Generalized Bidding Games: Where Bidding and Stochastic Games Meet

Ali Asadi
Institute of Science and Technology Austria, Klosterneuburg, Austria

Thomas A. Henzinger
Institute of Science and Technology Austria, Klosterneuburg, Austria

Ehsan Kafshdar Goharshady
Institute of Science and Technology Austria, Klosterneuburg, Austria

Pavol Kebis
Institute of Science and Technology Austria, Klosterneuburg, Austria

Kaushik Mallik
IMDEA Software Institute, Spain

Abstract

Two-player games on graphs are a classical framework for analyzing strategic decision making. In turn-based games, two players move a token along the edges of the graph, and the right to move the token is determined by the current vertex. In traditional bidding games – referred to as *pure* bidding games – the right to move the token is determined at each step through bidding; here we consider Richman bidding, where the winning player of a bid pays the losing player. The winner is decided based on a temporal or quantitative specification evaluated over the resulting infinite play.

In this work, we combine turn-based games and pure bidding games into *generalized* bidding games, with player-1 vertices, player-2 vertices, and bidding vertices. This natural and simple generalization of bidding games has far-reaching consequences. First, we show that, as a model, generalized bidding games are more expressive than pure bidding games, and we provide several applications. Second, and most importantly, we show that generalized Richman bidding games are structurally equivalent to simple stochastic games, a well-studied model: they are linearly interreducible to each other. As was previously known, the special case of pure Richman bidding games corresponds to random-turn games. In other words, generalized bidding games extend pure bidding games in the same way that simple stochastic games extend random-turn games. We use this connection to solve generalized Richman bidding games for temporal (parity) and quantitative (mean-payoff and discounted-sum) specifications. From a computational perspective, we establish that generalized bidding games with parity and mean-payoff specifications retain the best known upper bounds for turn-based games and pure bidding games, namely NP ∩ coNP.

Finally, we study a repair problem that asks whether bidding vertices can be assigned "owners" so as to bring the threshold budget required to win the game below a given target. This problem has direct applications in compositional policy synthesis for multi-objective settings, and we show it to be NP-complete.

2012 ACM Subject Classification Theory of computation → Algorithmic game theory; Computing methodologies → Stochastic games

Keywords and phrases Bidding Games, Stochastic Games

Digital Object Identifier 10.4230/LIPIcs.CONCUR.2026.13

Related Version *Full Version*: `https://arxiv.org/abs/2606.29420` [1]

Funding The research was partially supported by Austrian Science Fund (FWF) 10.55776/COE12, ERC CoG 863818 (ForM-SMArt), FWF-2022-SFB F8502 (SPyCoDe), ERC-2020-AdG 101020093 (VAMOS), and the RYC2024-049116-I grant funded by MICIU/AEI/10.13039/501100011033 and ESF+.

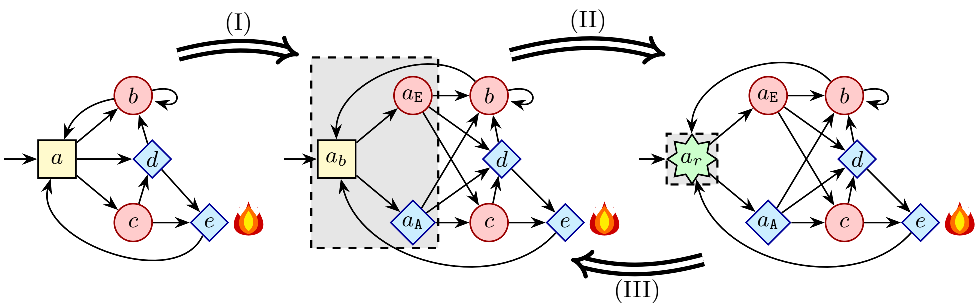

Figure 1 An overview of our translation from generalized bidding games (GBG) to simple stochastic games (SSG). LEFT: the original GBG; CENTER: the equivalent simple GBG (SGBG), i.e., a GBG whose bidding vertices have exactly two successors; RIGHT: the equivalent SSG. The yellow squares are the bidding vertices, the green star is a random vertex, the red circles are Eve's vertex, and the blue diamonds are Adam's vertices. The grey box highlights the main change in each step while going from left to right. Eve's specification is to avoid the unsafe vertex e forever.

1 Introduction

Games on graphs are central in computer science, with applications from automata theory to the synthesis of reactive systems. In the classical turn-based setting, two players, Eve and Adam, move a token on a graph, generating an infinite path; Eve wins if the path satisfies a temporal or quantitative specification, and the goal is to compute a winning strategy for her.

An important extension is *bidding games*, where turns are determined by auctions rather than fixed ownership. Players bid using finite budgets to gain control of the next move, with payments governed by a model such as first-price Richman bidding, where the players are initially allocated budgets whose sum is 1, and after each bidding, the higher bidder picks the next move and pays the bid amount to the opponent, so that the total budget in the game remains 1. Bidding games model situations where resources must be invested while making strategic decisions in the presence of opponents. Applications include online advertisement platforms [5] and decoupled control of cyber-physical systems [8].

Often, decision-making problems involve the interleaving of bidding and turn-based mechanisms, but there is no existing model that combines them in one framework. To model this, we introduce *generalized* bidding games (GBG), where bidding takes place in a given subset of vertices – called the *bidding vertices*, while each non-bidding vertex – called a *control vertex* – is either owned by Eve or by Adam. When the token lands on a bidding vertex, who moves the token is decided by (first-price Richman) bidding. When the token lands on a control vertex, no bidding takes place. Instead, it is the owner's turn to choose where the token goes next. This way, our model unifies bidding and turn-based games into a new class of games that is more general than each of the two. To mark the distinction, we refer to the traditional bidding games as *pure* bidding games or BG in short, because turns are assigned purely based on bidding.

We claim that GBGs are strictly more powerful than BGs and turn-based games. One direction is easy: GBGs are at least as expressive as BGs and turn-based games, because these are restricted variants. For the other direction, we show that the solution of GBGs requires a *quantitative* analysis (which is also sufficient as we will show in the subsequent sections), while qualitative analyses suffice for the two restricted cases.

▶ **Example 1** (GBGs versus BGs and turn-based games)**.** Consider the GBG shown in the left diagram in Figure 1, where Eve's goal is to avoid the unsafe vertex e at all times (safety). In this game, Eve wins from a if her budget is strictly more than $\frac{1}{2}$, because Adam's budget is less than $\frac{1}{2}$ (since the sum of players' budgets is 1), so that Eve can outbid Adam at a, move the token to b, and then produce the sequence b^ω, thereby surely avoiding e at all time. Dually, Eve loses from a if her budget is strictly less than $\frac{1}{2}$, because then Adam can outbid Eve, move to d, and then reach e, violating Eve's safety specification. We say that the threshold budget in a is $\frac{1}{2}$.

If instead the game were a BG, with all vertices being bidding vertices, Eve would not be able to guarantee safety in the long-run with any amount of initial budget. This is because the underlying graph is strongly connected, for which the analysis of thresholds is qualitative: if the set of unsafe vertices is non-empty, Eve loses with *any* budget, and if the set of unsafe vertices is empty, Eve can win with *any* budget [3]. In general, on strongly connected graphs, a qualitative analysis of the specification suffices for solving BGs, whereas GBGs require a more involved quantitative analysis. We formalize this claim in Appendix B.

Similarly, if the game were a turn-based game, with all vertices being control vertices, the discussion of budgets would not be necessary, and every vertex would either be winning or losing for Eve. Finding the winning vertices in any turn-based game is based on a combinatorial (qualitative) analysis of the graph, which is again less involved than the quantitative analysis necessary for GBGs. ◀

Like BGs, the analysis of GBGs revolves around the concept of *threshold budget*, which is the necessary and sufficient budget of Eve to win from a given initial vertex. For instance, the threshold budget in Example 1 was $\frac{1}{2}$. The existence of threshold budgets is a fundamental theoretical question, which is a variant of determinacy.[1] We prove that threshold budgets exist for GBGs with parity, discounted-sum, and mean-payoff specifications. Incidentally, we are the first to consider discounted-sum specifications for any form of bidding games.

Our main result is that *simple* SGBGs, where every bidding vertex has exactly two outgoing edges, are *structurally equivalent* to *simple stochastic games* (SSG), which have been extensively studied in the literature [19, 17]. By structural equivalence we refer to a simple reinterpretation of the same game graph. In SSGs, the game graph contains the usual control vertices – owned by Eve and Adam – and *random* vertices with two outgoing edges, such that from every random vertex, the token moves to one of the two successors with equal probabilities. The *value* of an SSG is the maximal probability with which Eve can fulfill her specification from the initial vertex against any strategy of Adam. We show that, by simply replacing each bidding vertex of each SGBG with a random vertex, as illustrated in Step (II) of Figure 1, the value of the resulting SSG equals 1 minus the threshold of the original GBG. The opposite direction also holds true: by replacing every random vertex of a given SSG with a bidding vertex, as illustrated in Step (III) of Figure 1, the threshold of the resulting SGBG equals 1 minus the value of the original SSG. An important consequence is that any tool developed for either SGBG or SSG supports the other out of the box, without involving any intermediate reduction step.

While the structural equivalence of SGBGs and SSGs is uniform for all specifications we consider – both qualitative specifications such as parity and quantitative specifications such as discounted-sum and mean-payoff – we need to prove the equivalence for each class of specifications separately. In particular, the different specifications require different proof techniques to establish the preservation of strategies and threshold budgets.

It was already known that a similar structural equivalence (except for discounted-sum specifications, which were not studied before in this context) exists between BGs and *random-turn games* (RTG), where RTGs are the special subclass of SSGs in which every random vertex v is succeeded by an Eve's vertex $v_{\mathtt{E}}$ and an Adam's vertex $v_{\mathtt{A}}$, and the edges originating

[1] The determinacy of zero-sum games implies every state is winning for one of the players.

from $v_{\mathtt{E}}$ and $v_{\mathtt{A}}$ connect to identical sets of vertices. To put our result into perspective, SGBGs extend BGs in the same way that SSGs extend RTGs. However, while little is known about the exact relation between SSGs and RTGs, we show in our second main result, that every GBG (independent of the out-degree of bidding vertices) can be linearly translated into a SGBG (with bidding out-degree 2). This establishes a linear reduction of all GBGs to SSGs; the reduction gadget is shown in Step (I) in Figure 1. The translation is such that the size of the resulting stochastic game is linear in the size of the original bidding game, and moreover, for parity, discounted-sum, and mean-payoff specifications, all thresholds and winning strategies remain intact. Incidentally, a similar translation is unlikely to exist for BGs, because BGs with out-degree 2 (called simple BG or SBG) are equivalent to RTGs with out-degree 2 and reduce to Markov chains, which can be solved in polynomial time; therefore, if a reduction from BGs to SBGs were possible, we would have a polynomial-time algorithm for BGs (with arbitrary out-degree) – resolving a long-standing open problem (the best known complexity for the general out-degree case is NP ∩ coNP).

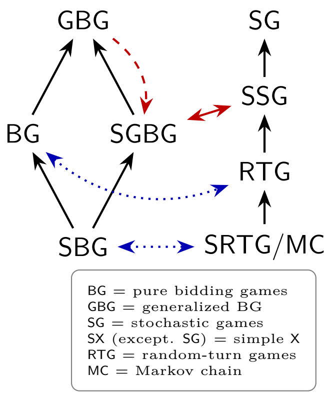

Figure 2 Connection between bidding games and stochastic games. The red arrow (our result) and dotted blue arrows (known) are structural equivalences. The red dashed arrow is our reduction (polynomial-time). The black arrows go from special cases to more general cases.

Complexity-wise, for reachability, safety, parity, and mean-payoff specifications, computing values for both SSGs and RTGs are known to be in NP ∩ coNP, and the membership in P is a long-standing open problem. Therefore, our results imply that computing thresholds of GBGs is in NP ∩ coNP for all considered specification classes. On the other hand, the best known algorithms for BGs are also in NP ∩ coNP, which is a corollary of their connection to RTGs. Summarizing, not only do GBGs subsume and extend BGs, but also they allow a seamless transition to the extensively studied SSGs, without paying any additional computational blow-up. The relationships obtained between different classes of GBGs and stochastic games is shown in Figure 2.

Our work is the first to consider discounted-payoff specifications in the context of any form of bidding games, and we show that the structural equivalence between SSGs and SGBGs also holds in this case. However, this does not provide an immediate upper complexity bound, because finding values for SSGs with discounted-sum specifications is an open problem.[2] Nonetheless, our result indicates that solving this problem for either SGBGs or SSGs will immediately provide a solution for the other category.

[2] Here, we are dealing with the quantitative satisfaction variant of the problem, aka the "quantile"

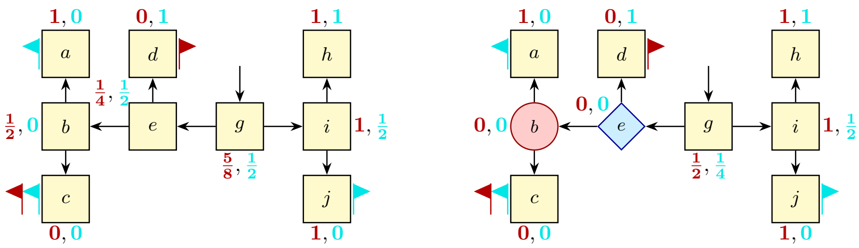

Figure 3 Application of GBGs in multi-objective control via auction-based scheduling [8]; the left and right subfigures show the solutions using BG and GBG, respectively. Eve's and Adam's objectives are to reach any of the red and the blue flags, respectively. The numbers next to the vertices represent the personal threshold budgets of Eve and Adam as if each of them were playing separately against an adversary. While composition fails in the BG (since the sum of thresholds is more than 1), it succeeds using GBGs.

Motivating Example: Application in Multi-Objective Decision-Making

Auction-based scheduling [8] provides a decoupled approach to multi-objective path planning, where independent strategies for objectives φ and φ' are combined via bidding. Each objective is solved as a zero-sum bidding game, yielding local strategies and threshold budgets θ and θ'. If $\theta + \theta' < 1$, the budget can be split so that both objectives are satisfied under composition.

A limitation is that treating the opponent as fully adversarial can make $\theta + \theta' \geq 1$. This can be alleviated by assigning permanent control of some vertices to players, lowering thresholds for critical objectives, though at the cost of added coordination; this is illustrated in Figure 3. This is a direct application of our model of GBGs.

This application leads to the *threshold repair problem*: given a GBG, a target threshold $t \in (0, 1)$, and a budget n, can we select at most n bidding vertices to assign to Eve so that her threshold drops below t? A dual version for Adam's threshold can be posed in the same way. We show this problem is NP-complete for reachability, safety, and parity specifications, and it can be encoded as a mixed-integer logical program (as shown in the full version [1]) and solved efficiently in practice.

Related Work

Bidding games have a long history, going back to the seminal work of Lazarus et al. [22] in the nineties. This work studied bidding games with reachability specifications, where various different bidding mechanisms were introduced, including Richman, poorman, taxman, etc. Later, bidding games were extended to richer classes of specifications, like temporal (parity, etc.) and quantitative specifications (mean-payoff, etc.) [4, 5], richer classes of arenas, like Markov decision processes [7], and richer bidding mechanisms, like games with discrete bidding currencies [21, 9] and games with charging [2], where the players can top-up their budgets by collecting rewards during the game. All these extensions are orthogonal to our setting, and therefore they can be combined together.

In terms of the characteristics of our GBGs, the closest models are bidding games with charging [2] and discrete bidding games [9]. As in our case, both of these models exhibit non-trivial (other than 0 or 1) thresholds for safety and parity specifications on strongly

problem, where the goal is to determine the maximum *probability* of securing a discounted-sum above a given threshold. This should not be confused with the problem of maximizing the *expected* discounted sum, for which solutions exist in the literature.

connected game graphs. However, these games introduce additional mechanisms to the bidding step, and they do not provide a clean connection to stochastic games like our approach does. For instance, in discrete-bidding games [9], resolving bidding ties play a central role while computing strategies. In bidding games with charging [2], a normalization step is required to ensure that the sum of budgets of the players remain 1 after every round of charging. In contrast, our model offers a very simple and clean extension of BGs.

2 Generalized Bidding Games

We introduce generalized bidding games as a novel model for strategic decision-making involving auctions. The two opposing players will be referred to as Eve and Adam, respectively.

▶ **Definition 2** (Game graphs with bidding and control vertices). *A game graph with bidding and control vertices is of the form $\mathcal{G} = (V, V_{\mathtt{E}}, V_{\mathtt{A}}, V_{\mathtt{b}}, E)$, where V is the finite set of vertices, $V_{\mathtt{E}}$, $V_{\mathtt{A}}$, and $V_{\mathtt{b}}$ form a partition of V, and $E \subseteq V \times V$ is the set of edges. The vertices $V_{\mathtt{E}}$ and $V_{\mathtt{A}}$ are controlled by Eve and Adam, respectively, and they will be collectively called the* control *vertices. On the other hand, the vertices $V_{\mathtt{b}}$ are called the* bidding *vertices.*

We will use E interchangeably as a relation (like defined) and as a function, where $E(v) \coloneqq \{v' \in V \mid (v, v') \in E\}$ for every $v \in V$.

Semantics. Initially, a token is placed on a vertex v^0, and some non-negative monetary budgets are allocated to the players, where Eve's initial budget is $B^0_{\mathtt{E}}$, and Adam's initial budget is $B^0_{\mathtt{A}} = 1 - B^0_{\mathtt{E}}$, so that $B^0_{\mathtt{E}} + B^0_{\mathtt{A}} = 1$. We say $(v^t, B^t_{\mathtt{E}})$ is the *configuration* at time $t \geq 0$, where v^t and $B^t_{\mathtt{E}}$ are the token's location and Eve's budget, respectively, and it will be guaranteed by the auction mechanism that Adam's budget at time t is implicitly $B^t_{\mathtt{A}} = 1 - B^t_{\mathtt{E}}$.

At configuration $(v^t, B^t_{\mathtt{E}})$, if $v^t \in V_{\mathtt{E}}$ then Eve picks $v^{t+1} \in E(v^t)$, if $v^t \in V_{\mathtt{A}}$ then Adam picks $v^{t+1} \in E(v^t)$, and otherwise, if $v^t \in V_{\mathtt{b}}$, an auction takes place, where the players use their available budgets to bid for the privilege of choosing the successor, and whoever bids higher pays the bid amount to the opponent and picks $v^{t+1} \in E(v^t)$.

Towards formalizing the above, we introduce some notation. A *path* (in $\mathcal{G}$) is a vertex sequence $v^0 v^1 \ldots$, such that for every $t \geq 0$, $(v_t, v_{t+1}) \in E$. Paths are either finite or infinite, and $\mathrm{PathsFin}^{\mathcal{G}}_v$ and $\mathrm{PathsInf}^{\mathcal{G}}_v$ respectively denote the sets of all finite and infinite paths in $\mathcal{G}$ starting at v. For every finite path $\rho = v^0 \ldots v^k$, $\mathit{last}(\rho) = v^k$. A *trace* is a (finite or infinite) sequence of configurations $(v^0, B^0_{\mathtt{E}})(v^1, B^1_{\mathtt{E}}) \ldots$ such that $v^0 v^1 \ldots$ is a path in $\mathcal{G}$, and moreover for every $t \geq 0$, if v^t is a control vertex then the budgets remain unchanged, i.e., $B^t_{\mathtt{E}} = B^{t+1}_{\mathtt{E}}$. The path *induced* by the trace $(v^0, B^0_{\mathtt{E}})(v^1, B^1_{\mathtt{E}}) \ldots$ is simply $v^0 v^1 \ldots$.

Strategies are recipes describing the next moves of the players, defined as follows.

▶ **Definition 3** (Strategies). *A strategy of Eve is a function $\pi_{\mathtt{E}}$ such that for every finite path ρ and current budget $B_{\mathtt{E}} \in [0, 1]$ of Eve, if $\mathit{last}(\rho) \in V_{\mathtt{E}}$ then $\pi_{\mathtt{E}} \colon (\rho, B_{\mathtt{E}}) \mapsto v \in E(\mathit{last}(\rho))$, and if $\mathit{last}(\rho) \in V_{\mathtt{b}}$ then $\pi_{\mathtt{E}} \colon (\rho, B_{\mathtt{E}}) \mapsto (b, v) \in [0, B_{\mathtt{E}}] \times E(\mathit{last}(\rho))$. Strategies of Adam are defined analogously. Let $\Pi_{\mathtt{E}}$ and $\Pi_{\mathtt{A}}$ be the sets of all strategies of Eve and Adam, respectively.*

Intuitively, when $\mathit{last}(\rho) \in V_{\mathtt{E}}$ (no bidding), $\pi_{\mathtt{E}}(\rho, B_{\mathtt{E}})$ prescribes the next vertex, and when $\mathit{last}(\rho) \in V_{\mathtt{b}}$, $\pi_{\mathtt{E}}(\rho, B_{\mathtt{E}})$ prescribes a bid value $b \in [0, B_{\mathtt{E}}]$ and the next vertex to pick if the bidding is won.

Suppose $\pi_{\mathtt{E}} \in \Pi_{\mathtt{E}}$ and $\pi_{\mathtt{A}} \in \Pi_{\mathtt{A}}$ are arbitrary policies of the players. Given an initial vertex v^0 and initial budget $B^0_{\mathtt{E}} \in [0, 1]$ of Eve, $\pi_{\mathtt{E}}$ and $\pi_{\mathtt{A}}$ generate a unique infinite trace as follows:

- The initial configuration is $(v^0, B_{\mathtt{E}}^0)$.
- At time $t \geq 0$, let the configuration be $(v^t, B_{\mathtt{E}}^t)$. There are three possibilities for obtaining the next configuration $(v^{t+1}, B_{\mathtt{E}}^{t+1})$:
  1. if $v^t \in V_{\mathtt{E}}$, and if $\pi_{\mathtt{E}}(v^0 \ldots v^t, B_{\mathtt{E}}^t) = v'$, then $v^{t+1} = v'$ and $B_{\mathtt{E}}^{t+1} = B_{\mathtt{E}}^t$;
  2. if $v^t \in V_{\mathtt{A}}$, and if $\pi_{\mathtt{A}}(v^0 \ldots v^t, 1 - B_{\mathtt{E}}^t) = v'$, then $v^{t+1} = v'$ and $B_{\mathtt{E}}^{t+1} = B_{\mathtt{E}}^t$;
  3. if $v^t \in V_{\mathtt{b}}$, then an auction takes place: Let $\pi_{\mathtt{E}}(v^0 \ldots v^t, B_{\mathtt{E}}^t) = (b_{\mathtt{E}}, v')$ and $\pi_{\mathtt{A}}(v^0 \ldots v^t, 1 - B_{\mathtt{E}}^t) = (b_{\mathtt{A}}, v'')$. If $b_{\mathtt{E}} \geq b_{\mathtt{A}}$, then Eve wins the bidding, pays Adam the bid amount $b_{\mathtt{E}}$, so that $B_{\mathtt{E}}^{t+1} = B_{\mathtt{E}}^t - b_{\mathtt{E}}$, and the next vertex is the one chosen by Eve, i.e., $v^{t+1} = v'$. Otherwise, if $b_{\mathtt{A}} > b_{\mathtt{E}}$, Adam wins the bidding, pays Eve the bid amount $b_{\mathtt{A}}$, so that $B_{\mathtt{E}}^{t+1} = B_{\mathtt{E}}^t + b_{\mathtt{A}}$, and the next vertex is chosen by *Adam*, i.e., $v^{t+1} = v''$.

Note that bidding ties are resolved in favor of Eve; later we will point out (see Remark 13) that how the ties are resolved is immaterial for the technical results that we will present in this paper. The infinite path generated by $\pi_{\mathtt{E}}$ and $\pi_{\mathtt{A}}$ and the initial budget $B_{\mathtt{E}}^0$ is the path induced by the unique trace generated by $\pi_{\mathtt{E}}$ and $\pi_{\mathtt{A}}$, and it is written as $\mathrm{path}^{\mathcal{G}}(\pi_{\mathtt{E}}, \pi_{\mathtt{A}}, B_{\mathtt{E}}^0) = v^0 v^1 \ldots$.

Specifications. A *specification* is a given *set* of infinite paths of $\mathcal{G}$. We categorized the specifications into *qualitative* and *quantitative* specifications. We will consider the following qualitative specifications: (1) *reachability* specifications contain all paths visiting a given target $T \subseteq V$, written as: $\mathsf{Reach}^T := \left\{ v^0 v^1 \ldots \in \mathrm{PathsInf}_{v^0}^{\mathcal{G}} \mid \exists t \geq 0 \,.\, v^t \in T \right\}$; (2) *safety* specifications contain all paths staying inside a given set of safe vertices $T \subseteq V$, written as: $\mathsf{Safe}^T := \left\{ v^0 v^1 \ldots \in \mathrm{PathsInf}_{v^0}^{\mathcal{G}} \mid \forall t \geq 0 \,.\, v^t \in T \right\}$; (3) *parity* specifications assume each vertex v is assigned a color $C(v) \in \mathbb{N}$, and it contains all paths in which the maximum infinitely occurring color is even, written as: $\mathsf{Parity}^C := \left\{ v^0 v^1 \ldots \in \mathrm{PathsInf}_{v^0}^{\mathcal{G}} \mid \max \{ c \in \mathbb{N} \mid \forall s \geq 0 \,.\, \exists t > s \,.\, C(v^t) = c \} \text{ is even} \right\}$.

For quantitative specifications, we assume a reward assignment $R(v) \in \mathbb{R}$ to each vertex v and $r \in \mathbb{R}$. We will consider the following quantitative specifications: (1) *discounted sum* specifications, given a discount factor $\lambda \in (0, 1)$, contain all paths whose discounted sum of rewards is at least r, i.e., $\mathsf{DS}_{\geq r}^{\lambda, R} := \left\{ v^0 v^1 \ldots \in \mathrm{PathsInf}_{v^0}^{\mathcal{G}} \mid \sum_{t=0}^{\infty} \lambda^t R(v^t) \geq r \right\}$; (2) *mean-payoff* specifications contain all paths whose mean-payoff is at least r, i.e. $\mathsf{MP}_{\geq r}^R := \left\{ v^0 v^1 \ldots \in \mathrm{PathsInf}_{v^0}^{\mathcal{G}} \mid \limsup_{k \to \infty} \frac{1}{k} \sum_{t=0}^{k-1} R(v^t) \geq r \right\}$.

Generalized bidding games. We formalize GBGs and SGBG; examples are in Figure 1.

▶ **Definition 4** (Generalized bidding game). *A generalized bidding game (GBG) is a pair $(\mathcal{G}, \varphi)$, where $\mathcal{G}$ is a game graph, and φ is a specification on $\mathcal{G}$.*

▶ **Definition 5** (Simple generalized bidding game). *A simple GBG (SGBG) is a GBG $(\mathcal{G}, \varphi)$ where for every bidding vertex v of $\mathcal{G}$, it holds that $|E(v)| = 2$.*

The traditional bidding games – which we refer to as *pure* bidding games or just BG in short – are special cases of GBGs with only bidding vertices. Similarly, a simple pure bidding game (SBG) is a special case of BG whose every (bidding) vertex has two outgoing edges.

Like BGs, the central concept in GBGs is the *threshold budget* of Eve, which is the amount of initial budget that is necessary and sufficient for Eve to fulfill the specification.

▶ **Definition 6** (Threshold budgets). *Suppose $(\mathcal{G},\varphi)$ is a GBG. The threshold budget is a function $Th_{\mathcal{G},\varphi}\colon V \to [0,1]$ such that for every vertex $v \in V$, the following hold:*

$$\forall B^0_{\mathtt{E}} > Th_{\mathcal{G},\varphi}(v) \,.\, \exists \pi_{\mathtt{E}} \in \Pi_{\mathtt{E}} \,.\, \forall \pi_{\mathtt{A}} \in \Pi_{\mathtt{A}} \,.\, \mathrm{path}^{\mathcal{G}}_v(\pi_{\mathtt{E}},\pi_{\mathtt{A}},B^0_{\mathtt{E}}) \in \varphi,$$
$$\forall B^0_{\mathtt{E}} < Th_{\mathcal{G},\varphi}(v) \,.\, \exists \pi_{\mathtt{A}} \in \Pi_{\mathtt{A}} \,.\, \forall \pi_{\mathtt{E}} \in \Pi_{\mathtt{E}} \,.\, \mathrm{path}^{\mathcal{G}}_v(\pi_{\mathtt{E}},\pi_{\mathtt{A}},B^0_{\mathtt{E}}) \notin \varphi.$$

Intuitively, Eve wins from v if her initial budget $B^0_{\mathtt{E}}$ is above her threshold budget $Th_{G,\varphi}(v)$ at v, and loses if her budget is lower than $Th_{G,\varphi}(v)$. When $B^0_{\mathtt{E}} = Th_{G,\varphi}(v)$, the outcome of the game (i.e., who has a winning strategy) depends on the tie-breaking mechanism; we will point it out in Remark 13.

We first establish the following connection between GBGs and SGBGs:

▶ **Proposition 7** (Equivalence of GBGs and SGBGs, Proof in the full version [1]). *Computing thresholds for GBGs reduces in linear time to computing thresholds for SGBGs.*

Since SGBGs are special cases of GBGs, the other direction trivially holds, implying that SGBGs and GBGs are equivalent to each other modulo a linear blow-up.

Decision problems. For each specification type $M \in \{\mathsf{Reach}, \mathsf{Safe}, \mathsf{Parity}, \mathsf{DS}, \mathsf{MP}\}$, the question of whether the threshold is below 0.5 is the main problem we study, and it is equivalent to the membership of the set $\mathit{THRESH}^{\leq 0.5}_M := \{(\mathcal{G},\varphi,v) \mid (\mathcal{G},\varphi) \text{ is a GBG and } v \text{ is a vertex of } \mathcal{G} \wedge \varphi \text{ is of type } M \wedge Th_{\mathcal{G},\varphi}(v) \leq 0.5\}$. The two fundamental questions related to threshold budgets are:

1. *Does threshold budget exist for every generalized bidding game with reachability, safety, parity, discounted sum, and mean-payoff specifications?* We show that the answer is "yes."
2. *What is the complexity of deciding membership in $\mathit{THRESH}^{\leq 0.5}_M$, for various specification types M?* We will answer this question by establishing bridges between GBGs and simple stochastic games, a class of extensively studied games which we summarize next.

3 Bridging Generalized Bidding Games and Simple Stochastic Games

3.1 Preliminaries on Simple Stochastic Games

We recall some basic concepts on simple stochastic games (SSGs) from the literature [18, 16].

▶ **Definition 8** (Simple stochastic games). *A simple stochastic game (SSG) is a pair $(\mathcal{S},\varphi)$, where $\mathcal{S} = (V, V_{\mathtt{E}}, V_{\mathtt{A}}, V_{\mathtt{r}}, E)$ is the game graph, in which V is a finite set of vertices, $V_{\mathtt{E}}, V_{\mathtt{A}}$, and $V_{\mathtt{r}}$ form a partition of V, with $V_{\mathtt{E}}$ and $V_{\mathtt{A}}$ being the control vertices owned respectively by Eve and Adam and $V_{\mathtt{r}}$ being the random vertices, $E \subseteq V \times V$ is the set of edges where each random vertex $r \in V_{\mathtt{r}}$ has exactly two successors,[3] i.e., $|E(r)| = 2$, and φ is a specification.*

SSGs have similar gameplay as GBGs for the control vertices – Eve chooses successors from $V_{\mathtt{E}}$ and Adam chooses successors from $V_{\mathtt{A}}$ – whereas at the random vertices in $V_{\mathtt{r}}$, each of the two successors is chosen with probability 0.5.

Random-turn games (RTG) are played between Eve and Adam on finite graphs. In each turn, who moves the token is determined by a (fair) coin toss, i.e., each player gets to move the token with probability 0.5. RTGs can be modeled by SSGs where random

[3] The more standard definition of SSGs from the literature imposes the two out-degree restriction on *all* vertices, not just the random ones. Our formalism allows a more succinct representation.

vertices alternate with the control vertices, i.e., $E \subseteq (V_{\mathtt{r}} \times (V_{\mathtt{E}} \cup V_{\mathtt{A}})) \cup ((V_{\mathtt{E}} \cup V_{\mathtt{A}}) \times V_{\mathtt{r}})$, and every random vertex $v \in V_{\mathtt{r}}$ has exactly two successors $v_{\mathtt{E}} \in V_{\mathtt{E}}$ and $v_{\mathtt{A}} \in V_{\mathtt{A}}$, such that $E(v_{\mathtt{E}}) = E(v_{\mathtt{A}})$ (i.e., v assigns the turn randomly, but afterwards the players have the same options).

Paths in SSGs are defined as in GBGs, and a *strategy* $\sigma_{\mathtt{E}}$ of Eve maps finite paths only ending at Eve's vertices to their successors (i.e., strategies involve no bidding); strategies of Adam are defined analogously, and the sets of their strategies are $\Sigma_{\mathtt{E}}$ and $\Sigma_{\mathtt{A}}$, respectively. Every pair of strategies $\sigma_{\mathtt{E}} \in \Sigma_{\mathtt{E}}$ and $\sigma_{\mathtt{A}} \in \Sigma_{\mathtt{A}}$ in an SSG $\mathcal{S}$ resolves all non-determinism and induces a Markov chain $\mathcal{S}^{\sigma_{\mathtt{E}},\sigma_{\mathtt{A}}}$ (with possibly infinite state space). For a given measurable set of paths Δ, we write $P^{\sigma_{\mathtt{E}},\sigma_{\mathtt{A}}}(\Delta \mid v) \in [0,1]$ to denote the probability that a randomly sampled path of $\mathcal{S}^{\sigma_{\mathtt{E}},\sigma_{\mathtt{A}}}$ starting at v belongs to Δ. With this, for a given specification φ, we define the optimal *value* of every vertex $v \in V$ as $val_{\mathcal{S},\varphi}(v) := \sup_{\sigma_{\mathtt{E}} \in \Sigma_{\mathtt{E}}} \inf_{\sigma_{\mathtt{A}} \in \Sigma_{\mathtt{A}}} P^{\sigma_{\mathtt{E}},\sigma_{\mathtt{A}}}(\varphi \mid v)$, where the measurability for different types of φ that we consider is established [13, 23, 10].

Deciding whether the value is above 0.5 is a fundamental problem for SSGs, and, for a given specification type $M \in \{\mathsf{Reach}, \mathsf{Safe}, \mathsf{Parity}, \mathsf{DS}, \mathsf{MP}\}$, it is equivalent to the membership of the set $VALUE_M^{\geq 0.5} := \{(\mathcal{S}, \varphi, v) \mid (\mathcal{S}, \varphi) \text{ is an SSG and } v \text{ is a vertex of } \mathcal{S} \wedge \varphi \text{ is of type } M \wedge val_{\mathcal{S},\varphi}(v) \geq 0.5\}$.

Central to the value estimation for SSGs is the *Bellman operator* $\mathcal{B}^S \colon [0,1]^V \to [0,1]^V$, which maps a given value assignment $X \in [0,1]^V$ to an updated assignment as follows:

$$\mathcal{B}^S(X)(v) = \begin{cases} X(v) & v \in S \\ \max_{u \in E(v)} X(u) & v \in V_{\mathtt{E}} \setminus S \\ \min_{u \in E(v)} X(u) & v \in V_{\mathtt{A}} \setminus S \\ \frac{\min_{u \in E(v)} X(u) + \max_{u \in E(v)} X(u)}{2} & v \in V_{\mathtt{r}} \setminus S. \end{cases}$$

3.2 Exchanging Bidding Vertices with Random Vertices

Given a BG $(\mathcal{G}, \varphi)$, its corresponding random-turn game $(RT(\mathcal{G}), \varphi)$ is obtained by reinterpreting each vertex of $\mathcal{G}$ as the vertex in a random-turn game. The following theorem is a well-known result in the literature that relates the thresholds of BGs to the values of RTGs.

▶ **Theorem 9** (Structural equivalence between BGs and RTGs [3]). *Let $(\mathcal{G}, \varphi)$ be a BG where φ is a* Reach, Safe, Parity, *or* MP *specification. Then, the threshold $Th_{\mathcal{G},\varphi}$ exists, and $(\mathcal{G}, \varphi)$ is structurally equivalent to the RTG $(RT(\mathcal{G}), \varphi)$, i.e., $Th_{\mathcal{G},\varphi}(v) = 1 - val_{RT(\mathcal{G}),\varphi}(v)$.*

We extend this connection in the context of SGBGs and SSGs. Given an SGBG $(\mathcal{G}, \varphi)$, let $(\Gamma(\mathcal{G}), \varphi)$ be the identical SSG that reinterprets the bidding vertices of $\mathcal{G}$ as random vertices. Conversely, for an SSG $(\mathcal{S}, \varphi)$, let $(\Gamma^{-1}(\mathcal{S}), \varphi)$ be the identical SGBG obtained by reinterpreting the random vertices of $\mathcal{S}$ as bidding vertices. It is easy to see that, for every SGBG $(\mathcal{G}, \varphi)$, $\Gamma^{-1}(\Gamma(\mathcal{G})) = \mathcal{G}$.

Our main result is Theorem 10, proved in Section 4 and Section 5. It is a generalization of Theorem 9 to SGBGs with parity, discounted-sum, and mean-payoff specifications.

▶ **Theorem 10** (Structural equivalence between SGBGs and SSGs). *Let $(\mathcal{G}, \varphi)$ be an SGBG where φ is a* Reach, Safe, Parity, DS, *or* MP *specification. Then, the threshold $Th_{\mathcal{G},\varphi}$ exists, and $(\mathcal{G}, \varphi)$ is structurally equivalent to the SSG $(\Gamma(\mathcal{G}), \varphi)$, i.e., $Th_{\mathcal{G},\varphi}(v) = 1 - val_{\Gamma(\mathcal{G}),\varphi}(v)$.*

Proposition 7 then gives rise to a structural equivalence between GBGs and SSGs, modulo a linear blow-up. This result yields immediately a computational equivalence between the problems $THRESH_{\varphi}^{\leq 0.5}$ and $VALUE_{\varphi}^{\geq 0.5}$, where φ is among the listed types.

4 Qualitative Specifications

In this section, we study the existence and computation of threshold budgets in GBGs with reachability, safety, and parity specifications. In particular, for all these specifications, we show that threshold budgets exist, and moreover, we establish a computational procedure by showing the structural equivalence between GBGs and SSGs, as claimed in Section 3.2.

Although parity specifications subsume both reachability and safety specifications, we first state and prove our results for reachability and safety (Section 4.1) because they are more intuitive. Building on top of this, we will extend our results to the more general case of parity specifications in Section 4.2.

4.1 Reachability and Safety Specifications

Suppose $\mathcal{G} = (V, V_{\mathtt{E}}, V_{\mathtt{A}}, V_{\mathtt{b}}, E)$ is a game graph and Reach^T is a reachability specification on $\mathcal{G}$, i.e. Eve's goal is to reach T and Adam's goal is to prevent T from being reached. We define $rTh\colon V \to [0,1]$ where if Eve's budget exceeds $rTh(v)$ when the game is in vertex v, she has a winning strategy. We then take the point-of-view of Adam and define $sTh\colon V \to [0,1]$ where if Adam's budget exceeds $sTh(v)$, he can prevent the reachability of T. We will then show that $rTh(v) = 1 - sTh(v)$, which implies $Th_{\mathcal{G},\mathsf{Reach}^T} = rTh(v) = 1 - Th_{\mathcal{G},\mathsf{Safe}^{V\setminus T}} = 1 - sTh(v)$.

We first introduce the bidding operator $\mathcal{L}^S\colon \mathbb{R}^V \to \mathbb{R}^V$ as follows:

$$\mathcal{L}^S(X)(v) := \begin{cases} X(v) & v \in S \\ \min_{u \in E(v)} X(u) & v \in V_{\mathtt{E}} \setminus S \\ \max_{u \in E(v)} X(u) & v \in V_{\mathtt{A}} \setminus S \\ \frac{\min_{u \in E(v)} X(u) + \max_{u \in E(v)} X(u)}{2} & v \in V_{\mathtt{b}} \setminus S \end{cases}$$

We show three useful properties of the bidding operator in the full version [1]: (i) $\mathcal{L}^S$ is monotone in X, (ii) if for $v \in V \setminus S$, Eve's budget is above $\mathcal{L}^S(X)(v)$ when the game is in v, then she can play so that her budget is above $X(u)$ if u is reached next, and (iii) $\mathcal{L}^S(\mathbf{1} - X) = \mathbf{1} - \mathcal{B}^S(X)$ where $\mathcal{B}$ is the bellman operator of $\Gamma(\mathcal{G})$.

Next, we illustrate the finite-horizon reachability thresholds $rTh^i\colon V \to [0,1]$, where if Eve's budget exceeds $rTh^i(v)$ when the token is on v, she has a strategy to reach T in at most i steps. The threshold rTh^i is defined recursively, with $rTh^0(v) := 0$ if $v \in T$, $rTh^0(v) := 1$ if $v \notin T$, and $rTh^i(v) := \mathcal{L}^T(rTh^{i-1})(v)$ for every $i > 0$.

Intuitively, Eve can win in 0 steps, if and only if the game is already in T. Moreover, for $i > 0$, she can guarantee to win the game in i steps, if she can guarantee to have enough budget in the next step to win in $i-1$ steps. We then define $rTh(v) := \lim_{i\to\infty} rTh^i(v)$ and show that it is well-defined. We also show how Eve can win when her budget is above $rTh(v)$.

▶ **Theorem 11** (Existence of Reach Thresholds). *For any GBG $(\mathcal{G}, \mathsf{Reach}^T)$ it holds that:*

- *The limit $rTh(v) = \lim_{i\to\infty} rTh^i(v)$ is well-defined.*
- *If Eve's budget exceeds $rTh^i(v)$ when the token is on v, she has a strategy to force reachability of T in at most i steps.*
- *If Eve's budget exceeds $rTh(v)$ when the token is on v, she has a strategy to win Reach^T.*

Proof. We provide a proof sketch for each statement. The full proofs are in the full version [1].

Well-Definedness of rTh. The first statement follows by the monotonicity of the operator $\mathcal{L}^T$ and the fact that the sequence is bounded in $[0,1]$.
The second statement is proved by induction on i. The full proof is available in the full version [1].

Reachability Winning Strategy. Suppose the game starts at v and Eve's budget $B_{\mathtt{E}}$ strictly exceeds $rTh(v)$. Since $rTh^i(v)$ is a non-increasing sequence converging to $rTh(v)$, there exists some finite $k \geq 0$ where $B_{\mathtt{E}} > rTh^k(v)$. Due to the second statement, Eve has a finite-horizon strategy to reach T in at most k steps. Therefore, following this strategy, Eve ensures reachability of T and wins the qualitative specification Reach^T. ◀

Similarly, taking the point-of-view of Adam, we can define the finite-horizon safety thresholds $sTh^i : V \to [0, 1]$ as the amount of budget he needs in order to prevent T from being reached for at least i steps (i.e. $\mathsf{Safe}^{V \setminus T}$ for i steps). Formally, sTh^i is defined recursively, with $sTh^0(v) = 1$ if $v \in T$, $sTh^0(v) = 0$ if $v \notin T$, and $sTh^i(v) = \mathcal{L}^T(sTh^{i-1})(v)$ for every $i > 0$. The next theorem shows the correctness of the sTh^i as the threshold budget for finite-horizon safety. Moreover, it shows that the limit $sTh(v) \coloneqq \lim_{i\to\infty} sTh^i(v)$ exists and Adam can win when his budget exceeds $sTh(v)$.

▶ **Theorem 12** (Existence of Safe Thresholds). *The following hold for any GBG $(\mathcal{G}, \mathsf{Reach}^T)$:*

- *If Adam's budget exceeds $sTh^i(v)$ when the token is on v, he can prevent T from being reached for at least i steps.*
- *The limit $sTh(v) = \lim_{i\to\infty} sTh^i(v)$ is well-defined.*
- *If Adam's budget exceeds $sTh(v)$ when the token is on v, he can prevent reachability of T.*

Proof. The first two statements are shown in the full version [1]. We prove the third statement:

Safety Winning Strategy. Suppose Adam's initial budget satisfies $B_{\mathtt{A}} > sTh(v)$. Since $sTh^i(v) = \mathcal{L}^T(sTh)$, it follows by the properties of the Bellman operator (see the full version [1]) that he has a strategy to ensure his budget exceeds $sTh(u)$ if u is reached in the next step. Thus, he can ensure that his budget is always above $sTh(v)$, for any v that is reached during the gameplay. Given that $sTh(v) = 1$ for $v \in T$, and Adam's budget cannot exceed 1, it follows that T is never reached. ◀

▶ Remark 13. Like pure bidding games [22, 2], only when Eve's budget coincides with the threshold budget rTh, the tie-breaking rule matters. Assuming ties are always resolved in favor of Eve, she has a winning strategy iff her initial budget is at least rTh. If ties were resolved in favor of Adam, then Eve would require more budget than rTh to win.

By induction, we can show that $rTh^i(v) = 1 - sTh^i(v)$ for all i and $v \in V$, which leads to:

▶ **Proposition 14.** *For any $T \subseteq V$ we have $Th_{\mathcal{G},\mathsf{Reach}^T} = rTh = 1 - sTh = 1 - Th_{\mathcal{G},\mathsf{Safe}^{V\setminus T}}$.*

Thus, computing threshold budgets for reachability and safety in GBGs are equivalent problems. Moreover, considering the complete lattice $([0, 1]^V, \leq)$ with the pointwise definition of $\leq$, the above definition of rTh shows that rTh is the greatest fixed-point X of the operator $\mathcal{L}^T(X)$ where $X_{|T} = \mathbf{0}$ is fixed. It is also a classical result for SSGs [20] that the optimal value function val in an SSG with reachability specification is the least fixed-point of the operator $\mathcal{B}^T(X)$ where $X_{|T} = \mathbf{1}$ is fixed. This implies the following:

▶ **Theorem 15** (Equivalence to SSGs). *Let $(\mathcal{G}, \mathsf{Reach}^T)$ be an SGBG. Then for all $v \in V$, it holds that $rTh(v) = 1 - val_{\Gamma(\mathcal{G}),\mathsf{Reach}^T}(v)$. Similarly, $sTh(v) = 1 - val_{\Gamma(\mathcal{G}),\mathsf{Safe}^{V\setminus T}}$.*

Proof. The dual relationship $\mathcal{L}^T(\mathbf{1} - X) = \mathbf{1} - \mathcal{B}^T(X)$ holds (See the full version [1]). Consequently, by applying the Knaster-Tarski fixed-point theorem, the bijection $X \mapsto \mathbf{1} - X$ maps the least fixed-point of $\mathcal{B}^T$ to the greatest fixed-point of $\mathcal{L}^T$.

It is established [20] that the maximal reachability value function $val_{\Gamma(\mathcal{G}),\mathsf{Reach}^T}$ acts as the *least* fixed-point of the operator $\mathcal{B}^T(X)$ subject to the target condition $X_{|T} = \mathbf{1}$. Applying the complementing map, the value vector $\mathbf{1} - val_{\Gamma(\mathcal{G}),\mathsf{Reach}^T}$ calculates the corresponding *greatest* fixed-point of $\mathcal{L}^T(X)$ under the inverted target condition $X_{|T} = \mathbf{0}$. Thus, $rTh(v) = 1 - val_{\Gamma(\mathcal{G}),\mathsf{Reach}^T}(v)$ for all $v \in V$. The last is due to $rTh = 1 - sTh$ for Reach^T. ◀

Proposition 7 then implies that the reachability threshold computation in GBGs is linearly equivalent to that of SGBGs, which is structurally equivalent to computing values in SSGs. On the algorithmic side, it is known that the membership problem of $VALUE^{\geq 0.5}_{\mathsf{Reach}}$ in SSGs is in $NP \cap coNP$ [18]. Theorem 15 implies that membership in $THRESH^{\leq 0.5}_{\mathsf{Reach}}$ is decidable in $NP \cap coNP$ as well. Moreover, if any or both of $V_{\mathtt{E}}$ or $V_{\mathtt{A}}$ are empty in an SSG, the SSG becomes an MDP. Computing the value function in MDPs is possible in polynomial time using linear programming [10]. Hence, if one of $V_{\mathtt{E}}$ or $V_{\mathtt{A}}$ is empty, $THRESH^{\leq 0.5}_{\mathsf{Reach}^T}$ is decidable in polynomial time. The following proposition summarises our complexity results:

▶ **Corollary 16** (Threshold verification complexities for Reach and Safe)**.** *The following hold:*

- *The membership problems of $THRESH^{\leq 0.5}_{\mathsf{Reach}}$ and $THRESH^{\leq 0.5}_{\mathsf{Safe}}$ are in $NP \cap coNP$.*
- *If $V_{\mathtt{E}} = \emptyset$ or $V_{\mathtt{A}} = \emptyset$, then both $THRESH^{\leq 0.5}_{\mathsf{Reach}}$ and $THRESH^{\leq 0.5}_{\mathsf{Safe}}$ membership are in P.*

4.2 Parity Specifications

In this section, we show the existence of parity thresholds in GBGs by constructing a series of recursive sequences whose limit converges to the parity threshold. This will later be used to show how the parity thresholds in GBGs relate to parity values in stochastic games.

Let $(\mathcal{G}, \mathsf{Parity}^C)$ be a GBG where $C\colon V \to \{0, \dots, d\}$ is a function that assigns a color $C(v)$ to each vertex v. For $k \leq d$ let $T_k = \{v \in V | C(v) = k\}$ be the set of vertices with color k. Moreover, for $\trianglelefteq \in \{<, \leq, >, \geq, \neq\}$, let $T_{\trianglelefteq k} = \{v \in V | C(v) \trianglelefteq k\}$.

We show the existence of the parity threshold function $pTh\colon V \to [0,1]$ where $pTh(v)$ is the amount of budget Eve needs at v to win the game. Let $Z_k \in [0,1]^{T_k}$ be a vector of *candidate* thresholds for vertices in T_k. We first define the *frugal* parity threshold function $fpTh_k(Z_{k+1}, Z_{k+2}, \dots, Z_d)\colon T_{\leq k} \to [0,1]$ where for $v \in T_{\leq k}$, if Eve's budget exceeds $fpTh_k(Z_{k+1}, Z_{k+2}, \dots, Z_d)(v)$ when the token is on v, she has a strategy to ensure either:

- **Escape:** The game reaches a vertex $u \in T_{>k}$ while her budget exceeds $Z_{C(u)}$, or,
- **Internal Win:** The game stays in $T_{\leq k}$ and the greatest color visited infinitely is even.

This way, $fpTh_d(\cdot)(v)$ will be equal to the actual parity threshold $pTh(v)$ since the escape condition is impossible when $k = d$. The recursive construction of $fpTh_k$ is in the full version [1].

▶ **Lemma 17** (Well-definedness and correctness of $fpTh$)**.** *Suppose $k \in \{0, \dots, d\}$ and $Z_{k+1}, \dots Z_d$ are candidate thresholds. The following statements hold:*

- *$fpTh_k(Z_{k+1}, \dots, Z_d)(v)$ is well-defined for all $v \in V$.*
- *If Eve has budget more than $fpTh_k(Z_{k+1}, \dots, Z_d)(v)$ when the token is on v, she has a strategy to ensure either (i) the game stays in $T_{\leq k}$ and she wins, or (ii) the game reaches a vertex $u \in T_{>k}$ while her budget exceeds $Z_{C(u)}(u)$.*

Lemma 17 uses the point-of-view of Eve and shows the existence of winning strategies for her whenever her budget is above $fpTh_k$. One could add one unit to the color of every vertex, i.e. have $C'(v) = C(v) + 1$, to obtain a new coloring C'. This switches the role of players and by the definition of $fpTh$, it follows that $fpTh^C_k = 1 - fpTh^{C'}_{k+1}$. This immediately results in the first main result of this section, stated below. Consequently, $Th_{\mathcal{G},\mathsf{Parity}^C}(v) = pTh(v)$.

▶ **Theorem 18** (Existence of Parity thresholds). *Consider the GBG $(\mathcal{G}, \mathsf{Parity}^C)$. When the token is on v, if Eve's budget exceeds $pTh(v) = fpTh_d(\cdot)(v)$, she has a strategy to win. If her budget is less than $pTh(v)$, Adam has a winning strategy.*

The recursive definition of $pTh(v) = fpTh_d(\cdot)(v)$ reveals that the parity threshold is computed as a *nested quantitative fixed-point* of the operator $\mathcal{L}^{\varnothing}$. Let μ and ν denote the least and greatest fixed-point operators, respectively. At each color k, the frugal threshold $fpTh_k$ is defined as μ of $\mathcal{L}^{T_{\geq k}}$ when k is even and as ν of $\mathcal{L}^{T_{\geq k}}$ when k is odd. The full parity threshold pTh is thus given by alternating least and greatest fixed-points, nested from the innermost color 0 up to the outermost color d, instantiated with the operator $\mathcal{L}$. Formally, $pTh = \Theta_d X_{T_d} \ldots \Theta_0 X_{T_0} \mathcal{L}^{\varnothing}(X)$, where $\Theta_i = \mu$ for all even i, and $\Theta_i = \nu$ for all odd i.

This nested fixed-point structure has a precise counterpart in the theory of simple stochastic games, utilizing the SSG Bellman operator $\mathcal{B}$ defined earlier. It was shown in [20] that the optimal value function val of $\Gamma(\mathcal{G})$ with parity specification Parity^C is computed as an alternating nested fixed-point of $\mathcal{B}$, using ν at even priority levels and μ at odd priority levels – the exact mirror of the alternation used in the threshold computation. Formally, $val^* = \Omega_d X_{T_d} \ldots \Omega_0 X_{T_0} \mathcal{B}^{\varnothing}(X)$, where $\Omega_i = \nu$ for all even i, and $\Omega_i = \mu$ for all odd i.

The bridge between the two nested fixed-point computations is the pointwise duality between the two operators. This duality effectively exchanges least fixed-points (μ) and greatest fixed-points (ν) under the complementation map $X \mapsto \mathbf{1} - X$. As the threshold computation uses μ exactly where the SSG value computation uses ν (and vice versa), this equivalence propagates through the entire alternating sequence from level 0 up to d:

▶ **Theorem 19** (Equivalence to SSGs, Proof in the full version [1]). *Let $(\mathcal{G}, \mathsf{Parity}^C)$ be an SGBG. Then for all $v \in V$, it holds that $pTh(v) = 1 - val_{\Gamma(\mathcal{G}), \mathsf{Parity}^C}(v)$.*

Combining Theorem 19 with Proposition 7 it is implied that computing parity thresholds in GBGs is linearly equivalent to that of SGBGs, which is structurally equivalent to computing parity values in SSGs. In particular, given that $VALUE^{\geq 0.5}_{\mathsf{Parity}}$ is in $NP \cap coNP$ [11, 18], it follows that $THRESH^{\leq 0.5}_{\mathsf{Parity}}$ is also in $NP \cap coNP$. Moreover, many algorithms have been developed for computing the optimal values of parity stochastic games [15, 17], and all of these algorithms can be used to compute the parity thresholds of GBGs. As stated earlier, in case $V_{\mathtt{E}} = \emptyset$ or $V_{\mathtt{A}} = \emptyset$, the corresponding SSG $\Gamma(\mathcal{G})$ is an MDP. Computing parity values in MDPs reduces to finding maximal end-components (MECs) and computing the maximum probability of reaching winning MECs [10], which totally takes polynomial time. Thus, in such cases, the parity thresholds of GBGs can be computed in polynomial time as well.

▶ **Corollary 20** (Threshold verification complexity for Parity). *The following results hold:*

- *The membership problem of $THRESH^{\leq 0.5}_{\mathsf{Parity}}$ is in $NP \cap coNP$.*
- *If $V_{\mathtt{E}} = \emptyset$ or $V_{\mathtt{A}} = \emptyset$, then $THRESH^{\leq 0.5}_{\mathsf{Parity}}$ membership is in P.*

Strategy complexity. The optimal strategies we constructed above need to account for various kinds of information from the past: For reachability, only the current configuration suffices, similar to the case of BGs. For safety, only the current vertex suffices, irrespective of the budget; recall that for BGs, safety has trivial solutions. For parity, the path seen so far and the current budget suffices; recall that for BGs, parity reduces to reachability. We conjecture, only the current configuration would suffice for parity on GBGs.

5 Quantitative Specifications

In this section, we consider GBGs with the quantitative specifications, namely mean payoff and discounted-sum. We show that threshold budgets exist, and prove the structural equivalence between SGBGs and SSGs, as claimed in Section 3.2. For mean payoff, this gives us the complexity of computing the thresholds in GBGs, due to known results from the SSG literature. For discounted-sum, the complexity question remains open.

5.1 Mean Payoff Specifications

In this subsection, we consider GBGs with the mean payoff specification, that is, given a reward function R and a number r, a path $\rho = v^0 v^1 \cdots$ is winning for Eve iff $\limsup_{k\to\infty} \frac{1}{k}\sum_{t=0}^{k-1} R(v^t) \geq r$. This specification has been studied in the literature [12, 14]. It is noteworthy that the specification of Eve is not to maximize the expected payoff, but rather to maximize the probability of having mean payoff at least r. Since this specification is Borel, sub-mixing, and prefix-independent, positional strategies suffice:

▶ **Theorem 21** (Memoryless determinacy of mean-payoff SSGs [12, Proposition 1])**.** *SSGs with mean payoff specification admit optimal positional strategies for Eve and Adam.*

The connection between random-turn games and pure bidding games has already been proved [4]. Their result relies on the fact that strongly connected components in pure bidding games have threshold budget either 0 or 1. We generalize this claim to any SGBG for which every vertex of the game has the same expected mean payoff value. Combining this claim with the results in [4], we obtain the following theorem. More details and the proof are given in the full version [1].

▶ **Theorem 22** (Existence of MP thresholds and equivalence to SSGs)**.** *Let $(\mathcal{G}, \mathsf{MP}^R_{\geq r})$ be an SGBG with a mean payoff specification and a reward function $R : V \to \mathbb{Q}$. The threshold budget function exists. Furthermore, for all $r \in \mathbb{R}$ and all $v \in V$, $Th_{\mathcal{G},\mathsf{MP}^R_{\geq r}}(v) = 1 - val_{\Gamma(\mathcal{G}),\mathsf{MP}^R_{\geq r}}(v)$.*

By Theorem 21, SSGs with mean payoff specification admit positional strategies. Computing the value for an MDP can be done in polynomial time [12]. Thus we get the following:

▶ **Corollary 23** (Threshold verification complexity for MP)**.** *The following results hold:*
- *The membership problem of $THRESH^{\leq 0.5}_{\mathsf{MP}^R_{\geq r}}$ is in NP $\cap$ coNP.*
- *If $V_{\mathtt{E}} = \emptyset$ or $V_{\mathtt{A}} = \emptyset$, then $THRESH^{\leq 0.5}_{\mathsf{MP}^R_{\geq r}}$ membership is in P.*

5.2 Discounted-Sum Specifications

We are the first to consider discounted-sum specifications in the context of bidding games. We prove that the thresholds admit the same equivalence with SSGs as shown for other specifications. Towards this goal, first, we describe a fixed-point characterization of the values of SSGs with discounted-sum specifications using a Bellman operator. Similar characterizations have been observed for Markov Decision Processes [24]. Notice that the goal of Eve is to maximize the probability of achieving payoff at least r. This should not be confused with the goal of maximizing the expected payoff commonly studied in the literature. Without loss of generality, we assume all rewards in the game are non-negative. We can assume this since adding c to every reward changes the payoff of an infinite play by $\frac{c}{1-\lambda}$.

▶ **Definition 24** (Bellman operator for discounted-sum SSGs). *Let* $(\mathcal{G} = (V, V_{\mathtt{E}}, V_{\mathtt{A}}, V_{\mathtt{r}}, E), \mathsf{DS}^{\lambda,R}_{\geq r})$ *be an SSG with* $\lambda \in (0,1)$. *For* $\psi \in V \times \mathbb{R} \to [0,1]$, *we define* $\mathcal{B} : (V \times \mathbb{R} \to [0,1]) \to (V \times \mathbb{R} \to [0,1])$ *such that for every* $(v,r) \in V \times \mathbb{R}$, *and* $r' = \frac{r - R(v)}{\lambda}$,

$$\mathcal{B}[\psi](v,r) \;=\; \begin{cases} 1 & r \leq R(v), \\ \max_{v' \in E(v)} \psi(v', r') & v \in V_{\mathtt{E}}, \\ \min_{v' \in E(v)} \psi(v', r') & v \in V_{\mathtt{A}}, \\ \frac{\psi(v_1, r') + \psi(v_2, r')}{2} & v \in V_{\mathtt{r}}, \{v_1, v_2\} = E(v). \end{cases}$$

The value of discounted-sum SSGs is characterized by taking the left-continuous closure of the limit of iterates of $\mathcal{B}$. The proof is provided in the full version [1]. To the best of our knowledge, it is not known whether the function $(v,r) \to val_{\Gamma(\mathcal{G}),\mathsf{DS}^{\lambda,R}_{\geq r}}(v)$ is computable. The left-continuous closure is necessary, as it is demonstrated in Example 28 in Appendix C.

▶ **Theorem 25** (Values in discounted-sum SSGs). *For every* $v \in V$ *and every* $r \in \mathbb{R}$, *it holds that* $val_{\Gamma(\mathcal{G}),\mathsf{DS}^{\lambda,R}_{\geq r}}(v) \;=\; \lim_{\varepsilon \to 0^+} \lim_{k \to \infty} \mathcal{B}^k[\psi_0](v, r - \varepsilon)$, *where* $\psi_0(v,r) = 0$.

Next, we prove the correspondence between the thresholds of the SGBGs and the values of the simple stochastic game. The proof uses the following idea: Let $\rho = v^0 v^1 \cdots v^k$ be a finite history of a play and inductively define $r^0 = r$ and $r^{n+1} = \frac{r^n - R(v^n)}{\lambda}$ for any $n \in \mathbb{N}$. If Eve starts the game in a vertex v with budget more than $1 - val_{\Gamma(\mathcal{G}),\mathsf{DS}^{\lambda,R}_{\geq r}}(v)$, then she can maintain the following invariant: For any k, Eve has strictly more budget than $1 - val_{\Gamma(\mathcal{G}),\mathsf{DS}^{\lambda,R}_{\geq r^k}}(v^k)$. Thanks to this invariant, she can guarantee overall payoff at least r. The situation is almost symmetric for Adam if Eve has budget less than $1 - val_{\Gamma(\mathcal{G}),\mathsf{DS}^{\lambda,R}_{\geq r}}(v)$ but we need to use the left-continuity property of the value function.

▶ **Theorem 26** (Existence of DS thresholds and equivalence to SSGs). *Let* $(\mathcal{G}, \mathsf{DS}^{\lambda,R}_{\geq r})$ *be a SGBG with a discounted-sum specification where* $\lambda \in (0,1)$. *The threshold budget function exists. Furthermore,* $Th_{\mathcal{G},\mathsf{DS}^{\lambda,R}_{\geq r}}(v) = 1 - val_{\Gamma(\mathcal{G}),\mathsf{DS}^{\lambda,R}_{\geq r}}(v)$ *for all* $r \in \mathbb{R}$ *and all* $v \in V$.

We remark that the computational complexity of the $\mathit{THRESH}^{\leq 0.5}_{\mathsf{DS}^{\lambda,R}_{\geq r}}$ problem is open since the corresponding decision problem for SSGs is still open.

Strategy complexity. The optimal strategies for discounted-sum specifications may need to account for the path seen so far. Intuitively, since the specification is not prefix-independent, the player needs to know the accumulated payoff. We illustrate this in the full version [1]. This property also translates to SSGs, which is a curious observation.

6 The Threshold Repair Problem

Inspired by the potential application of GBGs in auction-based scheduling for multi-objective decision-making (see the motivating example in the Introduction), we introduce the following threshold repair problem, where we want to change a small set of bidding vertices to Eve's vertices in order to reduce the threshold to below a given target value.

▶ **Problem 1** (Threshold repair). *Suppose we are given an SGBG* $(\mathcal{G} = (V, V_{\mathtt{E}}, V_{\mathtt{A}}, V_{\mathtt{b}}, E), \varphi)$, *a set of bidding vertices* $C \subseteq V_{\mathtt{b}}$ *called the constraint set, an integer* $k > 0$, *and a target threshold* $\gamma \in (0,1)$ *for some vertex* $v \in V$. *Does there exist a new SGBG* $(\mathcal{G}' = (V, V'_{\mathtt{E}}, V_{\mathtt{A}}, V'_{\mathtt{b}}, E), \varphi)$, *only differing in Eve's and bidding vertices, such that:*

(i) *only some* bidding *vertices from* $V_{\mathtt{b}}$ *got converted to Eve's vertices in* $V'_{\mathtt{E}}$*, i.e.,* $V'_{\mathtt{b}} \subseteq V_{\mathtt{b}}$*, implying* $V_{\mathtt{E}} \subseteq V'_{\mathtt{E}}$ *and* $V'_{\mathtt{E}} \subseteq V_{\mathtt{E}} \cup V_{\mathtt{b}}$*;*
(ii) *all bidding vertices in* $V_{\mathtt{b}}$ *that got converted to Eve's vertices in* $V'_{\mathtt{E}}$ *are in the constraint set* C*, i.e.,* $C \cap V'_{\mathtt{E}} = \emptyset$*;*
(iii) *the number of vertices converted to Eve's vertices is at most* k*, i.e.,* $|V'_{\mathtt{E}} \setminus V_{\mathtt{E}}| \leq k$*;*
(iv) *the threshold of the new game is at most* t*, i.e.,* $Th_{\mathcal{G}',\varphi}(v) \leq t$*.*

From the equivalence between SGBGs and SSGs, the problem has a direct analogue in the SSG setting, namely how can we increase the value of an SSG by changing a small set of random vertices with Eve's vertices? This problem could be of independent interest in applications where the randomness in the SSG could be replaced by deterministic control for a price. Below we state that the repair problem is NP-complete; the complexity proof and an MILP-based algorithm can be found in the full version [1].

▶ **Theorem 27** (Threshold repair complexity, proof in the full version [1])**.** *The threshold repair problem is NP-complete for GBGs with reachability, safety, and parity specifications.*

7 Conclusions and Future Directions

We introduce generalized bidding games (GBG) that subsume and extend both pure bidding games (BG) and turn-based games. We establish that thresholds exist for GBGs for a variety of qualitative and quantitative specifications, like parity and discounted-sum, and establish a structural equivalence between the thresholds of simple GBGs and SSGs. As the complexity of computing thresholds for GBGs is no more than the best-known complexity for BGs, we hope GBGs will become the standard model for studying bidding games in the future.

Several questions remain unresolved. Firstly, we only considered first-price Richman bidding, but many other alternatives exist in the literature. We conjecture that, apart from the structural equivalence result with SSGs, the existence and computation of thresholds will be extendible to other settings in a straightforward manner. Secondly, we established that the threshold computation problem for discounted-sum specifications turns out to be equivalent to the quantitative satisfaction value problem in SSGs. Since the latter is open, resolving either problem is an interesting open question that would simultaneously advance the other.

References

1. Ali Asadi, Thomas A. Henzinger, Ehsan Kafshdar Goharshady, Pavol Kebis, and Kaushik Mallik. Generalized bidding games: Where bidding and stochastic games meet, 2026. `arXiv:2606.29420`.
2. Guy Avni, Ehsan Kafshdar Goharshady, Thomas A. Henzinger, and Kaushik Mallik. Bidding games with charging. In *35th International Conference on Concurrency Theory, CONCUR 2024, Calgary, Canada, September 9-13, 2024*, volume 311 of *LIPIcs*, pages 8:1–8:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. `doi:10.4230/LIPIcs.CONCUR.2024.8`.
3. Guy Avni and Thomas A. Henzinger. A survey of bidding games on graphs (invited paper). In *31st International Conference on Concurrency Theory, CONCUR 2020, Vienna, Austria (Virtual Conference), September 1-4, 2020*, volume 171 of *LIPIcs*, pages 2:1–2:21. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. `doi:10.4230/LIPIcs.CONCUR.2020.2`.
4. Guy Avni, Thomas A. Henzinger, and Ventsislav Chonev. Infinite-duration bidding games. *J. ACM*, 66(4):31:1–31:29, 2019. `doi:10.1145/3340295`.
5. Guy Avni, Thomas A. Henzinger, and Rasmus Ibsen-Jensen. Infinite-duration poorman-bidding games. In *Web and Internet Economics - 14th International Conference, WINE 2018, Oxford, UK, December 15-17, 2018, Proceedings*, volume 11316 of *Lecture Notes in Computer Science*, pages 21–36. Springer, 2018. `doi:10.1007/978-3-030-04612-5_2`.

6 Guy Avni, Rasmus Ibsen-Jensen, and Josef Tkadlec. All-pay bidding games on graphs. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 1798–1805. AAAI Press, 2020. doi:10.1609/AAAI.V34I02.5546.

7 Guy Avni, Martin Kurecka, Kaushik Mallik, Petr Novotný, and Suman Sadhukhan. Bidding games on markov decision processes with quantitative reachability objectives. In *Proceedings of the 24th International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2025, Detroit, MI, USA, May 19-23, 2025*, pages 161–169. International Foundation for Autonomous Agents and Multiagent Systems / ACM, 2025. doi:10.5555/3709347.3743528.

8 Guy Avni, Kaushik Mallik, and Suman Sadhukhan. Auction-based scheduling. In *Tools and Algorithms for the Construction and Analysis of Systems - 30th International Conference, TACAS 2024, Luxembourg City, Luxembourg, April 6-11, 2024, Proceedings, Part III*, volume 14572 of *Lecture Notes in Computer Science*, pages 153–172. Springer, 2024. doi:10.1007/978-3-031-57256-2_8.

9 Guy Avni and Suman Sadhukhan. Computing threshold budgets in discrete-bidding games. *TheoretiCS*, 4, 2025. doi:10.46298/THEORETICS.25.5.

10 Christel Baier and Joost-Pieter Katoen. *Principles of model checking.* MIT Press, 2008.

11 Raphaël Berthon, Joost-Pieter Katoen, and Zihan Zhou. A direct reduction from stochastic parity games to simple stochastic games. In *36th International Conference on Concurrency Theory, CONCUR 2025, Aarhus, Denmark, August 26-29, 2025*, volume 348 of *LIPIcs*, pages 9:1–9:21. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025. doi:10.4230/LIPIcs.CONCUR.2025.9.

12 Tomás Brázdil, Václav Brozek, and Kousha Etessami. One-counter stochastic games. In *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2010, Chennai, India, December 15-18, 2010*, volume 8 of *LIPIcs*, pages 108–119. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2010. doi:10.4230/LIPIcs.FSTTCS.2010.108.

13 Krishnendu Chatterjee. Concurrent games with tail objectives. *Theor. Comput. Sci.*, 388(1-3):181–198, 2007. doi:10.1016/J.TCS.2007.07.047.

14 Krishnendu Chatterjee and Laurent Doyen. Perfect-information stochastic games with generalized mean-payoff objectives. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science, LICS '16, New York, NY, USA, July 5-8, 2016*, pages 247–256. ACM, 2016. doi:10.1145/2933575.2934513.

15 Krishnendu Chatterjee and Thomas A. Henzinger. Strategy improvement and randomized subexponential algorithms for stochastic parity games. In *STACS 2006, 23rd Annual Symposium on Theoretical Aspects of Computer Science, Marseille, France, February 23-25, 2006, Proceedings*, volume 3884 of *Lecture Notes in Computer Science*, pages 512–523. Springer, 2006. doi:10.1007/11672142_42.

16 Krishnendu Chatterjee and Thomas A. Henzinger. A survey of stochastic ω-regular games. *J. Comput. Syst. Sci.*, 78(2):394–413, 2012. doi:10.1016/J.JCSS.2011.05.002.

17 Krishnendu Chatterjee, Marcin Jurdzinski, and Thomas A. Henzinger. Quantitative stochastic parity games. In *Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2004, New Orleans, Louisiana, USA, January 11-14, 2004*, pages 121–130. SIAM, 2004. URL: http://dl.acm.org/citation.cfm?id=982792.982808.

18 Anne Condon. The complexity of stochastic games. *Inf. Comput.*, 96(2):203–224, 1992. doi:10.1016/0890-5401(92)90048-K.

19 Anne Condon. On algorithms for simple stochastic games. In *Advances in Computational Complexity Theory*, volume 13 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 51–72. American Mathematical Society, Providence, RI, 1993. doi:10.1090/dimacs/013/04.

20 Luca de Alfaro and Rupak Majumdar. Quantitative solution of omega-regular games. *J. Comput. Syst. Sci.*, 68(2):374–397, 2004. doi:10.1016/J.JCSS.2003.07.009.

21 Mike Develin and Sam Payne. Discrete bidding games. *Electron. J. Comb.*, 17(1), 2010. doi:10.37236/357.

22 Andrew J Lazarus, Daniel E Loeb, James G Propp, Walter R Stromquist, and Daniel H Ullman. Combinatorial games under auction play. *Games and Economic Behavior*, 27(2):229–264, 1999. doi:10.1006/game.1998.0676.

23 Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley Series in Probability and Statistics. Wiley, 1994. doi:10.1002/9780470316887.

24 Douglas J White. Minimizing a threshold probability in discounted markov decision processes. *Journal of mathematical analysis and applications*, 173(2):634–646, 1993. doi:10.1006/jmaa.1993.1093.

A Motivating Example: The Game Bid-Tac-Toe

Consider a bidding variant of the game tic-tac-toe, which we call *bid*-tac-toe; the game can be modeled as a GBG as shown in Figure 4. Let Eve's symbol be circles and Adam's symbol be crosses. Unlike the standard variant where the players take turns for placing their symbols, at each step, a bidding takes place, and the winner of the bidding decides who plays next. To model the bidding, we consider the first-price Richman variant, where the players initially start with budgets whose sum is 1. In each round, both players submit their bids simultaneously and concurrently, the highest bidder chooses who plays next, and pays the bid amount to the opponent. All bidding games in the rest of the paper will use the first-price Richman mechanism. The game ends if either all cells are filled, or if one of the players has put three symbols in a line. Eve wins if she manages to form a line (see vertex k), and loses otherwise (see vertex l), regardless of whether Adam forms a line. While bid-tac-toe can be easily modeled using GBGs, as shown in Figure 4, it cannot be modeled using a classical bidding game. This is because in classical bidding games, every vertex is a bidding vertex, and the outgoing edges are independent of the identity of the highest bidder. This would mean that, in the classical encoding of bid-tac-toe, Eve could place crosses and Adam could place circles, which is not allowed by the rule of the game.[4]

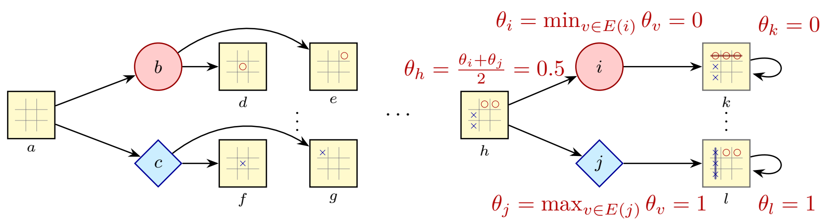

Figure 4 Snippets of the GBG modeling bid-tac-toe; see Figure 1 for vertex legends. The bidding vertices show the current board configuration. Eve and Adam respectively win by reaching vertices like k and l. For vertices h–l, we show the threshold budget computation (details in Section 4), where $E(x)$ and θ_x respectively denote the set of successors and the threshold of a given vertex x.

[4] While existing literature did use the classical model to represent bid-tac-toe [6], strictly speaking, it did not conform with the syntax of the game. In particular, Eve was allowed to place crosses and Adam was allowed to place circles. Incidentally, it did not matter, because the specification of Eve implicitly discourages her to put crosses, and vice versa.

B Distinction between GBGs and Pure Bidding Games

Our main goal in this section is to show that GBGs are strictly more general than pure bidding games. We first briefly recall the syntax of Alternating Temporal Logic (ATL) and its extension, ATL* and then show an ATL* formula that is satisfied by some GBG but cannot be satisfied by any pure bidding game.

ATL. Introduced to reason about multi-agent systems, ATL replaces the path quantifiers of CTL with strategic path quantifiers $\langle\langle A\rangle\rangle$, meaning "the coalition of agents A has a joint strategy to ensure". To directly connect this framework with GBGs, where initial budgets uniquely determine the players' abilities to enforce outcomes, we extend the standard ATL structures to incorporate an initial budget allocation. Focusing on the two-player setting where $\Sigma = \{Eve, Adam\}$ and the players' budgets always sum to 1, we augment the strategic quantifier with a budget parameter $b \in [0, 1]$. We write $\langle\langle A\rangle\rangle^{>b}$ to denote that when Eve's budget is above b, coalition A has a strategy to ensure the specified behavior against any strategy of the opposing coalition.

Given a set of atomic propositions AP, the extended *Budgeted ATL* syntax is defined inductively as follows:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \langle\langle A\rangle\rangle^{>b} X\varphi \mid \langle\langle A\rangle\rangle^{>b} G\varphi \mid \langle\langle A\rangle\rangle^{>b} F\varphi \mid \langle\langle A\rangle\rangle^{>b} \varphi\mathcal{U}\varphi$$

where $p \in AP$, $A \subseteq \Sigma$, and $b \in [0, 1]$. The temporal operators X, G, F, and $\mathcal{U}$ represent "next", "globally", "eventually", and "until", respectively.

In the context of GBGs, satisfying a formula such as $\langle\langle Eve\rangle\rangle^{>b}\psi$ at a given vertex v maps directly to our exact definition of threshold budgets (see Section 2): it evaluates to true if and only if the threshold budget required for Eve to enforce the path specification ψ from v satisfies $Th_{\mathcal{G},\psi}(v) \leq b$.

*Budgeted ATL** generalizes this naturally by decoupling the strategic quantifiers from the temporal operators, allowing unstructured nesting and boolean combination of path properties. It recursively distinguishes between state formulas (φ) and path formulas (ψ):

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \langle\langle A\rangle\rangle^{>b}\psi$$
$$\psi ::= \varphi \mid \neg\psi \mid \psi \vee \psi \mid X\psi \mid G\psi \mid F\psi \mid \psi\mathcal{U}\psi$$

where $p \in AP$, $A \subseteq \Sigma$, and $b \in [0, 1]$.

Our main goal in this section is to show that GBGs are strictly more general than pure bidding games. It is easy to see that pure bidding games are special cases of GBGs. We show an ATL* property that is satisfied by some GBGs but not by any pure bidding game. This shows that pure bidding games a strict subclass of GBGs.

Let $AP = \{p_1, p_2, p_3\}$ be the set of atomic propositions. Consider the following ATL* formulas:

$$\varphi_1 = \langle\langle Eve, Adam\rangle\rangle^{>0} \bigwedge_{1\leq i,j\leq 3} G(p_i \Rightarrow Fp_j)$$
$$\varphi_2 = \neg\langle\langle Eve, Adam\rangle\rangle^{>0} FG(\neg p_1 \wedge \neg p_2 \wedge \neg p_3)$$
$$\varphi_3 = \langle\langle Eve\rangle\rangle^{>0.5} G(\neg p_1)$$

Intuitively, φ_1 holds when every p_i vertex has a path to a p_j vertex for every i, j. On the other hand, φ_2 is satisfied when Eve and Adam cannot collaborate to ensure that no p_i is reached from some point on. Consequently, every SCC of the game graph must contain at

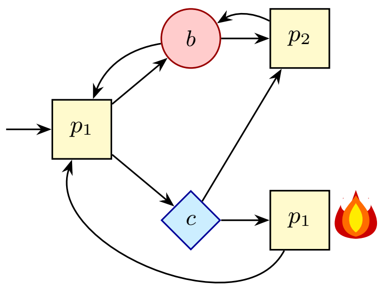

Figure 5 A GBG that satisfies $\Phi = \varphi_1 \wedge \varphi_2 \wedge \varphi_3$.

least one p_i vertex for some i. φ_1 and φ_2 together imply that every bottom SCC of the game graph has at least one p_i vertex for each i. Lastly, φ_3 states that when Eve's initial budget is above 0.5, she has a strategy to ensure that p_1 is never reached, i.e. $Th_{\mathcal{G},\mathsf{Safe}(p_1)} \leq 0.5$.

We show a GBG that satisfies $\Phi = \varphi_1 \wedge \varphi_2 \wedge \varphi_3$. We also show that no pure bidding game can satisfy Φ.

Consider the GBG in Figure 5. It satisfies φ_1, because the whole game graph is strongly connected. It satisfies φ_2, because the only SCC of the game graph contains all p_1, p_2 and p_3. Finally, it satisfies φ_3, because if Eve's budget is above 0.5, she can bid all in at p_1, win the first bidding (because Adam's budget is less than 0.5), and move the game to vertex b, where she can trap the game in $\{b, p_2\}$ indefinitely and satisfy $G(\neg p_1)$.

On the other hand, assume for the same of contradiction that a pure bidding game $\mathcal{G}$ satisfies Φ. As shown above, every bottom SCC of $\mathcal{G}$ must contain all p_1, p_2 and p_3. This means that Adam can always force the game to reach p_1, if his budget is non-zero [3]. This is in direct contradiction with φ_3 being satisfied, which shows no such pure bidding game $\mathcal{G}$ can exist.

C Appendix for Section 5.2

▶ **Example 28** (Discounted-sum specifications do not admit memoryless strategies)**.** Consider a discounted-sum SSG with discount $\lambda = \frac{1}{2}$, a single state owned by Eve and two actions: action (a) gives a payoff either -1 or 3 with probability $\frac{1}{2}$ and $\frac{1}{2}$, respectively; action (b) gives a payoff 0 with probability 1. The objective of Eve is to maximize the probability of gaining payoff at least 1. Taking action (b) cannot satisfy the specification. Instead, taking action (a) has a non-zero value. If the payoff outcome is 3, she switches to play action (b) which guarantees overall payoff 3. If the payoff outcome is -1, she plays action (a) again as she needs to compensate for the loss and there is non-zero probability of obtaining an overall payoff at least 1; playing action (b) cannot compensate for the loss. Therefore, the optimal strategy is not memoryless. The corresponding SGBG exhibits the same behaviour, i.e., chosen actions depend on the bidding outcomes.