rabin condition $\tilde{R}$ as follows: $\tilde{R} := R \cup R^f$.

Once $\tilde{G}$ and $\tilde{R}$ are obtained, one could use the fixpoint algorithm of Pnueli and Piterman for “normal” two-player Rabin games. This whole process yields a symbolic algorithm for fair adversarial Rabin games with $2(k + l) + 1$ alternations of fixpoint operators on a set of $(n + l)$ vertices that runs in time $O((n + l)^{k+1}(k + l)!))$. In contrast, our main theorem shows that we get a symbolic fixpoint expression with $2(k + 1)$ alternations that runs in time $O(n^{k+1}k!)$. In many applications, we expect $l = \Theta(n)$, for which our algorithm is significantly faster.

Remark 1. As already mentioned in the introduction, not all strong fairness assumptions (Streett assumptions) can be translated into live edges (see e.g., [2, p.264]). As an example, consider the two-player game graph depicted in Fig. 2. Player 0 and Player 1 vertices are indicated by a circle and a box, respectively. Now consider the following one-pair Streett assumption

$$\varphi_A := \Box \Diamond \{a, b, c\} \rightarrow \Box \Diamond \{d\} \lor \Box \Diamond \{a\}. \hspace{1cm} (14)$$

This fairness assumption states that it is not possible for a game to infinitely stay inside the set $\{a, b, c\}$ if Player 0 decides to not transition from $b$ to $a$ anymore from some point onward. We see that we cannot model this behavior by a fair edge leaving a Player 1 (boxed) state. If we mark the edge $(c, d)$ live, any fair play will transition to $d$ no matter if $a$ is visited infinitely often or not. Let us call this fair edge assumption $\varphi_A$. Then we see that $\alpha_A \rightarrow \varphi_A$ but not vice versa. This means that for a given guarantee $\varphi_G := \Diamond d$, a winning strategy in the fair adversarial game over $\alpha_A \rightarrow \varphi_G$ is not a winning strategy in

---

\[\text{Figure 1: Left: A live edge } (v, v') \text{ in } G^f. \text{ Right: The gadget used to replace } (v, v') \text{ in } \tilde{G}. \text{ The vertex named } vv' \text{ is a newly added vertex in } \tilde{G}; v \text{ belongs to } \tilde{V}_1, vv' \text{ belongs to } \tilde{V}_0, \text{ but } v' \text{ belongs to either } \tilde{V}_0 \text{ or } \tilde{V}_1.\]

\[\text{Figure 2: Counterexample to the equality of strong transition fairness and strong fairness (compassion).}\]
the original assume/guarantee game $\varphi_A \rightarrow \varphi_G$. In the latter, Player 1 might never transition from $c$ to $d$.

### 3.4 Specialized Rabin Games

This section shows, how the fixed point for fair adversarial Rabin games simplifies for different winning conditions. In particular, we show that the known fixed point for Rabin chain, Parity and Generalized Co-Büchi winning conditions allow for the same “synthactic trick” as in the Rabin case to get the right fixed point for their fair adversarial version. We prove these claims by reducing the fixed point in (7) to the special cases induced by the named winning conditions.

It should be noted that the fixpoint algorithm for fair adversarial Rabin games in (7) reduces to the normal fixpoint for different winning conditions. In particular, we show that the known fixed point for Rabin chain, Parity and Generalized Co-Büchi winning conditions allow for the same “synthactic trick” as in the Rabin case to get the right fixed point for their fair adversarial version. We prove these claims by reducing the fixpoint in (7) to the special cases induced by the named winning conditions.

Intuitively, the fixpoint algorithm computing $Z^*$ in (7) simplifies to a single permutation sequence, namely $Z^* = \{\langle F_2, F_3, \ldots, F_{2k}, \emptyset \rangle\}$ s.t. $F_i := \bigcup_{j=1}^{2k} C_j$. (18)

That is, the maximal color visited infinitely often along $\pi$ is even. A Parity winning condition $C$ with $2k$ colors corresponds to the Rabin chain winning condition

$$\{\langle F_2, F_3, \ldots, F_{2k}, \emptyset \rangle\} \quad \text{s.t. } F_i := \bigcup_{j=1}^{2k} C_j,$$

which has $k$ pairs. Due to $C$ forming a partition of the state space one can further simplify the fixpoint algorithm in (16). Indeed, the resulting fixpoint algorithm coincides with the one obtained from applying our synthactic trick to the well-known algorithm for Parity games (see (2)). This is formalized in the next theorem, which is proven in App. D.2.

**Theorem 3.5.** Let $G^f = \langle G, E^f \rangle$ be a game graph with live edges and $C$ be a Parity condition over $G$ with $2k$ colors. Further, let

$$Z^* := \{vY_0, \mu X_0, vY_1, \mu X_1, \ldots, \mu X_1, \bigcup_{j=1}^{k} \tilde{C}_j\}.$$ (16a)

where $\tilde{C}_j := R_j \cap \{\langle G_j \cap \text{Cpre}(Y_j) \rangle \cup \text{Apref}(Y_j, X_j)\}$. (16b)

with $G_{p_0} := \emptyset$ and $R_{p_0} := \emptyset$.

Then $Z^*$ is equivalent to the winning region $W$ of Player 0 in the fair adversarial game over $G^f$ for the winning condition $\varphi$ in (6). Moreover, a memoryless winning strategy for Player 0 can be extracted from this fixpoint algorithm.

**Fair Adversarial (Generalized) Co-Büchi Games:** A Co-Büchi winning condition is defined by a subset $A \subseteq V$ of vertices of $G$. A play $\pi$ satisfies the Co-Büchi condition $A$ if $\pi$ satisfies the LTL formula

$$\varphi := \diamond \Box A.$$ (20)

A Generalized Co-Büchi winning condition is defined by a set $A = \{A_1, \ldots, A_r\}$, where each $A_i \subseteq V$ is a subset of vertices of $G$. A play $\pi$ satisfies the Generalized Co-Büchi condition $A$ if $\pi$ satisfies the LTL formula

$$\varphi := \bigvee_{a \in [1, r]} \diamond \Box A_a.$$ (21)

Generalized Co-Büchi winning conditions correspond to a particular Rabin condition $R$ with $r$ pairs such that

$$\forall j \in [1, r] \quad R_j := \tilde{A}_j \quad \text{and} \quad G_j := V.$$ (22)

Intuitively, the fact that $G_j \models V$ for all $j$ leads to a cancelation of all Apref terms in $C_j$ and all terms become ordered, i.e., we have $C_{p_1} \subseteq C_{p_0}$ for every permutation sequence used in (7). As we take the union over all $C_{p_1}$’s in (7a), the term $C_{p_0}$ absorbs all others for every permutation sequence. Hence, for every permutation sequence we only have two terms left, one for $j = 0$ (over the artificially introduced Rabin pairs $G_{p_0} = R_{p_0} = \emptyset$) and one for the first choice $p_1$ made in this particular permutation. This is formalized in the following theorem which is proven in App. D.3.
Theorem 3.6. Let \( G^f = (G, E^f) \) be a game graph with live edges and \( A \) be a generalized Co-Büchi winning condition \( G \) with \( r \) pairs. Further, let
\[
Z^* := v Y_0, \mu X_0, \bigcup_{a \in [1, r]} v Y_a, Apre(Y_0, X_0) \cup (\overline{A}_a \cap Cpre(Y_a)).
\]
(23)

Then \( Z^* \) is equivalent to the winning region \( W \) of Player 0 in the fair adversarial game over \( G^f \) for the winning condition \( \varphi \) in (21). Moreover, a memoryless winning strategy for Player 0 can be extracted from this fixpoint algorithm.

4 GENERALIZED RABIN GAMES

Within this section we slightly generalize our main result, Thm. 3.1, to fair adversarial generalized Rabin games. That is, for each Rabin pair, we allow the goal set \( G_i \) to rather be a set of goal states \( G_i = \{G_{i1}, \ldots, G_{im_i}\} \). Then a play fulfills the winning condition if there exists one generalized Rabin pair \( (G_i, R_i) \) such that the play eventually remains within \( R_i \) and visits all sets \( G_i \) infinitely often.

The motivation of this generalization is to show that our syntactic trick also works for fair adversarial games with a generalized reactivity winning condition of rank 1 (GR(1) games for short). Generalized Rabin games allow to see a GR(1) winning condition as a particularly simple instantiation of a Rabin game as shown in Sec. 4.2.

4.1 Fair Adversarial Generalized Rabin Games

Generalized Rabin Conditions: A generalized Rabin condition is defined by a set \( \mathcal{R} = \{G_{i1}, R_{i1}\}, \ldots, \{G_{ik}, R_{ik}\} \) where each \( G_i = \{G_{i1}, \ldots, G_{im_i}\} \) is a finite set s.t. \( G_j \subseteq V \) for all \( j \in [1, k] \) and all \( l \in [1, m_l] \). We say that \( \mathcal{R} \) has global index set \( P = [1, k] \). A play \( \pi \) satisfies the generalized Rabin condition \( \mathcal{R} \) if \( \pi \) satisfies the LTL formula
\[
\varphi := \bigvee_{j \in P} \left( \Diamond \mathcal{R}_j \land \bigwedge_{l \in [1, m_l]} \Box \Diamond (G_j) \right).
\]
(24)

Recalling the discussion of Sec. 3.1, we know that the proof of Thm. 3.1 fundamentally relies on the correctness of our “syntactic trick” for safe Büchi (Thm. 3.2) and safe reachability (Thm. 3.3) games. Similarly, one needs to prove correctness of our “syntactic trick” for generalized Büchi games in the case of generalized Rabin games.

Safe Generalized Büchi Games: A safe generalized Büchi condition is defined by a tuple \( (\mathcal{F}, Q) \) where \( Q \subseteq V \) is a set of safe states and \( \mathcal{F} = \{F_1, \ldots, F_r\} \) is a set of goal sets. A play \( \pi \) satisfies the safe generalized Büchi condition \( (\mathcal{F}, Q) \) if \( \pi \) satisfies the LTL formula
\[
\varphi := \Box Q \land \bigwedge_{l \in [1, s]} \Box \Diamond F_1.
\]
(25)

Now we can apply our “syntactic trick” to the usual fixpoint algorithm for solving safe generalized Büchi Games and proof its correctness for all fair adversarial plays. This is formalized in the next theorem which is proven in App. E.1.

Theorem 4.1. Let \( G^f = (G, E^f) \) be a game graph with live edges and \( (\mathcal{F}, Q) \) with \( \mathcal{F} = \{F_1, \ldots, F_r\} \) a safe generalized Büchi winning condition. Further, let
\[
Z^* := v Y, \mu X, Q \cap (\bigcup_{b \in [1, s]} b X b) \cup (\bigcup_{b \in [1, s]} b F b). \quad \text{(26)}
\]

Then \( Z^* \) is equivalent to the winning region \( W \) of Player 0 in the fair adversarial game over \( G^f \) for the winning condition \( \varphi \) in (25). Moreover, a finite-memory winning strategy for Player 0 can be extracted from this fixpoint algorithm.

Intuitively, the proof of Thm. 4.1 reduces to Thm. 3.2 in a similar manner as the proof of Thm. 3.2 reduces to Thm. 3.3. However, the challenge in proving Thm. 4.1 is to show that it is indeed sound to use the fixed-point variable \( Y \) which is actually the intersection of fixpoint variables \( X \) both within \( Cpre \) and \( Apre \). The proof of this correctness essentially requires to show that upon termination we have \( Y^* = b X \) for all \( b \in [1, s] \) (see App. E.1 for a formal proof).

The Symbolic Algorithm: By knowing that (26) allows to correctly solve safe generalized Büchi games, we can immediately generalize this observation to Rabin games. This is formalized in the following theorem which is an immediate consequence of Thm. 3.1 and Thm. 4.1.

Theorem 4.2. Let \( G^f = (G, E^f) \) be a game graph with live edges and \( \mathcal{R} \) be a generalized Rabin condition over \( G \) with index set \( P = [1, k] \). Further, let
\[
Z^* := v Y_0, \mu X_0, \bigcup_{p_i \in P} v Y_{p_1} \cap \bigcup_{l_i \in [1, m_{p_1}]} \mu b X_{p_1} \cap \bigcup_{r_i \in [1, m_{p_1}]} b F_{p_1} \cap Cpre(Y_{p_1}) \cup Apre(Y_{p_1}). \quad \text{(27a)}
\]

where
\[
\begin{align*}
\mu b X_{p} & := \left( \bigcup_{i=0}^{j} \mathcal{R}_{p_i} \right) \cap \left( \left( \mathcal{R}_{p_j} \cap Cpre(Y_{p_j}) \right) \cup Apre(Y_{p_j}) \right) \quad \text{and} \quad \\
\mathcal{R}_{p_j} & := \left( \bigcup_{i=0}^{j} \mathcal{R}_{p_i} \right). \quad \text{(27b)}
\end{align*}
\]
within the section we show how fair adversarial Rabin
where for every Thm. 3.1. By adding the last tuple
satisfies the GR(1) condition
make the fixpoint representation more compact. It is not part of
\[ e \]
\[ p \]
\[ b \]
fulfill a (generalized) chain condition (compare (15)). That
principle, we would need to consider both possible orderings of
each conjunct of (16), i.e.,
condition, we essentially need to consider two indices in
form
fixed-point in Thm. 3.6 only needs to consider single indices
ized Co-Büchi condition (compare (22)) which can be solved
pairs with trivial goal sets actually correspond to a general-
ized reactivity winning condition of rank
Games can be reduced to fair adversarial games with
gener-
completeness we provide the proof of Thm. 4.2 in App. E.2.

4.2 Fair Adversarial GR(1) Games
Within this section we show how fair adversarial Rabin
Games can be reduced to fair adversarial games with gen-
eralized reactivity winning condition of rank
(Gr(1) for short).

| **GR(1) winning condition**:
| A GR(1) winning condition is defined by two sets \( \mathcal{A} = \{A_1, \ldots, A_r\} \) and \( \mathcal{F} = \{F_1, \ldots, F_s\} \),
| where for every \( i \in [1;r] \) and \( j \in [1:s]\), \( A_i, F_j \subseteq V \). A play \( \pi \)
satisfies the GR(1) condition \((\mathcal{A}, \mathcal{F})\) if it satisfies the LTL
| formula
| \[
\varphi := (\bigwedge_{a \in [1;r]} \square A_a) \rightarrow (\bigwedge_{b \in [1:s]} \Box F_b) \quad (28)
\]
By comparing \( \varphi \) in (28) with \( \varphi \) in (24) we see that a GR(1)
condition \((\mathcal{A}, \mathcal{F})\) can be transformed into a generalized Rabin
condition \( \tilde{R} \) with \( k = r + 1 \) pairs, such that
| \[ \forall j \in [1;r]. R_j := A_j \quad \text{and} \quad G_j := \{V\}, \quad (29a) \]
| \[ R_k := \emptyset \quad \text{and} \quad G_k := \mathcal{F}. \quad (29b) \]

**Fixpoint Algorithm:** We first observe that the first \( r \) Rabin
pairs with trivial goal sets actually correspond to a gen-
eralized Co-Büchi condition (compare (22)) which can be solved
by the fixed-point in Thm. 3.6 (see Sec. 3.4). Intuitively, the
fixed-point in Thm. 3.6 only needs to consider single indices
form \( P = [1;r] \) rather then full permutation sequences as in
Thm. 3.1. By adding the last tuple \( (G_k, R_k) \) to the winning
condition, we essentially need to consider two indices in
each conjunct of (16), i.e., \( p_j \) (with \( j \in [1;r] \)) and \( p_k \).
In principle, we would need to consider both possible orderings of
these two indices (compare (27)). However, by inspecting (29)
we see that the sets corresponding to these indices always
fulfill a (generalized) chain condition (compare (15)). That
is, we have \( R_j \supseteq R_k \) and \( V = \cup_{G_j} \supseteq \cup_{F_b} \) for any \( j \in [1;r] \)
and \( b \in [1:s] \). Hence, we only need to consider the permuta-
tion sequence \( p_j p_k \) (compare (16)). Using this insight along
with some additional simplifications we indeed yield the
\[ Z^* = \exists \nu \in \mathcal{Y}_k. \bigcup_{b \in [1:s]} \mu^b X_b \cup \bigcup_{a \in [1;r]} \nu Y_a. \]

Theorem 4.3. Let \( G^f = \langle G, E^f \rangle \) be a game graph with live
edges and \((\mathcal{A}, \mathcal{F})\) a GR(1) winning condition. Further, let
| \[
( F_b \cap \text{Cpre}(Y_k) ) \cup \text{Apre}(Y_k, b^k) \cup ( \bar{A}_a \cap \text{Cpre}(Y_a) ).
\]
Then \( Z^* \) is equivalent to the winning region \( W \) of Player 0
in the fair adversarial game over \( G^f \) for the winning condi-
tion \( \varphi \) in (28). Moreover, a finite-memory winning strategy for
Player 0 can be extracted from this fixpoint algorithm.

In particular, the extracted strategy is the same as the
strategy extracted for a "normal" GR(1) game in [23].

Remark 2. Svoreňová et al. presented a symbolic fixpoint
algorithm for stochastic games (which can be modeled using
fair adversarial games, see Sec. 5) with respect to GR(1) winning
conditions [29]. While one can show that the output of their
algorithm coincides with the output of our newly derived
fixpoint algorithm in (30), their algorithm is structurally more
involved. On a conceptual level, we feel our insight about simply
"swapping" predecessor operators in the right manner is
insightful even if one can also use their algorithm to find a
solution to this problem.

Fair Adversarial vs. Environmentally-Friendly GR(1)
Games: The idea of the simple "predecessor operator swapping
trick" shares resemblance with environmentally-friendly
GR(1) synthesis, proposed by Majumdar et al. [18]. There,
the authors show a direct symbolic algorithm to compute
Player 0 strategies which do not win a given GR(1) game vac-
uously, by rendering the assumptions false. More precisely,
given a synthesis game for the specification \( \varphi := (\varphi_A \rightarrow \varphi_G) \)
with \( \varphi_A \) and \( \varphi_G \) being LTL formulas modeling respectively
environment assumptions and system guarantees, Player 0
can win by violating \( \varphi_A \) and thereby satisfying \( \varphi \) vacuously.
Environmentally-friendly synthesis rules out such undesired
strategies by only computing so called non-conflicting win-
ning strategies. Interestingly, the fixpoint algorithm intro-
duced by Majumdar et al. [18] also swaps Cpre and Apre
operators, but in a slightly different way.

The GR(1) fragment considered by Majumdar et al. [18]
corresponds to a specification \( \varphi_A \rightarrow \varphi_G \) where both \( \varphi_A \)
and \( \varphi_G \) can be realized by a deterministic generalized Büchi
automaton. Hence, they provide an algorithm to compute
non-conflicting winning strategies in a deterministic gen-
eralized Büchi game under deterministic generalized Büchi
assumptions. If the used deterministic Büchi assumptions

---

\[^2\]The generalized Rabin pair \((G_{p_b}, R_{p_b})\) in (7) is artificially introduced to
make the fixpoint representation more compact. It is not part of \( \tilde{R} \).
can be translated into strong transition fairness assumptions, the resulting fair adversarial game is a generalized Büchi game (not a GR(1) game), solvable by the fixed point in (26) for \( Q = V \).

By reducing a GR(1) game to a fair adversarial game, one transforms the given assumption into one expressed by fair edges which cannot be falsified by Player 0 and therefore yields a simpler algorithm to compute non-conflicting strategies. However, the direct relationship between deterministic generalized Büchi assumptions and strong fair assumptions is not known, i.e., we do not know if we can reduce all environmentally-friendly GR(1) games to strong adversarial generalized Büchi games.

Finally, we want to point out that fair adversarial GR(1) games compute winning strategies that are only non-conflicting with respect to the environment assumptions encoded in the live edges. Player 0 can still win a fair adversarial GR(1) game vacuously by falsifying \( \phi_A \), i.e., never visiting any set \( A_i \) in \( \mathcal{A} \) (see (28)) infinitely often.

5 STOCHASTIC GENERALIZED RABIN GAMES

We present an important application of our fixpoint algorithm in solving stochastic two-player games, commonly known as 2\( \frac{1}{2} \)-player games. 2\( \frac{1}{2} \)-player games form an important subclass of abstract stochastic games, and has been studied quite extensively in the literature [5, 7, 31]. They can be seen as a generalization of two-player games by additionally capturing the environmental randomness inside the game. In order to do so, in addition to Player 0 and Player 1 vertices as in a two-player game, we introduce a new set of vertices called the random vertices. Whenever the game reaches a random vertex, one of the outgoing edges is picked randomly. We say that Player 0 wins the game almost surely if it wins the game with probability 1: we call the respective Player 0 strategy the almost sure winning strategy. Without loss of generality, we restrict ourselves to uniform distribution over the edges from a random vertex, since it is shown that such games with other distributions have exactly the same almost sure winning sets as the ones with uniform distributions [5].

We present a reduction from the computation of almost sure winning strategies in 2\( \frac{1}{2} \)-player generalized Rabin games to the computation of winning strategies in fair adversarial generalized Rabin games. This gives us a direct symbolic algorithm for solving 2\( \frac{1}{2} \)-player generalized Rabin games.

5.1 Preliminaries of 2\( \frac{1}{2} \)-player games

We start by introducing the basic setup of the 2\( \frac{1}{2} \)-player games.

The game graph: We consider usual 2\( \frac{1}{2} \)-player games played between Player 0, Player 1, and a third player representing environmental randomness. Formally, a 2\( \frac{1}{2} \)-player game graph is a tuple \( 
\mathcal{G} = (V, V_0, V_1, V_r, E) \) where (i) \( V \) is a finite set of vertices, (ii) \( V_0, V_1, \) and \( V_r \) are subsets of \( V \) which form a partition of \( V \), and (iii) \( E \subseteq V \times V \) is the set of directed edges. The vertices in \( V_r \) are called random vertices, and the edges originating in a random vertex are called random edges. The set of all random edges is denoted by \( E_r := E(V_r) \).

Strategies and plays: We will define strategies for Player 0 and Player 1 in exactly the same way as the strategies in two-player games. The new part is when the 2\( \frac{1}{2} \)-player game reaches a random vertex, the game chooses one of the random edges uniformly at random. A play is, as usual, an infinite sequence of vertices \( (v^0, v^1, \ldots) \) that satisfies the edge relation between two consecutive vertices in the sequence. Due to the presence of random edges, given an initial vertex \( v^0 \in V \) and a given pair of strategies \( \rho_0 \) and \( \rho_1 \) of Player 0 and Player 1 respectively, we will obtain a probability distribution over the set of plays. We denote the set of strategies of Player 0 and Player 1 by \( \Pi_0 \) and \( \Pi_1 \), respectively.

Almost sure winning: Let \( \phi \) be any \( \omega \)-regular specification over \( V \). Let us denote the event that the runs of a 2\( \frac{1}{2} \)-player game graph \( \mathcal{G} \) satisfies \( \phi \) using the symbol \( \mathcal{G} \models \phi \). For a given initial vertex \( v^0 \in V \) and for a given pair of strategies \( \rho_0 \) and \( \rho_1 \) of Player 0 and Player 1, we denote the probability of the occurrence of the event \( \mathcal{G} \models \phi \) by \( p_{v^0}^{\rho_0, \rho_1}(\mathcal{G} \models \phi) \). We define the set of almost sure winning states of Player 0 for the specification \( \phi \) as the set of vertices \( W_{a.s.} \subseteq V \) such that for every \( v \in W_{a.s.} \),

\[
\sup_{\rho_0 \in \Pi_0} \inf_{\rho_1 \in \Pi_1} p_{v^0}^{\rho_0, \rho_1}(\mathcal{G} \models \phi) = 1. \tag{31}
\]

5.2 The reduction

Suppose \( \mathcal{G} \) is a 2\( \frac{1}{2} \)-player game graph and \( \bar{\mathcal{R}} \) is a generalized Rabin winning condition. To obtain the reduced two-player game graph, we simply reinterpret the random vertices as Player 1 vertices and the random edges as live edges. Let us first formalize this notion of the reduced game graph.

**Definition 5.1 (Reduction to two-player game with live edges).**

Let \( \mathcal{G} = (V, V_0, V_1, V_r, E) \) be a 2\( \frac{1}{2} \)-player game graph. Define \( \text{Derand}(\mathcal{G}) := \langle (V, V_0, V_1, E'), E' \rangle \) as follows:

- \( V' = V \),
- \( V'_0 = V_0 \),
- \( V'_1 = V_1 \cup V_r \),
- \( E' = E \), and
- \( E'_r = E_r \).

It remains to show that the almost sure winning set of Player 0 in \( \mathcal{G} \) for the generalized Rabin winning condition \( \bar{\mathcal{R}} \)
is the same as the winning set of Player 0 in the fair adversarial game over Derand(\(G\)) for the winning condition \(\mathcal{R}\). This is formalized in the following theorem, which is proven in App. F. The proof essentially shows that the random edges of \(G\) simulate the live edges of Derand(\(G\)), and vice versa.

**Theorem 5.2.** Let \(G\) be a 2\(\frac{1}{2}\)-player game graph, \(\mathcal{R}\) be a generalized Rabin condition, \(\varphi \subseteq V^\omega\) be the corresponding LTL specification (Eq. (24)) over the set of vertices \(V\) of \(G\), and Derand(\(G\)) be the reduced two-player game graph. Let \(W \subseteq V\) be the set of all the vertices from where Player 0 wins the fair adversarial game over Derand(\(G\)) for the winning condition \(\varphi\), and \(W^{a.s.}\) be the almost sure winning set of Player 0 in the game graph \(G\) for the specification \(\varphi\). Then, \(W = W^{a.s.}\). Moreover, a winning strategy in Derand(\(G\)) is also a winning strategy in \(G\), and vice versa.

**REFERENCES**

[1] C. Baier and J.-P. Katoen. *Principles of Model Checking*. MIT Press, 2008.
[2] Christel Baier and Joost-Pieter Katoen. *Principles of model checking*. MIT press, 2008.
[3] Romain Brenguier, Guillermo A Pérez, Jean-François Raskin, and Ocan Sankur. Abssynthe: abstract synthesis from succinct safety specifications. *arXiv preprint arXiv:1407.5961*, 2014.
[4] J. Richard Buchi and Lawrence H. Landweber. Solving sequential conditions by finite-state strategies. *Transactions of the American Mathematical Society*, 138:295–311, 1969.
[5] Krishnendu Chatterjee, Luca de Alfaro, and Thomas A. Henzinger. The complexity of stochastic rabin and streett games. *In Luís Caires, Giuseppe F. Italiano, Luís Monteiro, Catuscia Palamidessi, and Moti Yung, editors, Automata, Languages and Programming, 32nd International Colloquium, ICALP 2005, Lisbon, Portugal, July 11-15, 2005, Proceedings*, volume 3580 of *Lecture Notes in Computer Science*, pages 878–890. Springer, 2005.
[6] Alonzo Church. Logic, arithmetic, and automata. *Proceedings of the International Congress of Mathematicians*, 1962, pages 23–35, 1963.
[7] Anne Condon. The complexity of stochastic games. *Information and Computation*, 96(2):203–224, 1992.
[8] Luca De Alfaro. *Formal verification of probabilistic systems*. Number 1601. Citeseer, 1997.
[9] Rüdiger Ehlers and Vasumathi Raman. Slugs: Extensible gr (1) synthesis. In *International Conference on Computer Aided Verification*, pages 333–339. Springer, 2016.
[10] E Allen Emerson and Charanjit S Jutla. The complexity of tree automata and logics of programs. In *FoCS*, volume 88, pages 328–337, 1988.
[11] E Allen Emerson and Charanjit S Jutla. On simultaneously determinizing and complementing omega-automata (extended abstract). In *Proceedings of the Fourth Annual Symposium on Logic in Computer Science*, pages 333–342. IEEE Computer Society, 1989.
[12] E Allen Emerson and Charanjit S Jutla. Tree automata, mu-calculus and determinacy. In *FoCS*, volume 91, pages 368–377, 1991.
[13] P. Francez. *Fairness*. Springer, Berlin, 1986.
[14] Yuri Gurevich and Leo Harrington. Trees, automata, and games. In *Proceedings of the fourteenth annual ACM symposium on Theory of computing*, pages 60–65, 1982.
[15] Dexter Kozen. Results on the propositional \(\mu\)-calculus. *Theoretical Computer Science*, 27(3):333 – 354, 1983. International Colloquium on Automata, Languages and Programming (ICALP).
[16] Orna Kupferman and Moshe Y Vardi. Safraless decision procedures. In *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS’05)*, pages 531–540. IEEE, 2005.
[17] David E Long, Anca Browne, Edmund M Clarke, Somesh Jha, and Wilfredo R Marreiro. An improved algorithm for the evaluation of fixpoint expressions. In *International Conference on Computer Aided Verification*, pages 338–350. Springer, 1994.
[18] Rupak Majumdar, Nir Piterman, and Anne-Kathrin Schmuck. Environmentally-friendly GR(1) synthesis. In *Tools and Algorithms for the Construction and Analysis of Systems*, pages 229–246, Cham, 2019. Springer International Publishing.
[19] Oded Maler, Amir Pnueli, and Joseph Sifakis. On the synthesis of discrete controllers for timed systems. In *Annual Symposium on Theoretical Aspects of Computer Science*, pages 229–242. Springer Berlin Heidelberg, 1995.
[20] Thibaud Michaud and Maximilien Colange. Reactive synthesis from ltl specification with spot. In *Proceedings of the 7th Workshop on Synthesis, SYNT@ CAV*, 2018.
A GENERAL LEMMAS

We first introduce some useful general lemmas.

**Lemma A.1.** If $Y \supseteq X$ then $Cpre(Y) \cup Apre(Y, X) = Cpre(Y)$.

**Proof.** The claim follows from the following derivation

\[
Cpre(Y) \cup Apre(Y, X) = Cpre(Y) \cup Cpre(X) \cup \left( \text{Pre}^\exists_X(Y) \right)
\]

\[
= Cpre(Y) \cup \left( \text{Pre}^\exists_X(Y) \cap \text{Pre}^\forall_Y(Y) \right)
\]

\[
= \left( Cpre(Y) \cup \text{Pre}^\exists_X(Y) \right) \cap \left( Cpre(Y) \cup \text{Pre}^\forall_Y(Y) \right)
\]

\[
= \left( Cpre(Y) \cup \text{Pre}^\exists_X(Y) \right) \cap Cpre(Y)
\]

\[
= Cpre(Y)
\]

where the second line follows from $Cpre(X) \subseteq Cpre(Y)$ (as $X \subseteq Y$) and the forth line follows as $Cpre(Y) = \text{Pre}^\exists_X(Y) \cup \text{Pre}^\forall_Y(Y)$. \hfill \Box

**Lemma A.2.** If $Y \subseteq X$ then $Apre(Y, X) = Cpre(X)$.

**Proof.** The claim follows from the following derivation

\[
Apre(Y, X) = Cpre(X) \cup \left( \text{Pre}^\exists_Y(X) \cap \text{Pre}^\forall_Y(Y) \right)
\]

\[
= \left( Cpre(X) \cup \text{Pre}^\exists_Y(X) \right) \cap \left( Cpre(X) \cup \text{Pre}^\forall_Y(Y) \right)
\]

\[
= \left( Cpre(X) \cup \text{Pre}^\exists_Y(X) \right) \cap Cpre(X)
\]

\[
= Cpre(X)
\]

where the fourth line follows as $Cpre(X) = \text{Pre}^\exists_Y(X) \cup \text{Pre}^\forall_Y(Y) \supseteq \text{Pre}^\forall_Y(Y)$ as $Y \subseteq X$. \hfill \Box

**Lemma A.3.** Let $f(X, Y)$, $g(X, Y)$, $h_a(X, Y)$ and $h_b(X, Y)$ be functions which are monotone in both $X \subseteq V$ and $Y \subseteq V$. Further, let

\[
Z := \nu Y_a. \mu X_a. \nu Y_b. \mu X_b. \left( h_a(X_a, Y_a) \cup f(X_a, Y_a) \right) \cup (h_b(X_b, Y_b) \cup g(X_b, Y_b))
\]

\[
\tilde{Z} := \nu \tilde{Y}_a. \mu \tilde{X}_a. \nu \tilde{Y}_b. \mu \tilde{X}_b. \left( h_a(\tilde{X}_a, \tilde{Y}_a) \cup f(\tilde{X}_a, \tilde{Y}_a) \right) \cup g(\tilde{X}_b, \tilde{Y}_b)
\]

\[
\check{Z} := \nu \check{Y}_a. \mu \check{X}_a. \nu \check{Y}_b. \mu \check{X}_b. \left( h_a(\check{X}_a, \check{Y}_a) \cup f(\check{X}_a, \check{Y}_a) \right) \cup (h_b(\check{X}_b, \check{Y}_b) \cup g(\check{X}_b, \check{Y}_b))
\]

Then

(i) $Z = \check{Z}$ if $h_b(X, Y) \subseteq h_a(X, Y)$ for all $X, Y \subseteq V$,

(ii) $Z = \tilde{Z}$ if $h_a(X, Y) \subseteq h_b(X, Y)$ for all $X, Y \subseteq V$, and

(iii) $Z = \check{Z} = \tilde{Z}$ if $h_a(X, Y) = h_b(X, Y)$ for all $X, Y \subseteq V$.

**Proof.** We first observe that (iii) is a direct consequence of (i) and (ii). We prove (i) in step 1-2 and (ii) in step 3-4 below.
\textbf{Step 1:} First we show that if }h_b(X, Y) \subseteq h_a(X, Y)\text{ it holds for all } l > 0\text{ that}
\begin{equation}
Y_a^{l+1} = h_a(Y_a^{l+1}, Y_a^l) \cup f(Y_a^{l+1}, Y_a^l) \cup g(Y_a^{l+1}, Y_a^l).
\end{equation}
\text{For this purpose observe that}
\begin{equation*}
X_b^{ljm+1} = h_a(X_b^{lj}, Y_b^l) \cup f(X_b^{lj}, Y_b^l) \cup h_b(0, Y_b^{ljm}) \cup g(0, Y_b^{ljm}).
\end{equation*}
\text{Now recall that }X_b^{ljm0} = \emptyset \subseteq X_b^{ljm1}\text{. We can generalize this inclusion to arbitrary } i > 1\text{ by utilizing the monotonicity of } f, g\text{ to observe the following derivation:}
\begin{equation*}
X_b^{ljmi+1} = h_a(X_b^{lj}\cup f(X_b^{lj}, Y_b^l) \cup h_b(X_b^{ljmi}, Y_b^{ljm}) \cup g(X_b^{ljmi}, Y_b^{ljm})
\end{equation*}
\text{obviously implying } X_b^{ljmi} \subseteq X_b^{ljmi+1}\text{ for all } i \geq 0.\text{ With this we have}
\begin{equation}
y_b^{ljmi+1} = \bigcup_{i \geq 0} X_b^{ljmi} = X_b^{ljm}
\end{equation}
\begin{equation*}
= h_a(X_b^{lj}, Y_b^l) \cup f(X_b^{lj}, Y_b^l) \cup h_b(X_b^{ljmi}, Y_b^{ljm}) \cup g(X_b^{ljmi}, Y_b^{ljm}).
\end{equation*}
\text{Here, } X_b^{ljm} : = X_b^{ljmi} \text{ where } i^1\text{ is the iteration in which the fixed-point over } X_b^{ljmi}\text{ is attained.}
\text{Now recall that } Y_b^{lj0} = V \text{ and therefore } Y_b^{lj0} \subseteq Y_b^{lj}.\text{ Hence, we can assume that } Y_b^{ljm} \subseteq Y_b^{ljm-1}\text{ to see that } X_b^{ljmi} \subseteq X_b^{ljmi-1}\text{ and therefore, subsequently, that } Y_b^{ljmi+1} \subseteq Y_b^{ljm}\text{ (again due to the monotonicity of } h, f, g).\text{ With this we see that}
\begin{equation}
X_a^{ljm+1} = \bigcup_{m \geq 0} Y_b^{ljm} = Y_b^{lj}
\end{equation}
\begin{equation}
= h_a(X_a^l, Y_a^l) \cup f(X_a^l, Y_a^l) \cup h_b(X_a^{ljmi}, Y_b^{ljm}) \cup g(X_a^{ljmi}, Y_b^{ljm}).
\end{equation}
\text{Again, } Y_b^{lj} = Y_b^{ljm} \text{ where } m^1\text{ is the iteration in which the fixed-point over } Y_b^{ljm}\text{ is attained. As we know that } Y_b^{ljm} = X_b^{ljm+1}\text{ we can replace } X_b^{ljmi+1}\text{ with } Y_b^{lj}\text{ above. With this, we have proven that}
\begin{equation}
X_a^{lj+1} = h_a(X_a^l, Y_a^l) \cup f(X_a^l, Y_a^l) \\
\quad \cup h_b(X_a^{lj+1}, X_a^{lj+1}) \cup g(X_a^{lj+1}, X_a^{lj+1})
\end{equation}
\text{for any } j \geq 0.\text{ Using the same monotonicity argument as before we again get } X_a^{lj+1} \supseteq X_a^{lj}\text{ and therefore}
\begin{equation*}
y_a^{lj+1} = \bigcup_{j \geq 0} X_a^{lj+1} = Y_a^l
\end{equation*}
\begin{equation*}
= h_a(Y_a^{lj+1}, Y_a^l) \cup f(Y_a^{lj+1}, Y_a^l) \\
\quad \cup h_b(Y_a^{lj+1}, Y_a^l) \cup g(Y_a^{lj+1}, Y_a^l)
\end{equation*}
\text{Utilizing the monotonicity argument one more time we get that } Y_a^l \supseteq Y_b^{lj+1}.\text{ With this it follows from the monotonicity of } h_b\text{ that}
\begin{equation}
h_b(Y_a^{lj+1}, Y_a^l) \subseteq h_b(Y_a^{lj+1}, Y_a^l) \subseteq h_a(Y_a^{lj+1}, Y_a^l)
\end{equation}
\text{and therefore}
\begin{equation}
y_a^{lj+1} = h_a(Y_a^{lj+1}, Y_a^l) \cup f(Y_a^{lj+1}, Y_a^l) \cup g(Y_a^{lj+1}, Y_a^l)
\end{equation}
\text{what proves the claim.}
\textbf{Step 2:} By utilizing (36), we now show that } Z = \overline{Z}.\text{ As } \overline{Z} = Y_a^1,\text{ it follows from the fixpoint equation defining } \overline{Z}\text{ that } \overline{Z} \text{ is the unique largest set of states s.t. } Y_a^1 = Y_a^1\text{ and }
\overline{Z} = h(\overline{Z}, \overline{Z}) \cup f(\overline{Z}, \overline{Z}) \cup g(\overline{Z}, \overline{Z}).
\end{equation}
\text{It now follows from (36) and the fact that } Z = Y_a^1,\text{ that } Z \text{ is also the unique largest set s.t. } Y_a^1 = Y_a^1\text{ and thereby fulfills equation (37). Hence } Z \text{ and } \overline{Z}\text{ must be equivalent.}
\textbf{Step 3:} We now show that if } h_b(X, Y) \subseteq h_b(X, Y)\text{, it holds for } l = l^1\text{ the corresponding } f = \overline{f}\text{ and any } m \geq 0\text{ that}
\begin{equation}
y_b^{lj,j+1} = f(X_a^{lj}, Y_a^l)
\end{equation}
\begin{equation}
\cup h_b(X_b^{ljmi}, Y_b^{ljm}) \cup g(X_b^{ljmi}, Y_b^{ljm}).
\end{equation}
\text{First observe that for arbitrary } l, j, m, i \geq 0\text{ it holds that}
\begin{equation}
X_b^{ljmi+1} = h_a(X_b^{lj}, Y_b^l) \cup f(X_b^{lj}, Y_b^l) \\
\quad \cup h_b(X_b^{ljmi}, Y_b^{ljm}) \cup g(X_b^{ljmi}, Y_b^{ljm}).
\end{equation}
\text{When re-initializing the inner FP, we have } X_b^{lj} \supseteq X_b^{l0} = \emptyset \text{ and } Y_a^l \subseteq Y_b^{lj} = V.\text{ Hence, the two } h\text{-terms are incomparable. However, we see that whenever } X_b^{ljmi} \supseteq X_b^{lj}\text{ (while still } Y_a^l \subseteq Y_b^{lj})\text{ we have}
\begin{equation}
h_a(X_b^{lj}, Y_b^l) \subseteq h_b(X_b^{ljmi}, Y_b^{ljm}) \subseteq h_b(X_b^{ljmi}, Y_b^{ljm})
\end{equation}
\text{and we get}
\begin{equation}
X_b^{ljmi+1} = f(X_b^{lj}, Y_b^l) \cup h_b(X_b^{ljmi}, Y_b^{ljm}) \cup g(X_b^{ljmi}, Y_b^{ljm}).
\end{equation}
\text{Now we know from the structure of this fixed-point that we keep increasing } X_b\text{ until}
\begin{equation}
X_b^{ljmi+1} = f(X_b^{lj}, Y_b^l) \cup h_b(X_b^{ljmi}, Y_b^{ljm}) \cup g(X_b^{ljmi}, Y_b^{ljm})
\end{equation}
\text{where } Y_b^{ljmi} = X_b^{ljmi} \subseteq Y_b^{ljm}.\text{ Further, we know that the fixed-point is attained over } Y_b\text{ if equality holds. It remains to show that for } l = l^1\text{ and } j = j^1\text{ the set } Y_b\text{ will never be empty.}
get smaller then $Y_a$ (as this would render the two $h$ terms incomparable again). I.e., we need to show that for all $m$ we have $Y_a^{l,m} \subseteq Y_b^{l,m}$. To see this, recall that $l$ and $j$ are such that the fixed-point over $X_a$ and $Y_a$ is already attained. That is, we know that $X_a^{l,j+1} = X_a^{l,j}$. It further follows from the structure of the fixed-point that $Y_b^{l,j+1} = X_a^{l,j+1}$. With this, it follows from the monotonicity of the fixed-point that $Y_b^{l,m} \subseteq Y_b^{l,m}$ for every $m \geq 0$.

**Step 4:** By utilizing (38), we now show that $Z = \tilde{Z}$.

Immediately follows from the structure of the fixpoint equation defining $\tilde{Z}$ that we similarly have

$$
\tilde{Y}_b^{l,j+1} = f(\tilde{X}_a^{l,j}, \tilde{Y}_a^{l,j}) \\
\cup h(\tilde{Y}_b^{l,j+1}, \tilde{Y}_b^{l,j+1}) \cup (\tilde{Y}_b^{l,j+1}, \tilde{Y}_b^{l,j+1}).
$$

(39)

By further observing that $Z = Y_b^{l,j} = X_b^{l,j}$ and $\tilde{Z} = \tilde{Y}_b^{l,j} = \tilde{X}_b^{l,j}$ we see that both $Z$ and $\tilde{Z}$ are the unique largest sets s.t. the inner fixpoint computations over $X_b$ and $Y_b$ (resp. $\tilde{X}_b$ and $\tilde{Y}_b$) converge to $Z$ (resp. $\tilde{Z}$) via the same fix-point equation in (38) (resp. (39)). This proves that $Z = \tilde{Z}$. □

**Lemma A.4.** Let $f(X, Y)$ and $g(X, Y)$ be two functions which are monotone in both $X \subseteq V$ and $Y \subseteq V$. Further, let

$$
Z := \nu Y_a, \mu X_a, \nu Y_b, \mu X_b, f(X_a, Y_a) \cup g(X_b, Y_b)
$$

$$
\tilde{Z} := \nu Y_a, \mu X_a, \nu Y_b, \mu X_b, g(X_a, Y_a) \cup f(X_b, Y_b)
$$

$Z_c := \nu Y_c, \mu X_c, f(X_c, Y_c)

Then it holds that

(i) $Z_c \subseteq Z$

(ii) $Z_c \subseteq \tilde{Z}$

If, in addition, $g(X, Y) \subseteq f(X, Y)$ for all $X, Y \subseteq V$, then it holds that

(iii) $Z = Z_c$

(iv) $\tilde{Z} = Z_c$.

**Proof.** We prove all claims separately:

- **(i)** $Z_c \subseteq Z_a^*$: First, consider a stage of the fixed point evaluation where $Y_a$ and $X_a$ have their initialisation value $Y_0^X = V$ and $X_0^X = \emptyset$ (here, the notation $X_a^{l,k}$ refers to the value of $X_a$ computed in the $k$th iteration over $X_a$ using the value for $Y_a$ computed in the $l$th iteration over $Y_a$). Then we see that $X_a^{l,k} = Y_b^{l,k}$ where $Y_b^{l,k} = f(0, V) \cup (Y_b^{l,k}, Y_b^{l,k})$. We therefore see that $X_a^{l,1} \supseteq X_a^{l,1} = f(0, V)$. With this, it follows from the monotonicity of $f$ and $g$ that $Y_a^{l,1} \supseteq X_a^{l,1} = Y_c^{l,1}$. With this, we see that $X_a^{l,m} = X_c^{l,m}$ for all $m > 0$ and therefore $Z_a = Y_a^* \supseteq Y_c^* = Z_c$.

- **(ii)** $Z_c \subseteq Z_b^*$: Consider arbitrary values $Y_b^m$ and $X_b^m$ and assume that $Y_b$ and $X_b$ have their initialisation value, i.e., $Y_b^0 = V$ and $X_b^0 = \emptyset$. Then we have

$$
X_b^{mn} = g(X_a^m, Y_a^m) \cup f(0, V) \supseteq X_c^{l,1}.
$$

Using the same reasoning as in the previous part, we see that this implies $Y_b^{mn} \supseteq Y_c^* = Z_c$. As this holds for any $m$ and $n$ it also holds when the fixed point over $Y_b$ and $X_a$ is obtained, i.e., when we have $Z_a = Y_a^* = Y_b^{mn}$, which proves the statement.

- **(iii)-(iv)** This is a simple consequence of Lem. A.3 (iii). It follows by choosing $f(X, Y) = \emptyset$ and $g(X, Y) = \emptyset$ in Lem. A.3 and interpreting $h_a$ as $f$ and $h_b$ as $g$ to show (iii) and $h_a$ as $g$ and $h_b$ as $f$ to show (iv). □

**B ADDITIONAL PROOFS FOR SEC. 3**

**B.1 Proof of Thm. 3.3**

We denote by $Y^m$ the $m$-th iteration over the fixpoint variable $Y$ in (12), where $Y^0 = V$. Further, we denote by $X^{m,i}$ the set computed in the $i$-th iteration over the fixpoint variable $X$ in (12) during the computation of $Y^m$ where $X^{m,0} = \emptyset$. Then it follows form (12) that

$$
X^{m,1} = X^{m,0} \cup T \cup (Q \cap \text{Apree}(Y^{m,1}, X^{m,0}))
$$

$$
= \emptyset \cup T \cup (Q \cap \text{Apree}(Y^{m,0}, \emptyset)) = T,
$$

$$
X^{m,2} = X^{m,1} \cup T \cup (Q \cap \text{Apree}(Y^{m,1}, X^{m,1}))
$$

$$
= T \cup (Q \cap \text{Apree}(Y^{m-1}, X^i)) \supseteq X^{m-1},
$$

and therefore, in general,

$$
X^{m,i+1} = T \cup (Q \cap \text{Apree}(Y^{m-1}, X^{m,i})) \supseteq X^{m,i}.
$$

With this, the fixpoint over $X$ corresponds to the set $X^{m,*} = \bigcup_{i=0}^{\infty} X^i = X^{m,i_{\text{max}}}$, where $i_{\text{max}}$ is the iteration where the fixpoint over $X$ is attained.

Now consider the computation of $Y$. Here we have $Y^0 = V$ and $Y^m = Y^{m-1} \cap X^{m,*} \subseteq Y^{m-1}$ where equality holds when a fixpoint is reached. Hence, in particular we have $Y^* = X^{m,*} = Z^*$. For simplicity we denote $X^{i,*}$ by $X^i$.

**Strategy construction.** In order to construct a winning strategy for Player 0 from (12), we construct a ranking over $V$ by choosing

$$
\text{rank}(v) = \begin{cases} 
1 & \text{if } v \in E \setminus X^{l-1} \\
\infty & \text{if } v \notin Z^*.
\end{cases}
$$

(40)

As $X^0 = \emptyset, X^1 = T$ (from above) and $Z^* = \bigcup_{i=0}^{\infty} X^i$, it follows that $\text{rank}(v) = 1$ iff $v \in T$ and $1 < \text{rank}(v) < \infty$ iff $v \in Z^* \setminus T$. Using this ranking we define a Player 0 strategy $\rho_0 : V_0 \rightarrow V$ s.t.

$$
\rho_0(v) = \min_{(c, w) \in E} \text{rank}(w).
$$

(41)

We next show that this player 0 strategy is actually winning w.r.t. $\psi$ (in (11)) in every fair adversarial play over $G^f$.

**Soundness.** To prove soundness, we need to show $Z^* \subseteq W$. That is, we need to show that for all $v \in Z^*$ there exists a strategy for player 0 s.t. the goal set $T$ is eventually reached along all live plays $\pi$ starting at $v$ while staying in $Q$. We choose $\rho_0$ in (41) and show that the claim holds.
First, it follows from the definition of $Apre$ that for a vertex $v \in Z^*$ exactly one of the following cases holds:
(a) $v \in T$ and hence $\text{rank}(v) = 1$,
(b) $v \in (V_0 \cap Z^*) \setminus T$, i.e., $1 < \text{rank}(v) < \infty$ and $v \in Q$ and there exists a $v' \in E(v)$ with $\text{rank}(v') < \text{rank}(v)$,
(c) $v \in ((V_1 \setminus V_2) \cap Z^*) \setminus T$, i.e., $1 < \text{rank}(v) < \infty$ and $v \in Q$ and for all $v' \in E(v)$ it holds that $\text{rank}(v') < \text{rank}(v)$, or
((l) $v \in (V_r \cap Z^*) \setminus T$, i.e., $1 < \text{rank}(v) < \infty$ and $v \in Q$ and there exists a $v' \in E^l(v)$ with $\text{rank}(v') < \text{rank}(v)$ and $E(v) \subseteq Z^*$.

We see that $\rho_0(v)$ chooses one existentially quantified edge in (b) vertices. In all other cases player 1 chooses the successor.

Further, we see that any play $\pi$ which starts in $\pi(0) = v \in Z^*$ and obeys $\rho_0$ has the property that $\pi(k) \in Z^* \setminus T$ implies $\pi(k) \in Q$ and $\pi(k+1) \in Z^*$ for all $k \geq 0$. This, in turn, means that for any such state $v = \pi(k) \in Z^* \setminus T$ as well as for its successor $\pi(k+1)$ a rank is defined, i.e., $\pi(k) \in X^1$ for some $0 < i < \infty$ and exactly one of the cases (b)-(l) applies. We call a vertex for which case (a) applies, an (a) vertex.

Now observe that the above reasoning implies that whenever an (a) vertex is hit along a play $\pi$ the claim holds. We therefore need to show that any play starting in $v \in Z^*$ eventually reaches an (a) vertex. First, consider a play in which no (l) vertex occurs. Then constantly hitting (b) and (c) vertices always reduces the rank of visited states (as we assume that $\pi$ obays $\rho_0$ in (41)). As the maximal rank is finite, we see that we must eventually hit a state with rank 1 which is an (a) state.

Note that the same argument holds when only a finite number of (l) vertices are visited along $\pi$. In this case we know that from some time onward no more (l) vertex occurs. As the last (l) vertex has a finite rank, there can only be a finite sequence of (b) and (c) vertices afterwards until finally an (a) vertex is reached.

We are therefore left with showing that on every path with an infinite number of (l) vertices, eventually an (a) vertex will be reached. We prove this claim by contradiction. I.e., we show that there cannot exist a path with infinitely many (l) vertices and no (a) vertex.

We first show that infinitely many (l) vertices and no (a) vertices in $\pi$ imply that vertices with rank 2 can only occur finitely often along $\pi$.

> Now assume that $v \in V_1 \cap Z^*$ with $\text{rank}(v) = 2$. If $v$ is a (c) vertex all successor states will have rank 1. With the same reasoning as before, this cannot occur.

> Now assume that $v \in V_1 \cap Z^*$ with $\text{rank}(v) = 2$ is labeled with $(l)$. In this case there surely exists a successor $v'$ of $v$ s.t. $(v, v') \in E^l$ and rank$(v') = 1$. But there might also exist another successor $v''$ of $v$ i.e., $(v'' \in E(v))$ s.t. rank$(v'') > 1$.

If there does not exist such a successor $v''$, all successors have rank 1 and we again cannot visit $v$.

> Now assume that $v \in V_1 \cap Z^*$ with $\text{rank}(v) = 2$, labeled with $(l)$ and there exists a successor $v'' \in E(v)$ s.t. rank$(v'') > 1$. Now let us assume that such a state $v$ is visited infinitely often along $\pi$. As $\pi$ is a live-sufficient play over $G$ we know that visiting $v$ infinitely often along $\pi$ implies that $v''$ with $(v, v') \in E^l$ and rank$(v') = 1$ (which surely exists by the definition of $Apre$) will also be visited infinitely often along $\pi$. This is again a contradiction to the above hypothesis and implies that such $v$'s can only be visited finitely often.

> As $V$ is a finite set, the set of states with rank 2 is finite. Hence, the occurrence of infinitely many states with rank 2 along $\pi$ implies that one of the above cases must occur infinitely often, which gives a contradiction to the above hypothesis. Using the same arguments, we can inductively show that states with any fixed rank can only occur finitely often if states with rank 1 (i.e., (a)-labeled vertices) never occur. As the maximal rank is finite (due to the finiteness of $V$) this contradicts the assumption that $\pi$ is an infinite play.

We therefore conclude that along any infinite live-sufficient play $\pi$ with infinitely many vertices labeled by $(l)$ we will eventually see a vertex labeled by (a).

Completeness. We now show why the fixpoint algorithm in (12) is complete. For this purpose, let $\mathcal{W}$ be the set of states from which Player 0 has a strategy, such that every fair adversarial play compliant with this strategy is winning w.r.t. $\psi$ in (11). Then we show that $\mathcal{W} \subseteq Z^*$ holds.

In particular, we show that $\mathcal{W} \subseteq Y^m$ for all $m > 0$, where $Y^m$ is the set computed in the $m$-th iteration over $Y$ in (12), initialized by $Y^0 := V$. This obviously implies $\mathcal{W} \subseteq \bigcap_{m > 0} Y^m := Z^*$. It is easy to see that the base case for the induction over $m$ is obvious as $Y^0 = V$, hence $\mathcal{W} \subseteq Y^0$.

We therefore only need to show the induction step over $m$, i.e., we show that $\mathcal{W} \subseteq Y^{m-1}$ implies $\mathcal{W} \subseteq Y^m$.

Now observe that for any $v \in \mathcal{W}$ we know that there exists a system strategy $\rho_0$ s.t. $\psi$ holds on every live play compliant with $\rho_0$ starting in $v$. I.e., there exists a finite sequence of states which are all contained in $Q$ and successively leading to $T$. We can therefore label all states in $\mathcal{W}$ inductively by their distance to $T$. I.e., we define a labeling function $\lambda$ s.t. the following holds:
(a) If $v \in T$ we have $\lambda(v) = 1$.
(b) If $v \in (V_0 \cap Q) \setminus T$ we have $\lambda(v) = 1 + \min_{(v', v') \in E} \lambda(v')$.
(c) If $v \in ((V_1 \setminus V_2) \cap Q)) \setminus T$ we have $\lambda(v) = 1 + \max_{(v', v') \in E} \lambda(v')$.
(d) If $v \in (V_r \cap Q) \setminus T$ we have $\lambda(v) = 1 + \min_{(v', v') \in E} \lambda(v')$ if $\max_{(v', v') \in E} \lambda(v') < \infty$ and $\lambda(v) = \infty$ otherwise.
(e) Else \(\lambda(v) = \infty\).
Further we define \(\Lambda := \{v \in V \mid \lambda(v) < \infty\}\).

The remainder of the completeness proof consists of three steps. Remember that we need to prove the induction step over \(m\), i.e., we assume that \(W \subseteq Y^{m-1}\) and show that this implies that also \(W \subseteq Y^m\). To utilize the induction hypothesis, we first show that \(\Lambda \subseteq W\) (Step 1) which implies \(\Lambda \subseteq Y^{m-1}\). We then show that \(W \subseteq \Lambda\) (Step 2) which allows us to reduce the proof for \(W \subseteq Y^m\) to proving \(\Lambda \subseteq Y^m\) (Step 3). Within Step 3 we then use the induction hypothesis \(\Lambda \subseteq Y^{m-1}\).

**Step 1: Show \(\Lambda \subseteq W\):** We choose \(v \in \Lambda\), i.e., \(\lambda(v) < \infty\) and need to prove that there exists a strategy \(\rho_0\) s.t. on all compliant plays \(Q\) is true until \(T\) is eventually visited. We first show that for \(\lambda(v) = 2\) the claim holds.

- Let \(v \in V_0\) s.t. \(\lambda(v) = 2\), i.e., case (b) holds for \(v\). Then we know that \(v \in Q\) and there must exist a strategy \(\rho_0\) s.t. the next state compliant with \(\rho_0\) has label 1. i.e., \(T\) is visited along all compliant plays.

- Let \(v \in V_1\) \(\setminus V_0\) s.t. \(\lambda(v) = 2\), i.e., case (c) holds for \(v\). Then we know that \(v \in Q\) and all successor states of \(v\) (which are defined by all compliant with any \(\rho_0\)) have label 1. i.e., \(T\) is visited along all compliant plays.

- Let \(v \in V_2\) s.t. \(\lambda(v) = 2\), i.e., case (d) holds for \(v\). In this case \(v \in Q\) and there exists a successor \(v' \in E'(v)\) s.t. \(\lambda(v') = 1\). But there might also exist another successor \(v'' \in E(v)\) s.t. \(1 < \lambda(v'') < \infty\). If there does not exist such a successor \(v''\), all successors have label 1 and the claim again holds for any strategy \(\rho_0\).

- Now assume that \(v \in V_2\) s.t. \(\lambda(v) = 2\) (i.e., \(v \in Q\)) and there exists a successor \(v'' \in E(v)\) s.t. \(\lambda(v'') > 1\). First, observe that also \(v'' \in Q\). For the purpose of contradiction, let us assume that a state \(v \in V_2\) s.t. \(\lambda(v) = 2\) is visited infinitely often along \(\pi\) without visiting \(T\). As \(\pi\) is a fair adversarial play over \(G\) we know that visiting \(v\) infinitely often along \(\pi\) implies that \(v'' \in E'(v)\) with \(\lambda(v') = 1\) (which surely exists by construction) will also be visited infinitely often along \(\pi\). As the state space is finite, by contradiction, \(\pi\) must eventually visit \(T\) (while always remaining in \(Q\) before by construction). I.e., \(v \in W\). Using the same reasoning inductively, we can show that whenever a state \(v \in V_2\) s.t. \(\lambda(v) = n < \infty\) is visited infinitely often along a fair adversarial play \(\pi\), \(\pi\) always remains in \(Q\) and a state in \(T\) is eventually reached. I.e., \(v \in W\). As the state space is finite, this implies that every infinite fair adversarial play \(\pi\) starting in \(\Lambda\) fulfills \(\psi\).

**Step 2: Show \(W \subseteq \Lambda\):** Observe that the claim \(W \subseteq \Lambda\) is equivalent to the inverse statement \(V \setminus \Lambda \subseteq V \setminus W\). Using this insight, we show that all states \(v\) with \(\lambda(v) = \infty\) are contained in \(V \setminus W\). That is, for any strategy \(\rho_0\) there exists a play compliant with \(\rho_0\) s.t. \((QUT)\) holds along \(\pi\) what implies that also the weaker property \(\square \neg T\) holds along \(\pi\).

We can therefore show this claim by constructing a play \(\pi\) compliant with \(\rho_0\) and starting with \(\pi(0) = v\) s.t. \(\lambda(\pi(l)) = \infty\) for all \(l \in \mathbb{N}\) under an arbitrary but fixed winning strategy \(\rho_0\). As \(\lambda(v') = 1\) for all \(v' \in T\), this proves that \(\square \neg T\) holds along \(\pi\).

We construct \(\pi\) inductively. The base case is simply \(\pi(0) = v\) and we have assumed that \(\lambda(v) = \infty\). Now let \(\lambda(\pi(l)) = \infty\), then we show how to construct \(\pi\) s.t. \(\lambda(\pi(l+1)) = \infty\). We have the following cases:

- Let \(\pi(l) \in V_0\) and define \(\pi(l+1) = \rho_0(\pi(l))\) for some arbitrary strategy \(\rho_0\). This implies \(\pi(l+1) \in E(\pi(l))\). Now, recall that \(\lambda(\pi(l+1)) = 1 + \min_{v' \in E(\pi(l))} \lambda(v') = \infty\). With this, it follows that \(\lambda(\pi(l+1)) = \infty\).

- Let \(\pi(l) \in V_1\), then any \(v'\) s.t. \(\pi(l), v' \in E\) is compliant with any arbitrary strategy \(\rho_0\). Further, it follows from (c) and (d) that we can only have \(\lambda(\pi(l)) = \infty\) if there exists a successor \(v'\) s.t. \(\lambda(v') = \infty\). We pick \(\pi(l+1) = v'\) s.t. \(\lambda(v') = \infty\) and get \(\lambda(\pi(l+1)) = \infty\).

**Step 3: Show \(\Lambda \subseteq Y^m\) given that \(\Lambda \subseteq Y^{m-1}\):** Let \(X^{m,n}\) be the set computed in the \(n\)-th iteration over \(X\) during the \(m\)-th iteration over \(Y\) where \(X^{0,0} = \emptyset\) for all \(m \geq 0\). Hence, it follows from the structure of the fixpoint algorithm in (12) that

\[
X^{m,n} = T \cup (Q \cap \text{Apref}(Y^{m-1}, X^{m,n-1})).
\]

Further, observe that the structure of the fixpoint algorithm implies \(Y^m = \bigcup_{n>0} X^{m,n}\). In order to prove \(\Lambda \subseteq Y^m\) we therefore prove that \(\lambda(v) = n\) implies \(v \in X^{m,n}\) for all \(0 < n < \infty\). We show this claim by an induction over \(n\).

- (base case) For \(\lambda(v) = 1\) we have \(v \in T\) and it trivially follows from (42) that \(v \in X^1\).

- (induction step) Now assume that for any \(v\) s.t. \(\lambda(v) = n-1\) holds that \(v \in X^{m,n-1}\) and show that for any \(v\) s.t. \(\lambda(v) = n\) also holds that \(v \in X^{m,n}\). We have the following cases:

  (i) Let \(1 < \lambda(v) = n < \infty\) and \(v \in V_0\). As \(v \in \Lambda\) we know that case (b) holds and hence \(v \in Q\) and there exists \(v' \in E(v)\) s.t. \(\lambda(v') = n-1\), hence \(v' \in X^{m,n-1}\). With this it follows from the definition of Cpre and Apref that \(v \in \text{Pre}^2(X^{m,n-1}) \subseteq \text{Cpre}(X^{m,n-1}) \subseteq \text{Apref}(X^{m,n-1})\). With this, it follows from (42) that \(v \in X^{m,n}\).

  (ii) Let \(1 < \lambda(v) = n < \infty\), \(v \in V_1 \setminus V_T\). Then case (c) holds and we know that \(v \in Q\) and for all \(v' \in E(v)\) holds that \(\lambda(v') \leq n-1\) and therefore \(v' \in X^{m,n-1}\). Therefore \(v \in \text{Pre}^2(X^{m,n-1}) \subseteq \text{Cpre}(X^{m,n-1}) \subseteq \text{Apref}(X^{m,n-1})\). With this, it follows from (42) that \(v \in X^{m,n}\).

  (iii) Let \(1 < \lambda(v) = n < \infty\), \(v \in V_2\) \(\setminus T\). Then we know that \(v \in Q\) and there exists a successor \(v' \) s.t. \(\lambda(v') \in E^f\) and \(\lambda(v') = n-1\) and hence \(v' \in X^{m,n-1}\). This implies \(v \in \text{Pre}^2(X^{m,n-1})\). Further, we know that for all successors \(v'' \in E(v)\) holds that \(\lambda(v'') < \infty\) and hence \(v'' \in \Lambda\). With the induction hypothesis \(\Lambda \subseteq Y^{m-1}\) (derived from Step 1 together with the hypothesis \(W \subseteq Y^{m-1}\)) we further have

Symbolic Algorithms for Omega-Regular Games
MPI-SWS-2020-007, Technical Report, December 2020
(4) that $v \in X^{m,n}$. 

B.2 Proof of Thm. 3.2

In order to simplify the proof of Prop. B.2, we first prove the following lemma.

**Lemma B.1.** Let $Q, G \subseteq V$ and

\[
Z^* := \nu Y. \mu X. Q \cap \{(G \cap \text{Cpre}(Y)) \cup \text{Apres}(Y, X)\} \\
\bar{Z}^* := \nu \bar{Y}. \nu Y. \mu X. Q \cap \left[\left(\bar{G} \cap \text{Cpre}(\bar{Y})\right) \cup \text{Apres}(Y, X)\right].
\]

Then $Z^* = \bar{Z}^*$.

**Proof.** We prove this lemma by a reduction to Lem. A.3 (iii). For this purpose, we define

\[
f(\bar{X}, \bar{Y}) := 0, \\
h_a(\bar{X}, \bar{Y}) := Q \cap G \cap \text{Cpre}(\bar{Y}), \\
g(X, Y) := Q \cap \text{Apres}(Y, X), \\
h_b(X, Y) := Q \cap G \cap \text{Cpre}(Y).
\]

With this we see that (43a) can be equivalently written as

\[
\nu \bar{Y}. \nu X. f(\bar{X}, \bar{Y}) \cup h_b(X, Y) \cup g(X, Y)
\]

while (43b) can be written as

\[
\nu \bar{Y}. \nu X. f(\bar{X}, \bar{Y}) \cup h_a(X, \bar{Y}) \cup g(X, Y).
\]

With this, it follows from Lem. A.3 (iii) that both equations are equivalent. \hfill \square

With Lem. B.1 in place, we can use (43b) instead of (10) to prove Thm. 3.2. Further, let us define $Z'(\langle T, Q \rangle)$ to be the set of states computed by the fixpoint algorithm in (12). Then we know that upon termination we have

\[
\bar{Z}^* = \bar{Y}^* = Z'((Q \cap G \cap \text{Cpre}(\bar{Y}^*), Q)).
\]

(44)

Now we will use (44) to prove soundness and completeness of Thm. 3.2.

**Soundness** Let us now define $T := Q \cap G \cap \text{Cpre}(\bar{Y}^*)$. Pick any state $v \in Z^*$ and the strategy $\rho_0$ defined as in (41) over the sets $X^i$ computed in the last iteration over $X$ when computing $Z'(\langle T, Q \rangle)$. Further, let $\pi$ be an arbitrary fair adversarial play starting in $v$ and being compliant with $\rho_0$. Then we need to show that $\pi$ fulfills $\psi$ in (8).

Using (44) and the fact that $v \in \bar{Z}^*$ we know from Thm. 3.3 that $\pi$ fulfills $\text{QU}T$. That is, there exists a $k \in \mathbb{N}$ s.t. $\pi(i) \in Q$ for all $i < k$ and $\pi(k) \in T = Q \cap G \cap \text{Cpre}(\bar{Y}^*)$. With this we know that (a) $\pi(k) \in Q$, (b) $\pi(k) \in G$ and (c) $v \in \text{Cpre}(\bar{Y}^*)$.

Now we have two cases: (c.1) If $\pi(k) \in V^1$, then it follows from the definition of Cpre that $E(\pi(k)) \subseteq \bar{Y}^*$. As $\bar{Y}^* = \bar{Z}^*$, we know $\pi(k + 1) \in \bar{Z}^*$. (c.2) If $\pi(k) \in V^0$ we know that $\text{rank}(\pi(k)) = \min_{v \in E(\pi(k))} \text{rank}(v')$. Now recall that $\bar{Z}^* = \bar{Y}^* = Y^* = \bigcup_{i>0} X^i$. Hence, any state with rank $0 < n \leq \infty$ is contained in $\bar{Z}^*$ and hence, we have $\pi(k + 1) \in \bar{Z}^*$. With this, we can successively re-apply Thm. 3.3 to $\pi(k + 1)$. This shows that $\bar{G}$ is visited infinitely often along $\pi$ while $\pi$ always remains within $Q$.

**Completeness** We pick a state $v \in W$ and show that $v \in Z^*$. As $v \in W$ we know that there exists a strategy $\rho_0$ s.t. every play $\pi$ compliant with $\rho$ fulfills $\psi$. That is, we know that $\pi$ always remains in $W$, where $W$ is the winning set for $\psi := \text{Q} \cup (Q \cap G)$. As $\psi$ must therefore also hold at states $\pi(k) \in Q \cap G$, and $\pi(k + 1) \in W$ (as $\pi$ stays always in $Q$ and always eventually re-visits $G$) we can strengthen $\psi$ to $\psi := \text{Q} \cup (Q \cap G \cap \text{Cpre}(W))$. Now it follows from Thm. 3.3 that $W = Z'((Q \cap G \cap \text{Cpre}(W), Q))$. It therefore follows from (44) that $W \subseteq Y^*$. As we have shown that $W \subseteq W$, the claim is proven.

C PROOF OF THM. 3.1

This section contains the proof of Thm. 3.1 which is inspired by the proof of Piterman and Pnueli [22] for “normal” Rabin games. We first give a construction of a ranking induced by the fixpoint algorithm in (7) in Sec. C.1, and use this ranking to define a memoryless Player $0$ strategy. As part of the soundness proof for Thm. 3.1 in Sec. C.2, we then show that this extracted strategy is indeed a winning strategy of Player $0$ in the fair adversarial game over $G'$ w.r.t. $\varphi$. Finally, we show in Sec. C.3 that the fixpoint algorithm in (7) is also complete, that is $W \subseteq Z^*$. Intuitively, completeness shows that if $Z^*$ is empty, there indeed exists no live-sufficient winning strategy (with arbitrary memory) for the given fair adversarial Rabin game. Additional lemmas and proofs can be found in Appendix C.4.

C.1 Strategy Extraction

Our strategy extraction is adapted from the ranking in [22, Sec. 3.1]. Recall, that we consider the set of Rabin pairs $\mathcal{R} = \{(G_1, R_1), \ldots, (G_k, R_k)\}$ with index set $P = \{1, \ldots, k\}$ and the artificial Rabin pair $(G_0, R_0)$ s.t. $G_0 = R_0 = 0$. A permutation of the index set $P$ is an one-to-one and onto function from $P$ to $P$; as usual, we write $p_1 \ldots p_n$ to denote the permutation mapping $i$ to $p_i$, for $i = 1, \ldots, k$. We define $\Pi(P)$ to be the set of all permutations over $P$. The configuration domain of the Rabin condition $\mathcal{R}$ is defined as

\[
D(\mathcal{R}) := \{p_0 \ldots p_k | i_j \in [0; n],
\]

\[
p_0 = 0, p_1 \ldots p_k \in \Pi(P) \cup \{\infty\}
\]

(45)

where $n < \infty$ is a natural number which is larger then the maximal number of iterations needed in any instance of the
fixed point computation in (7) which is known to be finite. If \( R \) is clear from the context, we write \( D \) instead of \( D(R) \).

**Intuition:** We first explain the intuition behind the chosen ranking. For this we consider the definition of ranks for states \( v \in Z^* \) in an iterative fashion. First, consider the last iteration over \( X_{p_0} \) converging to the fixed point \( Z^* = Y_{p_0} = \bigcup_{i_0 > 0} X_{p_0}^{i_0} \) where \( X_{p_0}^0 := \emptyset \). By flattening (7) we see that for all \( i_0 > 0 \) we have

\[
X_{p_0}^{i_0} = \text{Apref}(Y_{p_0}^*, X_{p_0}^{i_0-1}) \cup \mathcal{A}_{p_0,i_0}
\]  

(46a)

where \( \mathcal{A}_{p_0,i_0} \) collects all remaining terms of the fixpoint algorithm in (7) and will be specified later. For now, we want to assign a “minimal rank” to all states added to \( Z^* \) via the first term in (46a). Let us assume that the right “minimal rank” for these states is

\[
d = p_0 i_0 p_1 i_2 p_2 0 \ldots p_k 0 \quad \text{with} \quad p_1 < p_2 < \ldots < p_k \quad \text{and} \quad i_0 > 0.
\]

We assign this rank to \( v \) iff \( v \in \text{Apref}(Y_{p_0}^*, X_{p_0}^{i_0-1}) \setminus X_{p_0}^{i_0-1} \), i.e., if \( v \) is not already added to the fixed point in a previous iteration. The intuition behind this rank choice is that we want to remember that we have added \( v \) to \( Z^* \) in the \( i_0 \)'s computation over \( X_{p_0} \), which sets the counter for \( p_0 \) in \( d \) to \( i_0 \). We keep all other counters at 0 because there is no actual contribution of terms involving variables \( X_{p_i} \) for \( p_i \in P \) for the “adding” of \( v \).

Now recall that

\[
X_{p_0}^{i_0} = \bigcup_{p_1 \in P} Y_{p_1}^* = \bigcup_{p_1 \in P} \bigcup_{i_1 > 0} X_{p_1}^{i_1}.
\]

Further, we know that

\[
\text{Apref}(Y_{p_0}^*, X_{p_0}^{i_0-1}) \subseteq X_{p_1}^{i_1} \quad \text{for all} \quad p_1 \in P \quad \text{and} \quad i_1 > 0.
\]  

(46b)

Hence, any state added to the fixed point via \( X_{p_0}^{i_0} \) (which is not contained in \( X_{p_0}^{i_0-1} \)) is either added via \( \text{Apref}(Y_{p_0}^*, X_{p_0}^{i_0}) \) or via any other remaining term within \( X_{p_1}^{i_1} \) for at least one \( p_1 \) and \( i_1 > 0 \). So let us explore the ranking in the latter case.

For this, let us proceed by going over all \( X_{p_1}^{i_1} \) in increasing order over \( P \), i.e., we start with selecting \( p_1 = 1 \). Further, we remember that we compute the next iteration over \( X_{p_0} \), (i.e., \( X_{p_1}^{i_1} \) given \( X_{p_1}^{i_1-1} \)) as part of computing the set \( X_{p_0}^{i_0} \). I.e., we remember the computation-prefix \( \delta = p_0 i_0 \) in the computation of \( X_{p_1}^{i_1} \). To make \( \delta \) explicit, we denote \( X_{p_1}^{i_1} \) by \( X_{\delta p_1}^{i_1} \). Now, we again consider the last iteration over \( X_{\delta p_1} \) converging to the fixed point \( Y_{\delta p_1}^* \) (for the currently considered computation-prefix \( \delta \)). Then we have

\[
X_{\delta p_1}^{i_1} = \text{Apref}(Y_{\delta p_1}^*, X_{\delta p_1}^{i_1-1})
\]

\[
= S_{\delta} \cup \bigcup_{p_1 \in P} \bigcup \left( \{ G_p \cap \text{Cpre}(Y_{\delta p_1}^*) \} \cup \text{Apref}(Y_{\delta p_1}^*, X_{\delta p_1}^{i_1-1}) \right)
\]

\[
= C_{\delta p_1 i_1} \cup \mathcal{A}_{\delta p_1 i_1}.
\]

We now want to assign the “minimal rank” to all states that are added to the fixed point via \( C_{\delta p_1 i_1} \). The immediate choice of this rank is

\[
d = p_0 i_0 p_1 i_2 p_2 0 \ldots p_k 0 = \delta p_1 i_2 p_0 \ldots p_k 0 \quad \text{(46c)}
\]

with \( p_2 < \ldots < p_k \) and \( i_0, i_1 > 0 \).

(Note that we do not necessarily have \( p_1 < p_2 \)!)  

We only want to assign this rank to states that are already added to the fixed point via \( C_{\delta p_1 i_1} \), i.e., do not already have a rank assigned. First, all states \( v \in S_{\delta} \) already have an assigned rank (as discussed before). Second, for \( i_1 > 1 \) all states in \( C_{\delta p_1 i_1-1} \) have already an assigned rank. But, third, also all states that have been added by considering a different \( X_{p_1} \), with \( p_1 \in P \) being smaller then the currently considered \( p_1 \) also have an already assigned rank.

Now consider the ranking choices suggested in (46b) and (46c). Then we see that all already assigned ranks are smaller (in terms of the lexicographic order over \( D \)) than the one in (46c). To see this, first consider a state \( v \in S_{\delta} \). Either, \( v \in X_{p_0}^{i_0-1} \) in which case its 0’th counter is smaller then \( i_0 \) (i.e., \( i_0 - 1 < i_0 \)) or \( v \) has been added via \( S_{\delta} \), in which case the 0’th counter is equivalent but the first counter is 0 and therefore smaller then \( i_1 \) in (46c) (as, \( i_1 > 0 \)). Now consider a state \( v \in X_{p_1} \) with \( p_1 < p_1 \). In this case we see that 0’th counter is equivalent but the first permutation index is smaller (as \( p_1 < p_1 \)).

We can therefore avoid specifying exactly in which set \( v \) should not be contained to be a newly added state. We can simply collect all possible rank assignments for every state and then, post-process this set to select the smallest rank in this set. Let us now generalize this idea to all possible configuration prefixes.

**Proposition C.1.** Let \( \delta = p_0 i_0 \ldots p_{j-1} i_{j-1} \) be a configuration prefix, \( p_j \in P \setminus \{ p_1, \ldots, p_{j-1} \} \) the next permutation index and \( i_1 > 0 \) a counter for \( p_j \). Then the flattening of (7) for this configuration prefix is given by

\[
X_{\delta p_j}^{i_j} = S_{\delta} \cup C_{\delta p_1 i_1} \cup \mathcal{A}_{\delta p_1 i_1}
\]

(47a)
As this flattening follows directly from the structure of the fixpoint algorithm in (7) and the definition of $C_p$, in (7b), the proof is omitted.

Using the flattening of (7) in (47) we can define a ranking function induced by (7) as follows.

**Definition C.1.** Given the premises of Prop. C.1, we define $\delta(x) := p_{j+1}\delta p_{j+2}0 \ldots p_k0$ with $p_{j+1} < p_{j+2} < \ldots < p_k$ to be the minimal configuration post-fix. Then we define the rank-set

$$ R := \{ v \in V \mid v \subseteq \delta(x) \} $$(49)

for any configuration prefix $\delta$, next permutation index $p_j$ and counter $i_j > 0$. Thereby, we ultimately also prove this claim for $p_j = p_0 = 0$ where $\delta$ is the empty string and $\delta(x) = \bigcup_{i>0} X_{\delta_1}^{i_1}$ coincides with $Z^*$ in (7), which proves the statement.

With this insight the proof of Thm. C.2 as well as the soundness part of Thm. 3.1 reduce to the following proposition.

**Proposition C.2.** For all $j \in [0, k]$, computation-prefixes $\delta = p_{j+1}0p_{j+2}0 \ldots p_k0$ with $p_{j+1} < p_{j+2} < \ldots < p_k$, and $\delta' = p_{j+1}np_{j+2}0 \ldots p_k0$ with $p_k < p_{k-1} < \ldots < p_{j+1}$ be the minimal and maximal post-fix, respectively. Then, for all $v \in X^{\delta}_{\delta'}$, exactly one of the following cases holds:

(a) $v \in S_0$ and rank$(v) \leq \delta_1 Y'$

(b) $v \in Q_{\delta_1} \cap G_{\delta_1} \cap \text{Cpre}(Y^*_\delta)$ and rank$(v) = \delta_1 Y'$

(c) $v \in Q_{\delta_1} \cap \text{Apre}(Y^*_\delta, X^{i_1}_{\delta_1})$ and rank$(v) = \delta_{p_j} i_{j+1} Y'$ s.t. $i_j > 1$, or

(d) $v \in A_{\delta_1}$ and there exists $Y' < Y' \leq \delta_{p_j} i_{j+1} Y'$.

Using Prop. C.3 we prove Prop. C.2 by an induction over $j$. 

**C.2 Soundness**

We now show why the fixpoint algorithm in (7) is sound, i.e., why $Z^* \subseteq W$ in Thm. 3.1 holds. In addition, we also show that Thm. C.2 holds.

We prove soundness by an induction over the nesting of fixed points in (7) from inside to outside. In particular, we iteratively consider instances of the flattening in (47), starting with $j = k$ as the base case, and doing an induction from $j + 1$ to $j$. To this end, we consider a local winning condition which refers to the current configuration-prefix $\delta = p_{j+1}0p_{j+2}0 \ldots p_k0$ in (47), namely

$$ \psi_{\delta_1} := \left( \bigvee_{v \in P_{\delta_1}} \bigoslash \bigoslash \bigotimes G_{\delta_1} \right) $$

(49)

Further, we denote by $W_{\delta_1}$ the set of states for which player 0 wins the fair adversarial game over $G^\delta$ w.r.t. $\psi_{\delta_1}$ in (49).

By recalling that for $p_j = p_0 = 0$ we have $Q_{\delta_1} = V, S_0 = \emptyset$ and $G_{\delta_1} = \emptyset$, we see that for $j = 0$ the condition in (49) simplifies to

$$ \psi_{\delta_0} = \left( \bigoslash \bigoslash \bigotimes G_{\delta_0} \right) $$

This implies that $\psi_{\delta_0}$ is equivalent to $\varphi$ in (6). Given this observation, the proof of soundness in Thm. 3.1 proceeds by inductively showing that

$$ X_\delta^{i_j} \subseteq W_{\delta_1} $$

(50)

for any configuration prefix $\delta$, next permutation index $p_j$ and counter $i_j > 0$. Thereby, we ultimately also prove this claim for $p_j = p_0 = 0$ where $\delta$ is the empty string and $X^*_\delta = \bigcup_{i>0} X^{i_1}_{\delta_1}$ coincides with $Z^*$ in (7), which proves the statement.

With this insight the proof of Thm. C.2 as well as the soundness part of Thm. 3.1 reduce to the following proposition.

**Proposition C.3.** Given the premises of Prop. C.2, let

$$ p_j := p_{j+1}0p_{j+2}0 \ldots p_k0 $$

with $p_{j+1} < p_{j+2} < \ldots < p_k$, and

$$ \bar{p}_j := p_{j+1}np_{j+2}0 \ldots p_k0 $$

with $p_k < p_{k-1} < \ldots < p_{j+1}$ be the minimal and maximal post-fix, respectively. Then, for all $v \in X^{\delta}_{\delta'}$, exactly one of the following holds:

(a) $v \in S_0$ and rank$(v) \leq \delta_1 Y'$

(b) $v \in Q_{\delta_1} \cap G_{\delta_1} \cap \text{Cpre}(Y^*_\delta)$ and rank$(v) = \delta_1 Y'$

(c) $v \in Q_{\delta_1} \cap \text{Apre}(Y^*_\delta, X^{i_1}_{\delta_1})$ and rank$(v) = \delta_{p_j} i_{j+1} Y'$ s.t. $i_j > 1$, or

(d) $v \in A_{\delta_1}$ and there exists $Y' < Y' \leq \delta_{p_j} i_{j+1} Y'$.
Proof of Prop. C.2. **Base case:** First, for \( j = k \) the last line of (49) disappears. Then the proof reduces to Thms. 3.3 and Thm. 3.2 in the following way. First, we fix all fixpoint variables \( Y^*_{p_0 \ldots p_I} \) and \( X^U_{p_0 \ldots p_I} \) for \( I < j \) as well as \( Y^*_p \). With this, we see that \( T := S_\delta \cup (Q_{\delta p_j} \cap G_{\delta p_j} \cap \text{Cpre}(Y^*_p)) \) becomes a fixed set of states and (47a) reduces to

\[
X^U_{\delta p_j} = \bigcup \{ Q_{\delta p_j} \cap \text{Apre}(Y^*_p, X^{U,i}_{\delta p_j}) \}
\]

where we know that \( X^U_{\delta p_j} \subseteq Y^*_p \). Further, it follows form Prop. C.3 that for all \( X^U_{\delta p_j} \), the ranking only differs by the \( i_j \) count. Hence, we can replace \( \rho_0 \) in (48) by the simpler strategy \( \rho_0 \) in (41) that only considers the \( i_j \) count as the rank of states in \( Y^*_p \). With this it follows from Thm. 3.3 that for any fair adversarial play \( \pi \) compliant with \( \rho_0 \) in (48) and starting in \( X^U_{\delta p_j} \) for some \( i_j \geq 0 \) it holds that \( Q_{\delta p_j} \cup T \). This implies that whenever such a play \( \pi \) eventually reaches a state in \( S_\delta \subseteq T \) the first line of (49) holds.

Now assume that \( \pi \) does not reach a state in \( S_\delta \subseteq T \). Then it reaches a state in \( Q_{\delta p_j} \cap \text{G}_{\delta p_j} \cap \text{Cpre}(Y^*_p) \) and therefore has a successor state \( \nu \in Y^*_p \) for some \( i_j \geq 0 \). Hence, \( \nu \in X^U_{\delta p_j} \) for some \( i_j \geq 0 \). By repeatedly applying this argument we see that \( \pi \) either eventually reaches a state in \( S_\delta \subseteq T \) or it remains infinitely in \( C_{\delta p_j} \). In the latter case, it follows from Thm. 3.2 that the second line of (49) holds.

**Induction step:** For the induction step (from \( j+1 \) to \( j \)) we first analyze the assumption. I.e., we know that for the longer computation prefix \( \delta^j = \delta p_{j+1} \) and any next permutation index \( \rho_{j+1} \in \rho_{j+1} \in P \setminus \{ p_1, \ldots, p_j \} \). Now recall that (47e) implies

\[
A_{\delta p_{j+1}} = \bigcup_{\rho_{j+1} \in \rho_{j+1}} Y^*_{\delta p_{j+1}} \setminus S_{\delta p_{j+1}}
\]

and therefore, we know that for all \( \nu \in A_{\delta p_{j+1}} \) there exists a \( p_{j+1} \) s.t. \( \nu \in W_{\delta p_{j+1}} \). That is, any fair adversarial play starting in \( \nu \) that is compliant with \( \rho_0 \) in (48) fulfills (49).

Therefore, whenever a fair adversarial play \( \pi \) starting in \( X^U_{\delta p_j} \) visits a vertex \( \nu \in A_{\delta p_{j+1}} \) (i.e., case (d) holds), we know that \( \pi \) could possibly come back to a state \( \nu \in S_{\delta p_{j+1}} \setminus C_{\delta p_{j+1}} \) (via the first line of (49a)).

In this case, Prop. C.3 ensures that the \( i_j \) count of the rank of states always stays constant while the play stays in \( A_{\delta p_{j+1}} \). Therefore, one can ignore these finite sequences of (d) vertices in \( \pi \) while applying the ranking arguments of Thm. 3.3 and Thm. 3.2. I.e., we can conclude that in this case either the first or the second line of (49) holds for \( \pi \). It remains to show that \( \pi \) fulfills the last line of (49) if \( \pi \) eventually stays within \( A_{\delta p_{j+1}} \) forever. First, observe that this is only possible if \( S_\delta \) is not visited along \( \pi \). Hence, we know that \( Q_{\delta p_j} \) holds along \( \pi \) until \( A_{\delta p_{j+1}} \) is entered and never left. Further, as \( A_{\delta p_{j+1}} \) is assumed to be never left after some time \( k > 0 \), we know that from that time onward there exists no \( p_{j+1} \) s.t. \( S_{\delta p_{j+1}} \) is visited again by \( \pi \). This implies that for all vertices \( \pi(k') \) with \( k' > k \) the last two lines of (49) must be true for at least one \( p_{j+1} \). Hence, \( \pi \) fulfills the property

\[
\Psi_{\delta p_j} := \bigcup_{\rho_{j+1} \in \rho_{j+1}} Y^*_{\delta p_{j+1}} \setminus S_{\delta p_{j+1}}
\]

With this, it remains to show that \( \Psi_{\delta p_j} \) implies that the last line of (49) is true for \( \pi \). In particular, we can show that both statements are equivalent, i.e.,

\[
\Psi_{\delta p_j} = \bigcup_{\rho_{j+1} \in \rho_{j+1}} Y^*_{\delta p_{j+1}} \setminus S_{\delta p_{j+1}}
\]

Equation (51) is proven in Sec. C.4.2. This concludes the proof.

\( \square \)

**C.3 Completeness**

We now show why the fixpoint algorithm in (7) is complete, i.e., \( W \subseteq Z^* \) in Thm. 3.1 holds.

We also prove completeness by an induction over the nesting of fixed points (7) from inside to outside. In particular, we iteratively consider the fixed points \( Y^*_p \) and show that \( Y^*_p \subseteq W_{\delta p_j} \). As \( Y^*_p \), simplifies to \( \phi \) in (6) for \( p_j = p_0 = 0 \), we ultimately show that \( W \subseteq Z^* \) in Thm. 3.1. With this insight the proof of the completeness part of Thm. 3.1 reduces to the following proposition.

**Proposition C.4.** For all \( j \in [0, k] \), computation-prefixes \( \delta = p_0 \ldots p_{j-1} i_{j-1} \), next permutation index \( p_j \in P \setminus \{ p_0, \ldots, p_j \} \) and counter \( m > 0 \) it holds that \( W_{\delta p_j} \subseteq Y^*_p \) with \( Y^*_p := V \).

Proof. The proof proceeds by a nested induction, first over \( j \) starting with \( j = k \) and then over \( m \), starting with \( m = 0 \). It is easy to see that the base case for the induction over \( m \) is obvious as \( Y^*_p = V \), hence \( W_{\delta p_j} \subseteq Y^*_p \). Hence, we only need to show the induction step over \( m \) for every \( j \), i.e. we show that \( W_{\delta p_j} \subseteq Y^*_p \) implies \( W_{\delta p_j} \subseteq Y^*_p \).

**Base case:** Recall that for \( j = k \) the last line of (49) disappears. Hence, for any state \( \nu \in W_{\delta p_j} \), either the first or the second line of (49) holds. Then the proof reduces to Thms. 3.3 and Thm. 3.2 in the following way.

First, we fix all fixpoint variables \( Y^*_p \ldots p_j \) and \( X^U_{p_0 \ldots p_I} \) for \( I < j \) as well as \( Y^*_p \). With this, we see that \( T := S_\delta \cup (Q_{\delta p_j} \cap G_{\delta p_j} \cap \text{Cpre}(Y^*_p)) \) becomes a fixed set of states and (47a) reduces to

\[
X^U_{\delta p_j} = \bigcup \{ Q_{\delta p_j} \cap \text{Apre}(Y^*_p, X^{U,i}_{\delta p_j}) \}
\]
where $X^{m,i}_j$ is the set computed in the $i_j$-th iteration over $X$ during the computation of $Y^m_{\delta_{p_j}} \cup \bigcup_{i > 0} X^{m,i}_j$. Then it follows from Thm. 3.3 that any state $v \in V$ for which there exists a fair adversarial play $\pi$ that is winning for the winning condition $Q_{\delta_{p_j}} UT$ is contained in $Y^m_{\delta_{p_j}}$. If indeed, the first line of (49) holds for $\pi$, the claim is proven.

Now assume that $Q_{\delta_{p_j}} UT$ holds for $\pi$ but $S_\delta$ is never reached. Hence, $Q_{\delta_{p_j}} UT (Q_{\delta_{p_j}} \cap G_{p_j} \cap Cpre(y_{\delta_{p_j}}^{-1}))$ holds for $\pi$ and we know that $\pi$ always stays within $X^{m,i}_j \subseteq X^{m,i}_{\delta_{p_j}}$. With this, it follows form Thm. 3.2 that any state $v \in V$ for which there exists a fair adversarial play $\pi$ for which the second line of (49) holds is contained in $Y^m_{\delta_{p_j}}$.

**Induction Step**: For the induction from "j+1" to "j" we first analyze the assumption. I.e., we know that for the longer computation prefix $\delta' = \delta p_j$ and any next permutation index $p_{j+1}$ we have that $W_{\delta(p_{j+1})} \subseteq Y_{\delta' p_{j+1}}^{s'}$. Further, observe that $\Psi_{\delta_{p_j}}^{s'} \subseteq \bigcup_{p_{j+1} \in P \setminus \{p_0, \ldots, p_j\}} W_{\delta_{p_{j+1}}} \setminus S_{\delta_{p_{j+1}}}$ by construction. We therefore have

$$\Psi_{\delta_{p_j}}^{s'} \subseteq \bigcup_{p_{j+1} \in P \setminus \{p_0, \ldots, p_j\}} Y_{\delta' p_{j+1}}^{s'} \setminus S_{\delta_{p_{j+1}}} = A_{\delta_{p_{j+1}}}.$$ 

With this observation, we see that any fair adversarial play $\pi$ which fulfills the last line of (49) also fulfills the weaker condition $Q_{\delta_{p_{j+1}}} UT \subseteq A_{\delta_{p_{j+1}}}$. Therefore, the claim follows from the same reasoning as in the base case by re-defining $T$ to $T := S_\delta \cup (Q_{\delta_{p_{j+1}}} \cap G_{p_j} \cap Cpre(y_{\delta_{p_j}}^{-1})) \cup A_{\delta_{p_{j+1}}}$.

### C.4 Additional Lemmas and Proofs

In this section we provide additional lemmas and proofs to support the proof of Thm. 3.1 and Thm. C.2.

**C.4.1 Proof of Prop. C.3.**

**Lemma C.3.** Given the premises of Prop. C.3, it holds for all $v \in X^{ij}_{\delta_{p_j}}$ that

- (i) $v \in S_{\delta_{p_j}}$ iff $\text{rank}(v) \leq \delta p_j \gamma$
- (ii) $v \in X^{ij}_{\delta_{p_j}}$ iff $\text{rank}(v) \leq \delta p_j i_{j+1} \gamma$
- (iii) $v \in Y^{ij}_{\delta_{p_j}}$ iff $\text{rank}(v) \leq \delta p_j n_{i_{j+1}}$
- (iv) $v \in A_{\delta_{p_{j+1}}}$ iff there exists $\gamma' \leq \gamma$ s.t. $\text{rank}(v) = \delta p_{j+1} i_{j+1} \gamma'$. 

**Proof of Lem. C.3.** We prove all claims separately.

(i) It immediately follows from Def. C.1 (i) that $\delta p_j \gamma \in R(v)$ iff $v \in S_{\delta_{p_j}}$. If it is the minimal element in $R(v)$ then $\text{rank}(v) = \delta p_j \gamma$, if not, there exists a smaller element in $R(v)$, and then $\text{rank}(v) < \delta p_j \gamma$ from the definition of rank.

(ii) First, observe that for $j = k$ it follows from (47a) that $X^{ik}_{\delta_{p_k}} = S_{p_k i_{k+1}}$ and therefore from (i) that $v \in X^{ik}_{\delta_{p_k}}$ iff $\text{rank}(v) \leq \delta p_{k+1}$. Now we do an induction, assuming that for any $p_{j+1} \in P \setminus \{p_0, \ldots, p_j\}$ and $0 < i_{j+1} \leq n$ it holds that $v \in X^{ij}_{\delta_{p_{j+1}}}$ iff $\text{rank}(v) \leq \delta' p_{j+1} i_{j+1} \gamma'$ (where $\delta'$ goes up to index $j$ and $\gamma'$ starts only at index $j + 2$). Now recall that

$$X^{ij}_{\delta_{p_j}} = \bigcup_{p_{j+1} \in P \setminus \{p_0, \ldots, p_j\}} Y^{ij}_{\delta_{p_{j+1}}} \setminus \bigcup_{i_{j+1} > 0} X^{ij}_{\delta_{p_{j+1}}}.$$

Hence, $v \in X^{ij}_{\delta_{p_j}}$ iff there exists $p_{j+1} \in P \setminus \{p_0, \ldots, p_j\}$ and $0 < i_{j+1} \leq n$ s.t. $v \in X^{ij}_{\delta_{p_{j+1}}}$ and from (ii) therefore $\text{rank}(v) \leq \delta p_{j+1} i_{j+1} \gamma'$. Again, the worst case is $i_{j+1} = n$, giving $\text{rank}(v) \leq \delta p_{j+1} n_{i_{j+1}}$. 

(iv) It follows from (47a) that $v \in A_{\delta_{p_{j+1}}}$ iff $v \in X^{ij}_{\delta_{p_j}} \setminus S_{\delta_{p_{j+1}}}$. 

Hence, it follows from (i) and (ii) that $\text{rank}(v) > \delta p_{j+1} \gamma$ and $\text{rank}(v) \leq \delta p_{j+1} n_{i_{j+1}}$, which is true iff there exists $\gamma' \leq \gamma$ s.t. $\text{rank}(v) = \delta p_{j+1} i_{j+1} \gamma'$, which proves the statement. 

Given these properties of the ranking function, we are ready to prove the suggested case split in Prop. C.3.

**Proof of Prop. C.3.** We call a vertex $v \in V$ that fulfills cases (a) in either Lem. C.3 or Prop. C.3 an (a)-vertex. First, observe that cases (i) and (iv) in Lem. C.3 coincide with cases (a) and (d), respectively, in Prop. C.3. Further, recall that $X^{1}_{\delta_{p_j}} = \emptyset$. Therefore, $X^{1}_{\delta_{p_j}}$ only contains (a)-(b)-(d) -vertices, as $\text{Aprere}(\cdot, \emptyset) = \emptyset$. Now we know from (ii) that for any $v \in X^{1}_{\delta_{p_j}}$ we have $\text{rank}(v) \leq \delta p_j 1_{i_{j+1}}$. Now excluding the rankings for (a)- and (d)-vertices we obtain that (b)-vertices must have rank $\text{rank}(v) \leq \delta p_j 1_{i_{j+1}}$. Similarly, for every $i_j > 1$ we know that $X^{ij}_{\delta_{p_j}}$ contains (a)-(b)-(c)- and (d-)vertices. Now excluding (a)-(b) - and (d)-vertices yields $\text{rank}(v) \leq \delta p_{j+1} 1_{i_{j+1}}$ for all (c)-vertices.

**C.4.2 Proof of (51).** Given the notation in Sec. C.2 we prove that the equality in (51) holds.

First recall that

$$\Psi^{s'}_{\delta_{p_{j+1}}} := \left( \square Q_{\delta_{p_{j+1}}} \land \square \Diamond G_{\delta_{p_{j+1}}} \land \bigvee_{i \in \tilde{p}_{j+1}} \left( \square \Diamond R_i \land \square \Diamond G_i \right) \right).$$

where $\tilde{p}_{j+1} := P \setminus \{p_1, \ldots, p_{j+1}\}$.

For the insertion of (52) into (51a) we have the following observations. First, observe that $\Diamond (B \lor C) = \Diamond B \lor \Diamond C$, i.e., we can distribute the eventuality operator preceding $\Psi^{s'}_{\delta_{p_{j+1}}}$ over both lines. Second, we can re-order the preceeding disjunction over $p_{j+1}$ in (51a) and the disjunction between the
two lines of (52). This yields to the following condition

$$
\Psi_{\delta_{p_{j+1}}} = \Box Q_{\delta p_{j+1}} \land \\
\left( \bigvee_{p_{j+1} \in \tilde{P}_{j}} \left[ \Diamond \left( Q_{\delta p_{j+1}} \land \Box \Box G_{p_{j+1}} \right) \right] \right)
$$

Now let us investigate both lines of the right side of (53) separately. For the first line, observe that $\Diamond \Box A = \Box \Diamond A$ and $\Diamond (A \land B) = \Diamond A \land \Diamond B$. Further we have $Q_{\delta p_{j+1}} = \Box Q_{\delta p_{j+1}} \land \Box \Box G_{p_{j+1}}$. Therefore, we can rewrite the first line in (53) into

$$
\Psi_{1} = \Box Q_{\delta p_{j+1}} \land \bigvee_{p_{j+1} \in \tilde{P}_{j}} \left( \Diamond \left( Q_{\delta p_{j+1}} \land \Box \Box G_{p_{j+1}} \right) \right)
$$

By using the equality $\Diamond \Box (A \land B) = \Diamond \Box A \land \Diamond B$ and the fact that $Q_{\delta p_{j+1}}$ is independent of the choice of $p_{j+1}$ we get

$$
\Psi_{1} = \Box Q_{\delta p_{j+1}} \land \bigvee_{p_{j+1} \in \tilde{P}_{j}} \left( \Diamond \left( Q_{\delta p_{j+1}} \land \Box \Box G_{p_{j+1}} \right) \right)
$$

To analyze the second line of (53), recall that the eventuality operator $\Diamond$ distributes over disjunctions. We can therefore move the inner disjunction over $i$ outside and get

$$
\Psi_{2} = \Box Q_{\delta p_{j+1}} \land \bigvee_{p_{j+1} \in \tilde{P}_{j}} \left( \left( \bigvee_{i \in \tilde{P}_{j+1}} \Diamond \left( Q_{\delta p_{j+1}} \land \Box \Box G_{i} \right) \right) \right)
$$

Now observe that $\left( \Diamond \Box \Box G_{i} \right) = \Diamond \left( \Box \Box G_{i} \right)$ and $\Diamond (A \land B) = \Diamond A \land \Diamond B$. Additionally using $Q_{\delta p_{j+1}} = \Box Q_{\delta p_{j+1}} \land \Box \Box G_{p_{j+1}} \subseteq Q_{\delta p_{j+1}}$ we get

$$
\Psi_{2} = \Box Q_{\delta p_{j+1}} \land \bigvee_{p_{j+1} \in \tilde{P}_{j}} \left( \left( \bigvee_{i \in \tilde{P}_{j+1}} \Diamond \left( Q_{\delta p_{j+1}} \land \Box \Box G_{i} \right) \right) \right)
$$

Now let us investigate both lines of the right side of (53) separately. For the first line, observe that $\Diamond \Box A = \Box \Diamond A$ and $\Diamond (A \land B) = \Diamond A \land \Diamond B$. Further we have $Q_{\delta p_{j+1}} = \Box Q_{\delta p_{j+1}} \land \Box \Box G_{p_{j+1}}$. Therefore, we can rewrite the first line in (53) into

$$
\Psi_{1} = \Box Q_{\delta p_{j+1}} \land \bigvee_{p_{j+1} \in \tilde{P}_{j}} \left( \Diamond \left( Q_{\delta p_{j+1}} \land \Box \Box G_{p_{j+1}} \right) \right)
$$

By using the equality $\Diamond \Box (A \land B) = \Diamond \Box A \land \Diamond B$ and the fact that $Q_{\delta p_{j+1}}$ is independent of the choice of $p_{j+1}$ we get

$$
\Psi_{1} = \Box Q_{\delta p_{j+1}} \land \bigvee_{p_{j+1} \in \tilde{P}_{j}} \left( \Diamond \left( Q_{\delta p_{j+1}} \land \Box \Box G_{p_{j+1}} \right) \right)
$$

To analyze the second line of (53), recall that the eventuality operator $\Diamond$ distributes over disjunctions. We can therefore move the inner disjunction over $i$ outside and get

$$
\Psi_{2} = \Box Q_{\delta p_{j+1}} \land \bigvee_{p_{j+1} \in \tilde{P}_{j}} \left( \left( \bigvee_{i \in \tilde{P}_{j+1}} \Diamond \left( Q_{\delta p_{j+1}} \land \Box \Box G_{i} \right) \right) \right)
$$

Now observe that $\left( \Diamond \Box \Box G_{i} \right) = \Diamond \left( \Box \Box G_{i} \right)$ and $\Diamond (A \land B) = \Diamond A \land \Diamond B$. Additionally using $Q_{\delta p_{j+1}} = \Box Q_{\delta p_{j+1}} \land \Box \Box G_{p_{j+1}} \subseteq Q_{\delta p_{j+1}}$ we get
line (Eq. (56)). As both lines are connected by a disjunction in (53), we can ignore the second line in (53) and obtain

$$\Psi_{\delta p_j} = \Psi_1 = Q_{\delta p_j} \land \bigvee_{p_{j+1} \in \delta P_j} \left( \delta \bigcirc R_{p_{j+1}} \land \bigcirc G_{p_{j+1}} \right). \quad (57)$$

This concludes the proof of (51) as (57) coincides with (51b).

D  ADDITIONAL PROOFS FOR SEC. 3.4

D.1 Fair Adversarial Rabin Chain Games

In this section we prove Thm. 3.4. That is, we prove that for Rabin Chain conditions, the fixpoint computing $Z^*$ in (7) simplifies to the one in (16). This is formalized in the next proposition.

**Proposition D.1.** Given the premisses of Thm. 3.4 let $Z^*$ be the fixed point computed by (7) and $\tilde{Z}^*$ the fixed point computed by (16). Then $Z^* = \tilde{Z}^*$.

If Prop. D.1 holds, we immediately see that Thm. 3.4 directly follows from Thm. 3.1. It therefore remains to prove Prop. D.1.

Similar to the soundness and completeness proof for Thm. 3.1 we prove Prop. D.1 by an induction over the nesting of fixpoints in (7) form inside to outside. Here, however, we do not need to explicitly refer to counters $i_j$ as in Prop. 3.4. Hence, we can look at permutation prefixes instead of configuration prefixes. We have the following proposition.

**Proposition D.2.** Let $P$ be the index set of the Rabin chain condition $\mathcal{R}$ in Thm. 3.4. Further, for any $j \in [0; k]$ let $\delta := p_0 p_1 \ldots p_{j-1}$ be a permutation prefix, $\mathcal{P}_{\delta} := P \setminus \{p_0, \ldots, p_{j-1}\}$ the reduced index set and $q_0 := p_j \in \mathcal{P}_{\delta}$ the current permutation index. Further, define\(^5\)

$$Z^*_{\delta p_j} := v Y_{q_j} \cdot \mu X_{q_j} \cdot \bigcup_{q_1 \in \mathcal{P}_{\delta}} v Y_{q_1} \cdot \mu X_{q_1} \cdot \ldots \bigcup_{q_n \in \mathcal{P}_{\delta}} v Y_{q_n} \cdot \mu X_{q_n}.$$  

where $n := k - j,$

$$C_{\delta q_j} := Q_{\delta} \cap \bigcup_{l=0}^{\ell} R_{q_l} \cap \left( (G_{q_l} \cap \text{Cpre}(Y_{q_l})) \cup (\text{Apre}(Y_{q_l}, X_{q_l})) \right),$$

$$Q_{\delta} := \cap_{i=0}^{j-1} R_{p_i}, \quad \text{and} \quad S_{\delta} := \bigcup_{x=0}^{n} C_{\delta q_l} \cup \left( \bigcup_{l=0}^{\ell} R_{p_l} \right).$$

Then it holds that

$$Z^*_{\delta p_j} = v Y_{r_1} \cdot \mu X_{r_1} \cdot v Y_{r_2} \cdot \mu X_{r_2} \cdot \ldots v Y_{r_n} \cdot \mu X_{r_n} \cdot S_{\delta} \cup \bigcup_{x=0}^{n} \tilde{C}_{\delta q_l}$$

where

$$C_{\delta r_l} := Q_{\delta p_j} \cap \mathcal{R}_{r_l} \cap \left( (G_{r_l} \cap \text{Cpre}(Y_{r_l})) \cup (\text{Apre}(Y_{r_l}, X_{r_l})) \right)$$

with $r_1 \in \mathcal{P}_{\delta} p_j$ for all $i \in [1; n]$ s.t. $r_1 > r_2 > \ldots > r_n$ and $r_0 = q_0 = p_j$.

It should be noted that Prop. D.2 needs to hold for any choice of $j$ and $\delta$. Further, we have slightly abused notation by not specifying the values of the fixpoint parameters used within $S_{\delta}$. This is, however, not relevant for the proof of Prop. D.2 and we should interpret $S_{\delta}$ as a term computed by an arbitrary choice of the involved fixpoint parameters.

Now, it should be obvious that for the choice $j = 0$ we get $\delta = \varepsilon$ and $S_{\delta} = 0$. Further, we see that in this case, we have $\mathcal{P}_{\delta p_0} = P$ which implies that $Z^*_{\delta p_0}$ in (58) coincides with $Z^*$ in (7). Further, as $\mathcal{P}_{\delta p_0} = P$ we must have $r_1 = k, r_2 = k - 1, \ldots, r_l = 1$ and $r_0 = p_0 = 0$ to fulfill the requirements on $r$. Further $Q_{\delta p_0} = \emptyset = \mathcal{R}$. Therefore $Z^*_{\delta p_0}$ in (59) coincides with $Z^*$ in (16) in this case. Hence, proving Prop. D.2 for any $j$ (including $j = 0$), immediately proves Prop. D.1.

In the remainder of this section we prove Prop. D.2 by an induction over $j$, starting with $j = k$ as the base case. Now observe that for $j = k$ we have $\mathcal{P}_{\delta p_j} = \emptyset$ and hence both (58) and (59) reduce to a two-nested fixed point over the variables $Y_{q_j}, X_{q_j}$ and $Y_{r_1}, X_{r_1}$, respectively, where $r_0 = q_0 = p_k$ by definition. Further, we see that $C_{\delta q_k} = \tilde{C}_{\delta r_k}$ by definition, which immediately proves the claim of Prop. D.2 for the base case.

In the remainder of this section we prove the induction step from "$j" to "$j - 1" in a series of definitions and lemmas.

**Definition D.1.** Let $\tilde{P} \subseteq \mathbb{N}$ be a set of $n$ indices and $\beta = q_1 \ldots q_n$ with $q_i \in \tilde{P}$ and $q_i \neq q_j$ for all $j \neq i$ a full permutation sequence of the elements from $\tilde{P}$. For $1 \leq j \leq l \leq n$ we call $\beta_{p_{i+l}} = q_j q_{j+1} \ldots a$ maximal decreasing sub-sequence of $\beta$ if (i) $q_j < q_{j+1} < \ldots < q_i$, (ii) $q_{j-1} > q_j$ or $j = 1$, and (iii) $q_i > q_{i+1}$ or $l = n$.

We see that, by definition, the first maximally decreasing sub-sequences of a permutation sequence $\beta$ starts with $q_1$. Intuitively, decreasing sub-sequences allow to immediately utilize the properties in (15) to simplify the fixpoint expression.

**Lemma D.2.** Let $\delta, \tilde{P}_{\delta}$ and $q_0 = p_j$ as in Prop. D.2, $\beta = q_1 \ldots q_n$ a full permutation sequence of $\tilde{P}_{\delta p_j}$ and $\beta_{p_j} = q_j q_{j+1} \ldots q_j$
a maximal decreasing sub-sequence of \( \beta \). Then
\[
\nu Y_{q_j}, \mu X_{q_j}, \ldots, \nu Y_{q_l}, \mu X_{q_l}, \bigcup_{i=j}^{l} C_{\delta q_i} = vY_{q_j}, \mu X_{q_j} \cup C_{\delta q_j} \tag{60}
\]
\[\text{Proof.} \text{ Let } \alpha := q_{0} \ldots q_{j-1} \text{ and observe that}
\]
\[
C_{\delta q_j} = Q_{\delta \alpha} \cap \left( \overline{R_j} \cap G_j \cap \text{Cpre}(Y_{q_j}) \cup \{\overline{R_j} \cap \text{Apre}(Y_{q_j}, X_{q_j})\} \right)
\]
\[C_{\delta q_{j+1}} = Q_{\delta \alpha} \cap \left( \overline{R_j} \cap \overline{R_{j+1}} \cap G_{q_{j+1}} \cap \text{Cpre}(Y_{q_j}) \cup \{\overline{R_j} \cap \text{Apre}(Y_{q_j}, X_{q_j})\} \right)
\]
where the simplification of \( C_{\delta q_{j+1}} \) follows from \( \overline{R_j} \subseteq R_{j+1} \) (see (15)). So \( C_{\delta q_j} \) and \( C_{\delta q_{j+1}} \) really only differ by the \( G_j \) (resp. \( G_{q_{j+1}} \)) term in the first term of the disjunct. As \( G_j \supseteq G_{q_{j+1}} \) (see (15)) and all terms in the first part of the disjunct are intersected, we see that \( C_{\delta q_j} \supseteq C_{\delta q_{j+1}} \). With this it follows from case (iii) in Lem. A.4 that
\[
\nu Y_{q_j}, \mu X_{q_j}, \nu Y_{q_{j+1}}, \mu X_{q_{j+1}}, \bigcup_{j=0}^{n} C_{\delta q_i} = vY_{q_j}, \mu X_{q_j} \cup C_{\delta q_j}.
\]
Applying this argument to all \( i \in [j, l] \) proves the claim. \( \square \)

**Definition D.3.** We say that a permutation sequence \( \beta \) has **chain index** \( m \) if it contains \( m \) maximal decreasing sub-sequences. For \( \beta = q_1 \ldots q_n \) with chain index \( m \) we define its reduction \( \beta_1 \) as \( \beta_1 := r_1 \ldots r_m \) s.t. \( r_m = q_l \) if \( \beta_j \) is the \( m \)th maximally decreasing sub-sequence of \( \beta \).

**Lemma D.4.** Let \( \delta, \overline{P_{\delta}} \) and \( q_{0} = p_{1} \) as in Prop. D.2, \( \beta = q_{1} \ldots q_{n} \) a full permutation sequence of \( \overline{P_{\delta p_{j}}} \) with chain index \( m \) and \( \beta_{1} := r_{1} \ldots r_{m} \). Then
\[
\nu Y_{q_{j}}, \mu X_{q_{j}}, \nu Y_{q_{j+1}}, \mu X_{q_{j+1}}, \ldots, \nu Y_{q_{n}}, \mu X_{q_{n}}, \bigcup_{j=0}^{n} C_{\delta q_{i}} = vY_{r_{0}}, \mu X_{r_{0}}, \nu Y_{r_{1}}, \mu X_{r_{1}}, \ldots, \nu Y_{r_{m}}, \mu X_{r_{m}}, \bigcup_{l=0}^{m} C_{\delta r_{l}} \tag{61}
\]
where \( q_{0} = r_{0} = p_{j} \).

**Proof.** First, observe that by construction we always have \( r_{1} = q_{1} \). Hence, \( Q_{\delta \alpha} \) in the proof of Lem. D.2 reduces to \( Q_{\delta q_{1}} \) in this case. Further, consider \( r_{2} = q_{1} \) and observe that in this case \( Q_{\delta \alpha} = Q_{\delta} \cap \bigcap_{i=0}^{1} \overline{R_{i}} = Q_{\delta p_{j}} \cap \bigcap_{i=0}^{1} \overline{R_{i}}, \) as \( q_{1} \ldots q_{j-1} \) is a maximal decreasing sub-sequence by construction. Iteratively re-applying this argument along with Lem. D.2 for every \( l \in [1, m] \) therefore proves the claim. \( \square \)

Note that the only full permutation sequence of \( \overline{P_{\delta p_{j}}} \) with chain index \( n \) is the one where \( q_{1} > q_{2} > \ldots > q_{n} \), giving \( \beta_{1} = \beta_{2} = \beta \). Hence, the sequence \( r_{1} \ldots r_{n} \) used in (59) is actually the maximal permutation sequence of \( \overline{P_{\delta p_{j}}} \). We see that all other full permutation sequences \( \gamma \) of \( \overline{P_{\delta p_{j}}} \) have chain index \( m \) s.t. \( 1 \leq m < n \). As the \( \overline{C} \) terms in (16b) do not depend on the history of permutation sequences from \( \overline{P_{\delta p_{j}}} \), we see that any term constructed for a non-maximal permutation sequence is contained in the term constructed for the maximal permutation sequence. This is formalized in the next lemma.

**Lemma D.5.** Let \( \delta, \overline{P_{\delta}} \) and \( q_{0} = p_{1} \) as in Prop. D.2, \( \beta = q_{1} \ldots q_{n} \) a full permutation sequence of \( \overline{P_{\delta p_{j}}} \) and \( \beta_{1} := r_{1} \ldots r_{m} \) its maximal reduced permutation sequence. Then
\[
\nu Y_{q_{j}}, \mu X_{q_{j}}, \nu Y_{q_{j+1}}, \mu X_{q_{j+1}}, \ldots, \nu Y_{q_{n}}, \mu X_{q_{n}}, \bigcup_{j=0}^{n} C_{\delta q_{i}} = vY_{r_{0}}, \mu X_{r_{0}}, \nu Y_{r_{1}}, \mu X_{r_{1}}, \ldots, \nu Y_{r_{m}}, \mu X_{r_{m}}, \bigcup_{l=0}^{m} \overline{C}_{\delta q_{i}} \tag{62}
\]

**Proof.** It follows from the definition of \( \beta_{1} \) and repeatedly applying Lem. D.4 that
\[
\nu Y_{q_{j}}, \mu X_{q_{j}}, \nu Y_{q_{j+1}}, \mu X_{q_{j+1}}, \ldots, \nu Y_{q_{n}}, \mu X_{q_{n}}, \bigcup_{j=0}^{n} C_{\delta q_{i}} = vY_{r_{0}}, \mu X_{r_{0}}, \nu Y_{r_{1}}, \mu X_{r_{1}}, \ldots, \nu Y_{r_{m}}, \mu X_{r_{m}}, \bigcup_{l=0}^{m} C_{\delta r_{l}} \tag{62}
\]
Now we have by definition that \( r_{0} = q_{0} \) and \( r_{1} = q_{1} \) and therefore \( C_{\delta r_{0}} = C_{\delta q_{0}} \) and \( C_{\delta r_{1}} = C_{\delta q_{1}} \) by definition. Recall that \( r_{1} > r_{2} \), hence \( \overline{R_{r_{1}}} \cap \overline{R_{r_{2}}} = \overline{R_{r_{2}}} \). Iteratively applying this argument gives \( C_{\delta r_{1}} = C_{\delta r_{1}} \) for all \( l \in [1, n] \), what proves the claim. \( \square \)

Now observe that we can re-apply Lem. D.4 to \( \beta_{1} \) and reduce it even more. That means, \( \beta_{1} \) could now again have maximal decreasing sub-sequences and we therefore can reduce it to \( (\beta_{i})_{1} \). This might again be reducible and so forth. We therefore define the **maximal reduced permutation sequence** \( \overline{\beta_{1}} = ((\beta_{i})_{1}) \) s.t. \( r_{1} \ldots r_{m} \) is its maximal reduced permutation sequence. Then
\[
\nu Y_{q_{j}}, \mu X_{q_{j}}, \nu Y_{q_{j+1}}, \mu X_{q_{j+1}}, \ldots, \nu Y_{q_{n}}, \mu X_{q_{n}}, \bigcup_{j=0}^{n} C_{\delta q_{i}} \tag{62}
\]

**Lemma D.6.** Let \( \delta, \overline{P_{\delta}} \) and \( q_{0} = p_{1} \) as in Prop. D.2 and let \( \beta = r_{1} \ldots r_{m} \) be the maximal permutation sequence of \( \overline{P_{\delta p_{j}}} \), that its \( \beta = \beta_{1} \). Further, let \( \gamma \neq \beta \) be a full permutation sequence
of $\tilde{P}_{\delta \beta}$ s.t. $y_0 = s_1 \ldots s_m$ with $m < n$. Then

$$vY_1, \mu X_1, \ldots vY_n, \mu X_n \subseteq \bigcup_{i=1}^{n} \tilde{C}_{\delta \beta i}$$

(63)

$$\subseteq vY_1, \mu X_1, \ldots vY_m, \mu X_m \subseteq \bigcup_{i=1}^{m} \tilde{C}_{\delta \beta j}$$

(64)

Proof. As $\beta$ is a full permutation sequence of $\tilde{P}_{\delta \beta}$ we know that for any $i \in [1; m]$ there exists one $j \in [1; n]$ s.t. $s_i = r_j$. Further, as $C$ does not depend on the history of the permutation sequence $\beta$ and $\gamma$ we see that $\tilde{C}_{\delta \beta j} = \tilde{C}_{\delta \gamma j}$ in this case. As $m < n$ we see that the first line of (64) contains the fixpoint variables and $\tilde{C}$ terms of the second line of (64). We can therefore apply Lem. A.4 (i) and (ii) which immediately proves the claim.

Using this result, we are finally ready to prove the induction step of Prop. D.2.

Proof of Prop. D.2. Recall that Prop. D.2 trivially holds for $j = k$ which constitutes the base case of an induction over $j$. Now let us prove the induction step. Hence, let us assume that Prop. D.2 holds for $j$. Now consider $j - 1^n$, i.e., consider the permutation prefix $\delta' = p_0 \ldots p_{j-2}$ and pick any $p_{j-1} \in P_{\delta'}$. By the induction hypothesis, we know that Prop. D.2 holds for $\delta = p_0 \ldots p_{j-1}$ and any choice of $p_j \in P_{\delta}$. That is, $Z_{\delta p_j}^* \subseteq \tilde{C}_{\delta \beta j}$. With this, the fixpoint algorithm in (58) for $\delta'$ and $p_{j-1}$ simplifies to

$$Z_{\delta' p_{j-1}}^* = Z_{\delta j}^* = vY_{p_{j-1}}, \mu X_{p_{j-1}}, \bigcup_{p_j \in P_{\delta}} Z_{\delta p_j}^*$$

(65)

Proof. Recall the fixpoint algorithm for Rabin chain games in (16), i.e.,

$$Z^* := vY_0, \mu X_0, vY_1, \mu X_1, vY_2, \ldots vY_k, \mu X_k, \bigcup_{\gamma \neq 0} C_{\gamma}$$

First, observe that $R_0 = C_0 = \emptyset$ have been artificially introduced, and result in $\tilde{C}_0 = \text{Apre}(Y_0, X_0)$. Further, as we have assumed that $C$ is such that $\bigcup_{i \in [1, 2k]} C_i = V$. We can equivalently write

$$\tilde{C}_0 = \bigcup_{i=1}^{2k} C_j \cup \text{Apre}(Y_0, X_0)$$

$$\tilde{C}_0 = (C_1 \cap \text{Apre}(Y_0, X_0)) \cup (C_2 \cap \text{Apre}(Y_0, X_0)) \cup \ldots \cup (C_{2k} \cap \text{Apre}(Y_0, X_0))$$

For $j > 0$, by using (66) we observe that the definition of $\tilde{C}_j$ in (16b) can be written as

$$\tilde{C}_j = (C_{2j} \cap \text{Cpre}(Y_j)) \cup \left( \bigcup_{i=1}^{2j} C_i \cap \text{Apre}(Y_j, X_j) \right)$$

$$\cup (C_j \cap \text{Apre}(Y_j, X_j))$$

$$\cup \ldots \cup (C_{2j} \cap \text{Apre}(Y_j, X_j))$$

which has $k$ pairs.

Translating the Rabin Chain condition induced by $C$ in (65) into a Rabin condition as in Thm. 3.1 we get the tuple $\mathcal{R} = \{ (G_1, R_1), \ldots, (G_k, R_k) \}$ s.t.

$$R_i = F_{2i} = \bigcup_{j=2i+1}^{2k} C_j$$

(66a)

$$\tilde{R}_i = \bigcup_{j=2}^{j} C_j$$

(66b)

$$G_i = F_{2i+1} = \bigcup_{j=2i+1}^{2k} C_j$$

(66c)

$$\tilde{R}_i \cap G_i = C_{2i}$$

(66d)

Using these properties, the fixpoint algorithm in (16) simplifies further to the fixpoint algorithm for Parity winning conditions in (19). This is formalized in the following proposition.

Proposition D.3. Let $\mathcal{R} = \{ (G_1, R_1), \ldots, (G_k, R_k) \}$ be a Rabin chain condition s.t. (66) holds. Further let $Z^*$ be the fixed point computed by (7) and $\tilde{Z}^*$ the the fixed point computed by (19). Then $Z^* = \tilde{Z}^*$.

Proof. Recall the fixpoint algorithm for Rabin chain games in (16), i.e.,

$$Z^* := vY_0, \mu X_0, vY_1, \mu X_1, vY_2, \ldots vY_k, \mu X_k, \bigcup_{\gamma \neq 0} C_{\gamma}$$

D.2 Fair Adversarial Parity Games

We now consider a Parity winning condition $C = \{ C_1, C_2, \ldots, C_{2k} \}$ of colors, where each $C_i \subseteq V$ is the set of states of $\mathcal{G}$ with color $i$. Further, $C$ partition’s the state space, i.e.,
With this, we obtain the following fixed-point equation

\[ Z^* := vY_0, \mu X_0, vY_1, \mu X_1, \ldots, vY_k, \mu X_k. \]  \tag{67}

\[ \text{Apre}(Y_0, X_0) \]
\[ \cup \left( C_2 \cap \text{Cpre}(Y_1) \right) \cup \left( (C_1 \cup C_2) \cap \text{Apre}(X_1, Y_1) \right) \]
\[ \cup \ldots \]
\[ \cup \left( C_{2k} \cap \text{Cpre}(Y_k) \right) \cup \left( (C_1 \cup \ldots \cup C_{2k}) \cap \text{Apre}(X_k, Y_k) \right) \]

Now consider Lem. A.3 and let us define

\[ f(X_0, Y_0) := (C_1 \cap \text{Apre}(Y_0, X_0)) \]
\[ h_0(X_0, Y_0) := (C_3 \cap \text{Apre}(Y_0, X_0)) \]
\[ g(X_1, Y_1) := (C_2 \cap \text{Cpre}(Y_1)) \cup (C_1 \cap \text{Apre}(Y_1, X_1)) \]
\[ h_1(X_1, Y_1) := (C_3 \cap \text{Apre}(Y_1, X_1)) \]
\[ \ldots \cup (C_{2k} \cap \text{Apre}(Y_k, X_k)) \]

Then the result of the first part of the FP in (67) over \( Y_0, X_0, Y_1, X_1 \) corresponds to the fixed-point defining \( \tilde{Z} \) in Lem. A.3. It therefore follows from Lem. A.3 (ii) that this computation remains unchanged if we change the term \( C_1 \) to

\[ \tilde{C}_1 = (C_2 \cap \text{Cpre}(Y_1)) \cup \left( \bigcup_{j=1}^{k} C_{2j} \cap \text{Apre}(Y_j, X_j) \right). \]

After this, we can iteratively proceed to the fixpoint over \( Y_1, X_1, Y_2, X_2 \) and so forth. Generalizing this argument, we see that for any \( j \geq 0 \) we can define

\[ f_j(X_j, Y_j) := (C_{2j} \cap \text{Cpre}(Y_j)) \cup (C_1 \cap \text{Apre}(Y_j, X_j)) \]
\[ h_j(X_j, Y_j) := (C_{2(j+1)} \cap \text{Apre}(Y_j, X_j)) \]
\[ g_{j+1}(X_{j+1}, Y_{j+1}) := (C_{2(j+1)} \cap \text{Cpre}(Y_{j+1})) \]
\[ \cup (C_1 \cap \text{Apre}(Y_{j+1}, X_{j+1})) \]
\[ \ldots \cup (C_{2k} \cap \text{Apre}(Y_k, X_k)) \]

and therefore applying Lem. A.3 (ii) iteratively for every \( j \geq 0 \) yields the fixpoint equation

\[ Z^* := vY_0, \mu X_0, vY_1, \mu X_1, \ldots, vY_k, \mu X_k. \]  \tag{68}

We can now invoke Lem. A.3 again in a back-wards manner, to first make sure that every \( \tilde{C} \) term for \( j < k \) ends up being

\[ \tilde{C}_j = (C_{2j} \cap \text{Cpre}(Y_j)) \cup (C_{2(j+1)} \cap \text{Apre}(X_j, Y_j)) \]

and all other terms for \( j < k \) correspond to

\[ \tilde{C}_j = (C_{2j} \cap \text{Cpre}(Y_j)) \cup ((C_{2j} \cap C_{2(j+1)}) \cap \text{Apre}(X_j, Y_j)). \]

With this, it is now obvious that the fixpoint algorithm in (68) can be equivalently written as

\[ Z^* := vY_0, \mu X_0, vY_1, \mu X_1, \ldots, vY_k, \mu X_k. \]

\[ (C_1 \cap \text{Apre}(X_0, Y_0)) \cup (C_1 \cap \text{Cpre}(Y_1)) \cup (C_2 \cap \text{Apre}(X_1, Y_1)) \]
\[ \cup \ldots \cup (C_{2k} \cap \text{Cpre}(Y_k)). \]

With this the claim follows by renaming the fixpoint variables accordingly. \( \Box \)

### D.3 Fair Adversarial Generalized Co-Büchi Games

In this section we prove Thm. 3.6. That is, we prove that for generalized Co-Büchi conditions, the fixpoint computing \( Z^\ast \) in (7) simplifies to the one in (23). This is formalized in the next proposition.

**PROPOSITION D.4.** Let \( \mathcal{R} = \{(G_1, R_1), \ldots, (G_k, R_k)\} \) be a Rabin condition s.t. (22) holds. Further let \( Z^\ast \) be the fixed point computed by (7) and \( Z^\ast \) the fixed point computed by (23). Then \( Z^\ast = Z^\ast. \)

**PROOF.** Now consider the flattening of (7) in (47) for \( \tilde{\mathcal{R}} \). Then we see that for all \( j > 0 \) we have

\[ C_{\delta p_{ij}} := \left(Q_{\delta p_{i}} \cap \text{Cpre}(Y_{\delta p_{j}}^*) \right) \cup \left(Q_{\delta p_{i}} \cap \text{Apre}(Y_{\delta p_{j}}^*, X_{\delta p_{j}}^{i-1}) \right) \]
\[ = Q_{\delta p_{i}} \cap \left(\text{Cpre}(Y_{\delta p_{j}}^*) \cup \text{Apre}(Y_{\delta p_{j}}^*, X_{\delta p_{j}}^{i-1}) \right) \]

With this the claim follows by renaming the fixpoint variables accordingly. \( \Box \)
and we always have $X^i_{dp_j} \subseteq Y^*_{dp_j}$. With this, it follows from Lem. A.1 that

$$C_{dp_{ij}} = Q_{dp_j} \cap Cpre(Y^*_{dp_j}) \quad (70)$$

for all $\delta, p_j$ and $i_j$ with $j > 0$.

Now observe that for $\delta' = \delta p_{ij}$ and all $p_{j+1} \in P \setminus \{p_0, \ldots, p_j\}$ we have

$$Q_{\delta p_j} = Q_{\delta p_{j+1}} \cap \bar{F}_{p_{j+1}} \subseteq Q_{\delta p_j}.$$

It further follows from the structure of the fixed point in (7) that

$$Y^*_{dp_j} = \bigcup_{i_j > 0} X^i_{dp_j} = \bigcup_{i_j > 0} \bigcup_{p_{j+1} \in P} Y^*_{dp_{j+1}}$$

and therefore

$$Y^*_{dp_j} \subseteq Y^*_{dp_{j+1}}.$$

With this we get

$$C_{dp_{j+1}} \subseteq C_{dp_{ij}}$$

for all $\delta, p_j$ and $i_j$ with $j > 0$. Then it follows from Lem. A.4 (iii) that for every permutation sequence $\delta = p_0 p_1 \ldots p_k$ the union over all $C'$ terms simplifies to two terms, one for $j = 0$ and one for $j = 1$. Using this insight, we see that for the particular Rabin condition $\bar{F}$ the fixpoint algorithm in (7) simplifies to

$$v Y_0, \mu X_0, \bigcup_{p \in P} v Y_{p_1}, \mu X_{p_1}, C_{p_1} \cup C_{p_1} \quad (71)$$

Now recalling that $C_{p_1}$ simplifies to $\bar{A}_a \cap Cpre(Y_a)$ for $a = p_1$ (see (70)) if (22) holds, and that $C_{p_0} = Apre(Y_0, X_0)$ as $R_0 = Q_0 = \emptyset$, we see that (71) coincides with (23). \hfill \Box

E ADDITIONAL PROOFS FOR SEC. 4

E.1 Proof of Thm. 4.1

Our goal is to prove Thm. 4.1 by a reduction to Thm. 3.2 and Thm. 3.3. We therefore first show that a similar construction of an extended fixed point $\bar{Z}$ as in (43) within the proof of Thm. 3.2 also works for the generalized case. This is formalized in the following proposition.

PROPOSITION E.1. Given the premises of Thm. 4.1, let

$$Z^* := v Y. \bigcup_{b \in [1, s]} \mu b Y. (b F \cap Cpre(Y)) \cup Apre(Y, b X) \quad (72a)$$

and

$$\bar{Z}^* := v Y. \bigcup_{b \in [1, s]} v b Y. \mu b X. (b F \cap Cpre(Y)) \cup Apre(b, b X). \quad (72b)$$

Then $\bar{Z}^* = Z^*$.

However, as in (72) a conjunction is used to update $Y$, the proof is not as straightforward as for (43). We therefore separately show for both equations (72a) and (72b) that, upon termination, we have $Y^* = b X^*$ for all $b \in [1, s]$. Both claims are formalized in Lem. E.1 and Lem. E.2, respectively.

LEMMA E.1. Given the premises of Prop. E.1, let $b X^i$ be the set computed in the i-th iteration over the fixpoint variable $b X$ in (72a) during the last iteration over $Y$, i.e., $Y = Z^*$ already.

Further, we define $b X^0 = \emptyset$ and $b X^i := \bigcup_{b \in [1, s]} b X^i$. Then it holds that $Z^* = b X^*$ for all $b \in [1, s]$.

PROOF. We fix $Y = Z^*$ and $b \subseteq [1, s]$ and observe from (72a) that

$$b X^0 = (b F \cap Cpre(Z^*))$$

and therefore

$$b X^i = b X^0 \cup (b F \cap Cpre(Z^*)) \cup \text{Apre}(Z^*, b X^0) \subseteq (b F \cap Cpre(Z^*)) \cup \text{Apre}(Z^*, b X^0) \subseteq b X^0$$

With this, we have in general that

$$b X^{i+1} = b X^i \cup \text{Apre}(Z^*, b X^i) = (b F \cap Cpre(Z^*)) \cup \text{Apre}(Z^*, b X^i)$$

which implies $b X^{i+1} \subseteq b X^i$. Hence, $b X^* := \bigcup_{i \in [1, i_{\text{max}}]} b X^i = b X^i_{\text{max}}$, and therefore, in particular

$$b X^* = (b F \cap Cpre(Z^*)) \cup \text{Apre}(Z^*, b X^*). \quad (73)$$

By recalling that $Z^* = \bigcap_b b X^*$ we see that $Z^* \subseteq b X^*$

For the inverse direction, we use the observation $Z^* \subseteq b X^*$ together with Lem. A.2 to see that $\text{Apre}(Z^*, b X^*) = Cpre(b X^*)$. With this $(b F \cap Cpre(Z^*)) \subseteq Cpre(b X^*) \subseteq Cpre(b X^*) = \text{Apre}(Z^*, b X^*)$, and hence (73) reduces to

$$b X^* = Cpre(b X^*) \supseteq Cpre(Z^*). \quad (74)$$

As the last equality holds for all $b \subseteq [1, s]$ we see that

$$Z^* = \bigcap_b b X^* = \bigcap_b Cpre(b X^*) \supseteq Cpre(Z^*). \quad (74)$$

We can now use (74) to prove that $Z^* \supseteq b X^*$ also holds. To show this, we pick a vertex $v \subseteq b X^*$ and prove that $v \in Z^*$. To that end, observe that either (i) $v \in (b F \cap Cpre(Z^*)) \subseteq Cpre(Z^*) \subseteq Z^*$ which immediately proves the statement, or (ii) $v \in \text{Apre}(Z^*, b X^*)$. If (ii) holds we again have two cases. Either (a) $v \in Cpre(b X^*)$ which implies that there exists a finite sequence $Cpre(Cpre(... Cpre(b X^i) ...))$ where $b X^1 = b F \cap Cpre(Z^*) \subseteq Cpre(Z^*) \subseteq Z^*$ and therefore $v \in Cpre(Cpre(... Cpre(Z^*) ...)) \subseteq Z^*$. Finally we could have (b) that $v \in Cpre(b X^i) \cap Cpre(b X^i) \subseteq Cpre(Z^*) \subseteq Cpre(Z^*) \subseteq Z^*$, which again proves the statement. \hfill \Box
LEMMA E.2. Given the premises of Prop. E.1, let \( b^i \) be the set computed in the \( i \)-th iteration over the fixpoint variable \( b^i \) in (72b) during the last iteration over \( Y \), i.e., \( Y = \bar{Z}^* \) already. Further, we define \( b^0 = V \) and \( b^i := \bigcap_{i>0} b^{i-1} \). Then it holds that \( \bar{Z}^* = b^V \) for all \( b \in [1; s] \).

Proof. Recall that \( \bar{Z}^* = \bigcap_{b} b^V \) from the structure of the fixpoint algorithm in (72b). To prove \( \bar{Z}^* = b^V \) for all \( b \in [1; s] \) it therefore suffices to show that \( b^V = b^V \) for any two \( b, b' \in [1; s] \) s.t. \( b \neq b' \).

Towards this goal, recall from Thm. 3.3 that \( b^V \) is exactly the set of states from which player 0 can win a fair adversarial reachability game with target \( b^T := b^F \cap \text{Cpre}(\bar{Z}^*) \). However, every state \( v \in b^T \) allows player 0 to force the game to a state \( v' \in \bar{Z}^* = \bigcap_{b'} b^V \). Therefore, by definition player 0 has a strategy to reach a state \( v' \in b^V \) from any state \( v \in b^V \) for any \( b' \in [1; s] \) s.t. \( b \neq b' \). As, however \( b^V \) is defined as the winning region of player 0 w.r.t. the goal set \( b^T := b^F \cap \text{Cpre}(\bar{Z}^*) \), we know that there actually exists a player 0 strategy to drive the game from any state \( v \in b^V \) to \( b^T \), and therefore, by definition \( b^V \subseteq b^V \). As this inclusion holds mutually for all \( b, b' \in [1; s] \) s.t. \( b \neq b' \) we have that \( b^V = b^V \). With this, it immediately follows that \( \bar{Z}^* = b^V \) for all \( b \in [1; s] \). \( \square \)

With Lem. E.1 and Lem. E.2 in place, we see that for every update of \( Y \) the structure of the fixed-point over \( b^V \) and \( b^X \) upon termination of \( b^V \) coincides with the one in (43). With this, Prop. E.1 immediately follows from Lem. B.1.

Using Prop. E.1 we know that (72a) and (72b) compute the same set. Hence, we can use (72b) instead of (26) to prove Thm. 4.1. This allows us to simply reduce the proof of Thm. 4.1 to Thm. 3.2 and Thm. 3.3 as formalized below.

Proof of Thm. 4.1. Soundness & Completeness: Let us define \( \bar{Z}^* = \Gamma \) to be the set of states computed by the fixpoint algorithm in (12). Then it follows from (72b) that

\[
\bar{Z}^* = vY. \bigcap_{b \in [1; s]} Z^*((Q \cap b^F \cap \text{Cpre}(Y), Q)).
\]

In particular, it follows from Lem. E.2 that

\[
\bar{Z}^* = Z^*((Q \cap b^F \cap \text{Cpre}(\bar{Z}^*), Q)) \forall b \in [1; s].
\]

Now let us define \( b^W \) to be the fair adversarial winning state set for \( b^V \)

\[
b^V = \square Q \wedge \square \emptyset \Rightarrow b^F.
\]

With this, it follows from Thm. 3.2 that \( \bar{Z}^* = b^W \) for all \( b \in [1; s] \). Therefore, we obviously have \( \bigcap_{b \in [1; s]} b^W = \bar{Z}^* \).

Now let \( W \) be the fair adversarial winning set w.r.t.

\[
\psi = \square Q \wedge \bigcap_{b \in [1; s]} \emptyset \Rightarrow \square \emptyset \Rightarrow b^F.
\]

(compare (24)). Then we always have \( W \subseteq \bigcap_{b \in [1; s]} b^W \) which immediately implies \( W \subseteq \bar{Z}^* \). However, as \( aW = b^W \) for all \( a, b \in [1; s] \), we know that \( \psi \) holds for all \( v \in \bar{Z}^* \), hence \( Z^* \subseteq W \).

Strategy construction: We can define a rank function for every \( b \) as in (41) within the proof of Thm. 3.3 (see App. B.1), i.e.,

\[
b_{rank}(v) = i \quad \text{iff} \quad v \in b^i \setminus b^{i-1}.
\]

Then, we have a different strategy, \( b_{\rho_0} \), which is defined via (41) (see App. B.1) using the corresponding \( b^i \) rank function. With this, we define a new strategy \( \rho \) which circles through all possible goal sets in a pre-defined order. That is

\[
\rho_0(v, b) = \begin{cases} b_{\rho_0}(v), & v \notin b^F \\ b_{\rho_0}(v), & v \in b^F \end{cases}
\]

(76)

where \( b^+ = b + 1 \) if \( b < s \) and \( b^+ = 1 \) if \( b = s \).

The strategy in (76) is obviously winning for \( \psi \) in (24) as every \( b_{\rho_0} \) is a winning strategy for \( \psi \) (from Thm. 3.2) and upon reaching \( b^F \) we know that the respective state \( v \) is also contained in \( \text{Cpre}(\bar{Z}^*) \) where \( \bar{Z}^* = b^V \). Now it follows from the definition of \( \text{Cpre} \) that \( \text{Cpre}(b^V) \subseteq b^V \), hence, allowing to apply \( b_{\rho_0} \) upon reaching \( b^F \). \( \square \)

E.2 Proof for Thm. 4.2

We show how the proof of Thm. 3.1 in App. C needs to be adapted in order to prove the generalized version of Thm. 3.1, namely Thm. 4.2, instead.

Strategy Construction: Similar to the finite-memory strategy constructed for generalized Büchi games in App. E.1, the strategy for generalized Rabin games needs to remember the index of all the goal sets currently “chaised” for each permutation index up to \( p_j \). To formalize this, we define the set of full goal chain sequences for a given generalized Rabin specification \( \bar{R} \) by

\[
\Phi(\bar{R}) := \{ \ell_0 \ldots \ell_k \mid \ell_0 = 1, \ell_j \in [0; m_j] \}.
\]

(77)

If \( \bar{R} \) is clear from the context we simply write \( \Phi \). Given a goal chain prefix \( \phi := \ell_0 \ell_1 \ldots \ell_{j-1} \) we can now construct a ranking for each such prefix, using the flattening of (27) instead of (7). This yields the following proposition which follows from Prop. C.1 by simply annotating all terms with the goal chain prefix \( \phi \).

PROPOSITION E.2. Let \( \delta = \rho_0 \ldots \rho_{j-1-i} \) be a configuration prefix, \( \phi := \ell_0 \ell_1 \ldots \ell_{j-1} \) a goal chain prefix, \( p_j \in P \setminus \{ p_1, \ldots, p_{j-1} \} \) the next permutation index, \( \ell_j \in [1; m_{p_j}] \) the next goal set and \( i_j > 0 \) a counter for \( p_j \). Then the flattening
of (27) for this configuration and goal prefix is given by
\[ \phi_iX_{\delta_{p_j}} = \phi_iS_\delta \cup \phi_iC_{\delta_{p_j}} \cup \phi_iA_{\delta_{p_j}} \]  
where
\[ Q_{p_0...p_a} := \bigcup_{b=0}^{a} \bar{R}_{p_b}, \]  
\[ \ell_iC_{\delta_{p_i}} := \left( Q_{\delta_p} \cap \ell_iC_{\delta_{p_i}} \cap \text{Cpre}(Y_{\delta_{p_i}}) \right) \]  
\[ \cup \left( Q_{\delta_p} \cap \text{Apre}(Y_{\delta_{p_i}}) \cup \phi_iX_{\delta_{p_i} - 1} \right) \]  
\[ \ell_i...\ell_iA_{\delta_{p_i}} := \bigcup_{b=0}^{a} \ell_iC_{p_b...p_a}, \]  
\[ \phi_iA_{\delta_{p_j}} := \left( \bigcup_{i=1}^{p_{j+1}} \left( \bigcup_{i,j+1>0} \phi_iX_{\delta_{p_j},p_j} \right) \right) \]  
Again we see that this flattening follows directly from the structure of the fixpoint algorithm in (27) and the definition of \( \&_{p_j} \) in (27b). Using the flattening of (27) in (78) we can define a ranking function for each goal chain prefix \( \phi \) identical to Def. C.1. That is, given the premises of Prop. E.2, we define \( \phi_iR : V \rightarrow 2^D \) s.t. (i) \( \phi_iR(v) \) for all \( v \in V \), and (ii) \( \phi_i\delta_{p_j} \in \phi_iR(v) \) iff \( v \in \phi_i\delta_{p_j} \). The ranking function \( \phi_i\text{rank} : V \rightarrow D \) is then again defined as in Def. C.1 s.t. \( \phi_i\text{rank} : v \mapsto \min\{ \phi_i\text{rank}(w) \} \). Similarly, we can define a memoryless winning strategy for every fixed goal sequence \( \phi \) as in (48). That is,
\[ \phi_{0}(v) := \min_{(v,w)\in E} (\phi_i\text{rank}(w)). \]  
Now, similar to the proof of Thm. 4.1 (see Sec. 4.1) we can “stack” these memoryless winning strategies to define a new strategy with finite memory which circles through all possible goal sets in a pre-defined order. That is
\[ p_{0}(v, \phi_{\ell_j}) := \begin{cases} \phi_{i}p_{0}(v) & \forall \notin \ell_{\ell_{F}} \\ \phi_{i}p_{0}(v) & \forall \ell_{\ell_{F}} \end{cases} \]  
where \( \ell_{\ell_{F}} := \ell_{j} + 1 \) if \( \ell_{j} < m_{p_j} \) and \( \ell_{\ell_{F}} := 1 \) if \( \ell_{j} = m_{p_j} \).

Using this goal chain dependent ranking function, the proof of soundness and completeness of (27) along with the proof that \( p_{0} \) in (80) is indeed a winning strategy for player 0 in the fair adversarial generalized Rabin game, follows exactly the same lines as the proof in App. C. That is, we iteratively consider instances of the flattening in (78), starting with \( j = k \) as the base case, and doing an induction from \( "j + 1" \) to \( "j" \). To this end, we consider a generalized local winning condition which refers not only to the current configuration-prefix \( \delta = p_{0}...p_{j-1}i_{j-1} \) but also to the current goal chain prefix \( \phi := \ell_{0}...\ell_{j-1}. \) Hence, (49) gets modified to
\[ \phi_{\psi}(\delta_{p_j}) := \begin{cases} Q_{\delta_p} \cup \phi_{i}S_\delta \cup \phi_{i}C_{\delta_{p_j}} \cup \phi_{i}A_{\delta_{p_j}} \cup \phi_{i}X_{\delta_{p_j}} & \forall \epsilon \phi_{G_{p_j}} \\ Q_{\delta_p} \cup \phi_{i}S_\delta \cup \phi_{i}C_{\delta_{p_j}} \cup \phi_{i}A_{\delta_{p_j}} \cup \phi_{i}X_{\delta_{p_j}} & \forall \epsilon \phi_{G_{p_j}} \end{cases} \]  
where \( \bar{R}_{j} = p_{\epsilon} \setminus \{ p_{0},...p_{j} \} \). With this, it becomes obvious that the proof of soundness, completeness and the winning strategy for Thm. 4.2 follows exactly the same reasoning as in App. C while additionally using Thm. 4.1 to reason about the conjunction over goal sets.

The only remaining part to be shown concerns the last line of \( \phi_{i}(\delta_{p_j}) \). For this, we recall from App. C.2 that the induction step from \( "j + 1" \) to \( "j" \) relies on the fact that
\[ \phi_{i}(\psi_{\delta_{p_j}}) := \bar{Q}_{\psi_{\delta_{p_j}}} \cup \left( \bigcup_{i=1}^{p_{j+1}} \phi_{i}(\psi_{\delta_{p_j+1}}) \right) \]  
is indeed equivalent to the last line of \( \phi_{\psi_{\delta_{p_j}} \psi_{\delta_{p_j+1}}} \), where \( \phi_{\psi_{\delta_{p_j+1}}} \) denotes the last two lines of \( \phi_{\psi_{\delta_{p_j}}} \), with \( \phi := \phi_{\ell_{j}} \) and \( \delta_{j} := \delta_{p_j} \).

For (non-generalized) Rabin games this equivalence is proven in App. C.4.2. It can be seen by inspection within this proof, that using a conjunction over goal sets instead of a single goal set within the second and third line of \( \phi_{\psi_{\delta_{p_j}}} \) does not change any step in the derivation. Therefore, the same derivation can be used in the generalized case and is therefore omitted. This concludes the proof of Thm. 4.2.

### E.3 Proof of Thm. 4.3

Within this section we prove Thm. 4.3. That is, we prove that for GR(1) winning conditions, the fixpoint computing \( Z^* \) in (27) simplifies to the one in (30). This is formalized in the next proposition.

**Proposition E.3.** Let \( \bar{R} \) be a generalized Rabin condition with \( k \) pairs s.t. (29) holds for \( r := k - 1 \). Further let \( Z^* \) be the fixed point computed by (27) and \( \bar{Z}^* \) the fixed point computed by (30). Then \( Z^* = \bar{Z}^* \).

If Prop. E.3 holds, we immediately see that Thm. 4.3 directly follows from Thm. 4.2. It therefore remains to prove that Prop. E.3 holds.

**Proof.** First, consider an arbitrary permutation sequence \( \delta = p_{0}...p_{k}. \) Then we know that there exists exactly one \( j > 0 \) s.t. \( p_j = k \) and all other indices come from the set \( \{1; r\} \). We can therefore define \( y' = p_{1}...p_{j+1} \) and \( y'' = p_{j+1}...p_{k} \) s.t. \( p_{i} \in \{1; r\} \) for all \( i \neq j \). We note that \( y' = \epsilon \) if \( j = 1 \) and \( y'' = \epsilon \) if \( j = k \). With this we have \( \delta = p_{0}y'y''. \)
By inspecting (29) we see that the first $r$ pairs of the
generalized Rabin condition induced by the GR(1) specification
actually form a Generalized Co-Büchi condition (compare
(22) in Sec. 3.4). Hence, given a permutation sequence
$\delta = \rho_0 Y I \rho_1 Y I \ldots$ we can use the same reasoning as in the proof
of Thm. 3.6 in App. D.3 to see that
\[ C_{p_1} \supseteq \ldots \supseteq C_{p_{j-1}} \text{ and } C_{p_{j+1}} \supseteq \ldots \supseteq C_{p_k}. \] (83)

Now recall from the proof of Thm. 3.4 in App. D.1 that
these inclusions allow to recursively apply Lem. A.4 to
delete all $C$ terms which are included in either $C_{p_0}$ or $C_{p_{j+1}}$, along
with the fixpoint variables used within these terms (compare
Lem. D.2 where now $y'$ and $y''$ are interpreted as decreasing
sub-sequences). Applying these simplifications to (27)
in exactly the same manner as these simplifications where
applied to (7) in the proof of Thm. 3.4) results in a simpler
fixpoint algorithm where all permutation sequences have
the form $\delta = 0 q_1 k q_2$ with $q_1 \neq q_2$ and $q_1, q_2 \in \{1; r\}$ (here $q_1$
and $q_2$ correspond to $p_1$ and $p_{j+1}$ in (83), and $k$ corresponds to $p_j$).

Now we can inspect (29) again to see that $R_i \supseteq R_0$
and $G_i \supseteq G_{p_j}$ for all $i \in \{1; r\}$ and $j \in \{1; s\}$. This can be under-
stood as a “generalized Rabin chain condition” (compare
(15) in Sec. 3.4). Hence, we can apply Lem. D.2 one more
time, now to the “decreasing sub-sequence” $q_1 k$
eq q_2$ within ev-
ery permutation sequence. Again, utilizing this argument
iteratively in (27) yields a simpler fixpoint algorithm which
only contains permutation sequences $\delta = 0 k a$ with $a \in \{1; r\}$.

This proves that $Z'$ is equivalent to the set
\[ vY_0, \mu X_0, vY_k : \bigcup_{b \in \{1; s\}} vY_a, \mu X_a : \bigcup_{a \in \{1; r\}} vY_a. \]

Now inserting the simplifications for terms from the gen-
eralized Co-Büchi part (see (70) in App. D.3) and using $R_0 = G_0 = \emptyset$, we obtain
\[ vY_0, \mu X_0, vY_k : \bigcup_{b \in \{1; s\}} vY_a, \mu X_a : \bigcup_{a \in \{1; r\}} vY_a. \]

Apre($Y_0, X_0$)
\[ \cup (\langle T \cap Cpre(Y_k) \rangle \cup Apre(Y_k, \langle X_k \rangle)
\]
\[ \cup (\langle A \cap Cpre(Y_a) \rangle). \]

Now we can apply Lem. A.4 (iii) again to remove the first
occurrence of the Apre term to obtain the same expression
as in (30). This concludes the proof. \[\square\]

F ADDITIONAL PROOFS FOR SEC. 5

F.1 Preliminaries

1/2-player game: A special case of 2/2-player game graphs
is a Markov Decision Process (MDP) or 1/2-player game, which
is obtained by assuming that every Player 0 vertex in $V_0$
has only one outgoing edge.\(^4\) Analogously to the 2/2-player
games, for a given 1/2-player game graph $G$, we use the
notation $P^p_\nu(\models \phi)$ to denote the probability of occurrence
of the event $\models \phi$ when the runs initiate at $v^0$ and when
Player 1 uses the strategy $p_1$.

Role of end components in 1/2-player game: Limiting
behaviors in a 1/2-player game can be characterized using
the structure of the underlying game graph. We summarize
one key technical argument in the following.

Let $G = (V, V_0, V_1, V_r, E)$ be a 1/2-player game graph. A
set of vertices $U \subseteq V$ is called closed if (1) for every $v \in
U \cap V_r, E(v) \subseteq U$, and (2) for every $v \in U \cap (V_0 \cup V_1),
E(v) \cap U \neq \emptyset$. A closed set of vertices $U$ induces a subgame
graph $(V', V_0', V_1', V_r', E')$, denoted by $G \downarrow U$, which is itself a
1/2-player game graph and is defined as follows:

- $V' = U$,
- $V_0' = U \cap V_0$,
- $V_1' = U \cap V_1$,
- $V_r' = U \cap V_r$, and
- $E' = E \cap (U \times U)$.

A set of vertices $U \subseteq V$ of a 1/2-player game graph $G$ is an
end component if (a) $U$ is closed, and (b) the subgame graph
$G \downarrow U$ is strongly connected.

Denote the set of all end components of $G$ by $\mathcal{E} \subseteq 2^V$.

The next lemma states that under every strategy $p_1$ (being
memoryless or not) of Player 1 in the 1/2-player game, the
set of states visited infinitely often along a play is an end
component with probability one.

**Lemma F.1.** [8, Thm. 3.2] For every 1/2-player game graph,
for every vertex $\nu \in V$, and every Player 1 strategy $p_1$,
\[ P^p_\nu(\models \phi) = \bigvee_{U \in \mathcal{E}} \left( \emptyset \cup U \land (\bigvee_{u \in U} \Box \phi u) \right) = 1. \] (84)

This lemma implies the following corollary, which is moti-
vated by similar claim for Rabin winning conditions in the
literature [5].

**Corollary 1.** For a given 1/2-player game, for a given ver-
tex $\nu \in V$, and for a given Player 1 strategy $p_1$, a generalized
Rabin condition $\mathcal{R} = \{G_1, R_1\}, \ldots, (G_k, R_k)\}$ is satisfied
almost surely if and only if for every end component $U \subseteq V$
reachable from $v^0$, there is a $j \in \{1, 2, \ldots, k\}$ such that $U \cap R_j = \emptyset$ and for every $l \in \{1; m_j\}, U \cap \Box \neg E_j \neq \emptyset$.

\(^4\)Alternatively, we could also define 1/2-player game graphs by restricting
the outgoing edges from the Player 1 vertices; our choice is actually tailored
for the content of the rest of the section.
F.2 Proof of Thm. 5.2

We define the fairness constraint on the random edges of $G$ as per Eq. (3):

$$q^f := \land_{(u,v') \in E_r} \square q_u \to \square (q_u \land q_{v'}).$$

We first show that $W \subseteq W^{a.s.}$. Consider an arbitrary initial vertex $v^0 \in W$ and an arbitrary strategy $\rho_1$ of Player 1 in $G$. Let $\rho^*_0$ be a corresponding winning strategy for Player 0 from $v^0$ for the fair adversarial game over Derand($G$) for the winning condition $\phi$. By definition, $\rho^*_0$ realizes the specification $\phi$, whenever the adversary satisfies the strong fairness condition on the live edges in Derand($G$). On the other hand, the live edges in Derand($G$) are exactly the random edges in $G$. In other words, we already know that if we apply the same strategy $\rho^*_0$ to $G$, then $\inf_{\rho_1 \in R_1} p^{\rho^*_0, \rho_1}(G \models \phi^f \to \phi) = 1$.

We first show that the random edges $E_r$ also satisfy the strong fairness condition $q^f$ almost surely; actually we show that the probability of violation of $q^f$ in $G$ is 0. Consider the following:

$$p^{\rho^*_0, \rho_1}_{v^0}(G \models \neg q^f)$$

$$= p^{\rho^*_0, \rho_1}_{v^0}(G \models \neg (\land (u,v') \in E_r) \square q_u \to \square (q_u \land q_{v'}))$$

$$= p^{\rho^*_0, \rho_1}_{v^0}(G \models \lor (u,v') \in E_r \square q_u \land \square \neg (q_u \land q_{v'}))$$

$$\leq \sum_{(u,v') \in E_r} p^{\rho^*_0, \rho_1}_{v^0}(G \models \square q_u \land \square \neg (q_u \land q_{v'})).$$

We show that the right-hand side of the last inequality equals to 0 by proving that for every $(u,v') \in E_r$,

$$p^{\rho^*_0, \rho_1}_{v^0}(G \models \land \square q_u \land \square \neg (q_u \land q_{v'})) = 0.$$

Consider any arbitrary $(u,v') \in E_r$ and assume that the probability of taking the edge $(u,v')$ from $v$ is $p_1$. Let $\pi$ be a play on $G$ and $(i_0, i_1, i_2, \ldots)$ be the infinite sequence of time indices when the vertex $v$ is visited. For every $i_k$, the probability of not visiting $v'$ for the next $l$ time steps ($(i_k + 1, \ldots, i_{k+l})$ is given by $(1 - p)^l$, which converges to 0 as $l$ approaches $\infty$. This proves that for every $i_k$, eventually there will be a $v'$ at $(i_k + 1)$ with probability 1; in other words $v'$ will be visited infinitely often with probability 1. Hence, it follows that $\sum_{(u,v') \in E_r} p^{\rho^*_0, \rho_1}_{v^0}(G \models \land \square q_u \land \square \neg (q_u \land q_{v'})) = 0$, which in turn establishes that $p^{\rho^*_0, \rho_1}_{v^0}(G \models \neg q^f) = 0$.

Now consider the following derivation:

$$p^{\rho^*_0, \rho_1}_{v^0}(G \models \phi^f \to \phi)$$

$$= p^{\rho^*_0, \rho_1}_{v^0}(G \models \neg \phi^f \lor \phi)$$

$$\leq p^{\rho^*_0, \rho_1}_{v^0}(G \models \neg \phi^f) + p^{\rho^*_0, \rho_1}_{v^0}(G \models \phi)$$

$$= 0 + p^{\rho^*_0, \rho_1}_{v^0}(G \models \phi).$$

Since we know that $p^{\rho^*_0, \rho_1}_{v^0}(G \models \phi^f \to \phi) = 1$, hence it follows that $p^{\rho^*_0, \rho_1}_{v^0}(G \models \phi) = 1$.

Next, we show that $W \supseteq W^{a.s.}$. Consider an arbitrary initial vertex $v^0 \in W^{a.s.}$. Let $\rho^*_0$ be a corresponding almost sure winning strategy for Player 0 from $v^0$ in the 2/1-player game $G$ with the specification $\phi$. We show that Player 0 wins the fair adversarial game over Derand($G$) for the winning condition $\phi$ from vertex $v^0$ using the strategy $\rho^*_0$.

Let $\rho_1 \in R_1$ be any arbitrary Player 1 strategy in the game Derand($G$) such that the unique resultant play $\pi = (v^0, v^1, \ldots)$ due to $\rho_0^*$ and $\rho_1$ satisfies the fairness assumption. We use the notation Inf($\pi$) to denote the set of infinitely occurring vertices along the play $\pi$, i.e., $\text{Inf}(\pi) := \{w \in V | \forall m \in \mathbb{N}_0 \exists n > m . w^n = w\}$. First we show that (i) the set of vertices Inf($\pi$) forms an end component in $G$, and moreover (ii) there exists a Player 1 strategy $\rho_1^*$ in the game $G$ such that $p^{\rho^*_0, \rho_1^*}_{v^0}(G \models \text{Inf}(\pi)) > 0$. Claim (i) follows by observing the following:

- For all $v \in \text{Inf}(\pi) \cap V_r, V_r(v) \subseteq \text{Inf}(\pi)$, as otherwise in Derand($G$) there would be a vertex in $E^f(v)$ and outside Inf($\pi$) which would be visited infinitely many times due to infinitely many visits to $v$.
- For every $v \in \text{Inf}(\pi) \cap (V_0 \cup V_r), E(v) \neq \emptyset$, as otherwise in Derand($G$) the play $\pi$ would reach a dead-end.
- The subgame graph $G \downarrow \text{Inf}(\pi)$ is strongly connected, as otherwise in Derand($G$) there would be two vertices $u, v \in \text{Inf}(\pi)$ so that $v$ would not be reachable from $u$, contradicting the assumption that both $u$ and $v$ are visited infinitely often by $\pi$.

Claim (ii) follows by defining a strategy $\rho_1^* \equiv \rho_1$ on $G$. Now observe that for every edge $(u,v')$ chosen by Player 1 from a vertex $v \in \text{dom}(E^f)$ in Derand($G$), there exists a corresponding positive probability edge $(u,v')$ in $G$. Since Inf($\pi$) is entered by $\pi$ after finite time steps, hence the Claim (ii) follows.

Now, from Cor. 1 it follows that there is a $j \in \{1, 2, \ldots, k\}$ such that $\text{Inf}(\pi) \cap R_j = \emptyset$ and for every $i \in \{1, \ldots, m_j\}$, $\text{Inf}(\pi) \cap G_j \neq \emptyset$. Thus the play $\pi$ satisfies the generalized Rabin condition $\mathcal{R}$. Since this holds for any arbitrary Player 1 strategy, hence $W \supseteq W^{a.s.}$ and $\rho^*$ is the corresponding winning strategy for Player 0.