Stopping Criteria for Value Iteration on Concurrent Stochastic Reachability and Safety Games

Marta Grobelna*, Jan Křetínský†*, Maximilian Weininger‡§*

*Technical University of Munich
Munich, Germany
marta.grobelna@tum.de

†Masaryk University
Brno, Czech Republic
jan.kretinsky@fi.muni.cz

‡Ruhr-University Bochum
Bochum, Germany
maximilian.weininger@rub.de

§Institute of Science and Technology Austria
Klosterneuburg, Austria

***Abstract*—We consider two-player zero-sum concurrent stochastic games (CSGs) played on graphs with reachability and safety objectives. These include degenerate classes such as Markov decision processes or turn-based stochastic games, which can be solved by linear or quadratic programming; however, in practice, value iteration (VI) outperforms the other approaches and is the most implemented method. Similarly, for CSGs, this practical performance makes VI an attractive alternative to the standard theoretical solution via the existential theory of reals.**

VI starts with an under-approximation of the sought values for each state and iteratively updates them, traditionally terminating once two consecutive approximations are ϵ-close. However, this stopping criterion lacks guarantees on the precision of the approximation, which is the goal of this work. We provide *bounded* (a.k.a. interval) VI for CSGs: it complements standard VI with a converging sequence of over-approximations and terminates once the over- and under-approximations are ϵ-close.

***Index Terms*—Formal methods, foundations of probabilistic systems and games, verification, model checking**

I. Introduction

Concurrent stochastic games (CSGs, e.g., [14]): We consider two-player zero-sum games played on a graph. Every vertex represents a *state*. Edges are directed, originating from one state and leading to one or several other states. An edge is associated with a *probability distribution* over the successor states. A *play* proceeds through the graph as follows: starting from an initial state, both players simultaneously and independently choose an *action*, determining the edge to follow. Then, the next state is sampled according to the probability distribution, and the process is repeated in the successor.

We focus on infinite-horizon reachability and safety objectives [14]. The goal of reachability is to maximize the probability of reaching a given goal state. In contrast, the safety objective aims to maximize the probability of staying within a given set of states. The two objectives are dual, as instead of maximizing the probability of reaching a set of target states, one can minimize the probability of staying within the set of non-target states. Thus, we refer to both types collectively as CSGs. Popular subclasses of CSGs include turn-based stochastic games (TSGs), where the players make decisions in turns, or Markov decision processes (MDPs), which involve only one player.

The value problem: In CSGs, memoryless (a.k.a. stationary) strategies suffice for both players, meaning they yield the same supremum probability as history-dependent strategies. However, unlike in TSGs, the strategies for CSGs require *randomization*, meaning players choose distributions over actions rather than single actions. Additionally, while the safety objective player, Player $\mathscr{S}$, can attain optimal strategies [39], the reachability objective player, Player $\mathscr{R}$, only possesses ε*-optimal* strategies for a given $\varepsilon > 0$ [21]. As a result, the problem of deciding whether the supremum probability (a.k.a. *the value*) is at least p for $p \in [0, 1]$, is thus more subtle than for the mentioned subclasses. While for MDPs the value problem is in P, and for TSGs it is known to be in $\mathsf{NP} \cap \mathsf{co\text{-}NP}$, for CSGs it can be elegantly encoded into the *existential theory of reals (ETR)*, which is only known to be decidable in PSPACE (although not known to be complete for it) [20]. Unfortunately, algorithms for ETR are practically even worse than the more general, doubly exponential methods for the first-order theory of reals [40]. *"Finding a practical algorithm remains a very interesting open problem"* [25].

Practical approximation: We focus on algorithms *approximating* the value with a predefined precision $\varepsilon > 0$. Both for MDPs and TSGs, dynamic programming techniques such as *value iteration (VI)* or strategy iteration (SI) are practically more efficient than mathematical programming (linear or quadratic, respectively) [27], [30]. Thus, VI algorithms are prevalently used and implemented in popular tools such as PRISM-GAMES [33], motivating the focus on VI here.

Problem and our contribution: In VI, the lowest possible value is initially assigned to each state and then iteratively improved, computing an under-approximation of the value, converging to it in the limit. The algorithm (in practical implementations) terminates once two consecutive approximations are ε-close. However, the result can then be arbitrarily imprecise [23]. In this work, we introduce *bounded value iteration for CSGs*, following its previous success for MDPs [1], [5] or TSGs [18]. Its main idea is to enhance standard VI by introducing an over-approximation of the values computed in parallel with the under-approximation. Once the upper and lower bounds are ε-close, VI terminates, ensuring that the true value is at most ε away from the obtained approximation. Since the naïve formulation of an upper bound does not converge to the value

This research was funded in part by the German Research Foundation (DFG) project 427755713 GOPro, the MUNI Award in Science and Humanities (MUNI/I/1757/2021) of the Grant Agency of Masaryk University, the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 101034413, and the ERC Starting Grant DEUCE (101077178).

in general, previous approaches, notably including [7], have attempted to fix this but have failed. In this paper, we finally provide a valid solution.

Technical challenge: The fundamental technical difficulties arise from the following. The non-convergence of upper-bound approximations is primarily due to cyclic components, so-called *end components (ECs)*, see [14], [18], [29]. Notably, non-convergence, for this reason, is an issue present already in MDPs and TSGs; see [18], [29]. Solutions have been developed over the past decade for these two subclasses. Indeed, for MDPs with reachability objective, these end components can effectively be removed from the graph without altering the value [6], [24]. Other objectives were considered in [1], [3]. TSGs with reachability objectives already require more careful analysis, decomposing the end components into sub-parts, so-called *simple ECs* [18]. A comprehensive framework for various quantitative objectives was proposed in [29]. Unfortunately, the idea of simple ECs is not easily extendable from TSGs to CSGs due to the absence of optimal strategies.

Summary of our contribution: We provide a stopping criterion for VI on CSGs, solving an open problem with erroneous solution attempts in the literature (see the related work in Subsec. I-A below). To this end, we unravel the recursive hierarchical structure of end components in CSGs (see Rem. 31) and adapt the bounded VI algorithm.

A. Related Work

Available approaches: The PSPACE-algorithm introduced in [20] for deciding whether the value of a given game is at least p, for $p \in [0,1]$, allows for a trivial stopping criterion by iteratively executing this algorithm for a suitable sequence of $(p_i)_{i\in\mathbb{N}}$ (intuitively, we choose p_i such that alternatingly, the value of the game is above and below, while the distance between two consecutive p_i's monotonically decreases). However, this criterion is impractical since it uses the existential theory of reals [20]. The best known complexity upper bound comes from [22] and states that the problem of approximating the value of a CSG is in TFNP[NP], i.e. total function from NP with an oracle for NP. However, the proposed algorithm is not practical, as it relies on guessing a floating point representation of the value and optimal strategies for both players. A recursive bisection algorithm was introduced in [26], which is also impractical as its time complexity is best-case doubly exponential in the number of states. For the algorithms commonly employed for in the non-concurrent case, VI and SI, [25] provide doubly exponential lower and upper bounds on the number of iterations that VI requires in the worst-case for computing an ε-approximation. Their counter-example uses a CSG where all states have value 1. Thus, the worst-case complexity of our approach is the same since an additional over-approximation does not speed up convergence in this example. Nonetheless, these results are worst-case bounds, i.e. they hold a priori for all games; earlier termination is possible, but necessarily requires a stopping criterion, which has so far been elusive. Finally, in [38], an algorithm is provided that, unlike all other known algorithms, only has a *single*-exponential dependency on the number of states. A practical comparison of our value iteration and [38] is an interesting future step, as better worst-case complexity of an algorithm need not translate to better practical performance on typical instances; for example, in MDPs, worst-case exponential VI and SI typically outperform the polynomial approach of linear programming [27].

Previous attempts at stopping criterion: A stopping criterion for SI and VI on CSGs was first presented in [7], but later found to contain an irreparable mistake [10]. Specifically, the algorithm returned over-approximations smaller than the actual values in certain situations, as detailed in [10]. We analyze the counter-example from [10] in App. D-A Later, [19] proposed a stopping criterion for VI, which also contains a fundamental flaw: it fails to converge for CSGs with ECs. We analyze the counter-example to this approach in App. D-B. *Our work thus delivers the first stopping criterion in this context.*

Further directions of related work: Variants of CSGs have appeared very early, under the names of Everett, Gilette, or Shapley game. See [26] for an explanation of all game types, their relations, and algorithms to solve them. These games also consider discounted payoff or limit-average payoff, generalizing the reachability and safety CSGs we consider here. A generalization of CSGs to ω-regular objectives has been considered in [8], [15]. An insightful characterization of optimal strategies in concurrent games with various objectives can be found in [4].

II. Preliminaries

A. Concurrent Stochastic Games

Probability Distributions: For a countable set X, a function $\mu\colon X \to [0,1]$ is called a *distribution* over X if $\sum_{x\in X}\mu(x) = 1$. The *support* of μ is $\mathsf{Supp}(\mu) := \{x \mid \mu(x) > 0\}$. The set of all distributions over X is denoted by $\mathsf{Dist}(X)$.

Concurrent Stochastic Games: A *concurrent stochastic game* (CSG) [16] is a tuple $\mathsf{G} := (\mathsf{S}, \mathcal{A}, \Gamma_{\mathscr{R}}, \Gamma_{\mathscr{S}}, \delta, \mathsf{s}_0, \mathsf{T})$, where S is a finite set of *states*, $\mathcal{A} := \mathsf{A} \times \mathsf{B}$ is a finite set of *actions* with $\mathsf{A} := \{\mathsf{a}_1, \dots, \mathsf{a}_l\}$ and $\mathsf{B} := \{\mathsf{b}_1, \dots, \mathsf{b}_m\}$ the sets of actions available for player $\mathscr{R}$ and $\mathscr{S}$, respectively, $\Gamma_{\mathscr{R}}\colon \mathsf{S} \to (2^{\mathsf{A}} \setminus \emptyset)$ and $\Gamma_{\mathscr{S}}\colon \mathsf{S} \to (2^{\mathsf{B}} \setminus \emptyset)$ are two *enabled actions* assignments and $\delta\colon \mathsf{S} \times \mathsf{A} \times \mathsf{B} \to \mathsf{Dist}(\mathsf{S})$ is a *transition function*, where $\delta(s,\mathsf{a},\mathsf{b})(s')$ gives the *probability of a transition* from state s to state s' when player $\mathscr{R}$ chooses action $\mathsf{a} \in \Gamma_{\mathscr{R}}(s)$ and player $\mathscr{S}$ action $\mathsf{b} \in \Gamma_{\mathscr{S}}(s)$, $\mathsf{s}_0 \in \mathsf{S}$ is an *initial state*, and $\mathsf{T} \subseteq \mathsf{S}$ is a set of *target states*. A CSG is *turn-based* if for every state s only one player has a meaningful choice, i.e. either $\Gamma_{\mathscr{R}}(s)$ or $\Gamma_{\mathscr{S}}(s)$ is a singleton; we call such game a turn-based stochastic game (TSG).

Example 1 (CSGs). Consider the CSG `Hide-Run-or-Slip` depicted in Fig. 1 (an adaption of the `Hide-or-Run` game in [16], [21], [31]). Circles represent states and black dots depict a probabilistic transition with uniform distribution. Each edge is labeled with a pair of actions, the left for player $\mathscr{R}$ and the right for player $\mathscr{S}$; $\square$ is a placeholder for an arbitrary action. We have $\mathsf{S} := \{s_{\mathsf{hide}}, s_{\mathsf{home}}, s_{\mathsf{wet}}\}$, with $\Gamma_{\mathscr{R}}(s_{\mathsf{hide}}) := \{\mathsf{hide}, \mathsf{run}\}$

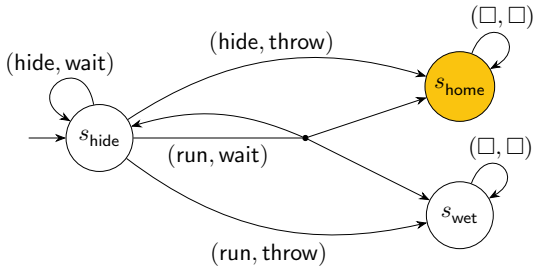

Fig. 1: Example CSG called `Hide-Run-or-Slip`.

and $\Gamma_{\mathscr{S}}(s_0) \coloneqq \{\text{wait}, \text{throw}\}$, s_0 is the initial state denoted by the arrow with no predecessor state, and $\mathsf{T} \coloneqq \{s_{\text{home}}\}$. The game has the following intuitive interpretation: Player $\mathscr{R}$ wants to get home without getting wet. Player $\mathscr{S}$ has a single snowball and can make player $\mathscr{R}$ wet by throwing it at player $\mathscr{R}$. If player $\mathscr{R}$ runs and player $\mathscr{S}$ throws the ball, player $\mathscr{R}$ gets wet. If player $\mathscr{R}$ runs but player $\mathscr{S}$ waits, with a probability of $\frac{1}{3}$ the player reaches home or the player slips instead and with a probability of $\frac{1}{3}$ does not move at all or with a probability of $\frac{1}{3}$ falls on the ground and gets wet. △

Plays: A *play* π of a CSG G is an infinite sequence of states $s_0 s_1 s_2 \cdots$, such that for all $i \in \mathbb{N}$ there are actions $\mathsf{a} \in \Gamma_{\mathscr{R}}(s_i)$ and $\mathsf{b} \in \Gamma_{\mathscr{S}}(s_i)$ with $\delta(s_i, \mathsf{a}, \mathsf{b})(s_{i+1}) > 0$. $\mathsf{Play}(\mathsf{G})$ is the set of all plays and $\mathsf{Play}_s(\mathsf{G})$ the set of all plays $s_0 s_1 s_2 \cdots$ with $s_0 = s$.

Strategies: A *strategy* for player $\mathscr{R}$ (or $\mathscr{S}$) is a function $\rho \colon \mathsf{S} \to \mathsf{Dist}(\mathsf{A})$ (or $\sigma \colon \mathsf{S} \to \mathsf{Dist}(\mathsf{B})$) that assigns a distribution over actions available to player $\mathscr{R}$ (or $\mathscr{S}$) to each state, i.e., for all $s \in \mathsf{S}$, $\mathsf{Supp}(\rho(s)) \subseteq \Gamma_{\mathscr{R}}(s)$ (or $\mathsf{Supp}(\sigma(s)) \subseteq \Gamma_{\mathscr{S}}(s)$).[1] We call a Player $\mathscr{R}$ (or Player $\mathscr{S}$) strategy ρ *pure* if all distributions it returns are Dirac distributions, i.e., at each $s \in \mathsf{S}$ we have a unique action $\mathsf{a} \in \Gamma_{\mathscr{R}}(s)$ (or $\mathsf{b} \in \Gamma_{\mathscr{S}}(s)$) such that $\rho(s)(\mathsf{a}) = 1$ (or $\sigma(s)(\mathsf{b}) = 1$). Otherwise, the strategy is *mixed*. For player $\mathscr{R}$ (or $\mathscr{S}$) we denote the set of strategies by $\mathcal{R}$ (or $\mathcal{S}$) and a single strategy by ρ (or σ).

Markov Decision Processes: Given a CSG G, if we fix a strategy $\rho \in \mathcal{R}$ of player $\mathscr{R}$, the game becomes a *$\mathscr{S}$-Markov Decision Process* (MDP, [41]) G_ρ with the transition function

$$\delta_\rho(s, \mathsf{b})(s') \coloneqq \sum_{\mathsf{a} \in \Gamma_{\mathscr{R}}(s)} \delta(s, \mathsf{a}, \mathsf{b})(s') \cdot \rho(s)(\mathsf{a}),$$

for all $s \in \mathsf{S}$ and $\mathsf{a} \in \Gamma_{\mathscr{R}}(s)$. The MDP induced by a fixed strategy $\sigma \in \mathcal{S}$ is defined analogously.

Markov Chains: Similarly, by fixing a pair of strategies $(\rho, \sigma) \in \mathcal{R} \times \mathcal{S}$, we obtain a *Markov chain* $\mathsf{G}_{\rho,\sigma}$ with the same state space S, the initial state s_0, and the transition probabilities P given by

$$\delta_{\rho,\sigma}(s)(s') \coloneqq \sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} \delta(s, \mathsf{a}, \mathsf{b})(s') \cdot \rho(s)(\mathsf{a}) \cdot \sigma(s)(\mathsf{b}).$$

[1]Since memoryless strategies are sufficient for the objectives considered in this paper, we do not introduce general history-dependent strategies to avoid clutter. We refer to [16] for more details.

Thus, a pair of strategies (ρ, σ) induces a unique probability measure $\mathbb{P}^{\rho,\sigma}_{\mathsf{s}_0}$ over plays in the Markov chain as usual, see [2, Chap. 10.1], where the set of paths starting in s_0 has measure 1.

Objectives: We partition S into T, denoting the set of states player $\mathscr{R}$ wants to reach, and $\mathsf{F} \coloneqq \mathsf{S} \setminus \mathsf{T}$, denoting the set of states player $\mathscr{S}$ wants to confine the game in. We denote the *reachability* objective by $\Diamond\mathsf{T} \coloneqq \{s_0 s_1 s_2 \cdots \mid \exists i \in \mathbb{N} : s_i \in \mathsf{T}\}$ and the *safety* objective by $\Box\mathsf{F} \coloneqq \{s_0 s_1 s_2 \cdots \mid \forall i \in \mathbb{N} : s_i \in \mathsf{F}\}$. The *value* of the objective $\Diamond\mathsf{T}$, i.e. $\mathsf{V}_{\mathscr{R}}(s)$, and the objective $\Box\mathsf{F}$, i.e. $\mathsf{V}_{\mathscr{S}}(s)$, at state s are given by

$$\mathsf{V}_{\mathscr{R}}(s) \coloneqq \sup_{\rho \in \mathcal{R}} \inf_{\sigma \in \mathcal{S}} \mathbb{P}^{\rho,\sigma}_s(\Diamond\mathsf{T})$$
$$\mathsf{V}_{\mathscr{S}}(s) \coloneqq \sup_{\sigma \in \mathcal{S}} \inf_{\rho \in \mathcal{R}} \mathbb{P}^{\rho,\sigma}_s(\Box\mathsf{F}).$$

By the determinacy of CSGs and the duality of these objectives [21], [34], it holds that $\mathsf{V}_{\mathscr{R}}(s) + \mathsf{V}_{\mathscr{S}}(s) = 1$. Consequently, the task of approximating $\mathsf{V}_{\mathscr{R}}(s)$ with a given precision is equivalent to approximating $\mathsf{V}_{\mathscr{S}}(s)$. Further, the objective of minimizing the reachability for T is equivalent to the objective of maximizing safety for F for the *same* player. Consequently, in the following we only focus on maximizing reachability as both minimization and the safety objectives can be reduced to it.

Example 2 (Optimal Strategies Need Not Exist). In CSGs, an optimal strategy for player $\mathscr{R}$ might not exist [31], meaning that at some states, the value is attainable only in the limit. Consider our running example from Fig. 1, `Hide-Run-or-Slip`. Assume for the moment that there is no chance of slipping, i.e. upon playing run and wait, the target state is reached. To win, Player $\mathscr{R}$ has to run eventually. However, Player $\mathscr{S}$ can utilize a strategy that throws with positive probability at all points in time. Thus, Player $\mathscr{R}$ cannot win almost surely.

However, Player $\mathscr{R}$ has the possibility of *limit-sure winning* in [16]: By running with vanishingly low probability ε in every round, the probability of winning is $1 - \varepsilon$. This is because Player $\mathscr{S}$ has the highest probability ε of hitting Player $\mathscr{R}$ by throwing in the first round; throwing in a later round n only has hitting probability ε^n. For any $\varepsilon > 0$, this strategy of Player $\mathscr{R}$ achieves $1 - \varepsilon$. The value, being the supremum over all strategies, is 1.

This notion of obtaining a value only in the limit is not restricted to sure winning: By adding the chance of slipping, the value of the game becomes 0.5. However, by the same argument as above, Player $\mathscr{R}$ cannot win with probability 0.5, but only with a probability $0.5 - \varepsilon$ for all $\varepsilon > 0$. △

For $s \in \mathsf{S}$, $\mathsf{a} \in \Gamma_{\mathscr{R}}(s)$ and $\mathsf{b} \in \Gamma_{\mathscr{S}}(s)$, the *set of potential successors* of s is denoted by $\mathsf{Post}(s, \mathsf{a}, \mathsf{b}) \coloneqq \mathsf{Supp}(\delta(s, \mathsf{a}, \mathsf{b}))$. We lift the notation to strategies $\rho \in \mathcal{R}$ and $\sigma \in \mathcal{S}$ by

$$\mathsf{Post}(s, \rho, \sigma) = \bigcup_{\mathsf{a} \in \mathsf{Supp}(\rho(s))} \bigcup_{\mathsf{b} \in \mathsf{Supp}(\sigma(s))} \mathsf{Post}(s, \mathsf{a}, \mathsf{b}).$$

We denote by $\mathsf{W}_{\mathscr{S}} \coloneqq \{s \in \mathsf{S} \mid \mathsf{V}_{\mathscr{R}}(s) = 0\}$ the *sure winning* region of player $\mathscr{S}$. It can be computed in at most $|\mathsf{S}|$ steps by iteration $\mathsf{W}^0_{\mathscr{S}} \coloneqq (\mathsf{S} \setminus \mathsf{T})$ and $\mathsf{W}^{k+1}_{\mathscr{S}} \coloneqq \{s \in \mathsf{S} \setminus \mathsf{T} \mid$

$\exists \mathsf{b} \in \Gamma_{\mathscr{S}}(s) : \forall \mathsf{a} \in \Gamma_{\mathscr{R}}(s) : \mathsf{Post}(s,\mathsf{a},\mathsf{b}) \subseteq \mathsf{W}^k_{\mathscr{S}}\}$ for all $k \in \mathbb{N}$ [15]. Consequently, we can assume without loss of generality that T and $\mathsf{W}_{\mathscr{S}}$ are both singletons and absorbing.

Example 3 (The Sets T, F, and $\mathsf{W}_{\mathscr{S}}$). In Fig. 1, player $\mathscr{R}$ wants to reach $\mathsf{T} = \{s_1\}$, while player $\mathscr{S}$ aims to stay in $\mathsf{F} = \{s_0, s_2\}$. Since s_2 is absorbing, $\mathsf{W}_{\mathscr{S}} = \{s_2\}$. △

Matrix Games: At each state of a CSG the players $\mathscr{R}$ and $\mathscr{S}$ play a two-player zero-sum *matrix game* [35], [43]. In general, a matrix game is a tuple $\mathsf{Z} := (N, A, u)$ [43] where, $N := \{1, \dots, n\}$ is a finite set of players, $A := \{\alpha_1, \dots, \alpha_m\}$ is a finite set of actions available to each player, and $u \colon A \to \mathbb{Q}$ is a utility function. In CSGs, the matrix game played at a specific state s can be represented by a matrix $\mathsf{Z}_{\mathsf{V}_{\mathscr{R}}}(s) \in \mathbb{Q}^{l \times m}$, where $\Gamma_{\mathscr{R}}(s) = \{a_1, \dots, a_l\}$ and $\Gamma_{\mathscr{S}}(s) = \{b_1, \dots, b_m\}$. The entries of the matrix correspond to the utility, i.e., the value attainable upon choosing a pair of actions $(\mathsf{a}_i, \mathsf{b}_j) \in A$. Thus, the i-th row and the j-th column is given by $\mathsf{Z}_{\mathsf{V}_{\mathscr{R}}}(s)(i,j) := \sum_{s' \in \mathsf{S}} \delta\big(s, a_i, b_j\big)\big(s'\big) \cdot \mathsf{V}_{\mathscr{R}}(s')$.

Example 4 (Matrix Game). Consider the CSG in Fig. 1. The matrix game played at state s_{hide} is given by the following matrix.

$$\mathsf{Z}_{\mathsf{V}_{\mathscr{R}}}(s_{\mathsf{hide}}) = \begin{array}{c} \begin{matrix} \text{throw} & \qquad \text{wait} \end{matrix} \\ \begin{pmatrix} 0 & \frac{1}{3} \cdot \mathsf{V}_{\mathscr{R}}(s_{\mathsf{hide}}) + \frac{1}{3} \\ 1 & \mathsf{V}_{\mathscr{R}}(s_{\mathsf{hide}}) \end{pmatrix} \end{array} \begin{matrix} \text{run} \\ \text{hide} \end{matrix} \tag{1}$$

Player $\mathscr{R}$ is the so called *row player* while Player $\mathscr{S}$ is the *column player*. △

In a matrix game, a player's *strategy* is a distribution over the available actions at a specific state. To distinguish between strategies of a CSG and strategies of a matrix game, we refer to strategies of a matrix game as *local strategies* and strategies of a CSG as *global* strategies. The set of all local strategies at a state s is denoted by $\mathcal{R}(s)$ or $\mathcal{S}(s)$ for player $\mathscr{R}$ or $\mathscr{S}$, respectively. The existence of optimal (local) strategies in a matrix game for both players is guaranteed by Nash's Theorem [36], [37]. The payoff that is attainable with an optimal local strategy is called *value* that we denote by $\mathsf{V}(\mathsf{Z}_{\mathsf{V}_{\mathscr{R}}})$ for a matrix game $\mathsf{Z}_{\mathsf{V}_{\mathscr{R}}}$. It can be calculated using linear programming (e.g., [28], see App. A-A).

End Components: A non-empty set of states $\mathsf{C} \subseteq \mathsf{S}$ is called an *end component* (EC) if (i) there exists a pair of strategies $(\rho, \sigma) \in \mathcal{R} \times \mathcal{S}$ such that for each $s \in \mathsf{C}$ it holds that $\mathsf{Post}(s, \rho, \sigma) \subseteq \mathsf{C}$; and (ii) for every pair of states $s, s' \in \mathsf{C}$ there is a play $s_0 s_1 \cdots$ such that $s_0 = s$ and $s_n = s'$ for some n, and for all $0 \leq i < n$, it holds $s_i \in \mathsf{C}$ and $s_{i+1} \in \mathsf{Post}(s_i, \rho, \sigma)$.

Intuitively, an EC is a set of states where a play can stay forever under some pair of strategies. In other words, the players can cooperate to keep the play inside the EC (this is the usual way to lift the definition of [14] from MDP to games). Thus, we can compute ECs in a CSG by computing ECs in the corresponding MDP with both players unified, i.e., every pair of actions is interpreted as an action in the MDP. Efficient algorithms for this exist [2], [12], [46]. An EC C is called *inclusion maximal* (short maximal) if there exists no EC C' such that $\mathsf{C} \subsetneq \mathsf{C}'$.

B. Value Iteration

Value iteration (VI, e.g. [11]) assigns an initial value estimate to each state and then iteratively updates it. In classical VI, which approximates the reachability value from below, the initial estimates are 1 for states in T and below the actual value otherwise, e.g. 0. Each iteration backpropagates the estimate by maximizing the expectation the value player $\mathscr{R}$ can ensure with respect to the previous estimate.

Formally, we capture estimates as valuations, where a *valuation* $\upsilon \colon \mathsf{S} \to [0,1]$ is a function mapping each state s to a real number representing the (approximate or true) value of the state. For two valuations υ, υ', we write $\upsilon \leq \upsilon'$ if $\upsilon(s) \leq \upsilon'(s)$ for every $s \in \mathsf{S}$.

To compute the expected value at a state s, the matrix game $\mathsf{Z}_\upsilon(s)$ has to be solved, meaning its value, $\mathsf{V}(\mathsf{Z}_\upsilon(s))$, has to be estimated. This computation is, especially in the turn-based setting, also referred to as *Bellman update*. Formally,

$$\mathsf{V}(\mathsf{Z}_\upsilon(s)) := \mathfrak{B}(\upsilon)(s) := \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\upsilon)(s, \rho, \sigma),$$

where $\mathfrak{B}(\upsilon)(s, \rho, \sigma) :=$

$$\sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathsf{S}} \rho(\mathsf{a}) \cdot \sigma(\mathsf{b}) \cdot \delta\big(s, \mathsf{a}, \mathsf{b}\big)\big(s'\big) \cdot \upsilon(s').$$

Convergent Under-approximation: We recall *VI from below* as in [9]: starting from the initial valuation L^0, we perform the Bellman update on every state to obtain a new valuation. We denote by L^k the valuation obtained in the k-th iteration. Formally:

$$\mathsf{L}^0(s) := \begin{cases} 1, \text{ if } s \in \mathsf{T}; \\ 0, \text{ else}, \end{cases} \qquad \mathsf{L}^{k+1}(s) := \mathfrak{B}(\mathsf{L}^k)(s). \tag{2}$$

Since T and $\mathsf{W}_{\mathscr{S}}$ are absorbing, for all $k \in \mathbb{N}$ we have $\mathsf{L}^k(s) = 1$ for all $s \in \mathsf{T}$, and $\mathsf{L}^k(s) = 0$ for all $s \in \mathsf{W}_{\mathscr{S}}$. The updated valuation, i.e. $\mathsf{L}^{k+1}(s)$, is computed by solving the corresponding matrix game.

Theorem 5 (VI converges from below [17, Thm. 1])**.** *VI from below converges to the value, i.e.* $\lim_{k \to \infty} \mathsf{L}^k = \mathsf{V}_{\mathscr{R}}$.

Bounded Value Iteration: While Thm. 5 proves that VI from below converges in the limit, this limit may not be reached in finitely many steps and can be irrational [17]. Thus, we do not know when to stop the algorithm to guarantee certain precision of the approximation, as there is no practical bound how close any valuation L^k is to the actual value. We merely have the worst-case bound of [25]: Running for a number of iterations that is doubly-exponential in the number of states allows to conclude that the lower bound is ε-close to the value.

To obtain a practical stopping criterion, we use the approach of Bounded Value Iteration (BVI), shown in Alg. 1. In addition to the lower bound L, it maintains an upper bound on the value

Algorithm 1 Bounded value iteration procedure for CSGs.

Algorithm BVI(CSG G, threshold $\varepsilon > 0$)
  $\mathsf{W}_{\mathscr{S}} \leftarrow \{s \in \mathsf{S} \mid \mathsf{V}_{\mathscr{R}}(s) = 0\}$ ▷ Winning region for $\mathscr{S}$
  $\mathsf{L}^0, \mathsf{U}^0$ initialized by Eq. (1) and (2), respectively
  $\mathsf{MEC} \leftarrow \mathsf{FIND\_MECs}(\mathsf{G})$ ▷ Find all MECs in the game
  $k \leftarrow 0$
  repeat
    for $s \in \mathsf{S}$ **do** ▷ Standard Bellman update of both bound
      $\mathsf{L}^{k+1}(s) \leftarrow \mathfrak{B}(\mathsf{L}^k)(s)$
      $\mathsf{U}^{k+1}(s) \leftarrow \mathfrak{B}(\mathsf{U}^k)(s)$
    for $\mathsf{C} \in \mathsf{MEC}$ **do**
      $\mathsf{U}^{k+1} \leftarrow \mathsf{DEFLATE}\ (\mathsf{G}, \mathsf{U}^{k+1}, \mathsf{C})$
    $k \leftarrow k + 1$
  until $\mathsf{U}^{k+1} - \mathsf{L}^{k+1} \leq \varepsilon$

U that is meant to converge to the value from above. Naïvely, this upper bound is defined as follows:

$$\mathsf{U}^0(s) \coloneqq \begin{cases} 0, \text{ if } s \in \mathsf{W}_{\mathscr{S}}; \\ 1, \text{ else,} \end{cases} \qquad \mathsf{U}^{k+1}(s) \coloneqq \mathfrak{B}(\mathsf{U}^k)(s). \quad (3)$$

Given a precision $\varepsilon > 0$, the algorithm terminates once the under- and the over-approximations are ε-close, i.e., when both approximations are at most ε-away from the actual value. However, applying Bellman updates does not suffice for the over-approximation to converge in the presence of ECs, as the following example shows:

Example 6 (Non-convergent Over-approximations). Consider the CSG `Hide-Run-or-Slip` in Fig. 1. To compute $\mathsf{U}^{k+1}(s_{\mathsf{hide}})$ according to Eq. (3), in each iteration we solve the matrix game $\mathsf{Z}_{\mathsf{U}^k}(s_{\mathsf{hide}})$ is given by Eq. (1) where the unknown $\mathsf{V}_{\mathscr{R}}$ is replaced by U^k, i.e.:

$$\mathsf{Z}_{\mathsf{U}^k}(s_{\mathsf{hide}}) = \begin{array}{c} \begin{matrix} \text{throw} & \text{wait} \end{matrix} \\ \begin{pmatrix} 0 & \frac{1}{3} \cdot \mathsf{U}^k(s_{\mathsf{hide}}) + \frac{1}{3} \\ 1 & \mathsf{U}^k(s_{\mathsf{hide}}) \end{pmatrix} \end{array} \begin{matrix} \text{run} \\ \text{hide} \end{matrix} \quad (4)$$

Table I shows the updates of the lower and upper bounds, $\mathsf{L}^k(s_{\mathsf{hide}})$ and $\mathsf{U}^k(s_{\mathsf{hide}})$, respectively. While the lower bound converges to 0.5, the upper bound stays at 1. This is because for player $\mathscr{R}$, action hide always "promises" a valuation of 1 in the next step, as the lower row of the matrix yields 1 for all player $\mathscr{S}$ strategies. △

III. Convergent Over-approximation: Overview

Here, we describe the structure of our solution. Ex. 6 shows that the naïve definition of the over-approximation need not converge to the true value. In particular, in the presence of ECs, Bellman updates do not have a unique fixpoint. Thus, our goal is to define a function DEFLATE (usage highlighted in Alg. 1, definition in Alg. 3) that, intuitively, decreases the "bloated" upper bounds inside each EC to a realistic value substantiated by a value promised *outside* of this EC. Formally, we ensure that Alg. 1 produces a monotonically decreasing sequence of valuations (i) over-approximating the reachability value and (ii) converging to it in the limit. This idea has been successfully applied for TSGs [18], [29].

TABLE I: BVI for the game `Hide-Run-or-Slip` (Fig. 1), where the over-approximations do not converge.

k	0	1	2	$\cdots$	∞
$\mathsf{L}^k(s_{\mathsf{hide}})$	0.0	0.25	0.36	$\cdots$	0.5
$\mathsf{U}^k(s_{\mathsf{hide}})$	1.0	1.0	1.0	$\cdots$	1.0

Remark 7 (Inflating for Safety). Since over-approximations need not converge for $\mathsf{V}_{\mathscr{R}}$, dually under-approximations need not converge for $\mathsf{V}_{\mathscr{S}}$ (as is the case in TSGs, see [29]). Thus, to directly solve a safety game, one needs an inflating operation dual to deflating. As described when introducing the objectives, we take the conceptually easier route of reducing everything to maximizing reachability objectives.

We proceed in two steps. First, in Sec. IV we prove that indeed ECs are the source of non-convergence, in particular what we call *Bloated End Components*. Intuitively, these are ECs where at each state both players prefer local strategies which all successor states belong to the EC. Second, in Sec. V, we define the DEFLATE algorithm, which essentially ensures that we focus on player $\mathscr{R}$ strategies that do not make the game stuck in a EC but rather progress towards the target. To this end, we lift the notion of *best exit* from TSGs [18, Definition 3] to CSGs.

IV. The Core of the Problem: Characterizing Bloated End Components

A locally optimal strategy of Player $\mathscr{R}$ does not coincide with an optimal global strategy of Player $\mathscr{R}$ because the latter must eventually leave ECs, while the former is under the illusion that staying is optimal. Thus, in this section, we want to find properties that local strategies (of both players) must fulfill in order to leave an EC in a way that is globally optimal. Thus, since this section mainly concerns *local* strategies, we use the word strategy to speak about local strategies and explicitly make clear when we talk about global ones.

Remark 8. Throughout the technical sections and the appendix, we always fix a CSG $\mathsf{G} \coloneqq (\mathsf{S}, \mathcal{A}, \Gamma_{\mathscr{R}}, \Gamma_{\mathscr{S}}, \delta, \mathsf{s}_0, \mathsf{T})$.

A. Convergence without ECs

As a first step, we prove that ECs are the only source of non-convergence, and without them, the naïve BVI using only Bellman updates converges.

Theorem 9 (Convergence without ECs — Proof in App. C-A)**.** *Let* G *be a CSG where all ECs are trivial, i.e. for every EC* C *we have* $\mathsf{C} \subseteq \mathsf{W}_{\mathscr{S}} \cup \mathsf{T}$*. Then, the over-approximation using only Eq.* (3) *converges, i.e.* $\lim_{k\to\infty} \mathsf{U}^k = \mathsf{V}_{\mathscr{R}}$.

Proof sketch. This proof is an extension of the proof of [18, Theorem 1] for turn-based games to the concurrent setting. The underlying idea is the same, and can be briefly summarized as follows: We assume towards a contradiction that $\lim_{k\to\infty} \mathsf{U}^k =:$

TABLE II: BVI for the CSG in Ex. 10, where the over-approximations converge.

k	0	1	2	3	$\cdots$	∞
$\mathsf{L}^k(s_{\mathsf{hide}})$	0.0	$\frac{1}{3}$	$\frac{4}{9}$	0.4815	$\cdots$	0.5
$\mathsf{U}^k(s_{\mathsf{hide}})$	1.0	$\frac{2}{3}$	$\frac{5}{9}$	0.5185	$\cdots$	0.5

$\mathsf{U}^\star \neq \mathsf{V}_{\mathscr{R}}$, and find a set $\mathcal{X}$ that maximizes the difference between upper bound and value. We show that every pair of strategies leaving the set $\mathcal{X}$ decreases the difference $\mathsf{U}^\star - \mathsf{V}_{\mathscr{R}}$. However, $\mathsf{V}_{\mathscr{R}}$ and $\mathsf{U}^\star$ are fixpoints of the Bellman update, from [17, Theorem 1] and Lem. 48, respectively. Consequently, optimal strategies need to remain in the set. However, in the absence of ECs, optimal strategies have to leave the set, which yields a contradiction and proves that $\mathsf{U}^\star = \mathsf{V}_{\mathscr{R}}$.

The key difference to the proof of [18, Theorem 1] is that we cannot argue about actions anymore, but have to consider mixed strategies. This significantly complicates notation. Additionally, and more importantly, the former proof crucially relied on the fact that for a state of Player $\mathscr{R}$, we know that its valuation is at least as large as that of any action, and dually for a state of Player $\mathscr{S}$, its valuation is at most as large as that of any action. In the concurrent setting, this is not true. The optimal strategies need not be maximizing nor minimizing the valuation and, moreover, they can be maximizing for one valuation and minimizing for another. Thus, we found a more general, and in fact simpler, way of proving that "no state in $\mathcal{X}$ can depend on the outside" [18, Statement 5] and deriving the contradiction. □

Interestingly, not all ECs cause non-convergence of Eq. (3) as the following example illustrates.

Example 10 (Unproblematic EC). We modify the CSG `Hide-Run-or-Slip` (Fig. 1) such that the matrix game played at s_{hide} is $\mathsf{Z}'_{\mathsf{U}^k}(s_{\mathsf{hide}})$ below.

$$\mathsf{Z}'_{\mathsf{U}^k}(s_{\mathsf{hide}}) = \begin{array}{c} \begin{matrix} \text{throw} & \text{wait} \end{matrix} \\ \begin{pmatrix} 1 & \frac{1}{3}\cdot \mathsf{U}^k(s_{\mathsf{hide}}) + \frac{1}{3} \\ 0 & \mathsf{U}^k(s_{\mathsf{hide}}) \end{pmatrix} \end{array} \begin{matrix} \text{run} \\ \text{hide} \end{matrix}$$

The difference is that $\mathsf{Z}'_{\mathsf{U}^k}(s_{\mathsf{hide}})(\mathsf{run}, \mathsf{throw}) = 1$ and $\mathsf{Z}'_{\mathsf{U}^k}(s_{\mathsf{hide}})(\mathsf{hide}, \mathsf{throw}) = 0$, switching the values as compared to the original CSG (see Eq. (4)). Here, both bounds converge to 0.5 despite the presence of the EC $\{s_{\mathsf{hide}}\}$, as shown in Table II. △

B. Towards Characterizing Bloated End Components

Intuition: In Ex. 10, the best strategy of Player $\mathscr{R}$ leaves the EC almost surely against all counter-strategies of Player $\mathscr{S}$, and hence BVI converges. In contrast, in Ex. 6, the best strategy of Player $\mathscr{R}$ is one where Player $\mathscr{S}$ has a counter-strategy that forces the play to stay inside the EC; this causes non-convergence. Generalizing these ideas, we see that a problem occurs if Player $\mathscr{R}$ has a strategy that is locally optimal but non-leaving, i.e. Player $\mathscr{S}$ has a counter-strategy that keeps the play inside an EC.

Outline: We formalize these ideas in the following definitions: First, Def. 12 formalizes optimal (local) strategies using *weakly dominant strategies* in matrix games, extending the standard definition (e.g. [35]) to sets of strategies. This extension is not straightforward, and there are several technical intricacies that we comment on. Next, Def. 15 captures leaving and staying strategies. We differentiate strategies that are leaving (irrespective of the opponent's strategy), staying (irrespective of the opponent's strategy), and non-leaving (where there exists an opponent's strategy that leads to staying) with respect to a given set of states. Based on this, we formally describe hazardous strategies in Def. 16, which are (locally) optimal strategies of Player $\mathscr{R}$ that are non-leaving; additionally, to be problematic for convergence, they are better than all leaving strategies. Using these, we can precisely characterize the Bloated End Components (BECs, Def. 18) that cause non-convergence.

Remark 11 (Additional Challenges Compared to TSGs). The core problem is the same as in TSGs: Player $\mathscr{R}$ is under the illusion that staying inside an EC yields a better valuation than leaving. However, in TSGs, the definitions of optimality and leaving are straightforward, since every state belongs to a single player and pure strategies are optimal; the definitions of hazardous and trapping strategies are not even necessary. In contrast, the definitions in CSGs are technically involved, as we have to take into account the interaction of the players and the possibility of optimal mixed strategies. In particular, [19] defined a straightforward extension of leaving based only on actions, not strategies. This is incorrect, as we demonstrate in App. D-B.

Definition 12 (Dominating Sets of Strategies)**.** Let υ be a valuation, $s \in \mathsf{S}$ a state, $\mathcal{R}_1, \mathcal{R}_2, \mathcal{R}' \subseteq \mathcal{R}(s)$ and $\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}' \subseteq \mathcal{S}(s)$ sets of local strategies. We now define two notions of domination for sets of strategies, namely *weak domination* and being *not worse*. Both of these depend on the player.

Definition for Player $\mathscr{R}$: We write $\mathcal{R}_2 \prec_{\upsilon,\mathcal{S}'} \mathcal{R}_1$ to denote that $\mathcal{R}_1$ *weakly dominates* $\mathcal{R}_2$ under the set of counter-strategies $\mathcal{S}'$ with respect to υ. Formally, $\exists \rho_1 \in \mathcal{R}_1 . \forall \rho_2 \in \mathcal{R}_2$:

(i) $\inf_{\sigma\in\mathcal{S}'} \mathfrak{B}(\upsilon)(s,\rho_2,\sigma) \leq \inf_{\sigma\in\mathcal{S}'} \mathfrak{B}(\upsilon)(s,\rho_1,\sigma)$, and
(ii) $\exists \sigma' \in \mathcal{S}'$ such that $\mathfrak{B}(\upsilon)(s,\rho_2,\sigma') < \mathfrak{B}(\upsilon)(s,\rho_1,\sigma')$.

If only Condition (i) is satisfied, we write $\mathcal{R}_2 \preceq_{\upsilon,\mathcal{S}'} \mathcal{R}_1$ to denote that the set $\mathcal{R}_1$ is *not worse* than $\mathcal{R}_2$ under $\mathcal{S}'$ with respect to υ.

Definition for Player $\mathscr{S}$: Dually, we write $\mathcal{S}_2 \prec_{\upsilon,\mathcal{R}'} \mathcal{S}_1$ to denote that set $\mathcal{S}_1$ *weakly dominates* $\mathcal{S}_2$ under $\mathcal{R}'$ with respect to υ. Formally, $\exists \sigma_1 \in \mathcal{S}_1 . \forall \sigma_2 \in \mathcal{S}_2$:

(i) $\sup_{\rho\in\mathcal{R}'} \mathfrak{B}(\upsilon)(s,\rho,\sigma_2) \geq \sup_{\rho\in\mathcal{R}'} \mathfrak{B}(\upsilon)(s,\rho,\sigma_1)$, and
(ii) $\exists \rho' \in \mathcal{R}'$ such that $\mathfrak{B}(\upsilon)(s,\rho',\sigma_2) > \mathfrak{B}(\upsilon)(s,\rho',\sigma_1)$.

If only Condition (i) is satisfied, we write $\mathcal{S}_2 \preceq_{\upsilon,\mathcal{R}'} \mathcal{S}_1$ to denote that the set $\mathcal{S}_1$ is *not worse* than $\mathcal{S}_2$ under $\mathcal{R}'$ with respect to υ.

Example 13 (Dominating Sets of Strategies). Consider the matrix game defined in Eq. (4) and the valuation $\mathsf{U}^k(s_{\mathsf{hide}}) =$

$\mathsf{U}^k(s_{\mathsf{home}}) = 1$ and $\mathsf{U}^k(s_{\mathsf{wet}}) = 0$. Here, for Player $\mathscr{R}$, the pure strategy $\{\mathsf{hide} \mapsto 1\}$ dominates the pure strategy $\{\mathsf{run} \mapsto 1\}$:

$$\{\mathsf{run} \mapsto 1\} \prec_{\mathsf{U}^k, \mathcal{S}(s_{\mathsf{hide}})} \{\mathsf{hide} \mapsto 1\}.$$

This is because when Player $\mathscr{S}$ throws the ball, hiding yields 1 while running yields 0. Note that this is in fact independent of the valuation, so it also holds for $\mathsf{V}_{\mathscr{R}}$.

Let $\mathsf{RunPositive} := \{(\mathsf{run} \mapsto \varepsilon, \mathsf{hide} \mapsto 1 - \varepsilon) \mid \varepsilon > 0\}$ be the set of all strategies that put positive probability on running. We have

$$\mathsf{RunPositive} \prec_{\mathsf{U}^k, \mathcal{S}(s_{\mathsf{hide}})} \{\mathsf{hide} \mapsto 1\}.$$

Again, this is true even when using $\mathsf{V}_{\mathscr{R}}$ as valuation. Note that we have this weak dominance even though the supremum over the set yields the optimum valuation, namely $\sup_{\rho \in \mathsf{RunPositive}} \inf_{\sigma \in \mathcal{S}(s_{\mathsf{hide}})} \mathfrak{B}(\mathsf{U}^k)(s, \rho, \sigma) = 1$. This exemplifies the strictness of our notion of dominance. It is crucial that our notion of domination can distinguish these sets of strategies: The set RunPostive contains all strategies that leave the EC. However, none of them is optimal (even though the supremum over all of them is), which is exactly the reason why VI chooses the staying strategy $\{\mathsf{hide} \mapsto 1\}$ for updating the valuation, and thus is stuck. △

We remark on several technicalities of Def. 12:

- "Weak" dominance: The term "weak" might be misleading. We choose to use the word for consistency with [35, Def. 4.12]. There, *weak* domination concerns Condition (ii), only requiring that there *exists* a counter-strategy where the inequality is strict; strict domination requires Condition (ii) *for all* counter-strategies. One might be tempted to use weak domination to denote what we call "not worse", i.e. only require that there is a strategy in the first set that has optimal valuation at least as good as all in the other set; or to think it only refers to the numerical comparators, e.g. $\geq$ and $>$ (as is sometimes the case when only comparing single strategies).
- Set-related challenges: The definition is challenging since we cannot speak about actions, but have to consider sets of — possible mixed — strategies. The exact quantification of the strategies is relevant. Further, it depends not only on the two sets we are comparing, but also on the counter-strategies of the opponent. Thus, we provide the definition explicitly for both players, to avoid confusion that could arise from just saying that they are analogous.
- All-quantification instead of optima: The definition uses all-quantification instead of optima. Concretely, weak dominance for Player $\mathscr{R}$ uses $\forall \rho_2 \in \mathcal{R}_2$ instead of writing $\sup_{\rho_2 \in \mathcal{R}_2}$. The latter definition cannot sufficiently distinguish sets of strategies, since the supremum of a set need not be contained in it, as we exemplified in Ex. 13. This fact is extremely important, as in the proof of Thm. 21, we pick the maximum from a set of strategies, and the existence of this maximum is guaranteed only because of the correct definition of dominance.
- Locally optimal strategies are not worse than any other: Formally, this claim is that for all locally optimal strategy $\rho \in \mathcal{R}(s)$ with respect to υ, we have $\mathcal{R}(s) \preceq_{\upsilon, \mathcal{S}(s)} \{\rho \mapsto 1\}$ (and dually for Player $\mathscr{S}$). This is immediate from Def. 12, since a locally optimal strategy maximizes $\inf_{\sigma \in \mathcal{S}'} \mathfrak{B}(\upsilon)(s, \rho, \sigma)$, and thus satisfies Condition (i) when compared to all other strategies. We will use this fact throughout the paper.
- Notation: When the valuation is clear from the context, we omit it for the sake of readability. Further, if we say that a strategy ρ_1 weakly dominates another strategy ρ_2 with respect to a counter-strategy σ, then we mean that $\{\rho_2\} \prec_{\{\sigma\}} \{\rho_1\}$.

We prove a lemma about the relation of weak domination and not being worse that is useful and instructive. The proof in App. C-B works by straightforward unfolding of definitions and rewriting.

Lemma 14 (Negating Weak Domination — Proof in App. C-B)**.** *Let υ be a valuation, $s \in \mathsf{S}$ a state, $\mathcal{R}_1, \mathcal{R}_2, \mathcal{R}' \subseteq \mathcal{R}(s)$ and $\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}' \subseteq \mathcal{S}(s)$ sets of local strategies.*

If for some sets of strategies we do not *have $\mathcal{R}_2 \prec_{\upsilon, \mathcal{S}'} \mathcal{R}_1$, then we have $\mathcal{R}_1 \preceq_{\upsilon, \mathcal{S}'} \mathcal{R}_2$. Analogously, not $\mathcal{S}_2 \prec_{\upsilon, \mathcal{R}'} \mathcal{S}_1$ implies $\mathcal{S}_1 \preceq_{\upsilon, \mathcal{R}'} \mathcal{S}_2$.*

To complete our intuitive understanding of the definition of domination, we point out a connection to the standard definition of weak domination: "A rational player does not use a dominated strategy." [35, Asm. 4.13] If a strategy is not dominated, by Lem. 14 it is not worse than any other strategy. This is exactly what we argued above: Locally optimal strategies are not worse than any other. We often use this fact throughout the paper.

Next, we formally define *leaving and staying strategies*. Given a set of states, a leaving strategy ensures that the set of successor states contains states outside the given set of states for all given counter-strategies of the opponent player. A strategy is staying if all successor states belong to the given set of states for all given counter-strategies of the opponent player. Note that a strategy can be neither leaving nor staying, if the set is exited for some, but not all counter-strategies of the opponent.

Definition 15 (Leaving and Staying Strategies)**.** Consider a set of states $\mathcal{X} \subseteq \mathsf{S}$ and a state $s \in \mathcal{X}$. Let $\mathcal{R}' \subseteq \mathcal{R}(s)$ and $\mathcal{S}' \subseteq \mathcal{S}(s)$ be sets of strategies of Player $\mathscr{R}$ and $\mathscr{S}$, respectively. The set of (local) *leaving strategies* for Player $\mathscr{R}$ with respect to $\mathcal{S}'$, is given by

$$\mathcal{R}_L(\mathcal{S}', \mathcal{X}, s) := \{\rho \in \mathcal{R}(s) \mid \forall \sigma \in \mathcal{S}'.(s, \rho, \sigma) \text{ leaves } \mathcal{X}\},$$

and for Player $\mathscr{S}$ with respect to $\mathcal{R}'$ by

$$\mathcal{S}_L(\mathcal{R}', \mathcal{X}, s) := \{\sigma \in \mathcal{S}(s) \mid \forall \rho \in \mathcal{R}'.(s, \rho, \sigma) \text{ leaves } \mathcal{X}\}.$$

A strategy that is not leaving is called *non-leaving*. The set of all non-leaving Player $\mathscr{R}$ strategies is denoted by $\mathcal{R}_{\overline{L}}(\mathcal{S}', \mathcal{X}, s)$ (or $\mathcal{S}_{\overline{L}}$ for Player $\mathscr{S}$).

In contrast, the set of *staying* strategies at a state $s \in \mathcal{X}$ for Player $\mathscr{R}$ with respect to $\mathcal{S}'$, is given by

$$\mathcal{R}_S(\mathcal{S}', \mathcal{X}, s) \coloneqq \{\rho \in \mathcal{R}(s) \mid \forall \sigma \in \mathcal{S}'.(s, \rho, \sigma) \text{ staysIn } \mathcal{X}\},$$

and for Player $\mathscr{S}$ with respect to $\mathcal{R}'$ by

$$\mathcal{S}_S(\mathcal{R}', \mathcal{X}, s) \coloneqq \{\sigma \in \mathcal{S}(s) \mid \forall \rho \in \mathcal{R}'.(s, \rho, \sigma) \text{ staysIn } \mathcal{X}\}.$$

Notation: If we consider leaving (or staying) strategies with respect to all counter-strategies, then we omit the set of counter-strategies, i.e. instead of $\mathcal{R}_L(\mathcal{S}(s), \mathcal{X}, s)$ (or $\mathcal{R}_S(\mathcal{S}(s), \mathcal{X}, s)$) we write $\mathcal{R}_L(\mathcal{X}, s)$ (or $\mathcal{R}_S(\mathcal{X}, s)$). We use the same shorthand notion for leaving (staying) strategies of Player $\mathscr{S}$.

We often speak about a leaving/staying pair of local strategies, so we provide the following shorthand notations: For a tuple $(s, \rho, \sigma) \in \mathsf{S} \times \mathcal{R} \times \mathcal{S}$, we say that (s, ρ, σ) leaves $\mathcal{X}$ if and only if $\mathsf{Post}(s, \rho, \sigma) \cap (\mathsf{S} \setminus \mathcal{X}) \neq \emptyset$. Analogously, we say that (s, ρ, σ) staysIn $\mathcal{X}$ if and only if $\mathsf{Post}(s, \rho, \sigma) \cap (\mathsf{S} \setminus \mathcal{X}) = \emptyset$ (or, equivalently, $\mathsf{Post}(s, \rho, \sigma) \subseteq \mathcal{X}$).

Intuition of Hazardous Strategies: Using the definitions of dominance and leaving or staying, we can now classify strategies of Player $\mathscr{R}$ that can lead to non-convergence. Intuitively, a hazardous strategy is one that Player $\mathscr{R}$ chooses, even though it can be staying for some counter-strategies. Thus, such a strategy (i) is non-leaving (i.e. there exist counter-strategies that make it staying), and (ii) it is not worse than any other strategy so that Player $\mathscr{R}$ may choose it. Moreover, to be problematic for convergence, (iii) the strategy weakly dominates all leaving strategies, i.e. leaving strategies are not chosen for the update.

Definition 16 (Hazardous Strategy)**.** Let $\mathcal{X} \subseteq \mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}})$ be as set of states, υ a valuation, and $s \in \mathcal{X}$. A strategy $\rho \in \mathcal{R}(s)$ is called hazardous with respect to υ if it satisfies:

(i) $\rho \in \mathcal{R}_{\overline{L}}(\mathcal{X}, s)$,
(ii) $\mathcal{R}(s) \setminus \{\rho\} \preceq_{\mathcal{S}(s)} \{\rho\}$, and
(iii) $\mathcal{R}_L(\mathcal{X}, s) \prec_{\mathcal{S}(s)} \{\rho\}$.

$\mathsf{Hazard}_\upsilon(\mathcal{X}, s)$ denotes the set of all hazardous strategies at state s with respect to a set of states $\mathcal{X}$ and a valuation υ.

We mention a corner case: In a state where Player $\mathscr{R}$ possesses no leaving strategies, all optimal strategies are hazardous (note in particular that Condition (iii) is trivially satisfied, since the dominated set of strategies $\mathcal{R}_L(\mathcal{X}, s)$ is empty).

Example 17 (Hazardous strategy). Consider again the matrix game defined in Eq. (4) and the initial valuation $\mathsf{U}^0(s_{\mathsf{hide}}) = \mathsf{U}^0(s_{\mathsf{home}}) = 1$ and $\mathsf{U}^0(s_{\mathsf{wet}}) = 0$. The strategy $\rho' \coloneqq \{\mathsf{hide} \mapsto 1\}$ is hazardous because: (i) It is non-leaving. (ii) It is an optimal strategy, i.e. it is not worse than any other strategy. (iii) It weakly dominates the set of all leaving strategies, see Ex. 13. △

Definition 18 (Bloated End Component (BEC))**.** An EC $\mathcal{X} \subseteq \mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}})$ is called *bloated end component* (BEC) with respect to a valuation υ if for all $s \in \mathcal{X}$ it holds that $\mathsf{Hazard}_\upsilon(\mathcal{X}, s) \neq \emptyset$.

Example 19 (Bloated End Component). Consider the CSG `Hide-Run-or-Slip` from Fig. 1. As discussed in Ex. 17, there exists a hazardous strategy in state s_{hide}. Moreover, $\{s_{\mathsf{hide}}\}$ is an EC, since under the pair of strategies that plays hide and wait, the play stays in it. Consequently, $\{s_{\mathsf{hide}}\}$ is a BEC and therefore VI does not converge in this state, see Ex. 6. △

We provide a lemma that captures the intuition of what it means to (not) be a BEC, and that is also useful in several proofs:

Lemma 20 (Negating Bloated — Proof in App. C-B)**.** *If an EC $\mathcal{X} \subseteq \mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}})$ is not bloated for a valuation υ, then there exists a state $s \in \mathcal{X}$ that has a locally optimal strategy that is leaving, formally $\exists \rho \in \mathcal{R}_L(\mathcal{X}, s).\mathcal{R}(s) \preceq_{\upsilon, \mathcal{S}(s)} \{\rho\}$.*

C. Convergence in the Absence of BECs

Now we can prove that BECs indeed are the reason that VI does not converge for over-approximations.

Theorem 21 (Non-convergence implies BECs — Proof in App. C-B)**.** *Let $\mathsf{U}^\star \coloneqq \lim_{k \to \infty} \mathsf{U}^k$ be the limit of the naïve upper bound iteration (Eq. (3)) on the CSG G. If VI from above does not converge to the value in the limit, i.e. $\mathsf{U}^\star > \mathsf{V}_{\mathscr{R}}$, then the CSG G contains a BEC in $\mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}})$ with respect to $\mathsf{U}^\star$.*

Proof sketch. This proof builds on the proof of Thm. 9. There, we constructed a set $\mathcal{X}$ maximizing the difference between $\mathsf{U}^\star$ and $\mathsf{V}_{\mathscr{R}}$ and showed that if there is a pair of optimal strategies leaving $\mathcal{X}$, then we can derive a contradiction: The upper bound decreases, which contradicts the fact that it is a fixpoint. In the context of the other proof, that allowed us to show that without ECs, VI converges, because without ECs it is impossible to have a set of states where all optimal strategies stay in that set.

In the presence of ECs, states can indeed have a positive difference between $\mathsf{U}^\star$ and $\mathsf{V}_{\mathscr{R}}$, see e.g. Ex. 6. Our goal is to prove that at least one of these ECs is bloated. Thus, we assume for contradiction that no EC is bloated under $\mathsf{U}^\star$. Then, by Lem. 20, there is an optimal leaving strategy for Player $\mathscr{R}$. Using that, we can repeat the argument from Thm. 9, showing that in this case $\mathsf{U}^\star$ would decrease. Again, this is a contradiction because it is a fixpoint of applying Bellman updates (Lem. 48). Thus, the initial assumption that no EC is bloated is false, and we can conclude that there exists a BEC. □

Remark 22 (Relation to [18]). Def. 18 of BEC is more general than the definition of BEC for TSGs in [18, Definition 4]. The differences are that in [18], an EC is only called bloated if it is bloated with respect to $\mathsf{V}_{\mathscr{R}}$, whereas we extended that definition to speak about a concrete valuation, similar to [29, Def. 3]. Further, the definition for TSGs speaks about the best exit value, which is the optimum among the available actions; in contrast, in CSGs the definition of best exit is technically involved and dependent on hazardous strategies, see Def. 27. Thus, our definition of BEC does not analyze the exit values, but instead uses a fundamental analysis of the strategies.

Key Contribution: The key novelty of this section is the correct definition of BEC that captures when VI from above does not converge. We highlight that this definition contains many technical intricacies: Lifting the notion of an exiting action [18, Section 2.2] from TSGs to CSGs requires considering sets of local strategies that leave against all opponent-strategies (Def. 15), and considering the additional complication that strategies can be neither leaving nor staying. Further, the exact definition of dominance is very important, as the supremum over all leaving strategies can be a staying strategy, see Ex. 13 and the related discussion in the item "All-quantification instead of optima" after the example.

V. Resolving Bloated End Components

A. Solution in TSGs

We have identified BECs as the cause of non-convergence. Our method for fixing the over-approximation is again based on the ideas for TSGs. We explain the intuition of their solution: Firstly, staying actions yield the valuation that is bloated; thus, we need an additional update of the over-approximation that depends only on leaving actions. The valuation to which we reduce the over-approximation is the *best exit* from the EC, which in TSGs simply is the leaving action attaining the highest value over all states of Player $\mathscr{R}$ [18, Definition 3]. Secondly, not all states in an EC need to have the same value, since Player $\mathscr{S}$ can prevent Player $\mathscr{R}$ from reaching the state that attains the best exit valuation. Hence, an EC is decomposed into parts that share the same value, called *simple* ECs (SECs) in [18, Definition 5]. Repeatedly finding these SECs and *deflating* their valuation by setting it to the best exit from the SEC suffices for convergence in TSGs.

When generalizing these ideas to CSGs, we encounter the following problems: Firstly, the definition of best exit is more involved, since staying and leaving depends not only on actions, but on strategies. Additionally, the supremum over all leaving strategies can be a non-leaving strategy, see Ex. 13. (This is also the reason why globally optimal strategies need not exist in CSGs, see Ex. 2). This was the fundamental reason why the solution proposed in [19] did not work, as it was based on actions. Secondly, we need to decompose the EC into parts. For this, we use a recursive approach, removing states that have been successfully deflated and checking whether there are further problematic states in the remainder of the EC.

Outline: In Sec. V-B, we develop a strategy-based definition of best exit (Def. 28), which relies on identifying the *trapping strategies* (Def. 24) that Player $\mathscr{S}$ uses to keep the play inside the BEC; and the *deflating strategies* (Def. 25), the best response of Player $\mathscr{R}$, namely the leaving strategies that should be played with arbitrarily small probability ε. In Sec. V-C, we provide the FIND_MBEC algorithm that finds all maximal BECs that are present in a given MEC. A maximal BEC is a set of states $\mathcal{X} \subseteq \mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}})$ that is a BEC and there exists no another set of states $\mathcal{X}' \subseteq \mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}})$ that is a BEC and $\mathcal{X} \subsetneq \mathcal{X}'$. Finally, Sec. V-D provides the full deflating procedure for CSGs in Alg. 3. In particular, it uses a recursive call to decompose a given MEC and deflate all relevant parts of it.

B. Defining the Best Exit

The key problem of a BEC is that in all states, all leaving strategies of Player $\mathscr{R}$ are dominated by hazardous strategies. Player $\mathscr{S}$ can play a trapping strategy and thereby make the Bellman update self-dependent. If the current valuation is too high, this prevents convergence. Intuitively, the "pressure" inside the BEC is too high, and we want to "deflate" it, by adjusting it to the pressure, i.e. valuation, outside of the BEC. Since non-trivial BECs neither contain target states nor belong to the winning region of Player $\mathscr{S}$, there has to exist a state in a BEC where the supremum over leaving and staying strategies is equal. To "equalize the pressure" between the BEC and the rest of the states, we need to estimate the pressure outside the BEC. To do so, we estimate the valuation attainable upon leaving the BEC at every state of the BEC, called exit value of the state. The best exit value is the maximum of all exit values. Reducing the upper bounds of the states inside the BEC to the best exit value, in case the best exit is smaller than the current valuation, "decreases the pressure". However, since the valuations of the exiting strategies can depend on the valuations of the states that belong to the BEC, this procedure may still only converge in the limit, already in TSGs. We provide an illustrative example:

Example 23 (Deflating BECs). Consider again the CSG `Hide-Run-or-Slip` (Fig. 1). Under the initial valuation (see Eq. (3)) the matrix game played at s_{hide} is given by

$$\mathsf{Z}_{\mathsf{U}^0}(s_{\mathsf{hide}}) = \begin{array}{c} \begin{matrix} \text{throw} & \text{wait} \end{matrix} \\ \begin{pmatrix} 0 & \frac{2}{3} \\ 1 & 1 \end{pmatrix} \end{array} \begin{matrix} \text{run} \\ \text{hide} \end{matrix}$$

Due to the hazardous strategy $\{\mathsf{hide} \mapsto 1\}$, the Bellman update cannot improve the initial upper bound of s_{hide}, but remains at 1. However, by the same argument as in Ex. 2, Player $\mathscr{R}$ can use the strategy of running with a probability $\varepsilon > 0$ that is arbitrarily small. This yields a value arbitrarily close to $\frac{2}{3}$. Consequently, we can deflate, i.e. decrease the upper bound of s_{hide} to $\frac{2}{3}$ which is the valuation attainable upon leaving the BEC at s_{hide}. After the Bellman update, the matrix game played at s_{hide} is given by

$$\mathsf{Z}_{\mathsf{U}^1}(s_{\mathsf{hide}}) = \begin{array}{c} \begin{matrix} \text{throw} & \text{wait} \end{matrix} \\ \begin{pmatrix} 0 & \frac{5}{9} \\ 1 & \frac{2}{3} \end{pmatrix} \end{array} \begin{matrix} \text{run} \\ \text{hide} \end{matrix}$$

The strategy $\{\mathsf{hide} \mapsto 1\}$ is still hazardous, but we can deflate the upper bound of s_{hide} to $\frac{5}{9}$. By repeating these steps, the upper bound converges to $\frac{1}{2}$ in the limit, and only in the limit, similar to the lower bound in Table I. △

How can we find the best exit value in general?: In Ex. 23, we used an argument about playing a leaving action with vanishingly small probability in order to figure out which entry in the matrix we choose for deflating. We provide an

alternative intuition: Player $\mathscr{R}$ plays a hazardous strategy most of the time. The best response of Player $\mathscr{S}$ is to play a weakly dominant strategy that stays in the EC, trapping the play. Thus, we can ignore the other strategies of Player $\mathscr{S}$ and consider only the columns of the matrix that correspond to *trapping strategies*. Now, we need to select a leaving strategy of Player $\mathscr{R}$ which then is played with vanishingly low probability. Thus, we restrict the matrix further to use only rows corresponding to leaving strategies. In the example, we end up with the top right entry. We now formalize how we can construct this sub-matrix, called the exiting sub-game.

Definition 24 (Trapping Strategy)**.** Let $\mathcal{X} \subseteq \mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}})$ be as set of states, υ a valuation, and $s \in \mathcal{X}$. A strategy $\sigma \in \mathcal{S}(s)$ is called *trapping strategy* if two conditions are satisfied:

(i) $\sigma \in \arg\min_{\sigma' \in \mathcal{S}(s)} \max_{\rho \in \mathcal{R}(s)} \mathfrak{B}(\upsilon)(s, \rho, \sigma')$, and
(ii) $\forall \rho \in \mathsf{Hazard}_\upsilon(\mathcal{X}, s) : (s, \rho, \sigma)\ \mathsf{staysIn}\ \mathcal{X}$.

$\mathsf{Trap}_\upsilon(\mathcal{X}, s)$ denotes the set of all trapping strategies at state s with respect to a set of states $\mathcal{X}$ and a valuation υ.

Definition 25 (Deflating Strategies)**.** Let $\mathcal{X} \subseteq \mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}})$ be a set of states. A Player $\mathscr{R}$ strategy $\rho \in \mathcal{R}(s)$ is called *deflating* if two conditions are satisfied:

(i) $\exists \sigma \in \mathsf{Trap}_\upsilon(\mathcal{X}, s)$ such that $(s, \rho, \sigma)\ \mathsf{leaves}\ \mathcal{X}$, and
(ii) $\mathsf{Supp}(\rho) \cap \bigcup_{\rho' \in \mathsf{Hazard}_\upsilon(\mathcal{X}, s)} \mathsf{Supp}(\rho') = \emptyset$.

$\mathsf{Defl}_\upsilon(\mathcal{X}, s)$ denotes the set of all deflating strategies at state s with respect to a set of states $\mathcal{X}$ and a valuation υ.

Definition 26 (Exiting sub-game)**.** Let $\mathcal{X} \subseteq \mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}})$ be a set of states, $s \in \mathcal{X}$ a state, and υ a valuation. Further, let $\mathsf{Z}_\upsilon(s)$ be the matrix game played at state $s \in \mathcal{X}$. If $\mathsf{Trap}_\upsilon(\mathcal{X}, s) \neq \emptyset$ then, the *exiting sub-game* played at state s, denoted by $\mathsf{Z}^{\mathsf{exit}}_\upsilon(s)$, is the matrix game where Player $\mathscr{R}$ has the actions in $\bigcup_{\rho \in \mathsf{Defl}_\upsilon(\mathcal{X}, s)} \mathsf{Supp}(\rho)$ and Player $\mathscr{S}$ has the actions in $\bigcup_{\sigma \in \mathsf{Trap}_\upsilon(\mathcal{X}, s)} \mathsf{Supp}(\sigma)$. The value of the exiting sub-game is given by $\mathsf{V}(\mathsf{Z}^{\mathsf{exit}}_\upsilon(s)) := \max(0, \sup_{\rho \in \mathsf{Defl}_\upsilon(\mathcal{X}, s)} \inf_{\sigma \in \mathsf{Trap}_\upsilon(\mathcal{X}, s)} \mathfrak{B}(\upsilon)(s, \rho, \sigma))$.

We explain how this exiting sub-game and its value are well-defined: The set of deflating strategies can be empty, namely in a state which has no leaving strategies. In this case, the value of the exiting sub-game is the supremum over an empty set, i.e. the smallest possible value, commonly minus infinity and 0 in our case. We highlight this possibility by explicitly taking the maximum of 0 and the value of the exiting subgame when we compute it. If the set of deflating strategies is non-empty, then the set of trapping strategies necessarily is non-empty, too, since the Item (i) of Def. 25 requires existence of a trapping strategy.

Definition 27 (Exit value)**.** Let $\mathcal{X} \subseteq \mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}})$ be a set of states, $s \in \mathcal{X}$ a state, and υ an over-approximation. Then, the *exit value* from $\mathcal{X}$ attainable at state s is given by

$$\mathsf{exitVal}_\upsilon(\mathcal{X}, s) := \begin{cases} \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\upsilon)(s, \rho, \sigma), & \text{if } \mathsf{Hazard}_\upsilon(\mathcal{X}, s) = \emptyset \\ & \quad \vee\, \mathsf{Trap}_\upsilon(\mathcal{X}, s) = \emptyset; \\ \mathsf{V}(\mathsf{Z}^{\mathsf{exit}}_\upsilon(s)), & \text{else.} \end{cases}$$

For BECs that consists of more than one state, the exit values attainable at different states within the BEC may differ. Consequently, to ensure that deflating does not reduce the value of any of the states in the BEC below its actual value we estimate the exit values at each state of the BEC, and finally select the maximal exit value, i.e. the best exit value, for deflating.

Definition 28 (Best Exit)**.** Let $\mathcal{X} \subseteq (\mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}}))$ be a set of states and υ a valuation. The *best exit value* with respect to a set $\mathcal{X}$ and a valuation υ is given by

$$\mathsf{bestExitVal}_\upsilon(\mathcal{X}) := \max_{s \in \mathcal{X}} \mathsf{exitVal}_\upsilon(\mathcal{X}, s).$$

The *best exit*, denoted by $\mathsf{bestExit}_\upsilon(\mathcal{X})$, is a state obtaining $\mathsf{bestExitVal}_\upsilon(\mathcal{X})$, and the set of all best exits is denoted by $\mathsf{bestExits}_\upsilon(\mathcal{X})$.

Lemma 29 ($\mathsf{bestExitVal}$ is sound)**.** *Let $\mathcal{X} \subseteq \mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}})$ be an EC, and $\mathsf{U} \in [0,1]^{|\mathsf{S}|}$ be a valuation with $\mathsf{U} \geq \mathsf{V}_{\mathscr{R}}$. Then, for all states $s \in \mathcal{X}$, we have $\mathsf{bestExitVal}_\mathsf{U}(\mathcal{X}) \geq \mathsf{V}_{\mathscr{R}}(s)$.*

Proof. This proof relies on the technical Lem. 49 stating that
(i) $\mathcal{X}' := \{s \in \mathcal{X} \mid \mathsf{V}_{\mathscr{R}}(s) \leq \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)\} \neq \emptyset$, and
(ii) $\max_{s \in \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s) \geq \max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$.
Let $\mathcal{X}' \subseteq \mathcal{X}$ be a set of states satisfying Condition (i). Further, choose $e \in \arg\max_{t \in \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(t)$ as one of the exits from $\mathcal{X}'$. By Item (ii) of Lem. 49 , we have that for all $s \in \mathcal{X}$: $\mathsf{V}_{\mathscr{R}}(s) \leq \mathsf{V}_{\mathscr{R}}(e)$. It remains to show $\mathsf{bestExitVal}_\mathsf{U}(\mathcal{X}) \geq \mathsf{V}_{\mathscr{R}}(e)$. We conclude as follows:

$$\begin{aligned} \mathsf{V}_{\mathscr{R}}(e) &\leq \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, e) && \text{(Since } e \in \mathcal{X}') \\ &= \mathsf{bestExitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}) && \text{(By the choice of } e) \\ &\leq \mathsf{exitVal}_\mathsf{U}(\mathcal{X}, e) && \text{(By Lem. 50)} \\ &\leq \mathsf{bestExitVal}_\mathsf{U}(\mathcal{X}). && \text{(By Def. 28)} \end{aligned}$$

We want to highlight that the statement of Lem. 50 is non-trivial and the proof is technically involved. □

C. Finding Maximal BECs

Since a BEC might contain other BECs, we want to find *maximal* BECs. A BEC $\mathcal{X}$ is maximal if there exists no BEC $\mathcal{X}'$ such that $\mathcal{X} \subsetneq \mathcal{X}'$. The existence of maximal BECs is proven in App. C-C.

Given a CSG G, a MEC C and the current upper bound estimate U, Alg. 2 finds all maximal BECs within C as follows: The set B contains all states for which hazardous strategies exist with respect to the set C. In case B is non-empty, it might consist of multiple disjoint MECs. Therefore, the algorithm calls the FIND_MECs procedure on B and returns all MECs

Algorithm 2 Algorithm for finding maximal BECs.

1: **Algorithm** FIND_MBECs(CSG G, MEC C, upper bound estimate U)
2: $B \coloneqq \{s \in \mathsf{C} \mid \mathsf{Hazard}_{\mathsf{U}}(\mathsf{C}, s) \neq \emptyset\}$
3: **if** $B \neq \emptyset$ **then** ▷ C contains BECs
4: **return** FIND_MECs(G, B)
5: **else**
6: **return** $\emptyset$ ▷ There exists no BEC in C

Algorithm 3 Algorithm for deflating BECs.

1: **Algorithm** DEFLATE(CSG G, upper bound estimate U, MEC C)
2: **for** $\mathcal{X} \in$ FIND_MBECs$(\mathsf{G}, \mathsf{C}, \mathsf{U})$ **do**
3: $\mathsf{u} \leftarrow \mathsf{bestExitVal}_{\mathsf{U}}(\mathcal{X})$
4: **for** $s \in \mathcal{X}$ **do**
5: $\mathsf{U}(s) \leftarrow \min(\mathsf{U}(s), \mathsf{u})$
6: **for** $\mathcal{E} \in$ FIND_MECs$(\mathcal{X} \setminus \mathsf{bestExits}_{\mathsf{U}}(\mathcal{X}))$ **do**
7: $\mathsf{U} \leftarrow$ DEFLATE$(\mathsf{G}, \mathsf{U}, \mathcal{E})$
▷ Recursively deflate sub-BECs
8: **return** U

that are in B. Otherwise, B is empty, which means that C does not contain any BECs, and the empty set is returned.

Lemma 30 (FIND_MBECs is correct— Proof in App. C-C)**.** *For a CSG, a MEC* C *and a valid upper bound* U*, it holds that* $\mathcal{X} \in$ FIND_MBECs$(\mathsf{G}, \mathsf{C}, \mathsf{U})$ *if and only if* $\mathcal{X}$ *is a BEC in* C *and there exists no* $T \subseteq \mathsf{C}$ *that is a BEC and* $\mathcal{X} \subsetneq T$*.*

Proof sketch. To prove the "⇒" direction, we assume towards a contradiction that $\mathcal{X}$ is a maximal EC but not a BEC. Then, all Player $\mathscr{R}$ strategies must violate at least one of the conditions posed by Def. 16. We consider each condition separately and derive a contradiction to the assumption that $\mathcal{X} \in$ FIND_MBECs$(\mathsf{G}, \mathsf{C}, \mathsf{U})$.

To prove the opposite direction, i.e., "⇐", we assume towards a contradiction that $\mathcal{X}$ is a maximal BEC but $\mathcal{X} \notin$ FIND_MBECs$(\mathsf{G}, \mathsf{C}, \mathsf{U})$. Here, we again make a case distinction. In the first case, some state $s \in \mathcal{X}$ was removed by the algorithm because at least one player has an optimal strategy that can leave $\mathcal{X}$. However, this is a contradiction to the assumption that $\mathcal{X}$ is a BEC. In the second case, we assume towards a contradiction that $\mathcal{X} \notin$ FIND_MECs(G, B) (where B is defined as in FIND_MBECs). This is then a contradiction to the assumption that $\mathcal{X}$ is a maximal EC. □

D. Convergent Bounded Value Iteration for CSGs and the Recursive Structure of ECs

Finally, Alg. 3 describes our main goal: the deflating procedure for BECs, to be plugged into Alg. 1. The algorithm takes a CSG G, the current upper bound estimate U, and a MEC C as input. First, the algorithm searches for all maximal BECs that might be contained in the MEC. The current upper bound estimate is returned if no BEC exists in the MEC. Otherwise, at least one BEC exists and must be deflated. If multiple BECs are found, each maximal BEC is deflated separately (see the for-loop in Line 2).

To deflate a maximal BEC $\mathcal{X}$, first, the *best* exit value $\mathsf{bestExitVal}_{\mathsf{U}}(\mathcal{X})$ is estimated. Next, each state of the BEC is considered and if the best exit value is smaller than the current upper bound estimate at that state, then it is reduced to the best exit value (as nothing better can be reached).

However, notice that within a BEC $\mathcal{X}$ there might exist another BEC $\mathcal{X}'$. From $\mathcal{X}'$, Player $\mathscr{R}$ might not be able to get to the best exit of $\mathcal{X}$ (the globally best one in $\mathcal{X}$) but only to a worse one, locally optimal for $\mathcal{X}'$. Hence, for such a sub-BEC $\mathcal{X}'$, more aggressive deflating to $\mathsf{bestExitVal}_{\mathsf{U}}(\mathcal{X}')$ is due. In other words, after deflating to $\mathsf{bestExitVal}_{\mathsf{U}}(\mathcal{X})$ we have handled all states where $\mathscr{R}$ can ensure reaching this best exit, and we can ignore these states for the moment; we can also ignore this best exit and have a fresh look at which states are *now* in a BEC and can reach the *second best* exit, i.e., the best exit in this remainder. Subsequently, we continue with the third best option etc.

Consequently, on Lines 6-7, DEFLATE is called *recursively* on all MECs that are contained in $\mathcal{X} \setminus \mathsf{bestExits}_{\mathsf{U}}(\mathcal{X})$, i.e. after removing all best exits. The procedure is repeated independently for each maximal BEC contained in C. A full example showing how DEFLATE works on a more complex BEC is included in App. B-B.

Remark 31 (Structure of ECs). This elucidates the hierarchical structure of ECs and their corresponding best exits. The recursive call of DEFLATE exposes the partial order over the ECs, their sub-ECs, and "internally" transient states (those not within sub-ECs after removing the best exit since they are bound to the just removed exit or another sub-EC that is a BEC with a lower value). This hierarchy captures (i) the ordering of exits by their values and (ii) the "independence" of exits with possibly different values when visiting one from another cannot be ensured.

Our goal for the remainder of this work is to show that Alg. 1 with deflation is correct and converges, i.e. that complementing Bellman updates $\mathfrak{B}$ with deflation results in a sequence of upper bounds U that converges to the value from above. For the sake of simplicity, we denote by $\mathfrak{D} : [0,1]^{|\mathsf{S}|} \to [0,1]^{|\mathsf{S}|}$ the operator that performs DEFLATE on a given valuation for all MECs in the CSG (which is reasonable since Alg. 1 performs DEFLATE on all MECs (in an arbitrary but fixed ordering)).

Remark 32 (Valid upper bonds). In the following, whenever we quantify an upper bound (a.k.a. over-approximation), we require it to be *valid*; meaning that it was obtained by iteratively applying deflating and the Bellman update on the initial over-approximation U^0 from Eq. (3), i.e. $\mathsf{U} = (\mathfrak{D} \circ \mathfrak{B})^k(\mathsf{U}^0)$ for some $k \in \mathbb{N}$. The reason for this restriction is that for convergence, we require $\mathfrak{D}$ to be order-preserving. While $\mathfrak{D}$ is order-preserving on valid upper bounds (see Lem. 36), in general it is not monotonic for arbitrary $\mathsf{U} \in \mathbb{R}^{|\mathsf{S}|}$. We illustrate this in App. B-C.

Definition 33 (Valid upper bound)**.** An upper bound $\mathsf{U} \in [0,1]^{|S|}$ is called *valid* if there exists $k \in \mathbb{N}$ such that $\mathsf{U} = (\mathfrak{D} \circ \mathfrak{B})^k(\mathsf{U}^0)$, or if $\mathsf{U} = \mathsf{V}_{\mathscr{R}}$.

We proceed as follows: After proving fundamental properties of both operators $\mathfrak{B}$ and $\mathfrak{D}$ in Lem. 34, we use these properties to show that valid upper bounds are indeed upper bounds, i.e. they are always greater or equal than the value in Lem. 35. With correctness established, Lem. 36 shows that on valid upper bounds, $(\mathfrak{D} \circ \mathfrak{B})$ is order-preserving, which is a necessary ingredient for convergence. Finally, Thm. 37 concludes by proving soundness and completeness of the full algorithm.

Lemma 34 (Properties of $\mathfrak{D}$ and $\mathfrak{B}$)**.** *Let $\upsilon \in [0,1]^{|S|}$ be a valuation. If $\upsilon \geq \mathsf{V}_{\mathscr{R}}$, then:*

(i) $\mathfrak{B}(\upsilon) \leq \upsilon$ *and* $\mathfrak{D}(\upsilon) \leq \upsilon$.
(ii) $\mathfrak{B}(\upsilon) \geq \mathsf{V}_{\mathscr{R}}$ *and* $\mathfrak{D}(\upsilon) \geq \mathsf{V}_{\mathscr{R}}$.

Proof. For the Bellman operator, both properties follow from the fact that the value $\mathsf{V}_{\mathscr{R}}$ is the least fixpoint of the Bellman operator, see [17, Thm. 1]. Thus, given a valuation greater than the value, it cannot increase, and it cannot become smaller than the value.

For $\mathfrak{D}$, observe that deflation only updates the valuation in Line 5 of Alg. 3 when setting the valuation of a state to $\min(\mathsf{U}(s), \mathsf{bestExitVal}_{\mathsf{U}}(\mathcal{X}))$ for some EC $\mathcal{X}$. Item (i) holds because of taking the minimum with the current valuation, so it can only decrease. For Item (ii), we show in Lem. 29 that for every EC $\mathcal{X}$ and state $s \in \mathcal{X}$, we have that $\mathsf{bestExitVal}_{\mathsf{U}}(\mathcal{X}) \geq \mathsf{V}_{\mathscr{R}}(s)$. This proves our goal, as the only update of DEFLATE keeps the valuation greater than $\mathsf{V}_{\mathscr{R}}$ in every state. The proof of Lem. 29 is technically involved, having to unfold many definitions in order to show the following intuitive fact: No state can have a larger value than that of some exit from the EC it is contained in. □

Lemma 35 (Soundness of valid upper bounds)**.** *For all $k \in \mathbb{N}$ it holds that $(\mathfrak{D} \circ \mathfrak{B})^k(\mathsf{U}^0) \geq \mathsf{V}_{\mathscr{R}}$.*

Proof. We proceed by induction over k.

Base case: $k = 0$, thus $(\mathfrak{D} \circ \mathfrak{B})^0(\mathsf{U}^0) = \mathsf{U}^0 \geq \mathsf{V}_{\mathscr{R}}$.

Induction hypothesis: For all $k \geq 0$, we assume that $(\mathfrak{D} \circ \mathfrak{B})^k(\mathsf{U}^0) \geq \mathsf{V}_{\mathscr{R}}$.

Induction step: To show: $(\mathfrak{D} \circ \mathfrak{B})^{k+1}(\mathsf{U}^0) \geq \mathsf{V}_{\mathscr{R}}$. We know by induction hypothesis that $(\mathfrak{D} \circ \mathfrak{B})^k(\mathsf{U}^0) \geq \mathsf{V}_{\mathscr{R}}$. Applying Item (ii) of Lem. 34 for $\mathfrak{B}$, we obtain $\mathfrak{B}((\mathfrak{D} \circ \mathfrak{B})^k(\mathsf{U}^0)) \geq \mathsf{V}_{\mathscr{R}}$. From this and Item (ii) for $\mathfrak{D}$, we get $\mathfrak{D}(\mathfrak{B}((\mathfrak{D} \circ \mathfrak{B})^k(\mathsf{U}^0))) \geq \mathsf{V}_{\mathscr{R}}$, proving our goal.

□

Lemma 36 ($(\mathfrak{D} \circ \mathfrak{B})$ is order-preserving on valid upper bounds)**.** *Let υ_1, υ_2 be valid upper bounds with $\upsilon_1 \geq \upsilon_2$. It holds that $(\mathfrak{D} \circ \mathfrak{B})(\upsilon_1) \geq (\mathfrak{D} \circ \mathfrak{B})(\upsilon_2)$.*

Proof. We know by Lem. 35 that all valid upper bounds are greater or equal to the value (including the value itself). Thus, if $\upsilon_2 = \mathsf{V}_{\mathscr{R}}$, the statement holds, and if $\upsilon_1 = \mathsf{V}_{\mathscr{R}}$, then $\upsilon_2 = \mathsf{V}_{\mathscr{R}}$ as well, since $\upsilon_1 \geq \upsilon_2$. It remains to consider the case that both valuations come from repeated application of the deflation and Bellman operators, i.e. $\upsilon_1 = (\mathfrak{D} \circ \mathfrak{B})^i(\mathsf{U}^0)$ and $\upsilon_2 = (\mathfrak{D} \circ \mathfrak{B})^j(\mathsf{U}^0)$. We assume $\upsilon_1 \neq \upsilon_2$, since otherwise the claim trivially holds.

By Item (i) of Lem. 34, every application of the operators can only decrease the resulting value; the item remains applicable, since the resulting valuations are always greater than or equal to the value. Consequently, $i < j$, as $\upsilon_1 > \upsilon_2$. Using this and applying Item (i) of Lem. 34 again, we conclude by stating

$$(\mathfrak{D} \circ \mathfrak{B})(\upsilon_1) = (\mathfrak{D} \circ \mathfrak{B})^{i+1}(\mathsf{U}^0) \geq (\mathfrak{D} \circ \mathfrak{B})^{j+1}(\mathsf{U}^0) = (\mathfrak{D} \circ \mathfrak{B})(\upsilon_2).$$

□

Theorem 37 (Soundness and completeness - Proof in App. C-D)**.** *For CSGs Alg. 1, using Alg. 3 as* DEFLATE*, produces monotonic sequences* L *under- and* U *over-approximating* $\mathsf{V}_{\mathscr{R}}$*, and terminates for every $\varepsilon > 0$.*

Proof sketch. Soundness and convergence of lower bounds is classical [17, Thm. 1], and our algorithm does not modify the computation of under-approximations. The soundness of the upper bounds is immediate from Lem. 35, since all upper bounds computed by the algorithm are valid, and thus greater or equal than the value. Proving the convergence of the upper bounds is the main challenge. First, in Lem. 52 we provide the technical proof that the upper bound in Alg. 1 indeed converges to a fixpoint, using that the operators are order-preserving (Lem. 36) and arguments from lattice theory. Then, we use the same idea we have utilized in the proofs of Thms. 9 and 21: We assume for contradiction that there exists a state where the difference between the fixpoint of the upper bound in Alg. 1 and the true value is strictly greater than zero. The states with the largest such difference contain a BEC (by Thm. 21), that eventually will be found and deflated; since deflation depends on the outside of the BEC, this decreases the upper bound. This causes a contradiction to the fact that the upper bound has converged to a fixpoint. Consequently, there can be no state with a positive difference, and the upper bounds converge, too. □

VI. Conclusion and Future Work

We have introduced a convergent over-approximation for concurrent stochastic games with reachability and safety objectives, thus giving value iteration the first sound stopping criterion and turning it into an anytime algorithm. Since the games are concurrent and (ε-)optimal strategies may need to be randomized, we could not use the technique of simple end components of [18]. Instead, we identify bloated end components where the play can get stuck forever and recursively deflate the over-approximations of these states to the best possible value attainable upon leaving the end component. We leave an efficient implementation for future work as an extension of the standard model checker PRISM-GAMES [32].

REFERENCES

[1] Pranav Ashok, Krishnendu Chatterjee, Przemysław Daca, Jan Křetínský, and Tobias Meggendorfer. Value Iteration for Long-Run Average Reward in Markov Decision Processes. In Rupak Majumdar and Viktor Kunčak, editors, *Computer Aided Verification*. Springer International Publishing, 2017.

[2] Christel Baier and Joost-Pieter Katoen. *Principles of Model Checking*. MIT Press, April 2008.

[3] Christel Baier, Joachim Klein, Linda Leuschner, David Parker, and Sascha Wunderlich. Ensuring the Reliability of Your Model Checker: Interval Iteration for Markov Decision Processes. In Rupak Majumdar and Viktor Kunčak, editors, *Computer Aided Verification*. Springer International Publishing, 2017.

[4] Benjamin Bordais, Patricia Bouyer, and Stéphane Le Roux. Subgame Optimal Strategies in Finite Concurrent Games with Prefix-Independent Objectives. In Orna Kupferman and Pawel Sobocinski, editors, *Foundations of Software Science and Computation Structures*, pages 541–560. Springer Nature Switzerland, 2023.

[5] Tomáš Brázdil, Krishnendu Chatterjee, Martin Chmelik, Vojtěch Forejt, Jan Křetínskỳ, Marta Kwiatkowska, David Parker, and Mateusz Ujma. Verification of Markov decision processes using learning algorithms. In *International Symposium on Automated Technology for Verification and Analysis*, pages 98–114. Springer, 2014.

[6] Tomáš Brázdil, Krishnendu Chatterjee, Martin Chmelík, Vojtěch Forejt, Jan Křetínský, Marta Kwiatkowska, David Parker, and Mateusz Ujma. Verification of Markov Decision Processes Using Learning Algorithms. In Franck Cassez and Jean-François Raskin, editors, *Automated Technology for Verification and Analysis*. Springer International Publishing, 2014.

[7] Krishnendu Chatterjee, Luca de Alfaro, and Thomas A Henzinger. Termination criteria for solving concurrent safety and reachability games. In *Proceedings of the twentieth annual ACM-SIAM symposium on Discrete algorithms*, pages 197–206. SIAM, 2009.

[8] Krishnendu Chatterjee, Luca De Alfaro, and Thomas A. Henzinger. Qualitative concurrent parity games. *ACM Transactions on Computational Logic*, July 2011.

[9] Krishnendu Chatterjee, Luca de Alfaro, and Thomas A Henzinger. Strategy improvement for concurrent reachability and safety games. *arXiv preprint arXiv:1201.2834*, 2012.

[10] Krishnendu Chatterjee, Luca de Alfaro, and Thomas A Henzinger. Strategy improvement for concurrent reachability and turn-based stochastic safety games. *Journal of computer and system sciences*, 79(5):640–657, 2013.

[11] Krishnendu Chatterjee and Thomas A. Henzinger. Value Iteration. In Orna Grumberg and Helmut Veith, editors, *25 Years of Model Checking: History, Achievements, Perspectives*. Springer, 2008.

[12] Costas Courcoubetis and Mihalis Yannakakis. The complexity of probabilistic verification. *Journal of the ACM*, 42, July 1995.

[13] B. A. Davey and H. A. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, Cambridge, 2 edition, 2002.

[14] Luca De Alfaro. How to specify and verify the long-run average behaviour of probabilistic systems. In *Proceedings. Thirteenth Annual IEEE Symposium on Logic in Computer Science (Cat. No. 98CB36226)*, pages 454–465. IEEE, 1998.

[15] Luca de Alfaro and Thomas A Henzinger. Concurrent omega-regular games. In *Logic in Computer Science, 2000. Proceedings. 15th Annual IEEE Symposium on*, pages 141–154. IEEE, 2000.

[16] Luca De Alfaro, Thomas A Henzinger, and Orna Kupferman. Concurrent reachability games. *Theoretical Computer Science*, 2007.

[17] Luca de Alfaro and Rupak Majumdar. Quantitative solution of omega-regular games. *Journal of Computer and System Sciences*, 68(2):374 – 397, 2004. Special Issue on STOC 2001.

[18] Julia Eisentraut, Edon Kelmendi, Jan Křetínský, and Maximilian Weininger. Value iteration for simple stochastic games: Stopping criterion and learning algorithm. *Information and Computation*, 2022.

[19] Julia Eisentraut, Jan Křetínský, and Alexej Rotar. Stopping Criteria for Value and Strategy Iteration on Concurrent Stochastic Reachability Games, 2019.

[20] Kousha Etessami and Mihalis Yannakakis. Recursive Concurrent Stochastic Games. *Logical Methods in Computer Science*, Volume 4, Issue 4, November 2008.

[21] H. Everett. *RECURSIVE GAMES*, pages 47–78. Princeton University Press, 1957.

[22] Søren Kristoffer Stiil Frederiksen and Peter Bro Miltersen. Approximating the Value of a Concurrent Reachability Game in the Polynomial Time Hierarchy. Springer, 2013.

[23] Serge Haddad and Benjamin Monmege. Interval iteration algorithm for mdps and imdps. *Theoretical Computer Science*, 735:111–131, 2018.

[24] Serge Haddad and Benjamin Monmege. Interval Iteration Algorithm for MDPs and IMDPs. *Theoretical Computer Science*, 2018.

[25] Kristoffer Arnsfelt Hansen, Rasmus Ibsen-Jensen, and Peter Bro Miltersen. The Complexity of Solving Reachability Games Using Value and Strategy Iteration. *Theory of Computing Systems*, 55, 2014.

[26] Kristoffer Arnsfelt Hansen, Michal Koucky, Niels Lauritzen, Peter Bro Miltersen, and Elias P. Tsigaridas. Exact algorithms for solving stochastic games: Extended abstract. In *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing*, 2011.

[27] Arnd Hartmanns, Sebastian Junges, Tim Quatmann, and Maximilian Weininger. A Practitioner's Guide to MDP Model Checking Algorithms. In Sriram Sankaranarayanan and Natasha Sharygina, editors, *Tools and Algorithms for the Construction and Analysis of Systems*. Springer Nature Switzerland, 2023.

[28] Frederick S. Hillier and Gerald J. Lieberman. *Introduction to Operations Research*. McGraw-Hill Higher Education, 2010.

[29] Jan Křetínský, Tobias Meggendorfer, and Maximilian Weininger. Stopping Criteria for Value Iteration on Stochastic Games with Quantitative Objectives. In *2023 38th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, 2023.

[30] Jan Křetínský, Emanuel Ramneantu, Alexander Slivinskiy, and Maximilian Weininger. Comparison of algorithms for simple stochastic games. *Information and Computation*, 2022.

[31] P. R. Kumar and T. H. Shiau. Existence of Value and Randomized Strategies in Zero-Sum Discrete-Time Stochastic Dynamic Games. *SIAM Journal on Control and Optimization*, 19, 1981.

[32] Marta Kwiatkowska, Gethin Norman, David Parker, and Gabriel Santos. Automated verification of concurrent stochastic games. In *International Conference on Quantitative Evaluation of Systems*, pages 223–239. Springer, 2018.

[33] Marta Kwiatkowska, Gethin Norman, David Parker, and Gabriel Santos. PRISM-games 3.0: Stochastic Game Verification with Concurrency, Equilibria and Time. In Shuvendu K. Lahiri and Chao Wang, editors, *Computer Aided Verification*. Springer International Publishing, 2020.

[34] Donald A. Martin. The determinacy of blackwell games. 1998.

[35] Michael Maschler, Shmuel Zamir, and Eilon Solan. *Game Theory*. Cambridge University Press, June 2020.

[36] John Nash. Non-Cooperative Games. 1951.

[37] John F. Nash. Equilibrium points in n-person games. 1950.

[38] Miquel Oliu-Barton. New algorithms for solving zero-sum stochastic games. *Mathematics of Operations Research*, 46(1):255–267, 2021.

[39] T. Parthasarathy. Discounted, positive, and noncooperative stochastic games. *International Journal of Game Theory*, 2(1):25–37, Dec 1973.

[40] Grant Olney Passmore and Paul B. Jackson. Combined decision techniques for the existential theory of the reals. In *Calculemus/MKM*, volume 5625 of *Lecture Notes in Computer Science*, pages 122–137. Springer, 2009.

[41] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley, 2009.

[42] T. E. S. Raghavan and J. A. Filar. Algorithms for stochastic games — A survey. *Zeitschrift für Operations Research*, 35(6):437–472, 1991.

[43] G. H. R. Santos. *Automatic Verification and Strategy Synthesis for Zero-Sum and Equilibria Properties of Concurrent Stochastic Games*. PhD thesis, University of Oxford, 2020.

[44] Dana Scott. Outline of a Mathematical Theory of Computation. *Kiberneticheskij Sbornik. Novaya Seriya*, 14, January 1977.

[45] Stephen Simons. Minimax Theorems and Their Proofs. In Ding-Zhu Du and Panos M. Pardalos, editors, *Minimax and Applications*. Springer US, 1995.

[46] Anton Wijs, Joost-Pieter Katoen, and Dragan Bošnački. Efficient GPU algorithms for parallel decomposition of graphs into strongly connected and maximal end components. *Formal Methods in System Design*, 48(3), June 2016.

APPENDIX A
FURTHER DEFINITIONS AND CONCEPTS

A. Solving Matrix Games

Given a valuation υ for all states in a two-player CSG, we can update the valuation at state s by solving a Linear Program (LP). Let the matrix game played at s be given by a matrix $\mathsf{Z} \in \mathbb{Q}^{l \times m}$, where l (resp. m) is the number of actions available to player $\mathscr{R}$ (resp. $\mathscr{S}$) at the state s. Then the LP that yields the value is the following [43]: Maximize $\upsilon(s)$ subject to the constraints:

$$\begin{aligned} \upsilon(s) &\leq x_1 \cdot z_{1j} + \cdots + x_l \cdot z_{lj} \text{ for } 1 \leq j \leq m \\ x_i &\geq 0 \text{ for } 1 \leq i \leq l \\ 1 &= x_1 + \cdots + x_l \end{aligned}$$

where $z_{ij} = \mathsf{Z}(s)(i,j) = \sum_{s' \in \mathsf{S}} \delta\big(s, a_i, b_j\big)\big(s'\big) \cdot \upsilon(s')$, and x_i is the probability that player $\mathscr{R}$ will take action i. Thus, by solving the LP, we not only obtain the value but also the optimal local strategy for player $\mathscr{R}$.

B. Domination of Strategies

Definition 38 (Domination [35, Def. 4.12])**.** Given a valuation υ, a state $s \in \mathsf{S}$ and sets of strategies $\mathcal{R}(s)$ and $\mathcal{S}(s)$ available at the state s for player $\mathscr{R}$ and $\mathscr{S}$, respectively.

- A strategy $\rho \in \mathcal{R}(s)$ is *weakly dominated* if there exists another strategy $\rho' \in \mathcal{R}(s)$ satisfying the following two conditions:
  - $\inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\upsilon)(s, \rho, \sigma) \leq \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\upsilon)(s, \rho', \sigma)$, and
  - $\exists \sigma' \in \mathcal{S}(s)$ such that $\mathfrak{B}(\upsilon)(s, \rho, \sigma') < \mathfrak{B}(\upsilon)(s, \rho', \sigma')$.
- Dually, a strategy $\sigma \in \mathcal{S}(s)$ is *weakly dominated* if there exists another strategy $\sigma' \in \mathcal{S}(s)$ satisfying the following two conditions:
  - $\sup_{\rho \in \mathcal{R}(s)} \mathfrak{B}(\upsilon)(s, \rho, \sigma) \geq \sup_{\rho \in \mathcal{R}(s)} \mathfrak{B}(\upsilon)(s, \rho, \sigma')$, and
  - $\exists \rho' \in \mathcal{R}(s)$ such that $\mathfrak{B}(\upsilon)(s, \rho', \sigma) > \mathfrak{B}(\upsilon)(s, \rho', \sigma')$.

APPENDIX B
FURTHER EXAMPLES

A. Best Exit

Example 39 (Deflating BECs). Consider the CSG depicted in Fig. 2. At each state both players can choose among two actions. The game contains one MEC $\mathsf{C} \coloneqq \{s_0, s_1, s_2\}$. The matrix games played at the states s_0, s_1 and s_2 with respect to an upper bound U^k (where $k \in \mathbb{N}$) are defined as follows.

$$\mathsf{Z}_{\mathsf{U}^k}(s_0) \coloneqq \begin{pmatrix} \overset{\mathsf{d}_1}{\mathsf{U}^k(s_0)} & \overset{\mathsf{d}_2}{\mathsf{U}^k(s_1)} \\ \frac{1}{2}(\mathsf{U}^k(s_0) + \alpha) & \alpha \end{pmatrix} \begin{matrix} \mathsf{a}_1 \\ \mathsf{a}_2 \end{matrix}, \qquad \mathsf{Z}_{\mathsf{U}^k}(s_1) = \begin{pmatrix} \overset{\mathsf{e}_1}{\gamma} & \overset{\mathsf{e}_2}{\beta} \\ \frac{1}{2}(\mathsf{U}^k(s_1) + \mathsf{U}^k(s_2)) & \mathsf{U}^k(s_0) \end{pmatrix} \begin{matrix} \mathsf{b}_1 \\ \mathsf{b}_2 \end{matrix},$$

$$\mathsf{Z}_{\mathsf{U}^k}(s_2) = \begin{pmatrix} \overset{\mathsf{f}_1}{\mathsf{U}^k(s_0)} & \overset{\mathsf{f}_2}{\frac{1}{2}(\mathsf{U}^k(s_0) + \mathsf{U}^k(s_1))} \\ \gamma & \mathsf{U}^k(s_2) \end{pmatrix} \begin{matrix} \mathsf{c}_1 \\ \mathsf{c}_2 \end{matrix}.$$

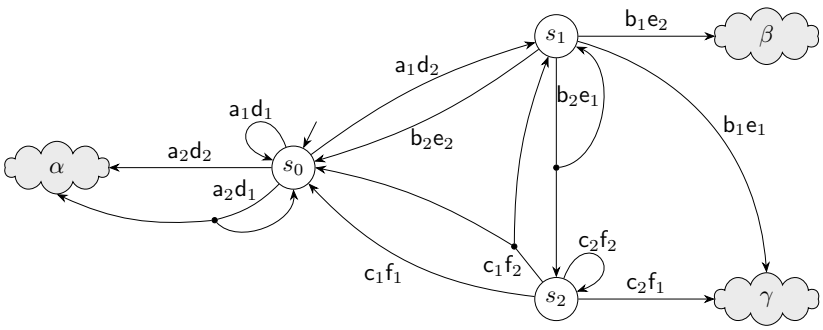

Fig. 2: CSG with non-single-state EC. The clouds represent the irrelevant parts of the game.

TABLE III: Full example of the BVI algorithm for the CSG depicted in Fig. 2. i is the i-th iteration of the main loop in Alg. 1, j is the j-th iteration of the while-loop in Alg. 3. C is the set of states among which a BEC $\mathcal{X}$ is searched. The best exit from $\mathcal{X}$ is underlined.

i	j	$\mathsf{U}^i(s_0)$	$\mathsf{U}^i(s_1)$	$\mathsf{U}^i(s_2)$	C	$\mathcal{X}$
0	0	1	1	1	$\{s_0, s_1, s_2\}$	$\{s_0, s_1, \underline{s_2}\}$
	1	0.9	0.9	0.9	$\{s_0, s_1\}$	$\{s_0, \underline{s_1}\}$
	2	0.7	0.7	0.9	$\{s_0\}$	$\{\underline{s_0}\}$
	3	0.2	0.7	0.9	$\emptyset$	$\emptyset$
1	0	0.2	0.7	0.9	$\{s_0, s_1, s_2\}$	$\{\underline{s_0}\}$
	1	0.2	0.7	0.9	$\{s_1, s_2\}$	$\{\underline{s_2}\}$
	2	0.2	0.7	0.45	$\{s_1\}$	$\emptyset$

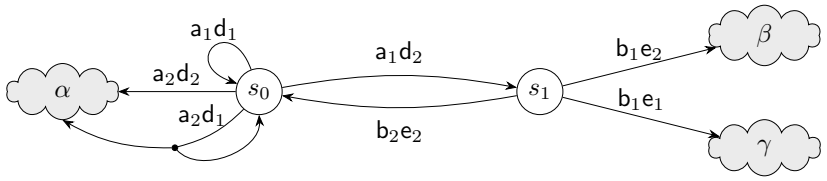

Fig. 3: The resulting game after removing s_2 from the game illustrated in Fig. 2.

For $k = 0$, $\alpha = 0.2, \beta = 0.7$ and $\gamma = 0.9$ and the initialization $\mathsf{U}^0(s_0) = \mathsf{U}^0(s_1) = \mathsf{U}^0(s_2) = 1$, the matrix games look as follows.

$$\mathsf{Z}_{\mathsf{U}^0}(s_0) = \begin{pmatrix} & \mathsf{d}_1 & \mathsf{d}_2 \\ \mathsf{a}_1 & 1 & 1 \\ \mathsf{a}_2 & 0.6 & 0.2 \end{pmatrix}, \qquad \mathsf{Z}_{\mathsf{U}^0}(s_1) = \begin{pmatrix} & \mathsf{e}_1 & \mathsf{e}_2 \\ \mathsf{b}_1 & 0.9 & 0.7 \\ \mathsf{b}_2 & 1 & 1 \end{pmatrix}, \qquad \mathsf{Z}_{\mathsf{U}^0}(s_2) = \begin{pmatrix} & \mathsf{f}_1 & \mathsf{f}_2 \\ \mathsf{c}_1 & 1 & 1 \\ \mathsf{c}_2 & 0.9 & 1 \end{pmatrix}.$$

Since at each state, there exist hazardous and trapping strategies, the three states form a BEC. To estimate the values of each exit, we need to solve three sub-matrix games played at each state of the BEC. The best exit value is then the maximum of the three solutions. The three linear programs that solve the three sub-matrix games are the following.

$\max \mathsf{U}^1(s_0)$ s.t.	$\max \mathsf{U}^1(s_1)$ s.t.	$\max \mathsf{U}^1(s_2)$ s.t.
$\mathsf{U}^1(s_0) \leq 0.6 \cdot x_2$	$\mathsf{U}^1(s_1) \leq 0.9 \cdot x_1$	$\mathsf{U}^1(s_2) \leq 0.9 \cdot x_2$
$\mathsf{U}^1(s_0) \leq 0.2 \cdot x_2$	$\mathsf{U}^1(s_1) \leq 0.7 \cdot x_1$	$\mathsf{U}^1(s_2) \leq 1 \cdot x_2$
$x_2 = 1$	$x_1 = 1$	$x_2 = 1$

Here x_1 and x_2 are the probabilities that player $\mathscr{R}$ chooses the first or second action available at the corresponding state. The best exit is $\max\{0.2, 0.7, 0.9\} = 0.9$, therefore, the upper bound of all the three states can be safely reduced to 0.9. $\triangle$

B. BVI Algorithm - Full Example

Example 40 (BVI). Consider the CSG depicted in Fig. 2, where $\alpha = 0.2, \beta = 0.7$, and $\gamma = 0.9$. The matrix games played at each state are given by the matrices $\mathsf{Z}_{\mathsf{U}^k}(s_0), \mathsf{Z}_{\mathsf{U}^k}(s_1)$ and $\mathsf{Z}_{\mathsf{U}^k}(s_2)$ defined as in Example 39. We choose $\varepsilon = 0.001$.

Table III summarizes the steps of the algorithm. Initially it holds that $\mathsf{U}^0(s_0) = \mathsf{U}^0(s_1) = \mathsf{U}^0(s_2) = 1$. Since $\mathsf{C} = \{s_0, s_1, s_2\}$ is a MEC, it will be found by Alg. 3. Within C the BEC $\mathcal{X} = \mathsf{C}$ is found. The best exit value from $\mathcal{X}$ is calculated, i.e. the three sub-matrix games are solved and the best exit value needed for deflating the BEC is the maximum among the solutions:

$$\mathsf{Z}^{\mathsf{exit}}_{\mathsf{U}^0}(s_0) := \begin{pmatrix} \mathsf{d}_1 & \mathsf{d}_2 \\ 0.6 & 0.2 \end{pmatrix}_{\mathsf{a}_2} \qquad \mathsf{Z}^{\mathsf{exit}}_{\mathsf{U}^0}(s_1) = \begin{pmatrix} \mathsf{e}_1 & \mathsf{e}_2 \\ 0.9 & 0.7 \end{pmatrix}_{\mathsf{b}_1} \qquad \mathsf{Z}^{\mathsf{exit}}_{\mathsf{U}^0}(s_2) = \begin{pmatrix} \mathsf{f}_1 & \mathsf{f}_2 \\ 0.9 & 1 \end{pmatrix}_{\mathsf{c}_1}$$

The best exit from $\mathcal{X}$ is s_2 (underlined in Table III) and the value of the exit is 0.9, so in the next step the over-approximations of those three states are deflated to 0.9 and the best exit is removed from C. The resulting sub-game is depicted in Fig. 3. For

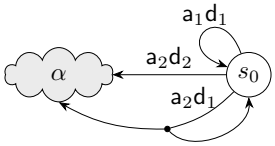

Fig. 4: The resulting game after removing s_1 from the game illustrated in Fig. 3.

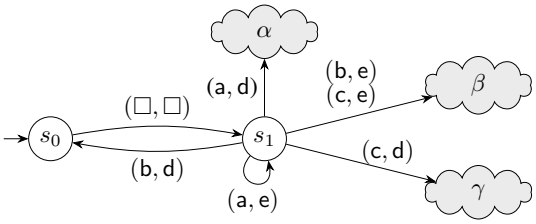

Fig. 5: Monotonicity is not guaranteed in general.

$\mathsf{U}'(s_0) = \mathsf{U}'(s_1) = \mathsf{U}'(s_2) = 0.9$, the set $\mathsf{C} = \{s_0, s_1\}$ contains the BEC $\mathcal{X} = \mathsf{C}$. Now, two linear programs need to be solved to solve the two sub-matrix games.

$$\mathsf{Z}^{\mathsf{exit}}_{\mathsf{U}'}(s_0) := \begin{array}{c} \begin{matrix} \mathsf{d}_1 & \mathsf{d}_2 \end{matrix} \\ \begin{pmatrix} 0.55 & 0.2 \end{pmatrix} \end{array} \begin{array}{c} \\ \mathsf{a}_2 \end{array} \qquad \mathsf{Z}^{\mathsf{exit}}_{\mathsf{U}'}(s_1) = \begin{array}{c} \begin{matrix} \mathsf{e}_1 & \mathsf{e}_2 \end{matrix} \\ \begin{pmatrix} 0.9 & 0.7 \end{pmatrix} \end{array} \begin{array}{c} \\ \mathsf{b}_1 \end{array}$$

The best exit is s_1 and the value of the exit 0.7, thus, the upper bounds of s_0 and s_1 are reduced to 0.7 and s_1 is removed from C. The resulting sub-game is depicted in Fig. 4. Finally, for $\mathsf{U}''(s_0) = 0.7$, the set $\mathsf{C} = \{s_0\}$ contains the BEC $\mathcal{X} = \{s_0\}$. To deflate it, we need to solve the following exiting sub-game.

$$\mathsf{Z}^{\mathsf{exit}}_{\mathsf{U}''}(s_0) := \begin{array}{c} \begin{matrix} \mathsf{d}_1 & \mathsf{d}_2 \end{matrix} \\ \begin{pmatrix} 0.45 & 0.2 \end{pmatrix} \end{array} \begin{array}{c} \\ \mathsf{a}_2 \end{array}$$

The best exit value is 0.2. The upper upper bound of s_0 is reduced to 0.2. After removing the state from $\mathcal{X}$ we obtain an empty set and the algorithm can proceed with the next MEC. Since, we assumed that the game has only one MEC, the deflating phase is finished. As $\mathsf{U}^0 - \mathsf{L}^0 > \varepsilon$ holds, the next iteration of the algorithm is executed.

The Bellman update returns the same valuation for all states, i.e. $\mathsf{U}^1(s_0) = 0.2$, $\mathsf{U}^1(s_1) = 0.7$ and $\mathsf{U}^1(s_2) = 0.9$. Alg. 3 again finds the MEC $\mathsf{C} = \{s_0, s_1, s_2\}$ that contains the two BECs $\mathcal{X}_1 := \{s_0\}$ and $\mathcal{X}_1 := \{s_2\}$. First $\mathcal{X} = \{s_0\}$ is deflated to 0.2 and next the BEC $\mathcal{X} = \{s_2\}$ is deflated. For this, the following exiting sub-game needs to be solved.

$$\mathsf{Z}^{\mathsf{exit}}_{\mathsf{U}^1}(s_0) := \begin{array}{c} \begin{matrix} \mathsf{d}_2 \end{matrix} \\ \begin{pmatrix} 0.45 \end{pmatrix} \end{array} \begin{array}{c} \\ \mathsf{a}_2 \end{array}$$

Notice, that here we only consider staying strategies for player $\mathscr{S}$, which is why we only consider action d_2. Therefore, the best exit value is 0.45. After removing s_0 and s_2 from $\mathcal{X}$, no further BECs are contained in the MEC. Since now $\mathsf{U}^1 - \mathsf{L}^1 < \varepsilon$ holds, the BVI algorithm terminates. △

C. Non-Monotonicity of Deflation

Example 41. Consider the CSG depicted in Fig. 5. The variables α, β and γ are placeholders indicating that upon leaving a certain valuation is obtained. The set of states $\{s_0, s_1\}$ is an EC. We consider two valuations, U and U', such that $\mathsf{U} \geq \mathsf{U}'$ and the EC $\{s_0, s_1\}$ is for both upper bounds a BEC. The upper bound U assigns the following valuations: $\mathsf{U}(s_0) = 0.6, \mathsf{U}(s_1) = 0.6, \mathsf{U}(\alpha) = 0.8, \mathsf{U}(\beta) = 0.5, \mathsf{U}(\gamma = 0.55)$. The upper bound U' assigns the following valuations: $\mathsf{U}'(s_0) = 0.6, \mathsf{U}'(s_1) = 0.45, \mathsf{U}'(\alpha) = 0.5, \mathsf{U}'(\beta) = 0.5, \mathsf{U}'(\gamma = 0.55)$. Then, the matrix games played for the two valuations at state s_1 are given by the following matrices:

$$\mathsf{Z}_{\mathsf{U}}(s_1) := \begin{array}{c} \begin{matrix} \mathsf{d} & \mathsf{e} \end{matrix} \\ \begin{pmatrix} 0.8 & 0.6 \\ 0.6 & 0.5 \\ 0.55 & 0.5 \end{pmatrix} \end{array} \begin{array}{c} \mathsf{a} \\ \mathsf{b} \\ \mathsf{c} \end{array}, \qquad \mathsf{Z}_{\mathsf{U}'}(s_1) := \begin{array}{c} \begin{matrix} \mathsf{d} & \mathsf{e} \end{matrix} \\ \begin{pmatrix} 0.5 & 0.45 \\ 0.6 & 0.5 \\ 0.55 & 0.5 \end{pmatrix} \end{array} \begin{array}{c} \mathsf{a} \\ \mathsf{b} \\ \mathsf{c} \end{array}$$

Then, under U the strategy $\{\mathsf{a} \mapsto 1\}$ is a hazardous strategy and the exit value is 0.5. In contrast for U' the strategy $\{\mathsf{b} \mapsto 1\}$ is hazardous and the exit value is 0.55. Thus, for arbitrary $\mathsf{U}, \mathsf{U}' \in \mathbb{R}^{|\mathsf{S}|}$ it might happen that $\mathsf{exitVal}_{\mathsf{U}}(\mathcal{X}) < \mathsf{exitVal}_{\mathsf{U}'}(\mathcal{X})$ for some BEC $\mathcal{X}$ although $\mathsf{U} \geq \mathsf{U}'$ holds. Intuitively, this arises because the sub-EC which forms a BEC changes when the relative ordering of exits is modified. However, the problem cannot occur when considering valid over-approximations (see Lem. 36) because then the upper bounds decrease in a well-behaved way when the relative ordering of the exits changes. △

APPENDIX C
PROOFS FOR SEC. III

Throughout the whole Appendix C, when proving convergence of BVI, we utilize definitions and theorems from lattice and fixpoint theory. Thus, we first briefly recall some necessary definitions (adjusting notation to our work) and theorems.

Definition 42 (Ordered set [13, Chapter 1.2])**.** A set P equipped with a relation $\preceq$: $P \times P$ is called an *ordered set* if and only if $\preceq$ is reflexive, antisymmetric and transitive.

Definition 43 (Directed set [13, Chapter 7.7])**.** Let P be an ordered set. A non-empty set $D \subseteq P$ is *directed* if and only if for every pair of elements $x, y \in \mathsf{D}$ there exists $z \in \mathsf{D}$ that is a lower bound for both, formally $z \preceq x$ and $z \preceq y$.

Definition 44 (Complete partial order (CPO) [13, Chapter 8.1])**.** An ordered set P is a complete partially ordered set (CPO) if and only if

(i) P has a top element, $\top := \mathsf{inf}_P \emptyset$, and
(ii) for every directed set $D \subseteq P$, we have $\mathsf{inf}_P D$ exists.

Definition 45 (Continuity [13, Chapter 8.6])**.** Let P and Q be two CPOs. A mapping $\varphi : P \to Q$ is *continuous* if

(i) for every directed set $D \subseteq P$, the subset $\varphi(D)$ of Q is also directed, and
(ii) it holds that $\varphi(\mathsf{inf}\, D) = \mathsf{inf}\, \varphi(D) := \mathsf{inf}\{\varphi(x) | x \in D\}$.

Definition 46 (Order-preserving [13, Chapter 1.34])**.** Let P and Q be ordered sets. A map $\varphi : P \to Q$ is *order-preserving* (also called monotone) if $x \preceq y$ in P implies $\varphi(x) \preceq \varphi(y)$ in Q.

Theorem 47 (Fixpoint Theorem [13, Chapter 8.15])**.** *Let P be a complete partial order, let F be an order-preserving and continuous self-map on P and define $\alpha := \mathsf{sup}_{n \geq 0} F^n(\top)$. Then α is the greatest fixpoint of F, i.e. the largest element of P satisfying $F(\alpha) = \alpha$.*

We remark that we inverted the definitions and the theorem: This is because we are interested in a greatest fixpoint, whereas the textbook [13] only speaks about least fixpoints. Inverting the comparator and replacing inf with sup yields the original definitions. With only these changes, the proof of [13, Chapter 8.15] yields our modified claim Thm. 47.

A. Convergence in the absence of end components

We start by proving a technical lemma that is also useful for several future proofs: The over-approximation computed using only Bellman updates converges to a fixpoint.

Lemma 48 (Upper bound converges to a fixpoint)**.** *Let $(\mathsf{U}^k)_{k \in \mathbb{N}_0}$ be the sequence of upper bounds computed by applying Eq. (3) on a CSG G. Let $\mathsf{U}^\star := \lim_{k \to \infty} \mathsf{U}^k$ be the limit of the sequence. This limit is a fixpoint of the Bellman update, i.e. for all $s \in \mathsf{S}$, $\mathfrak{B}(\mathsf{U}^*)(s) = \mathsf{U}^\star(s)$.*

Proof. This lemma is a consequence of the fixpoint theorem we just recalled. Thus, we proceed as follows: We explain that the domain of $\mathfrak{B}$ is a CPO and prove that $\mathfrak{B}$ is order-preserving and continuous. Then, Thm. 47 yields that $\mathsf{U}^\star$ is a (namely the greatest) fixpoint.

Complete partial order. The domain of $\mathfrak{B}$ are valuations, i.e. vectors $[0,1]^{|\mathsf{S}|}$ mapping every state to a number. Thus, we define the set P to be the set of all valuations. We use the standard point-wise comparisons as relation, i.e. $\upsilon_1 \preceq \upsilon_2$ if and only if for all states $s \in \mathsf{S}$ we have $\upsilon_1(s) \leq \upsilon_2(s)$. Thus, the top element $\top$ is the function that maps all states to 1. For every directed set D, a greatest lower bound $\ell = \mathsf{inf}_P D$ exists: Set $\ell(s) = \mathsf{inf}_{d \in D} d(s)$ for all $s \in \mathsf{S}$. It is a lower bound, as by point-wise comparison, it is smaller than all valuations in D; it is the greatest lower bound, since picking a larger number for any state would not be a lower bound any more. Thus, the set consisting of valuations $[0,1]^{|\mathsf{S}|}$ with this relation is a CPO.

Order-preserving. Recall that the Bellman operator on a state is defined as follows: $\mathfrak{B}(\upsilon)(s) := \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\upsilon)(s, \rho, \sigma)$, where

$$\mathfrak{B}(\upsilon)(s, \rho, \sigma) := \sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathsf{S}} \upsilon(s') \cdot \delta\big(s, \mathsf{a}, \mathsf{b}\big)\big(s'\big) \cdot \rho(s)(\mathsf{a}) \cdot \sigma(s)(\mathsf{b}).$$

Eq. (3) lifts it to valuations by applying it state-wise. Hence, for every state, we apply an operation consisting of multiplications and summations, which are order-preserving. Thus, overall, the Bellman operator is order-preserving.

Continuous. We just showed that the Bellman operator on valuations is an order-preserving self-map on the set P of valuations. Then, [13, Lemma 8.7 (i)] yields that for every directed subset $D \subseteq P$, we have that $\mathfrak{B}(D) \coloneqq \{\mathfrak{B}(d) \mid d \in D\}$ is a directed subset, which is Condition (i) of Def. 45. It remains to show Condition (ii): $\inf_{d\in D} \mathfrak{B}(d) = \mathfrak{B}(\inf D)$. Since the comparisons by the relation $\preceq$ are performed point-wise, we have to prove that for all states $s \in \mathsf{S}$, we have $\inf_{d\in D} \mathfrak{B}(d)(s) = \mathfrak{B}(\inf D)(s)$. Thus, fix an arbitrary state $s \in \mathsf{S}$, and conclude using the following chain of equations.

$$
\begin{aligned}
&\mathfrak{B}(\inf D)(s) \\
&= \sup_{\rho\in\mathcal{R}(s)} \inf_{\sigma\in\mathcal{S}(s)} \sum_{(\mathsf{a},\mathsf{b})\in\mathcal{A}} \sum_{s'\in\mathsf{S}} \\
&\qquad \left(\inf_{d\in D} d(s')\right) \cdot \delta\big(s,\mathsf{a},\mathsf{b}\big)\big(s'\big) \cdot \rho(s)(\mathsf{a}) \cdot \sigma(s)(\mathsf{b}) && \text{(Unfolding definition of Bellman operator)} \\
&= \sup_{\rho\in\mathcal{R}(s)} \inf_{\sigma\in\mathcal{S}(s)} \inf_{d\in D} \sum_{(\mathsf{a},\mathsf{b})\in\mathcal{A}} \sum_{s'\in\mathsf{S}} \\
&\qquad d(s') \cdot \delta\big(s,\mathsf{a},\mathsf{b}\big)\big(s'\big) \cdot \rho(s)(\mathsf{a}) \cdot \sigma(s)(\mathsf{b}) && \text{(Claim 1)} \\
&= \inf_{d\in D} \sup_{\rho\in\mathcal{R}(s)} \inf_{\sigma\in\mathcal{S}(s)} \sum_{(\mathsf{a},\mathsf{b})\in\mathcal{A}} \sum_{s'\in\mathsf{S}} \\
&\qquad d(s') \cdot \delta\big(s,\mathsf{a},\mathsf{b}\big)\big(s'\big) \cdot \rho(s)(\mathsf{a}) \cdot \sigma(s)(\mathsf{b}) && \text{(Claim 2)} \\
&= \inf_{d\in D} \mathfrak{B}(d)(s). && \text{(Collapsing the Bellman operator definition)}
\end{aligned}
$$

Claim 1: This step moves the $\inf_{d\in D}$ out of the summation, which is correct, since addition is a continuous operation. ▲

Claim 2: This step moves $\inf_{d\in D}$ to the front, first utilizing that infima can be switched. Then, to switch $\inf_{d\in D}$ and $\sup_{\rho\in\mathcal{R}(s)}$, we make use of the Minimax Theorem [45] which states that for a function $f : X \times Y \to \mathbb{R}$ that is concave-convex, it holds that $\sup_{x\in X} \inf_{y\in Y} f(x,y) = \inf_{y\in Y} \sup_{x\in X} f(x,y)$. f is concave-convex if f is concave for a fixed $y \in Y$ and convex for a fixed $x \in X$. This holds, in particular, for bilinear functions, i.e. functions that are linear in both arguments. The function considered at this point is the following:

$$
\begin{aligned}
f(\rho, d) = \inf_{\sigma\in\mathcal{S}(s)} \sum_{(\mathsf{a},\mathsf{b})\in\mathcal{A}} \sum_{s'\in\mathsf{S}} \\
d(s') \cdot \delta\big(s,\mathsf{a},\mathsf{b}\big)\big(s'\big) \cdot \rho(s)(\mathsf{a}) \cdot \sigma(s)(\mathsf{b}).
\end{aligned}
$$

This function is indeed bilinear, since addition and multiplication are linear functions. ▲

Overall, we have shown that the sequence $(\mathsf{U}^k)_{k\in\mathbb{N}_0}$ is the result of applying an order-preserving, continuous function to the top-element of a complete partial order, and thus it converges to a (the greatest) fixpoint.

□

Theorem 9 (Convergence without ECs — Proof in App. C-A)**.** *Let* G *be a CSG where all ECs are trivial, i.e. for every EC* C *we have* $\mathsf{C} \subseteq \mathsf{W}_{\mathscr{S}} \cup \mathsf{T}$*. Then, the over-approximation using only Eq.* (3) *converges, i.e.* $\lim_{k\to\infty} \mathsf{U}^k = \mathsf{V}_{\mathscr{R}}$.

Proof. This proof is an extension of the proof of [18, Theorem 1] for turn-based games to the concurrent setting. The underlying idea is the same, and can be briefly summarized as follows: We assume towards a contradiction that $\mathsf{U}^\star \neq \mathsf{V}_{\mathscr{R}}$, and find a set $\mathcal{X}$ that maximizes the difference between upper bound and value. Every pair of strategies leaving the set decreases the difference. However, $\mathsf{V}_{\mathscr{R}}$ and $\mathsf{U}^\star$ are fixpoints of the Bellman updates, from [17, Theorem 1] and Lem. 48, respectively. Consequently, optimal strategies need to remain in the set. However, in the absence of ECs, optimal strategies have to leave the set, which yields a contradiction and proves that $\mathsf{U}^\star = \mathsf{V}_{\mathscr{R}}$.

Main challenge. The key difference to the proof of [18, Theorem 1] is that we cannot argue about actions anymore, but have to consider mixed strategies. This significantly complicates notation. Additionally, and more importantly, the former proof crucially relied on the fact that for a state of player $\mathscr{R}$, we know that its valuation is at least as large as that of any action, and dually for a state of player $\mathscr{S}$, its valuation is at most as large as that of any action. In the concurrent setting, this is not true. The optimal strategies need not be maximizing nor minimizing the valuation and, moreover, they can be maximizing for one valuation and minimizing for another. Thus, we found a more general, and in fact simpler, way of proving that "no state in $\mathcal{X}$ can depend on the outside" [18, Statement 5] and deriving the contradiction. The crucial insight is that we can fix locally optimal strategies and then apply Claim 4.

Notation for Bellman operator. Before we begin the formal proof, we establish a condensed notation for a number of terms in the Bellman operator:

$$rest(s, a, b, s', \rho, \sigma) = \delta\big(s, \mathsf{a}, \mathsf{b}\big)\big(s'\big) \cdot \rho(s)(\mathsf{a}) \cdot \sigma(s)(\mathsf{b}).$$

Thus, the Bellman operator for some valuation υ and pair of strategies (ρ, σ) simplifies to

$$\mathfrak{B}(\upsilon)(s, \rho, \sigma) = \sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathsf{S}} \upsilon(s') \cdot rest(s, a, b, s', \rho, \sigma).$$

The set $\mathcal{X}$ with maximum difference. We define the *difference of a state* $s \in \mathsf{S}$ as $\Delta(s) \coloneqq \mathsf{U}^\star(s) - \mathsf{V}_{\mathscr{R}}(s)$. Recall that $\mathsf{V}_{\mathscr{R}}$ is the least fixpoint and $\mathsf{U}^\star$ the greatest fixpoint of $\mathfrak{B}$. Hence, we know that $\Delta(s) \geq 0$ for all states. Further, since we assume for contradiction that $\mathsf{U}^\star \neq \mathsf{V}_{\mathscr{R}}$, there exist states with $\Delta(s) > 0$. Thus, we can find a non-empty set of states with maximum difference: $\mathcal{X} \coloneqq \{s \in \mathsf{S} | \Delta(s) = \mathsf{max}_{s \in \mathsf{S}}\, \Delta(s)\}$.

A leaving pair of strategies decreases the difference. Let $s \in \mathcal{X}$ be a state in $\mathcal{X}$. Let $(\rho, \sigma) \in (\mathcal{R}(s) \times \mathcal{S}(s))$ be a pair of strategies such that (s, ρ, σ) leaves $\mathcal{X}$. Then, following this pair of strategies for one step decreases the difference, formally

$$\mathfrak{B}(\Delta)(s, \rho, \sigma) < \Delta(s). \tag{5}$$

We prove this using the following chain of equations:

$$\begin{aligned}
&\mathfrak{B}(\Delta)(s, \rho, \sigma) \\
&= \sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathsf{S}} \Delta(s') \cdot rest(s, a, b, s', \rho, \sigma) && \text{(Definition of Bellman operator)} \\
&< \Delta(s). && \text{(Claim 3)}
\end{aligned}$$

Claim 3: By assumption, we have that there exists a $t \in \mathsf{S} \setminus \mathcal{X}$ such that this t is reached with positive probability under the exiting strategies, i.e. $\sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} rest(s, a, b, t, \rho, \sigma) > 0$. For this t outside of $\mathcal{X}$, we have $\Delta(t) < \Delta(s)$, since $\mathcal{X}$ is defined as the set of all states with maximum difference. Further, no state can have a difference larger than $\Delta(s)$. Furthermore, the remaining terms in the sum that the differences $\Delta(s')$ are multiplied with are a probability distribution, formally $\sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathsf{S}} rest(s, a, b, s', \rho, \sigma) = 1$. Thus, if all differences were equal to $\Delta(s)$, the sum would yield $\Delta(s)$. As one of the summands is smaller than $\Delta(s)$ and all others are at most $\Delta(s)$, we get that the sum has to be smaller than $\Delta(s)$. ▲

Without non-trivial ECs, $\mathcal{X}$ must be left. We have that $\mathcal{X} \cap (\mathsf{W}_{\mathscr{S}} \cup \mathsf{T}) = \emptyset$: The difference is 0 for target states because both the value and the upper bound are equal to 1; and the difference is 0 for the sure winning region of player $\mathscr{S}$, since the upper bound and value are equal to 0 (see Eq. (3)). Thus, since by assumption there are no ECs in $\mathsf{S} \setminus (\mathsf{W}_{\mathscr{S}} \cup \mathsf{T})$, the set $\mathcal{X}$ cannot contain an EC. Consequently, there exists a state $s \in \mathcal{X}$ such that for all pairs of available actions $(a, b) \in \Gamma_{\mathscr{R}}(s) \times \Gamma_{\mathscr{S}}(s)$, we have $\mathsf{Supp}(\delta\big(s, a, b\big)) \cap (\mathsf{S} \setminus \mathcal{X}) \neq \emptyset$, i.e. there is a successor state outside of $\mathcal{X}$. This is the case because, if all states had a pair of actions that stays in $\mathcal{X}$, then there exists a pair of strategies that keeps the play inside a subset of $\mathcal{X}$, which would then form an EC. For a formal proof, we refer to [18, Lemma 2]. Note that, while their proof is for turn-based games, the definition of EC is a graph theoretic notion where, intuitively, the players "work together" (formally, it is only about the existence of an edge in the underlying hypergraph), and thus the proof is applicable to CSGs, too. In the following, we let s denote such a state where all strategies are leaving.

Notation for locally optimal strategies. For any state and valuation, locally optimal strategies for both players exist. We establish a shorthand for the locally optimal strategies in state s (the one obtained in the previous step) with respect to $\mathsf{U}^\star$ and $\mathsf{V}_{\mathscr{R}}$. For player $\mathscr{R}$ and valuation $\mathsf{U}^\star$, we denote a locally optimal strategy by

$$\rho_{\mathsf{U}} \in \operatorname*{arg\,max}_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\mathsf{U}^\star)(s, \rho, \sigma).$$

Similarly we denote a locally optimal strategy of player $\mathscr{S}$ with respect to $\mathsf{U}^\star$ by

$$\sigma_{\mathsf{U}} \in \operatorname*{arg\,min}_{\sigma \in \mathcal{S}(s)} \sup_{\rho \in \mathcal{R}(s)} \mathfrak{B}(\mathsf{U}^\star)(s, \rho, \sigma).$$

Analogously, we define locally optimal strategies with respect to $\mathsf{V}_{\mathscr{R}}$, namely ρ_{V} and σ_{V}, obtained by replacing $\mathsf{U}^\star$ with $\mathsf{V}_{\mathscr{R}}$ in the above definition.

Deriving the contradiction. Recall that $s \in \mathcal{X}$ is a state where all available pairs of actions, and thus all strategies, leave $\mathcal{X}$. We derive the contradiction $\Delta(s) < \Delta(s)$.

$$
\begin{aligned}
\Delta(s) &= \mathsf{U}^\star(s) - \mathsf{V}_{\mathscr{R}}(s) && \text{(Definition of } \Delta\text{)}\\
&= \mathfrak{B}(\mathsf{U}^\star)(s) - \mathsf{V}_{\mathscr{R}}(s) && (\mathsf{U}^\star \text{ is fixpoint by Lem. 48})\\
&= \mathfrak{B}(\mathsf{U}^\star)(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{U}}) - \mathsf{V}_{\mathscr{R}}(s) && ((\rho_{\mathsf{U}}, \sigma_{\mathsf{U}}) \text{ locally optimal w.r.t. } \mathsf{U}^\star)\\
&\leq \mathfrak{B}(\mathsf{U}^\star)(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{V}}) - \mathsf{V}_{\mathscr{R}}(s) && \text{(Claim 4)}\\
&= \mathfrak{B}(\mathsf{U}^\star)(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{V}}) - \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s) && (\mathsf{V}_{\mathscr{R}} \text{ is fixpoint [17, Theorem 1]})\\
&= \mathfrak{B}(\mathsf{U}^\star)(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{V}}) - \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho_{\mathsf{V}}, \sigma_{\mathsf{V}}) && ((\rho_{\mathsf{V}}, \sigma_{\mathsf{V}}) \text{ locally optimal w.r.t. } \mathsf{V}_{\mathscr{R}})\\
&\leq \mathfrak{B}(\mathsf{U}^\star)(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{V}}) - \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{V}}) && \text{(Claim 4)}\\
&= \mathfrak{B}(\Delta)(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{V}}) && \text{(Claim 5)}\\
&< \Delta(s). && ((s, \rho_{\mathsf{U}}, \sigma_{\mathsf{V}}) \text{ leaves } \mathcal{X} \text{ and Eq. (5)})
\end{aligned}
$$

Claim 4: This argument is used in two steps in the above chain of equations. For the first usage, observe that σ_{V} can be at most as good as the optimal σ_{U}. More formally, recall σ_{U} was chosen as the arg min of the Bellman operator with respect to $\mathsf{U}^\star$. Thus, $\mathfrak{B}(\mathsf{U}^\star)(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{U}}) \leq \mathfrak{B}(\mathsf{U}^\star)(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{V}})$.

For the second usage, by the analogous argument we have $\mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho_{\mathsf{V}}, \sigma_{\mathsf{V}}) \geq \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{V}})$. Since this term is the subtrahend of the subtraction, the overall expression can only become greater. ▲

Claim 5: This step follows from expanding the definition of the Bellman operator, rearranging the sums and collapsing the definition of Bellman operator. Formally, for all states q and strategy pairs (ρ, σ) it holds that

$$
\begin{aligned}
\mathfrak{B}(\Delta)(q, \rho, \sigma) &= \sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathsf{S}} \Delta(s') \cdot \mathit{rest}(s, a, b, s', \rho, \sigma) && \text{(Definition of Bellman operator)}\\
&= \sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathsf{S}} \left(\mathsf{U}^\star(s') - \mathsf{V}_{\mathscr{R}}(s')\right) \cdot \mathit{rest}(s, a, b, s', \rho, \sigma) && \text{(Definition of } \Delta\text{)}\\
&= \left(\sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathsf{S}} \mathsf{U}^\star(s') \cdot \mathit{rest}(s, a, b, s', \rho, \sigma) \right) -\\
&\quad \left(\sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathsf{S}} \mathsf{V}_{\mathscr{R}}(s') \cdot \mathit{rest}(s, a, b, s', \rho, \sigma) \right) && \text{(Splitting the sum)}\\
&= \mathfrak{B}(\mathsf{U}^\star)(q, \rho, \sigma) - \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(q, \rho, \sigma). && \text{(Definition of Bellman operator)}
\end{aligned}
$$

▲

Summary. Starting from the assumption that $\mathsf{U}^\star \neq \mathsf{V}_{\mathscr{R}}$, we derived that there exists a set of states $\mathcal{X}$ where the difference Δ between upper bound and value is maximized. Further, a pair of strategies leaving $\mathcal{X}$ decreases this difference. However, since there are no ECs in $\mathcal{X}$, there has to be a state where the optimal strategies for $\mathsf{V}_{\mathscr{R}}$ and $\mathsf{U}^\star$ leave, which allows us to derive a contradiction. Thus, the initial assumption is false, and we have $\mathsf{U}^\star = \mathsf{V}_{\mathscr{R}}$. □

B. Convergence without Bloated End Components

Lemma 14 (Negating Weak Domination — Proof in App. C-B)**.** *Let υ be a valuation, $s \in \mathsf{S}$ a state, $\mathcal{R}_1, \mathcal{R}_2, \mathcal{R}' \subseteq \mathcal{R}(s)$ and $\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}' \subseteq \mathcal{S}(s)$ sets of local strategies.*

If for some sets of strategies we do not *have $\mathcal{R}_2 \prec_{\upsilon,\mathcal{S}'} \mathcal{R}_1$, then we have $\mathcal{R}_1 \preceq_{\upsilon,\mathcal{S}'} \mathcal{R}_2$. Analogously, not $\mathcal{S}_2 \prec_{\upsilon,\mathcal{R}'} \mathcal{S}_1$ implies $\mathcal{S}_1 \preceq_{\upsilon,\mathcal{R}'} \mathcal{S}_2$.*

Proof. We only provide the proof for Player $\mathscr{R}$, as the other one is analogous by exchanging the names of the strategy sets, replacing $\mathcal{R}$ with $\mathcal{S}$ and vice versa.

We assume that we do not have $\mathcal{R}_2 \prec_{\upsilon,\mathcal{S}'} \mathcal{R}_1$ under the set of counter-strategies $\mathcal{S}'$ with respect to υ. Writing out the definition, this means that $\forall \rho_1 \in \mathcal{R}_1 . \exists \rho_2 \in \mathcal{R}_2$:

(i) $\inf_{\sigma \in \mathcal{S}'} \mathfrak{B}(\upsilon)(s, \rho_2, \sigma) > \inf_{\sigma \in \mathcal{S}'} \mathfrak{B}(\upsilon)(s, \rho_1, \sigma)$, or
(ii) $\forall \sigma' \in \mathcal{S}'$ we have $\mathfrak{B}(\upsilon)(s, \rho_2, \sigma') \geq \mathfrak{B}(\upsilon)(s, \rho_1, \sigma')$.

Our goal is to have that $\mathcal{R}_1 \preceq_{\upsilon,\mathcal{S}'} \mathcal{R}_2$, formally: Formally, $\exists \rho_2 \in \mathcal{R}_2 . \forall \rho_1 \in \mathcal{R}_1$:

$$\inf_{\sigma \in \mathcal{S}'} \mathfrak{B}(\upsilon)(s, \rho_1, \sigma) \leq \inf_{\sigma \in \mathcal{S}'} \mathfrak{B}(\upsilon)(s, \rho_2, \sigma).$$

Both conditions of negated weak dominance imply our goal. The only remaining problem is the order of quantifiers. However, the choice of ρ_2 does not depend on ρ_1, and we can always pick ρ_2 as the strategy that maximizes $\inf_{\sigma \in \mathcal{S}'} \mathfrak{B}(\upsilon)(s, \rho_2, \sigma)$. Thus, we can exchange the order of quantifiers and prove our goal. □

Lemma 20 (Negating Bloated — Proof in App. C-B). *If an EC $\mathcal{X} \subseteq \mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}})$ is not bloated for a valuation υ, then there exists a state $s \in \mathcal{X}$ that has a locally optimal strategy that is leaving, formally $\exists \rho \in \mathcal{R}_L(\mathcal{X}, s). \mathcal{R}(s) \preceq_{\upsilon, \mathcal{S}(s)} \{\rho\}$.*

Proof. Since $\mathcal{X}'$ is not a BEC, we know that at some $s \in \mathcal{X}'$ it holds that $\mathsf{Hazard}_{\mathsf{U}^\star}(\mathcal{X}', s) = \emptyset$. Fix s to be such a state. Every strategy $\rho' \in \mathcal{R}(\mathsf{S})$ must violate at least one of the 3 conditions of Def. 16. We write out the negations:

(i) $\rho' \notin \mathcal{R}_{\overline{L}}(\mathcal{X}', s)$, i.e. the strategy is leaving $\rho' \in \mathcal{R}_L(\mathcal{X}', s)$.

(ii) We do not have $\mathcal{R}(s) \setminus \{\rho'\} \preceq_{\mathcal{S}(s)} \{\rho'\}$. By contraposition of Lem. 14, this implies $\{\rho'\} \prec_{\mathcal{S}(s)} \mathcal{R}(s) \setminus \{\rho'\}$, so the strategy is sub-optimal.

(iii) We do not have $\mathcal{R}_L(\mathcal{X}', s) \prec_{\mathcal{S}(s)} \{\rho'\}$. By Lem. 14, this implies $\{\rho'\} \preceq_{\mathcal{S}(s)} \mathcal{R}_L(\mathcal{X}', s)$, i.e. there are leaving strategies that are not worse than ρ'. Note that in particular, this implies that $\mathcal{R}_L(\mathcal{X}', s)$ is non-empty.

Our assumption gives us the disjunction over the three violated conditions. We proceed by a case distinction, always assuming that all strategies violate a certain condition, which allows us to prove our goal, or, if there exists a strategy satisfying the condition, we continue with the next one.

Case "Not (i)": If all strategies violate Condition (i), that means all strategies are leaving, i.e. $\mathcal{R}_L(s) = \mathcal{R}(s)$. Thus, since there always exist optimal strategies, by picking an optimal $\rho \in \mathcal{R}$ we naturally have $\mathcal{R}(s) \preceq_{\upsilon, \mathcal{S}(s)} \{\rho\}$.

Case "(i), but not (iii)": We assume there exists strategies that satisfies Condition (i), but all strategies that satisfy it violate Condition (iii). This means that for every non-leaving strategy, the set of leaving strategies is not worse than it. As this holds for all non-leaving strategies, we have $\mathcal{R}_{\overline{L}}(s) \preceq_{\upsilon, \mathcal{S}(s)} \mathcal{R}_L(s)$. Using that $\mathcal{R}(s) = \mathcal{R}_{\overline{L}}(s) \cup \mathcal{R}_L(s)$ and the set of leaving strategies trivially is not worse than itself, we obtain: $\mathcal{R}(s) \preceq_{\upsilon, \mathcal{S}(s)} \mathcal{R}_L(s)$. Moreover, the set of leaving strategies is non-empty, since the definition of not worse requires that there exists a strategy in the right-hand set. This proves our goal.

Case "(i) and (iii), but not (ii)": We assume there exists strategies that satisfy Condition (i) and (iii), but all these strategies violate Condition (ii). This case cannot happen, and below we derive a contradiction. This completes our cast distinction, since every strategy has to violate at least one of the three conditions.

Our assumption is that there exists a non-leaving strategy that weakly dominates the set of all leaving strategies. However, every such strategy is suboptimal, as by violating Condition (ii) it is weakly dominated by all other strategies. This is a contradiction, because then there are no optimal strategies. More formally, if we assume the optimal strategy is leaving, this is a contradiction, because there exists a non-leaving strategy dominating the set of leaving strategies. And if we assume the optimal strategy is non-leaving, this is a contradiction, because every non-leaving strategy is weakly dominated by the set of others.

□

Theorem 21 (Non-convergence implies BECs — Proof in App. C-B). *Let $\mathsf{U}^\star := \lim_{k \to \infty} \mathsf{U}^k$ be the limit of the naïve upper bound iteration (Eq. (3)) on the CSG G. If VI from above does not converge to the value in the limit, i.e. $\mathsf{U}^\star > \mathsf{V}_{\mathscr{R}}$, then the CSG G contains a BEC in $\mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}})$ with respect to $\mathsf{U}^\star$.*

Proof. Intuition and outline:. This proof builds on the proof of Thm. 9. There, we constructed a set $\mathcal{X}$ maximizing the difference between $\mathsf{U}^\star$ and $\mathsf{V}_{\mathscr{R}}$ and showed that if there is a pair of optimal strategies leaving leaving $\mathcal{X}$, then we can derive a contradiction: The upper bound decreases, which contradicts the fact that it is a fixpoint. In the context of the other proof, that allowed us to show that without ECs, VI converges, because without ECs it is impossible to have a set of states where all optimal strategies stay in that set.

In the presence of ECs, states can indeed have a positive difference between $\mathsf{U}^\star$ and $\mathsf{V}_{\mathscr{R}}$, see e.g. Ex. 6. Our goal is to prove that at least one of these ECs is bloated. Thus, we assume for contradiction that no EC is bloated under $\mathsf{U}^\star$. Thus, by Lem. 20, there is an optimal leaving strategy for player $\mathscr{R}$. Using that, we can repeat the argument from Thm. 9, showing that in this case $\mathsf{U}^\star$ would decrease. Again, this is a contradiction because it is a fixpoint of applying Bellman updates (Lem. 48). Thus, the initial assumption that no EC is bloated is false, and we can conclude that there exists a BEC.

Establishing the Context. As in the proof of Thm. 9, let $\mathcal{X} := \{s \in \mathsf{S} \mid \Delta(s) = \max_{s \in \mathsf{S}} \Delta(s)\}$ be the set of states with maximum difference. We denote the maximum by $\Delta^{\max}$, and our assumption yields that $\Delta^{\max} > 0$. Note that this implies that $\mathcal{X} \cap (\mathsf{W}_{\mathscr{S}} \cup \mathsf{T}) = \emptyset$, since for those states, their value is set correctly by initialization, and their difference is 0. Thus, if we find a BEC that is a subset of $\mathcal{X}$, it also satisfies the additional condition of being non-trivial, i.e. not in $(\mathsf{W}_{\mathscr{S}} \cup \mathsf{T})$. The contraposition of Thm. 9 yields that there has to be an EC in $\mathcal{X}$.

Bottom MECs. To derive the contradiction, in the following, we consider ECs with a particular property, namely ECs that are *bottom in* $\mathcal{X}$. A MEC $\mathcal{X}'$ is bottom in $\mathcal{X}$ if the successors of a pair of strategies that leaves the MEC reaches states outside of $\mathcal{X}$ with positive probability. Intuitively, a bottom MEC in $\mathcal{X}$ is a MEC, such that after leaving it, none of the successors is

part of another MEC in $\mathcal{X}$. One can compute such ECs using the MEC decomposition of $\mathcal{X}$, ordering them topologically and picking one at the end of a chain.

Let $\mathcal{X}'$ be a bottom MEC with $\mathcal{X}' \subseteq \mathcal{X}$. $\mathcal{X}'$ exists because by assumption $\mathcal{X}$ contains at least one EC, thus, there also has to exist an EC that is bottom in $\mathcal{X}$.

Optimal Leaving Strategies in Non-BECs. We use the assumption for contradiction to say that $\mathcal{X}'$ is not bloated with respect to $\mathsf{U}^\star$. Then, using Lem. 20, we know that there exists a state $s \in \mathcal{X}'$ where an optimal strategy ρ_{U} exists that is leaving $\mathcal{X}'$. Moreover, since $\mathcal{X}'$ is a bottom MEC in $\mathcal{X}$, we also have that it is leaving with respect to $\mathcal{X}$.

Deriving the Contradiction. Using these facts, we can exactly repeat the argument used in the proof of Thm. 9 under the paragraph-heading "Deriving the Contradiction". Recall we denote locally optimal strategies with respect to $\mathsf{U}^\star$ by $\rho_{\mathsf{U}}, \sigma_{\mathsf{U}}$ (and we just proved ρ_{U} is leaving for all counter-strategies), and analogously locally optimal strategies with respect to $\mathsf{V}_{\mathscr{R}}$, by ρ_{V} and σ_{V}. We highlight that Eq. (5), Claim 4 and Claim 5 from Thm. 9 are applicable in the context of this proof, too.

$$
\begin{aligned}
\Delta(s) &= \mathsf{U}^\star(s) - \mathsf{V}_{\mathscr{R}}(s) && \text{(Definition of } \Delta\text{)} \\
&= \mathfrak{B}(\mathsf{U}^\star)(s) - \mathsf{V}_{\mathscr{R}}(s) && (\mathsf{U}^\star \text{ is fixpoint by Lem. 48}) \\
&= \mathfrak{B}(\mathsf{U}^\star)(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{U}}) - \mathsf{V}_{\mathscr{R}}(s) && ((\rho_{\mathsf{U}}, \sigma_{\mathsf{U}}) \text{ locally optimal w.r.t. } \mathsf{U}^\star) \\
&\leq \mathfrak{B}(\mathsf{U}^\star)(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{V}}) - \mathsf{V}_{\mathscr{R}}(s) && \text{(Claim 4 in Thm. 9)} \\
&= \mathfrak{B}(\mathsf{U}^\star)(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{V}}) - \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s) && (\mathsf{V}_{\mathscr{R}} \text{ is fixpoint [17, Theorem 1]}) \\
&= \mathfrak{B}(\mathsf{U}^\star)(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{V}}) - \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho_{\mathsf{V}}, \sigma_{\mathsf{V}}) && ((\rho_{\mathsf{V}}, \sigma_{\mathsf{V}}) \text{ locally optimal w.r.t. } \mathsf{V}_{\mathscr{R}}) \\
&\leq \mathfrak{B}(\mathsf{U}^\star)(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{V}}) - \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{V}}) && \text{(Claim 4 in Thm. 9)} \\
&= \mathfrak{B}(\Delta)(s, \rho_{\mathsf{U}}, \sigma_{\mathsf{V}}) && \text{(Claim 5 in Thm. 9)} \\
&< \Delta(s). && ((s, \rho_{\mathsf{U}}, \sigma_{\mathsf{V}}) \text{ leaves } \mathcal{X} \text{ and Eq. (5)})
\end{aligned}
$$

Now that we have derived a contradiction, our initial assumption that all ECs are not BECs is wrong, so we know there exists a BEC $\mathcal{X}' \subseteq (\mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}}))$ with respect to $\mathsf{U}^\star$. This concludes the proof.

As a side note, we remark that it is indeed possible that there is only one BEC that causes many states, even some not in an EC, to have a positive difference Δ. For an example, we refer to [18, Fig. 4].

□

C. Soundness of DEFLATE

The following is a technical lemma that is needed to show the soundness of deflation. Intuitively, it says that the value of all states in an EC needs to depend on some exit. Note that there can be states whose value is higher than their own exit value, namely if they can reach a better exit. However, this cannot be the case for all states, but there must be some whose value is less than or equal to their exit value (first condition), and in fact no state can have a higher value than these states that actually depend on exiting (second condition). The proof is very technical, as essentially it requires unfolding all the definitions, and thereby also unfolding all the included case distinctions.

Lemma 49 (No state has a larger value than that of an exit from its EC)**.** *Let $\mathcal{X} \subseteq \mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}})$ be an EC. Then, it holds that*

(i) $\mathcal{X}' \coloneqq \{s \in \mathcal{X} \mid \mathsf{V}_{\mathscr{R}}(s) \leq \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)\} \neq \emptyset$*, and*

(ii) $\max_{s \in \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s) \geq \max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$.

Proof. We prove the lemma by contradiction, i.e. we assume that one of the two conditions posed by the lemma is violated. We make the following case distinction:

(i) $\mathcal{X}' = \emptyset$; and
(ii) $\mathcal{X}' \neq \emptyset$ but $\max_{s \in \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s) < \max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$.

Case (i)

In this case it holds that $\mathcal{X}' = \emptyset$, i.e. for all $s \in \mathcal{X}$ we have $\mathsf{V}_{\mathscr{R}}(s) > \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)$. Recall that in Def. 27 if $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \emptyset$ is true at some state $s \in \mathcal{X}$, then the exit value is given by the value of the matrix game played at that state.

Consequently, as for all $s \in \mathcal{X}$ it holds that $\mathsf{V}_{\mathscr{R}}(s) > \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)$, at all states $s \in \mathcal{X}$ it must hold that $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) \neq \emptyset$ because otherwise we would obtain the following contradiction: $\mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma) = \mathsf{V}_{\mathscr{R}}(s)$.

We proceed with the assumption that for all $s \in \mathcal{X}$ it holds that $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) \neq \emptyset$. Since by the case assumption at all $s \in \mathcal{X}$ it holds that $\mathsf{V}_{\mathscr{R}}(s) > \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)$ the true values of the states in $\mathcal{X}$ are attainable with the hazardous and trapping strategies. More formally, at each $s \in \mathcal{X}$ the following chain of equations holds.

$$\begin{aligned}
\mathsf{V}_{\mathscr{R}}(s) &= \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma) && (\mathsf{V}_{\mathscr{R}} \text{ is a fixpoint \& case assumption: } \mathsf{V}_{\mathscr{R}}(s) > \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)) \\
&= \sup_{\rho \in \mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma) && (\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) \text{ are optimal by Def. 16}) \\
&= \sup_{\rho \in \mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)} \inf_{\sigma \in \mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s))} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma). && (\mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) \text{ are optimal by Def. 24})
\end{aligned}$$

Thus, for player $\mathscr{R}$ staying in $\mathcal{X}$ is optimal and since no target state is contained in $\mathcal{X}$ it has to hold that $\mathsf{V}_{\mathscr{R}}(s) = 0$ for all $s \in \mathcal{X}$. However, this is a contradiction to the assumption that $\mathsf{V}_{\mathscr{R}}(s) > \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)$ since $\mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) \geq 0$ (as all possible valuations are non-negative and the value of the exiting sub-game is given by the maximum between 0 and $\mathsf{exitVal}$ - see Def. 27 and Def. 26). Thus, the case assumption that $\mathcal{X}' = \emptyset$ must be false.

Case (ii)

In this case it holds that $\mathcal{X}' \neq \emptyset$ but $\max_{s \in \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s) < \max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$ is true.
We make the following case distinction: (ii.a) $\mathcal{X} \setminus \mathcal{X}'$ is a BEC; and (ii.b) $\mathcal{X} \setminus \mathcal{X}'$ is not a BEC.

Case (ii.a)

In this case $\mathcal{X} \setminus \mathcal{X}'$ is a BEC, i.e. for all $s \in \mathcal{X} \setminus \mathcal{X}'$ it holds that $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s) \neq \emptyset$.
In case it holds that $\mathsf{V}_{\mathscr{R}}(s) > \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s)$ for all $s \in \mathcal{X} \setminus \mathcal{X}'$, then since no target state is contained in $\mathcal{X}$ and therefore neither in $\mathcal{X} \setminus \mathcal{X}'$, it must be true that $\mathsf{V}_{\mathscr{R}}(s) = 0$ for all $s \in \mathcal{X} \setminus \mathcal{X}'$. However, similarly as in Case (i), this is a contradiction to the assumption that $\mathsf{V}_{\mathscr{R}}(s) > \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s)$ for all $s \in \mathcal{X} \setminus \mathcal{X}'$ as $\mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s) \geq 0$ for all $s \in \mathcal{X} \setminus \mathcal{X}'$.
Consequently, there has to exist $s \in \mathcal{X} \setminus \mathcal{X}'$ such that $\mathsf{V}_{\mathscr{R}}(s) \leq \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s)$.
We make another case distinction:

(ii.a.1) for all $s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$ it holds that $\mathsf{V}_{\mathscr{R}}(s') > \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s')$; and
(ii.a.2) there exists $s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$ such that $\mathsf{V}_{\mathscr{R}}(s') \leq \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s')$.

Case (ii.a.1) In this case for all $s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$ it holds that $\mathsf{V}_{\mathscr{R}}(s') > \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s')$.
Thus, at each $s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s')$ the following chain of equations holds.

$$\begin{aligned}
\mathsf{V}_{\mathscr{R}}(s') &= \sup_{\rho \in \mathcal{R}(s')} \inf_{\sigma \in \mathcal{S}(s')} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s', \rho, \sigma) && (\mathsf{V}_{\mathscr{R}} \text{ is a fixpoint}) \\
&= \sup_{\rho \in \mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s')} \inf_{\sigma \in \mathcal{S}(s')} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s', \rho, \sigma) && (\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s') \text{ are optimal by Def. 16}) \\
&= \sup_{\rho \in \mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s')} \inf_{\sigma \in \mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s'))} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s', \rho, \sigma). && (\mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s') \text{ are optimal by Def. 24})
\end{aligned}$$

Thus, at all states that attain the highest value among $\mathcal{X} \setminus \mathcal{X}'$, it is optimal for both players to choose strategies that together are staying in $\mathcal{X} \setminus \mathcal{X}'$. However, since $\mathcal{X}$ and thus $\mathcal{X} \setminus \mathcal{X}'$ does not belong to the winning region of Player $\mathscr{S}$, leaving $\mathcal{X} \setminus \mathcal{X}'$ has to be possible. Thus, from each state $s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$, Player $\mathscr{R}$ has to possess an optimal strategy that leads to a state $s'' \in \mathcal{X} \setminus \mathcal{X}'$ where leaving $\mathcal{X} \setminus \mathcal{X}'$ is possible, i.e. where $\mathsf{V}_{\mathscr{R}}(s'') \leq \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s'')$ holds. Thus, at state s'' the highest value also has to be attainable. However, this is a contradiction to the case assumption that for all $s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$ it holds that $\mathsf{V}_{\mathscr{R}}(s') > \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s')$.

Case (ii.a.2) In this case there exists $s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$ such that $\mathsf{V}_{\mathscr{R}}(s') \leq \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s')$. Let $\mathcal{X}'' := \{s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s) \mid \mathsf{V}_{\mathscr{R}}(s') \leq \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s')\}$. Then, we need to make another case distinction:

(ii.a.2.1) for all $s \in \mathcal{X}''$ it holds that $\mathsf{Defl}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s) = \emptyset$; and
(ii.1.2.2) there exists $s' \in \mathcal{X}''$ such that $\mathsf{Defl}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s') \neq \emptyset$.

Case (ii.a.2.1) In this case for all $s \in \mathcal{X}''$ it holds that $\mathsf{Defl}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s) = \emptyset$. Then, since $\mathcal{X}$ and so $\mathcal{X} \setminus \mathcal{X}'$ do not belong to the winning region of Player $\mathscr{S}$, there has to exist a state $s'' \in \mathcal{X} \setminus \mathcal{X}'$ such that $\mathsf{Defl}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s'') \neq \emptyset$

so leaving $\mathcal{X} \setminus \mathcal{X}'$ is possible. However, then $s'' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$ must hold which is a contradiction to the case assumption.

Case (ii.a.2.2) In this case there exists $s' \in \mathcal{X}''$ such that $\mathsf{Defl}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s') \neq \emptyset$. Then, the following chain of equations holds.

$$
\begin{aligned}
\mathsf{V}_{\mathscr{R}}(s') &\leq \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s') && \text{(Case assumption)} \\
&= \max(0, \mathsf{V}(\hat{\mathsf{Z}}(s'))) && \text{(By case assumption (ii.a): } \mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s') \neq \emptyset) \\
&= \mathsf{V}(\hat{\mathsf{Z}}(s')) && \text{(Case assumption: } \mathsf{Defl}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s') \neq \emptyset) \\
&= \sup_{\rho \in \mathsf{Defl}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s')} \inf_{\sigma \in \mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s')} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s', \rho, \sigma) \\
& && \text{(Value of the exiting sub-game — see Def. 27)} \\
&= \sup_{\rho \in \mathsf{Defl}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s')} \inf_{\sigma \in \mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s')} \sum_{(a,b) \in \mathcal{A}} \sum_{s \in \mathcal{X} \setminus \mathcal{X}'} \underbrace{\mathsf{V}_{\mathscr{R}}(s)}_{\leq \max_{s'' \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s'')} \cdot \delta(s', a, b)(s) \cdot \rho(a) \cdot \sigma(b) \\
&\quad + \sum_{s \in \mathcal{X}'} \underbrace{\mathsf{V}_{\mathscr{R}}(s)}_{< \max_{s'' \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s'')} \cdot \delta(s', a, b)(s) \cdot \rho(a) \cdot \sigma(b) \\
& && \text{(Def. of } \mathfrak{B} \text{ and case assumption (ii): } \max_{s \in \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s) < \max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)) \\
&< \max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s). && \text{(Everything sums up to 1)}
\end{aligned}
$$

Thus, from the case assumption we derived a contradiction because $s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$.

Case (ii.b)

In this case $\mathcal{X} \setminus \mathcal{X}'$ is not a BEC, i.e. there exists $s' \in \mathcal{X} \setminus \mathcal{X}'$ such that $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s') = \emptyset$.

We need another case distinction:

(ii.b.1) for all $s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$ it holds that $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s') \neq \emptyset$; and

(ii.b.2) there exists $s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$ such that $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s') = \emptyset$.

Case (ii.b.1) In this case we have that for all $s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$ it holds that $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s') \neq \emptyset$. Then, at each $s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$ the following chain of equations holds.

$$
\begin{aligned}
\mathsf{V}_{\mathscr{R}}(s') &= \sup_{\rho \in \mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s')} \inf_{\sigma \in \mathcal{S}(s')} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s', \rho, \sigma) && (\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s') \text{ are optimal by Def. 16}) \\
&= \sup_{\rho \in \mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s')} \inf_{\sigma \in \mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s'))} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s', \rho, \sigma). && (\mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s') \text{ are optimal by Def. 24})
\end{aligned}
$$

Thus, at all states that attain the highest value among $\mathcal{X} \setminus \mathcal{X}'$ it is optimal for both players to choose strategies that together are staying in $\mathcal{X} \setminus \mathcal{X}'$. However, since $\mathcal{X}$ and thus $\mathcal{X} \setminus \mathcal{X}'$ does not belong to the winning region of Player $\mathscr{S}$, leaving $\mathcal{X} \setminus \mathcal{X}'$ has to be possible. Thus, from each state $s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$, Player $\mathscr{R}$ has to possess an optimal strategy that leads to a state $s'' \in \mathcal{X} \setminus \mathcal{X}'$ where leaving $\mathcal{X} \setminus \mathcal{X}'$ is possible, i.e. where $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s'') = \emptyset$ holds. Thus, at state s'' the highest value also has to be attainable. However, this is a contradiction to the case assumption that for all $s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$ it holds that $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s') \neq \emptyset$.

Case (ii.b.2) In this case there exists $s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$ such that $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s') = \emptyset$. Then, the following chain of equations holds.

$$
\begin{aligned}
\mathsf{V}_{\mathscr{R}}(s') &= \sup_{\rho \in \mathcal{R}(s')} \inf_{\sigma \in \mathcal{S}(s')} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s', \rho, \sigma) && (\mathsf{V}_{\mathscr{R}} \text{ is a fixpoint}) \\
&= \sup_{\rho \in \mathcal{R}(s')} \inf_{\sigma \in \mathcal{S}(s')} \sum_{(a,b) \in \mathcal{A}} \sum_{s \in \mathcal{X} \setminus \mathcal{X}'} \underbrace{\mathsf{V}_{\mathscr{R}}(s)}_{\leq \max_{s'' \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s'')} \cdot \delta(s', a, b)(s) \cdot \rho(a) \cdot \sigma(b) \\
&\quad + \sum_{s \in \mathcal{X}'} \underbrace{\mathsf{V}_{\mathscr{R}}(s)}_{< \max_{s'' \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s'')} \cdot \delta(s', a, b)(s) \cdot \rho(a) \cdot \sigma(b) \\
&< \max_{s'' \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s'').
\end{aligned}
$$

(Everything sums up to 1 and $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X} \setminus \mathcal{X}', s') = \emptyset$ thus at least one successor state is in $\mathcal{X}'$)

Thus, from the case assumption we derived a contradiction because $s' \in \arg\max_{s \in \mathcal{X} \setminus \mathcal{X}'} \mathsf{V}_{\mathscr{R}}(s)$.

Thus, every case leads to a contradiction which concludes the proof. □

Using Lem. 49, we now prove that for every EC $\mathcal{X}$, we have that for an upper bound U, the best exit value of the EC is also an upper bound. This is the crucial ingredient for showing that deflation cannot decrease a valuation below the value $\mathsf{V}_{\mathscr{R}}$.

Lemma 50. *Let $\mathcal{X} \subseteq \mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}})$ be an EC, and $\mathsf{U} \in [0,1]^{|\mathsf{S}|}$ be a valuation with $\mathsf{U} \geq \mathsf{V}_{\mathscr{R}}$. Then, for all states $s \in \mathsf{bestExits}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X})$, we have $\mathsf{exitVal}_{\mathsf{U}}(\mathcal{X}, s) \geq \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)$.*

Proof. Let $s \in \mathsf{bestExits}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X})$. Our goal is to show that $\mathsf{exitVal}_{\mathsf{U}}(\mathcal{X}, s) \geq \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)$. Since for the estimation of $\mathsf{exitVal}_{\mathsf{U}}(\mathcal{X}, s)$ and $\mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)$ the sets of strategies, $\mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)$, $\mathsf{Trap}_{\mathsf{U}}(\mathcal{X}, s)$, $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)$, and $\mathsf{Hazard}_{\mathsf{U}}(\mathcal{X}, s)$, which can be empty or non-empty, we have to consider all possible combinations. We consider the following four main cases, which we further analyse with respect to the U sets of strategies where necessary:

Case (I) $\mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \emptyset$ and $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \emptyset$,
Case (II) $\mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \emptyset$ and $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) \neq \emptyset$,
Case (III) $\mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) \neq \emptyset$ and $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \emptyset$, and
Case (IV) $\mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) \neq \emptyset$ and $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) \neq \emptyset$.

For each case we show that either the case is impossible or that the statement of the lemma holds.

Case (I) $\mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \emptyset$ and $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \emptyset$.

Let $\sigma' \in \mathcal{S}(s)$ be an optimal Player $\mathscr{S}$ strategy under $\mathsf{V}_{\mathscr{R}}$. Since $\mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \emptyset$, at least on of the two conditions posed by the definition of trapping strategies (Def. 24) must be violated under $\mathsf{V}_{\mathscr{R}}$.
Since $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \emptyset$, Condition (ii) of Def. 24, i.e. $\forall \rho \in \mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) : (s, \rho, \sigma')\ \mathsf{staysIn}\ \mathcal{X}$, is trivially satisfied. Consequently, Condition (i) must be violated, i.e., it must hold that there exists no optimal strategy for Player $\mathscr{S}$ which cannot be true. Thus, this case is impossible.

Case (II) $\mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \emptyset$ and $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) \neq \emptyset$.

At state s it holds that $\mathsf{V}_{\mathscr{R}}(s) = \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)$, since we chose it to be in $\mathsf{bestExits}_{\mathsf{V}_{\mathscr{R}}}$. Thus, it is in the set $\mathcal{X}$, constructed in Condition (i) of Lem. 49. By Condition (ii) of Lem. 49, we know that all states in $\mathcal{X} \setminus \{s\}$ attain a value that is either smaller or equal $\mathsf{V}_{\mathscr{R}}(s)$. Using this and the fact that no staying strategy can be optimal (as the set of trapping strategies is empty), we can derive a contradiction as follows.

$$
\begin{aligned}
\mathsf{V}_{\mathscr{R}}(s) &= \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma) && (\mathsf{V}_{\mathscr{R}} \text{ is fixpoint of } \mathfrak{B}) \\
&= \sup_{\rho \in \mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma) && (\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) \text{ are optimal under } \mathsf{V}_{\mathscr{R}}) \\
&< \sup_{\rho \in \mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)} \inf_{\sigma \in \mathcal{S}_{\overline{L}}(\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s), \mathcal{X}, s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma) && (\mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \emptyset) \\
&= \sup_{\rho \in \mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)} \inf_{\sigma \in \mathcal{S}_{\overline{L}}(\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s), \mathcal{X}, s)} \sum_{(\mathsf{a}, \mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathcal{X}} \underbrace{\mathsf{V}_{\mathscr{R}}(s')}_{\leq \mathsf{V}_{\mathscr{R}}(s)} \cdot \delta(s, \rho, \sigma) \cdot \rho(\mathsf{a}) \cdot \sigma(\mathsf{b}) \\
& && (\text{Unfolding definition of } \mathfrak{B}) \\
&\leq \sup_{\rho \in \mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)} \inf_{\sigma \in \mathcal{S}_{\overline{L}}(\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s), \mathcal{X}, s)} \sum_{(\mathsf{a}, \mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathcal{X}} \mathsf{V}_{\mathscr{R}}(s) \cdot \delta(s, \rho, \sigma) \cdot \rho(\mathsf{a}) \cdot \sigma(\mathsf{b}) \\
& && (\mathsf{V}_{\mathscr{R}}(s') \leq \mathsf{V}_{\mathscr{R}}(s) \text{ by (ii) in Lem. 49}) \\
&= \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s) \\
&= \mathsf{V}_{\mathscr{R}}(s). && (\mathsf{V}_{\mathscr{R}} \text{ is fixpoint of } \mathfrak{B})
\end{aligned}
$$

Thus, overall $\mathsf{V}_{\mathscr{R}}(s) < \mathsf{V}_{\mathscr{R}}(s)$, a contradiction.

Case (III): $\mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) \neq \emptyset$ and $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \emptyset$.

Since $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \emptyset$ we know by Lem. 14 that there exists $\rho_L \in \mathcal{R}_L(\mathcal{X}, s).\mathcal{R}(s) \preceq_{\mathsf{V}_{\mathscr{R}}, \mathcal{S}(s)} \{\rho_L\}$. Our goal is to show that there exists $\rho^* \in \mathcal{R}(s)$ that is optimal under $\mathsf{V}_{\mathscr{R}}$ and $\rho^* \in \mathsf{Defl}_{\mathsf{U}}(\mathcal{X}, s)$. We distinguish two cases:

- (III.a): For all $\rho_M \in \mathcal{R}_L(\mathcal{X}, s)$ that are optimal under $\mathsf{V}_{\mathscr{R}}$, there exists $\rho_H \in \mathsf{Hazard}_{\mathsf{U}}(\mathcal{X}, s)$ such that $\mathsf{Supp}(\rho_M) \cap \mathsf{Supp}(\rho_H) \neq \emptyset$, and
- (III.b): There exists $\rho_L \in \mathcal{R}_L(\mathcal{X}, s)$ that is optimal under $\mathsf{V}_{\mathscr{R}}$ and it holds that $\mathsf{Supp}(\rho_L) \cap \bigcup_{\rho \in \mathsf{Hazard}_{\mathsf{U}}(\mathcal{X}, s)} \mathsf{Supp}(\rho) = \emptyset$.

Case (III.a) In this case, for all $\rho_M \in \mathcal{R}_L(\mathcal{X}, s)$ that are optimal under $\mathsf{V}_{\mathscr{R}}$, there exists $\rho_H \in \mathsf{Hazard}_{\mathsf{U}}(\mathcal{X}, s)$ such that $\mathsf{Supp}(\rho_M) \cap \mathsf{Supp}(\rho_H) \neq \emptyset$.

Let $\rho_H \in \mathsf{Hazard}_{\mathsf{U}}(\mathcal{X}, s)$ and $\rho_L \in \{\rho \in \mathcal{R}_L(\mathcal{X}, s) \mid \mathsf{Supp}(\rho_H) \cap \mathsf{Supp}(\rho_L) = \emptyset\}$ such that $\mathsf{Supp}(\rho_M) \cap \mathsf{Supp}(\rho_H) \neq \emptyset$ and $\mathsf{Supp}(\rho_M) \cap \mathsf{Supp}(\rho_L) \neq \emptyset$. In other words, only a strategy that mixes the supports of ρ_H and ρ_L is optimal under $\mathsf{V}_{\mathscr{R}}$.
Due to the case assumption that only strategies that mix with some hazardous strategy are optimal, ρ_L must be sub-optimal under $\mathsf{V}_{\mathscr{R}}$, as well as ρ_H.
More formally, for all $\rho_M \in \mathcal{R}_L(\mathcal{X}, s)$, such that $\mathcal{R}(s) \setminus \{\rho_M\} \preceq \{\rho_M\}$ it holds that

$$\exists \rho_H \in \mathsf{Hazard}_{\mathsf{U}}(\mathcal{X}, s). \exists \rho_L \in \{\rho \in \mathcal{R}_L(\mathcal{X}, s) \mid \mathsf{Supp}(\rho_H) \cap \mathsf{Supp}(\rho_L) = \emptyset\} :$$
$$\sup_{\mathsf{Dist}(\mathsf{Supp}(\rho_M) \setminus \mathsf{Supp}(\rho_L))} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma) \prec \{\rho_M\}$$
$$\text{and}$$
$$\sup_{\mathsf{Dist}(\mathsf{Supp}(\rho_M) \setminus \mathsf{Supp}(\rho_H))} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma) \prec \{\rho_M\}.$$

Let $\mathcal{S}^* \subseteq \mathcal{S}(s)$ be the optimal Player $\mathscr{S}$ strategies with respect to ρ_M. Since ρ_M has to mix ρ_L and ρ_H, the following must be true:

- there exists $\sigma_1 \in \mathcal{S}^*$ such that

$$\mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho_L, \sigma_1) < \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho_H, \sigma_1), \text{ and} \tag{6}$$

- there exists $\sigma_2 \in \mathcal{S}^*$ such that

$$\mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho_L, \sigma_2) > \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho_H, \sigma_2), \tag{7}$$

because otherwise either ρ_L or ρ_H would be optimal under $\mathsf{V}_{\mathscr{R}}$ thus mixing would not be necessary. Further, Player $\mathscr{S}$ also needs to mix the two strategies σ_1 and σ_2 to ensure optimality.
Our goal now is to show that $s \notin \mathsf{bestExits}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X})$ which would lead to a contradiction to the assumption that $s \in \mathsf{bestExits}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X})$, showing that at a best exit there cannot exist only optimal strategies that fulfill the same properties as ρ_M.
To estimate the best exit value from $\mathcal{X}$ under $\mathsf{V}_{\mathscr{R}}$, all exists from $s \in \mathcal{X}$ have to be estimated. By Eq. (6) and Eq. (7), we know that the part of the EC $\mathcal{X}$ that is reachable with (ρ_H, σ_1), say $\mathcal{X}' \subset \mathcal{X}$, attains a higher value than the one that is reachable with (ρ_H, σ_2).
More formally, the following holds. Let $\hat{s} \in \max_{s' \in \mathcal{X}'} \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s')$. At $\hat{s}$ leaving $\mathcal{X}'$ has to be possible, because otherwise, the values at all $s' \in \mathcal{X}'$ would be equal to 0, which would be a contradiction to the assumption that $\mathcal{X} \subseteq \mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}})$. Then, the following chain of equations holds.

$$\begin{aligned}
\mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) &= \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma) && (\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \emptyset) \\
&= \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho_M, \sigma) && (\rho_M \text{ is optimal under } \mathsf{V}_{\mathscr{R}}) \\
&= \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho_M, \sigma_1) && (\sigma_1 \text{ is optimal under } \mathsf{V}_{\mathscr{R}} \text{ with respect to } \rho_M) \\
&< \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho_H, \sigma_1). && (\text{Under } \sigma_1, \rho_H \text{ is optimal by Eq. (6)}) \\
&= \sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathsf{Post}(s, \rho_H, \sigma_1)} \mathsf{V}_{\mathscr{R}}(s') \cdot \delta(s, \mathsf{a}, \mathsf{b}) \cdot \rho(\mathsf{a}) \cdot \sigma(\mathsf{b}) && (\text{Unfolding the def. of } \mathfrak{B}) \\
&\leq \sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathsf{Post}(s, \rho_H, \sigma_1)} \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, \hat{s}) \cdot \delta(s, \mathsf{a}, \mathsf{b}) \cdot \rho(\mathsf{a}) \cdot \sigma(\mathsf{b}) \\
& && (\text{At } \hat{s} \text{ one can attain the highest value upon leaving}) \\
&= \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, \hat{s}). && (\text{Everything sums-up to 1})
\end{aligned}$$

Thus, under $\mathsf{V}_{\mathscr{R}}$, the state $\hat{s}$ attains a higher value upon leaving than s. This is a contradiction to the assumption that $s \in \mathsf{bestExits}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X})$.

Case (III.b) There exists $\rho_L \in \mathcal{R}_L(\mathcal{X}, s)$ that is optimal under $\mathsf{V}_{\mathscr{R}}$ and it holds that $\mathsf{Supp}(\rho_L) \cap \bigcup_{\rho \in \mathsf{Hazard}_{\mathsf{U}}(\mathcal{X}, s)} \mathsf{Supp}(\rho) = \emptyset$.
Then, ρ_L must belong to the set of deflating strategies under U, as it satisfies both conditions posed by the definition of deflating strategies (see Def. 25).

If $\mathsf{Trap}_{\mathsf{U}}(\mathcal{X}, s) \neq \emptyset$, then the following chain of equations holds.

$$\begin{aligned}
\mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) &= \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \sigma(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma) && \text{(By Def. 16 in case } \mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \emptyset\text{)} \\
&= \inf_{\sigma \in \sigma(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho_L, \sigma) && (\rho_L \text{ is optimal}) \\
&\leq \sup_{\rho \in \mathsf{Defl}_{\mathsf{U}}(\mathcal{X}, s)} \inf_{\sigma \in \sigma(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma) && (\{\rho_L\} \subseteq \mathsf{Defl}_{\mathsf{U}}(\mathcal{X}, s)\}) \\
&\leq \sup_{\rho \in \mathsf{Defl}_{\mathsf{U}}(\mathcal{X}, s)} \inf_{\sigma \in \mathsf{Trap}_{\mathsf{U}}(\mathcal{X}, s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma) && (\mathsf{Trap}_{\mathsf{U}}(\mathcal{X}, s) \subseteq \mathcal{S}(s)) \\
&\leq \sup_{\rho \in \mathsf{Defl}_{\mathsf{U}}(\mathcal{X}, s)} \inf_{\sigma \in \mathsf{Trap}_{\mathsf{U}}(\mathcal{X}, s)} \mathfrak{B}(\mathsf{U})(s, \rho, \sigma) && (\mathfrak{B} \text{ is order-preserving}) \\
&= \mathsf{exitVal}_{\mathsf{U}}(\mathcal{X}, s).
\end{aligned}$$

If $\mathsf{Trap}_{\mathsf{U}}(\mathcal{X}, s) = \emptyset$, then the following chain of equations holds.

$$\begin{aligned}
\mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) &= \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \sigma(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma) && \text{(By Def. 16 in case } \mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \emptyset\text{)} \\
&= \inf_{\sigma \in \sigma(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho_L, \sigma) && (\rho_L \text{ is optimal}) \\
&\leq \sup_{\rho \in \mathsf{Defl}_{\mathsf{U}}(\mathcal{X}, s)} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\mathsf{U})(s, \rho, \sigma) && (\mathfrak{B} \text{ is order-preserving}) \\
&= \mathsf{exitVal}_{\mathsf{U}}(\mathcal{X}, s).
\end{aligned}$$

Case (IV): $\mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) \neq \emptyset$ and $\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) \neq \emptyset$.

It holds that $s \in \mathsf{bestExits}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X})$, thus there exist no other state at which leaving $\mathcal{X}$ attains better value. By Lem. 49 we have that for all $s' \in \mathsf{bestExits}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X})$ it holds that $\mathsf{V}_{\mathscr{R}}(s') \leq \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s')$. Consequently, at a best exit it can only hold that $\mathsf{V}_{\mathscr{R}}(s') = \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s')$ because otherwise $\mathsf{V}_{\mathscr{R}}$ would not be a fixpoint of $\mathfrak{B}$. Therefore, the following chain of equations holds at s.

$$\begin{aligned}
\mathsf{V}_{\mathscr{R}}(s) &= \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma) && (\mathsf{V}_{\mathscr{R}} \text{ is fixpoint of } \mathfrak{B}) \\
&= \sup_{\rho \in \mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(s)} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma) && (\mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) \text{ are optimal under } \mathsf{V}_{\mathscr{R}}) \\
&= \sup_{\rho \in \mathsf{Hazard}_{\mathsf{V}_{\mathscr{R}}}(s)} \inf_{\sigma \in \mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma) && (\mathsf{Trap}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) \text{ are optimal under } \mathsf{V}_{\mathscr{R}}) \\
&= \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s).
\end{aligned}$$

(By Lem. 49 we have $\mathsf{V}_{\mathscr{R}}(s) \leq \mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s)$ however only "=" can hold as $\mathsf{V}_{\mathscr{R}}$ is fixpoint of $\mathfrak{B}$)

Thus, this case is equal to Case (III) where $\mathsf{exitVal}_{\mathsf{V}_{\mathscr{R}}}(\mathcal{X}, s) = \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\mathsf{V}_{\mathscr{R}})(s, \rho, \sigma)$, as the assumptions of cases (I) and (II) are not possible.

□

Lemma 51 (Existence of maximal BECs)**.** *Given a CSG* G *and let* $\mathsf{C} \subseteq (\mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}}))$ *be an EC and let* U *be a valuation. If under* U *there exists a BEC* $\mathcal{X} \subseteq \mathsf{C}$*, then there also exists a maximal BEC, i.e. there exists a BEC* $\mathcal{X}^{\mathsf{max}} \subseteq \mathsf{C}$ *such that* $\mathcal{X} \subseteq \mathcal{X}^{max}$ *and it holds that* $\mathcal{X}^{max} \cup \{s\}$ *is not a BEC for all* $s \in \mathsf{C} \setminus \mathcal{X}^{\mathsf{max}}$.

Proof. To prove the lemma it suffices to show the following: Let $\mathcal{X}_1 \subseteq (\mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}}))$ and $\mathcal{X}_2 \subseteq (\mathsf{S} \setminus (\mathsf{T} \cup \mathsf{W}_{\mathscr{S}}))$ be two ECs that are BEC under U with $\mathcal{X}_1 \cap \mathcal{X}_2 \neq \emptyset$. Then, it holds that $\mathcal{X}_1 \cup \mathcal{X}_2$ is also a BEC.

Let $s^{\cap} \in (\mathcal{X}_1 \cap \mathcal{X}_2)$, $s' \in \mathcal{X}_1$, and $s'' \in \mathcal{X}_2$. Since $\mathcal{X}_1$ is a BEC at each state $s \in \mathcal{X}_1$ it holds that $\mathsf{Hazard}_{\mathsf{U}}(\mathcal{X}_1, s) \neq \emptyset$ and $\mathsf{Trap}_{\mathsf{U}}(\mathcal{X}_1, s) \neq \emptyset$ for all $s \in \mathcal{X}_1$. Further, since $\mathcal{X}_1$ is an EC, there exists a play $s_0 s_1 \ldots$ such that $s_0 = s'$ and $s_n = s^{\cap}$ for some n and for all $0 \leq i < n$ it holds that $s_{i+1} \in \mathsf{Post}(s_i, \rho_i^*, \sigma_i^*)$ where $\rho_i^* \in \mathsf{Hazard}_{\mathsf{U}}(\mathcal{X}_1, s_i)$ and $\sigma_i^* \in \mathsf{Trap}_{\mathsf{U}}(\mathcal{X}_1, s_i)$.

Similarly, since $\mathcal{X}_2$ is also a BEC, there exists a play $s'_0 s'_1 \ldots$ such that $s'_0 = s^{\cap}$ and $s'_m = s''$ for some m and for all $0 \leq j < m$ it holds that $s'_{j+1} \in \mathsf{Post}(s'_j, \rho'_j, \sigma^*_j)$ where $\rho'_j \in \mathsf{Hazard}_{\mathsf{U}}(\mathcal{X}_2, s'_j)$ and $\sigma'_j \in \mathsf{Trap}_{\mathsf{U}}(\mathcal{X}_2, s'_j)$.

Therefore, for any $s, s' \in \mathcal{X}_1 \cup \mathcal{X}_2$ there exist a play $s''_0 s''_1 \ldots$ such that $s''_0 = s'$ and $s''_l = s''$ for some l and for all $0 \leq k < l$ it holds that $s''_{k+1} \in \mathsf{Post}(s''_k, \rho''_k, \sigma''_k)$ where $\rho''_k \in \mathsf{Hazard}_{\mathsf{U}}(\mathcal{X}_1 \cup \mathcal{X}_2, s''_k)$ and $\sigma''_k \in \mathsf{Trap}_{\mathsf{U}}(\mathcal{X}_1 \cup \mathcal{X}_2, s''_k)$.

Thus, we have shown that the $\mathcal{X}_1 \cup \mathcal{X}_2$ is an EC. Further, since at all states of $\mathcal{X}_1$ and $\mathcal{X}_2$ the two conditions (i) and (ii) of Def. 18 are fulfilled, also $\mathcal{X}_1 \cup \mathcal{X}_2$ fulfills them. Consequently, $\mathcal{X}_1 \cup \mathcal{X}_2$ is a BEC which in turn proves that there exist maximal BECs. □

Now that we have proven that maximal BECs indeed exist, the next step is to prove the correctness of FIND_MBECs, i.e. the algorithm that can find maximal BECs.

Lemma 30 (FIND_MBECs is correct— Proof in App. C-C)**.** *For a CSG, a MEC* C *and a valid upper bound* U*, it holds that* $\mathcal{X} \in \mathsf{FIND\_MBECs}(\mathsf{G}, \mathsf{C}, \mathsf{U})$ *if and only if* $\mathcal{X}$ *is a BEC in* C *and there exists no* $T \subseteq \mathsf{C}$ *that is a BEC and* $\mathcal{X} \subsetneq T$*.*

Proof. We prove the lemma in two steps. First, if for a set of states $\mathcal{X} \subseteq \mathsf{C}$ it holds that $\mathcal{X} \in \mathsf{FIND\_MBECs}(\mathsf{G}, \mathsf{C}, \mathsf{U})$ then $\mathcal{X}$ is a maximal BEC. Second, we will show that if a set of states $\mathcal{X} \subseteq \mathsf{C}$ is a maximal BEC, then $\mathcal{X} \in \mathsf{FIND\_MBECs}(\mathsf{G}, \mathsf{C}, \mathsf{U})$.

1. Direction "⇒" Let $\mathcal{X} \in \mathsf{FIND\_MBEC}(\mathsf{G}, \mathsf{C}, \mathsf{U})$. We need to show that $\mathcal{X}$ is a maximal BEC.
Let $B \coloneqq \{s \in \mathsf{C} \mid \mathsf{Hazard}_{\mathsf{U}}(\mathsf{C}, s) \neq \emptyset\}$. Since, $\mathcal{X} \in \mathsf{FIND\_MBEC}(\mathsf{G}, \mathsf{C}, \mathsf{U})$, then $\mathcal{X} \subseteq B$.
We show that $\mathcal{X}$ is a maximal BEC in C via a contradiction. In particular, we consider the following two cases:

Case (i) $\mathcal{X}$ is maximal EC in B but not BEC in C; and
Case (ii) $\mathcal{X}$ is a BEC in C but not a maximal one.

Case (i) Assume towards a contradiction that $\mathcal{X}$ is a maximal EC in B but not BEC in C. Then there exists $s' \in \mathcal{X}$ such that $\mathsf{Hazard}_{\mathsf{U}}(\mathcal{X}, s') = \emptyset$. Thus, every strategy $\rho' \in \mathcal{R}(s')$ must violate at least one of the three conditions of Def. 16. We write out the negatations.

(a) $\rho' \notin \mathcal{R}_{\overline{L}}(\mathcal{X}, s')$, i.e. the strategy is leaving $\rho' \in \mathcal{R}_L(\mathcal{X}, s)$.
(b) We do not have $\mathcal{R}(s') \setminus \{\rho'\} \preceq_{\mathcal{S}(s')} \{\rho'\}$. By contraposition of Lem. 14, this implies $\{\rho'\} \prec_{\mathcal{S}(s')} \mathcal{R}(s') \setminus \{\rho'\}$, so the strategy is sub-optimal.
(c) We do not have $\mathcal{R}_L(\mathcal{X}, s') \prec_{\mathcal{S}(s')} \{\rho'\}$. By Lem. 14, this implies $\{\rho'\} \preceq_{\mathcal{S}(s')} \mathcal{R}_L(\mathcal{X}, s')$, i.e. there are leaving strategies that are not worse than ρ'. Note that in particular, this implies that $\mathcal{R}_L(\mathcal{X}, s')$ is non-empty.

Our assumption gives us the disjunction over the three violated conditions. We proceed by a case distinction, always assuming that each strategy violates a certain condition, which allows us to prove our goal, or, if there exists a strategy satisfying the condition, we continue with the next one.

Case "all strategies satisfy (a)". In this case all strategies of player $\mathscr{R}$ must be leaving $\mathcal{X}$, i.e. $\mathcal{R}_L(\mathcal{X}, s') = \mathcal{R}(s')$. Thus, for all $\sigma \in \mathcal{S}(s')$ and for all $\rho \in \mathcal{R}(s')$ there exists $\hat{s} \in \mathsf{Post}(s', \rho, \sigma)$ such that $\hat{s} \notin \mathcal{X}$. Consequently, $\mathcal{X}$ is not an EC which is a contradiction to the assumption that $\mathcal{X}$ is an EC.

Case "some strategy violates (a) and satisfies (c)". We assume that ρ' violates (a) but satisfies condition (c). Thus, Player $\mathscr{R}$ has a non-leaving strategy, however it is not optimal. In other words, Player $\mathscr{R}$ has an optimal strategy $\rho \in \mathcal{R}(s')$ such that there exists $\hat{s} \in \mathsf{Post}(s', \rho, \sigma)$ such that $\hat{s} \notin \mathcal{X}$. Then, using the same argumentation as in the previous case we can derive a contradiction to the assumption that $\mathcal{X} \in \mathsf{FIND\_MBEC}(\mathsf{G}, \mathsf{C}, \mathsf{U})$.

Case "some strategy violate (a) and (c) but satisfy (b)". We assume that ρ' violates (a) and (c) but satisfies condition (b). Then, strategy ρ' is dominated by another non-leaving strategy, say $\rho \in \mathcal{R}_{\overline{L}}(\mathcal{X}, s')$, that in turn has to violate one of the two previous cases.

Case (ii) Now, assume towards a contradiction that $\mathcal{X}$ is a BEC in C but not maximal. Since $\mathcal{X} \in \mathsf{FIND\_MECs}(\mathsf{G}, B)$, $\mathcal{X}$ is maximal BEC in B. Then, due to the case assumption that $\mathcal{X}$ is not maximal, there has to exist a set of states $\hat{\mathsf{S}} \subseteq \mathsf{C} \setminus B$ such that $\mathcal{X} \cup \hat{\mathsf{S}}]$ is a maximal BEC in C. However, since for all $s \in \mathsf{C} \setminus B$ it holds that $\mathsf{Hazard}_{\mathsf{U}}(\mathsf{C}, s) = \emptyset$, $\mathcal{X} \cup \hat{\mathsf{S}}$ cannot be an EC in C which is a contradiction to the assumption that $\mathcal{X}$ is an EC in C.
Overall, we have shown that $\mathcal{X}$ is a BEC and it is also maximal.

2. Direction "⇐" Let $\mathcal{X} \subseteq \mathsf{C}$ be a maximal BEC in the MEC C. We need to show that $\mathcal{X} \in \mathsf{FIND\_MBEC}(\mathsf{G}, \mathsf{C}, \mathsf{U})$ holds. Let $B \coloneqq \{s \in \mathsf{C} \mid \mathsf{Hazard}_{\mathsf{U}}(\mathsf{C}, s) \neq \emptyset\}$. Assume towards a contradiction $\mathcal{X} \notin \mathsf{FIND\_MBEC}(\mathsf{G}, \mathsf{C}, \mathsf{U})$. Then, we distinguish two cases: (i) $\exists s \in \mathcal{X}$ such that $s \notin B$, or (ii) $\mathcal{X} \notin \mathsf{FIND\_MECs}(\mathsf{G}, B)$.

Case (i) If $\exists s \in \mathcal{X}$ such that $s \notin B$, then it holds that $\mathsf{Hazard}_{\mathsf{U}}(\mathsf{C}, s) = \emptyset$, by Claim 6 we also have $\mathsf{Hazard}_{\mathsf{U}}(\mathcal{X}, s) = \emptyset$ as $\mathcal{X} \subseteq \mathsf{C}$, a contradiction.

Case (ii) In this case it holds that $\forall s \in \mathcal{X}$ we have that $s \in B$, however, $\mathcal{X} \notin \mathsf{FIND\_MECs}(\mathsf{G}, B)$. We can again distinguish three cases: (ii.a) $\exists \mathcal{X}' \in \mathsf{FIND\_MECs}(\mathsf{G}, B)$ such that $\mathcal{X}' \subsetneq \mathcal{X}$; and (ii.b) $\exists \mathcal{X}' \in \mathsf{FIND\_MECs}(\mathsf{G}, B)$ such that $\mathcal{X}' \cap \mathcal{X} \neq \emptyset$ and $\exists s' \in \mathcal{X}' \setminus \mathcal{X}$; and (ii.c) $\forall \mathcal{X}' \in \mathsf{FIND\_MECS}(\mathsf{G}, B)$ it holds that $\mathcal{X}' \cap \mathcal{X} = \emptyset$.

Case (ii.a) Then there is a state in $\mathcal{X} \setminus B$, a contradiction to the assumption that $\mathcal{X}$ is a maximal BEC in B.

Case (ii.b) By "1. Direction "⇒" " we know that $\mathcal{X}'$ is a maximal BEC. In particular, for all $s \in \mathcal{X}'$ it holds that $\mathsf{Hazard}_{\mathsf{U}}(\mathcal{X}', s) \neq \emptyset$. Then, by Lem. 51 we also know that since $\mathcal{X}' \cap \mathcal{X} \neq \emptyset$, then $\mathcal{X}' \cup \mathcal{X}$ has to be also a BEC. However, this is a contradiction to the assumption that $\mathcal{X}$ is already a maximal BEC.

Case (ii.c) Since $\forall \mathcal{X}' \in \mathsf{FIND\_MECS}(\mathsf{G}, B)$ it holds that $\mathcal{X}' \cap \mathcal{X} = \emptyset$ then, for all $s' \in \mathcal{X}$ we have $s' \notin B$. However, $\mathcal{X}$ is a maximal BEC in the MEC C by assumption and therefore for all $s \in \mathcal{X}$ it holds that $\mathsf{Hazard}_{\mathsf{U}}(\mathcal{X}, s) \neq \emptyset$. Further, since $\mathcal{X} \subseteq \mathsf{C}$, by Claim 6 it also holds that $\mathsf{Hazard}_{\mathsf{U}}(\mathsf{C}, s) \neq \emptyset$. Consequently, such s' cannot exist and therefore such $\mathcal{X}'$ cannot exist, a contradiction.

Claim 6: For any ECs $\mathsf{C} \subseteq \mathsf{C}'$ we have $\mathsf{Hazard}_{\mathsf{U}}(\mathsf{C}, s) \subseteq \mathsf{Hazard}_{\mathsf{U}}(\mathsf{C}', s)$.

Proof. Let $\rho \in \mathcal{R}_{\overline{L}}(\mathsf{C}, s)$. We need to show that $\rho \in \mathcal{R}_{\overline{L}}(\mathsf{C}', s)$ holds.
First, strategy ρ satisfies condition (i) of Def. 16, thus, there has to exist $\hat{\sigma} \in \mathcal{S}(s)$, such that $(s, \rho, \hat{\sigma})$ $\mathsf{staysIn}$ C holds. Since $\mathsf{C} \subseteq \mathsf{C}'$, it also has to be true that $(s, \rho, \hat{\sigma})$ $\mathsf{staysIn}$ C'. Consequently, ρ satisfies condition (i) of Def. 16 with respect to C'.
Second, ρ satisfies condition (ii) of Def. 16, i.e. ρ is optimal. As the optimality of a strategy is independent of any set of states this condition is satisfied anyway.
Third, ρ satisfies condition (iii) of Def. 16, i.e. all strategies that are leaving with respect to C are sub-optimal. More precisely, it holds that $\mathcal{R}_L(\mathsf{C}, s) \prec_{\mathcal{S}(s)} \{\rho\}$. Since $\mathsf{C} \subseteq \mathsf{C}'$ holds, there cannot exist an optimal strategy that is leaving with respect to the greater set of states C'. Consequently, ρ also satisfies condition (iii) of Def. 16 with respect to C'.
Consequently, ρ satisfies all conditions posed by Def. 16 with respect to C', so it holds that $\mathsf{Hazard}_{\mathsf{U}}(\mathsf{C}, s) \subseteq \mathsf{Hazard}_{\mathsf{U}}(\mathsf{C}', s)$. □

▲

□

D. Soundness and Completeness

In order to prove the soundness and completeness of Alg. 1 with Alg. 3 as DEFLATE routine, we need to prove that the sequence of lower and upper bounds converges to a fixpoint. Therefore, before we can prove Thm. 37, first we need to prove the following lemma.

Lemma 52 (Alg. 1 converges to a fixpoint)**.** *The BVI algorithm (Alg. 1) converges to a fixpoint, i.e.* $\lim_{k\to\infty}(\mathfrak{B}^k(\mathsf{L}^0), (\mathfrak{D} \circ \mathfrak{B})^k\left(\mathsf{U}^0\right)) = (\mathfrak{B}(\lim_{k\to\infty} \mathfrak{B}^k(\mathsf{L}^0)), (\mathfrak{D} \circ \mathfrak{B})(\lim_{k\to\infty}(\mathfrak{D} \circ \mathfrak{B})^k\left(\mathsf{U}^0\right)))$.

Proof. We consider the domain $\mathbb{V} := [0,1]^{|\mathsf{S}|} \times [0,1]^{|\mathsf{S}|}$, i.e. every element consists of two vectors of real numbers, the under- and over-approximation. The bottom element of the domain, denoted by $\bot$, is $(\vec{0}, \vec{1})$, where for $a \in [0,1]$, $\vec{a}$ denotes the function that assigns a to all states. We further restrict the domain to exclude elements of the domain that are trivially irrelevant for the computation. In particular, we exclude all tuples (L, U) where $\mathsf{L}(s) < 1$ for a target state $s \in \mathsf{T}$ or $\mathsf{U}(s) > 0$ for a state with no path to the target state $s \in \mathsf{W}_{\mathscr{S}}$. Then the bottom element is $\bot = (\mathsf{L}^0, \mathsf{U}^0)$, i.e. the vector that we have before the first iteration of the main loop of Alg. 1. Concretely, $\mathsf{L}^0(s)$ is 1 for all $s \in \mathsf{T}$, i.e. target states, and 0 everywhere else, and $\mathsf{U}^0(s)$ is 0 for all $s \in \mathsf{W}_{\mathscr{S}}$, i.e. states where $\mathscr{S}$ can surely win, and 1 everywhere else.

We define a comparator $\sqsubseteq$ on $\mathbb{V}$, to compare two elements of the domain. We write $(\mathsf{L}^k, \mathsf{U}^k) \sqsubseteq (\mathsf{L}^{k+1}, \mathsf{U}^{k+1})$ if and only if both $\mathsf{L}^k \leq \mathsf{L}^{k+1}$ and $\mathsf{U}^k \geq \mathsf{U}^{k+1}$ hold with component-wise comparison. Intuitively, $(\mathsf{L}^k, \mathsf{U}^k) \sqsubseteq (\mathsf{L}^{k+1}, \mathsf{U}^{k+1})$ holds if $(\mathsf{L}^{k+1}, \mathsf{U}^{k+1})$ is a more precise approximation than $(\mathsf{L}^k, \mathsf{U}^k)$. The comparator $\sqsubseteq$ induces a complete partial order over the domain, since we have a bottom element and every direct subset has a supremum; the latter claim holds, because $\sqsubseteq$ reduces to component-wise comparison between real numbers from [0,1], where suprema exist. For more details on the definition of directed set and complete partial orders, we refer to [13].

Alg. 1 first applies the Bellman operator on the over- and under-approximation and subsequently applies the deflate operator on the over-approximation (i.e. upper bound). Thus, the operator that mimics the behavior of the algorithm is the following $\mathsf{BVI}(\mathsf{L}^k, \mathsf{U}^k) = (\mathfrak{B}(\mathsf{L}^k), (\mathfrak{D} \circ \mathfrak{B})(\mathsf{U}^k))$.

From Thm. 5 we know that the under-approximation converges. Also, by Lem. 36 we know that $(\mathfrak{D} \circ \mathfrak{B})$ is order-preserving. Thus, for the final argument it remains to show that $\mathfrak{D}$ is (Scott-)continuous.

A map is (Scott-)continuous if, for every directed set D in $(\mathsf{proj}_2(\mathbb{V}), \sqsubseteq)$, the subset $\mathfrak{D}(D)$ of $(\mathsf{proj}_2(\mathbb{V}), \sqsubseteq)$ is directed, and $\mathfrak{D}(\sup D) = \sup \mathfrak{D}(D)$. Let $s \in \mathsf{S}$, if s, under $\sup D$, does not belong to any BEC, then $\mathfrak{D}(\sup D)(s) = \sup D(s)$. Thus, we proceed with the assumption that under the valuation $\sup D$, $s \in \mathcal{X} \subseteq \mathsf{S}$ such that $\mathcal{X}$ is a BEC.

Let $\mathsf{Hazard}_{\sup D}(\mathcal{X}, s)$ be the set of hazardous strategies and let $\mathsf{Trap}_{\sup D}(\mathcal{X}, s)$ be the set of suitable counter strategies for player $\mathscr{S}$ (see Def. 18). Since $\mathcal{X}$ is a BEC both sets are non-empty, i.e. $\mathsf{Hazard}_{\sup D}(\mathcal{X}, s) \neq \emptyset$ and $\mathsf{Trap}_{\sup D}(\mathcal{X}, s) \neq \emptyset$, at all $s \in \mathcal{X}$. Thus, $\mathfrak{D}(\sup D)(s) = \min(\sup D(s), \mathsf{bestExitVal}_{\sup D}(\mathcal{X}))$. For the sake of readability let $\mathcal{R}'(s) := \mathsf{Dist}(\Gamma_{\mathscr{R}}(s) \setminus \bigcup_{\rho'' \in \mathsf{Hazard}_{\sup D}(\mathcal{X}, s)} \mathsf{Supp}(\rho''))$ and $\mathcal{S}'(s) := \mathsf{Trap}_{\sup D}(\mathcal{X}, s)$ for some $s \in \mathcal{X}$. Then, the following chain of equations holds for $s \in \mathcal{X}$.

$$
\begin{aligned}
&\mathfrak{D}(\sup D)(s) = \min(\sup D(s), \mathsf{bestExitVal}_{\sup D}(\mathcal{X})) \\
&= \min(\sup D(s), \max_{s' \in \mathcal{X}} \mathsf{exitVal}_{\sup D}(\mathcal{X}, s')) && \text{(By Def. 28 (best exit value))} \\
&= \min(\sup D(s), \max_{s' \in \mathcal{X}} \sup_{\rho \in \mathcal{R}'(s')} \inf_{\sigma \in \mathcal{S}'(s')} \mathfrak{B}(\sup_{d \in D} d)(s', \rho, \sigma)) && \text{(By Def. 27 (exit value))} \\
&= \min(\sup D(s), \max_{s' \in \mathcal{X}} \sup_{d \in D} \sup_{\rho \in \mathcal{R}'(s')} \inf_{\sigma \in \mathcal{S}'(s')} \mathfrak{B}(d)(s', \rho, \sigma)) && \text{(Bellman operator is Scott-continuous (see proof Theorem 9))}
\end{aligned}
$$

$$= \min(\sup D(s), \sup_{d\in D} \max_{s'\in\mathcal{X}} \sup_{\rho\in\mathcal{R}'(s')} \inf_{\sigma\in\mathcal{S}'(s')} \mathfrak{B}(d)(s',\rho,\sigma)) \qquad \text{(By Claim 7)}$$

$$= \min(\sup D(s), \sup_{d\in D} \max_{s'\in\mathcal{X}} \mathsf{exitVal}_d(\mathcal{X}, s')) \qquad \text{(By Def. 27 (exit value)))}$$

$$= \min(\sup D(s), \sup_{d\in D} \mathsf{bestExitVal}_d(\mathcal{X})) \qquad \text{(By Def. 28 (best exit value))}$$

$$= \sup_{d\in D} \min(d(s), \mathsf{bestExitVal}_d(\mathcal{X})) \qquad \text{(min is Scott-continuous)}$$

$$= \sup_{d\in D} \mathfrak{D}(d)(s). \qquad \text{(Since } \mathfrak{D} \text{ is the deflate operator)}$$

Claim 7: Since $\max_{s'\in\mathcal{X}} \mathsf{exitVal}_{\sup D}(\mathcal{X}, s')$ exists the following chain of equations holds:

$$\begin{aligned}
&\sup_{d\in D} \max_{s'\in\mathcal{X}} \sup_{\rho\in\mathcal{R}'(s')} \inf_{\sigma\in\mathcal{S}'(s')} \mathfrak{B}(d)(s',\rho,\sigma) \\
&= \sup_{d\in D} \sup_{s'\in\mathcal{X}} \sup_{\rho\in\mathcal{R}'(s')} \inf_{\sigma\in\mathcal{S}'(s')} \mathfrak{B}(d)(s',\rho,\sigma) && \text{(As the maximum exists)} \\
&= \sup_{(d,s')\in(D\times\mathcal{X})} \sup_{\rho\in\mathcal{R}'(s')} \inf_{\sigma\in\mathcal{S}'(s')} \mathfrak{B}(d)(s',\rho,\sigma) \\
&= \sup_{s'\in\mathcal{X}} \sup_{d\in D} \sup_{\rho\in\mathcal{R}'(s')} \inf_{\sigma\in\mathcal{S}'(s')} \mathfrak{B}(d)(s',\rho,\sigma) \\
&= \max_{s'\in\mathcal{X}} \sup_{d\in D} \sup_{\rho\in\mathcal{R}'(s')} \inf_{\sigma\in\mathcal{S}'(s')} \mathfrak{B}(d)(s',\rho,\sigma).
\end{aligned}$$

▲

We have shown that $\mathfrak{B}$ and $\mathfrak{D}$ are both Scott-continuous, thus the composition $(\mathfrak{D}\circ\mathfrak{B})$, i.e. the sequential application of the operators $\mathfrak{B}$ and $\mathfrak{D}$, is also continuous [44]. Further, $\mathbb{V}$ is a complete partial order. Therefore, Kleen's Fixpoint Theorem [13, Theorem 8.15] is applicable.

Then, we know that

$$\lim_{k\to\infty} (\mathfrak{D}\circ\mathfrak{B})^k(\bot) = \sup_{k\geq 0} (\mathfrak{D}\circ\mathfrak{B})^k(\bot) \qquad (\star)$$

holds and by the theorem we know that the fixpoint exists and is given by $\sup_{k\geq 0}(\mathfrak{D}\circ\mathfrak{B})^k(\bot)$. Now, we can finally conclude:

$$\begin{aligned}
(\mathfrak{D}\circ\mathfrak{B})(\lim_{k\to\infty} (\mathfrak{D}\circ\mathfrak{B})^k(\bot)) &\overset{(\star)}{=} (\mathfrak{D}\circ\mathfrak{B})(\sup_{k\geq 0}(\mathfrak{D}\circ\mathfrak{B})^k(\bot)) \\
&= \sup_{k\geq 0}(\mathfrak{D}\circ\mathfrak{B})((\mathfrak{D}\circ\mathfrak{B})^k(\bot)) && \text{(since } (\mathfrak{D}\circ\mathfrak{B}) \text{ is continuous)} \\
&= \sup_{k\geq 1}(\mathfrak{D}\circ\mathfrak{B})^k(\bot) \\
&= \sup_{k\geq 0}(\mathfrak{D}\circ\mathfrak{B})^k(\bot) && \text{(since } \bot \sqsubseteq (\mathfrak{D}\circ\mathfrak{B})^k(\bot) \text{ for all } k) \\
&= \lim_{k\to\infty} (\mathfrak{D}\circ\mathfrak{B})^k(\bot). && \text{by } (\star)
\end{aligned}$$

□

Now we can prove the following theorem.

Theorem 37 (Soundness and completeness - Proof in App. C-D)**.** *For CSGs Alg. 1, using Alg. 3 as* DEFLATE*, produces monotonic sequences* L *under- and* U *over-approximating* $\mathsf{V}_{\mathscr{R}}$*, and terminates for every* $\varepsilon > 0$*.*

Proof. We denote by L^k and U^k the lower and upper bound function after the k-th call of DEFLATE. L^k and U^k are monotonic under-, respectively, over-approximation of $\mathsf{V}_{\mathscr{R}}$ because they are updated via Bellman updates respectively $(\mathfrak{D}\circ\mathfrak{B})$-updates, which are order-preserving as shown in Lem. 36 and soundness as shown in Lem. 35 .

Since DEFLATE iterates over finite sets, the computations take a finite time. Thus, it remains to prove that the main loop of Alg. 1 terminates, i.e., for all $\varepsilon > 0$, there exists an $n \in \mathbb{N}$ such that for all $s \in \mathsf{S}$ $\mathsf{U}^n(s) - \mathsf{L}^n(s) \leq \varepsilon$. It suffices to show that $\lim_{k\to\infty} \mathsf{U}^k - \mathsf{V}_{\mathscr{R}} = 0$, because $\lim_{k\to\infty} \mathsf{L}^k = \mathsf{V}_{\mathscr{R}}$ (from e.g. [42]).

In the following let $\mathsf{U}^\star := \lim_{k\to\infty} \mathsf{U}^k$, that exists by Lemma 48, and $\Delta(s) := \mathsf{U}^\star(s) - \mathsf{V}_{\mathscr{R}}(s)$. Assume towards a contradiction that the algorithm does not converge, i.e., there exists a state $s \in \mathsf{S}$ with $\Delta(s) > 0$.

The proof is structured as follows.

- From $\Delta > 0$ we derive that there has to exist a BEC.

- The states with the maximal Δ contain BECs.
- Alg. 1 will find a BEC contained in $\mathcal{X} \subseteq \mathsf{S}$ and deflate it.
- Deflating will decrease the upper bound of that states contained in the BEC, which is a contradiction because by Lemma 52, $\mathsf{U}^\star$ is a fixpoint.

Let $\Delta^{\max} := \max_{s \in \mathsf{S}} \Delta(s)$ and let $\mathsf{C} := \{s \in \mathsf{S} \mid \Delta(s) = \Delta^{\max}\}$. If C does not contain any BECs, then the contraposition of Thm. 9 proves our goal. Thus, we continue with the assumption that C contains BECs.

Let $\mathcal{X} \subseteq \mathsf{C}$ be a BEC contained in C that Alg. 3 will eventually find and deflate. We now consider *bottom* BECs. A BEC $\mathcal{X}'$ is called bottom in $\mathcal{X}$ if none of the successors of a strategy that leaves the BEC $\mathcal{X}'$, is part of another BEC in $\mathcal{X}$. A bottom BEC can be computed by first finding the bottom MEC within $\mathcal{X}$, then identifying all BECs, ordering them topologically and finally picking the one at the end of a chain.

Let $\mathcal{X}' \subseteq \mathcal{X}$ be a bottom BEC that Algorithm 3 eventually finds. In order to deflate the BEC, we need to estimate the exit value for each $s \in \mathcal{X}'$, as defined in Definition 27. Further, $\mathsf{Hazard}_{\mathsf{U}^\star}(\mathcal{X}', s)$ is the set of hazardous strategies and since $\mathcal{X}'$ is a BEC, at all states $s \in \mathcal{X}'$, we have $\mathsf{Hazard}_{\mathsf{U}^\star}(\mathcal{X}', s) \neq \emptyset$.

We distinguish two cases: (i) $\mathsf{Trap}_{\mathsf{U}^\star}(\mathcal{X}', s) \neq \emptyset$; and (ii) $\mathsf{Trap}_{\mathsf{U}^\star}(\mathcal{X}', s) = \emptyset$.

Case (i) In this case we have that $\mathsf{Trap}_{\mathsf{U}^\star}(\mathcal{X}', s) \neq \emptyset$. Then $\mathsf{Defl}_{\mathsf{U}^\star}(\mathcal{X}', s) \neq \emptyset$ (and because $\mathcal{X} \cap \mathsf{W}_{\mathscr{S}} = \emptyset$), thus the following chain of equations holds.

$$
\begin{aligned}
\mathsf{U}^\star(s) &= \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\mathsf{U}^\star)(s, \rho, \sigma) && (\mathsf{U}^\star \text{ is a fixpoint}) \\
&= \sup_{\rho \in \mathsf{Hazard}_{\mathsf{U}^\star}(\mathcal{X}', s)} \inf_{\sigma \in \mathsf{Trap}_{\mathsf{U}^\star}(\mathcal{X}', s)} \mathfrak{B}(\mathsf{U}^\star)(s, \rho, \sigma) && (\mathsf{Hazard}_{\mathsf{U}^\star}(\mathcal{X}', s) \neq \emptyset) \\
&> \sup_{\rho \in \mathsf{Defl}_{\mathsf{U}^\star}(\mathcal{X}', s)} \inf_{\sigma \in \mathsf{Trap}_{\mathsf{U}^\star}(\mathcal{X}', s)} \mathfrak{B}(\mathsf{U}^\star)(s, \rho, \sigma) && (\mathsf{Defl}_{\mathsf{U}^\star}(\mathcal{X}', s) \text{ are sub-optimal with respect to } \mathsf{U}^\star) \\
&= \mathsf{exitVal}_{\mathsf{U}^\star}(\mathcal{X}', s) && (\text{By Def. 27})
\end{aligned}
$$

Since $\mathsf{bestExitVal}_{\mathsf{U}^\star}(\mathcal{X}') = \max_{s' \in \mathcal{X}'} \mathsf{exitVal}_{\mathsf{U}^\star}(\mathcal{X}', s')$, and above we have shown that for all $s \in \mathcal{X}'$ it holds $\mathsf{U}^\star(s) > \mathsf{exitVal}_{\mathsf{U}^\star}(\mathcal{X}', s)$, we obtain a contradiction to the assumption that $\mathsf{U}^\star$ is a fixpoint.

Case (ii) In this case it holds that $\mathsf{Trap}_{\mathsf{U}^\star}(\mathcal{X}', s) = \emptyset$. Thus, Player $\mathscr{S}$ prefers strategies that are leaving with respect to $\mathsf{Hazard}_{\mathsf{U}^\star}(\mathcal{X}, s)$. Since $\mathcal{X}'$ is a bottom BEC at least one successor state does not belong to $\mathcal{X}$. Therefore, the following chain of equations holds.

$$
\begin{aligned}
\Delta^{\max} - \mathsf{V}_{\mathscr{R}}(s) &= \mathsf{U}^\star(s) && (\text{By definition of } \Delta^{\max}) \\
&= \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \mathfrak{B}(\mathsf{U}^\star)(s, \rho, \sigma) && (\mathsf{U}^\star \text{ is a fixpoint}) \\
&= \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathsf{S}} \mathsf{U}^\star(s') \cdot \delta\big(s, \mathsf{a}, \mathsf{b}\big)\big(s'\big) \cdot \rho(\mathsf{a}) \cdot \sigma(\mathsf{b}) && (\text{Unfolding definition of } \mathfrak{B}) \\
&= \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathcal{X}} \mathsf{U}^\star(s') \cdot \delta\big(s, \mathsf{a}, \mathsf{b}\big)\big(s'\big) \cdot \rho(\mathsf{a}) \cdot \sigma(\mathsf{b}) \\
&\quad + \sum_{s'' \in \mathsf{S} \setminus \mathcal{X}} \mathsf{U}^\star(s'') \cdot \delta\big(s, \mathsf{a}, \mathsf{b}\big)\big(s''\big) \cdot \rho(\mathsf{a}) \cdot \sigma(\mathsf{b}) && (\text{Player } \mathscr{S} \text{ prefers leaving } \mathcal{X}' \text{ and thus } \mathcal{X}) \\
&= \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathcal{X}} \mathsf{U}^\star(s') \cdot \delta\big(s, \mathsf{a}, \mathsf{b}\big)\big(s'\big) \cdot \rho(\mathsf{a}) \cdot \sigma(\mathsf{b}) \\
&\quad + \sum_{s'' \in \mathsf{S} \setminus \mathcal{X}} \left(\underbrace{\Delta(s'')}_{< \Delta^{\max}} + \mathsf{V}_{\mathscr{R}}(s'') \right) \cdot \delta\big(s, \mathsf{a}, \mathsf{b}\big)\big(s''\big) \cdot \rho(\mathsf{a}) \cdot \sigma(\mathsf{b}) && (\text{By def. of } \Delta) \\
&< \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \sum_{(\mathsf{a},\mathsf{b}) \in \mathcal{A}} \sum_{s' \in \mathcal{X}} \mathsf{U}^\star(s') \cdot \delta\big(s, \mathsf{a}, \mathsf{b}\big)\big(s'\big) \cdot \rho(\mathsf{a}) \cdot \sigma(\mathsf{b}) \\
&\quad + \sum_{s'' \in \mathsf{S} \setminus \mathcal{X}} \big(\Delta^{\max} + \mathsf{V}_{\mathscr{R}}(s'')\big) \cdot \delta\big(s, \mathsf{a}, \mathsf{b}\big)\big(s''\big) \cdot \rho(\mathsf{a}) \cdot \sigma(\mathsf{b}) \\
&&& (\Delta \text{ for states outside } \mathcal{X} \text{ is strictly smaller than } \Delta^{\max}) \\
&= \mathsf{U}^\star(s) && (\text{Everything sums-up to 1})
\end{aligned}
$$

Thus, we get a contradiction to the assumption that $\mathsf{U}^\star$ is a fixpoint.

□

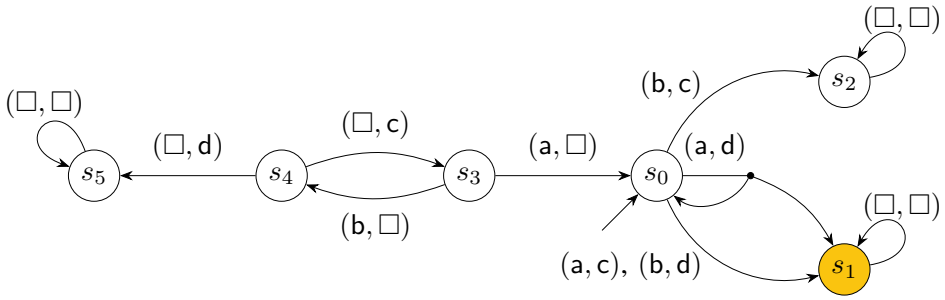

Fig. 6: Counter example for the BVI provided in [7].

APPENDIX D
COUNTER-EXAMPLES FOR PREVIOUS WORKS

A. Mistake in [9]

In [9], the authors provide an exemplary CSG that illustrates the incorrectness of the BVI algorithm presented in [7]. Fig. 6 shows the CSG. The value attainable at s_5 is 0.6 while the value attainable at s_0 is $2 - \sqrt{2}$. In our algorithm (Alg. 3) the set $\{s_4, s_3\}$ does not constitute a BEC for the valuation $\mathsf{U}(s_4) = \mathsf{U}(s_3) = \mathsf{U}(s_1) = 1$ and $\mathsf{U}(s_2) = 0$. Consequently, the while-loop of Alg. 3, which is responsible for deflating, is not executed and thus the values of all states are updated only using the Bellman operator. This yields the correct values by Thm. 9. Therefore, Alg. 3 correctly sets the valuation of states s_4 and s_3 to the value 0.6. In contrast, the algorithm presented in [7] correctly sets the value of state s_4 to 0.6 but reduces the value of state s_3 to $2 - \sqrt{2} < 0.6$.

B. Mistake in [19]

There was an attempt to fix the above problem in [19], however, also this solution is incorrect. An exemplary CSG that illustrates the incorrectness of the approach presented in [19] is the CSG illustrated in Fig. 2. While our approach correctly deflates the value of state s_0, the best exit as defined in [19] is falsely 1, i.e. the value of s_0 is never reduced.