ECAI 2023
K. Gal et al. (Eds.)

doi:10.3233/FAIA230264

Reachability Poorman Discrete-Bidding Games

Guy Avni[a], Tobias Meggendorfer[b,c], Suman Sadhukhan[a], Josef Tkadlec[d] and Đorđe Žikelić[c]

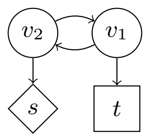
[a]University of Haifa
[b]Technical University of Munich
[c]Institute of Science and Technology Austria
[d]Harvard University

Abstract. We consider *bidding games*, a class of two-player zero-sum *graph games*. The game proceeds as follows. Both players have bounded budgets. A token is placed on a vertex of a graph, in each turn the players simultaneously submit bids, and the higher bidder moves the token, where we break bidding ties in favor of Player 1. Player 1 wins the game iff the token visits a designated target vertex. We consider, for the first time, *poorman discrete-bidding* in which the granularity of the bids is restricted and the higher bid is paid to the bank. Previous work either did not impose granularity restrictions or considered *Richman* bidding (bids are paid to the opponent). While the latter mechanisms are technically more accessible, the former is more appealing from a practical standpoint. Our study focuses on *threshold budgets*, which is the necessary and sufficient initial budget required for Player 1 to ensure winning against a given Player 2 budget. We first show existence of thresholds. In DAGs, we show that threshold budgets can be approximated with error bounds by thresholds under continuous-bidding and that they exhibit a periodic behavior. We identify closed-form solutions in special cases. We implement and experiment with an algorithm to find threshold budgets.

1 Introduction

Two-player zero-sum *graph games* are a fundamental model with numerous applications, e.g., in reactive synthesis [23] and multi-agent systems [2]. A graph game is played on a finite directed graph as follows. A token is placed on a vertex, and the players move it throughout the graph. We consider *reachability* games in which Player 1 wins iff the token visits a designated target vertex. Traditional graph games are *turn-based*: the players alternate turns in moving the token. We consider *bidding games* [17, 16] in which an "auction" (bidding) determines which player moves the token in each turn.

Several concrete bidding mechanisms have been defined. In all mechanisms, both players have bounded budgets. In each turn, both players simultaneously submit bids that do not exceed their budgets, and the higher bidder moves the token. The mechanisms differ in three orthogonal properties. *Who pays:* In *first-price* bidding only the winner pays the bid, whereas in *all-pay* bidding both players pay their bids. *Who is the recipient:* In *Richman* bidding (named after David Richman) payments are made to the other player, in *poorman* bidding payments are made to the "bank", i.e. the bid is lost. *Restrictions on bids:* In *continuous-bidding* no restrictions are imposed and bids can be arbitrarily small, whereas in *discrete-bidding* budgets and bids are restricted to be integers.



B_2	0	1	2	3	4	$5 \cdots$
$T_{v_1}(B_2)$	0	0	1	1	2	$3 \cdots$
$T_{v_2}(B_2)$	0	1	3	4	6	$8 \cdots$

Figure 1: Player 1's threshold budget as a function of Player 2's initial budget in the two intermediate vertices of the game on the left.

In this work, we study, for the first time, first-price poorman discrete-bidding games. This combination addresses two limitations of previously-studied models. First, most work on bidding games focused on continuous-bidding games, where a rich mathematical structure was identified in the form of an intriguing equivalence with a class of stochastic games called *random-turn games* [22], in particular for infinite-duration games [3, 4, 5, 7]. These results, however, rely on bidding strategies that prescribe arbitrarily small bids. Employing such strategies in practice is questionable – after all, money is discrete. Second, discrete-bidding games have only been studied under Richman bidding [13, 1, 10]. The advantage of Richman over poorman bidding is that, as a rule of thumb, the former is technically more accessible. In terms of modeling capabilities, however, while Richman bidding is confined to so called *scrip systems* that provide fairness using an internal currency, poorman bidding captures a wide range of settings since it coincides with the popular first-price auction.

The central quantity that we focus on is the *threshold budget* in a vertex, which is a necessary and sufficient budget for Player 1 to ensure winning the game. Formally, a *configuration* of a bidding game is a triple $\langle v, B_1, B_2 \rangle$, where v denotes the vertex on which the token is placed and B_i is Player i's budget, for $i \in \{1, 2\}$. For an initial vertex v, we call a function $T_v : \mathbb{N} \to \mathbb{N}$ the threshold budgets at v if for every configuration $c = \langle v, B_1, B_2 \rangle$, Player 1 wins from c if $B_1 \geq T_v(B_2)$ and loses from c if $B_1 \leq T_v(B_2) - 1$. We stress that we focus only on *pure* strategies.

Example 1. Consider the game that is depicted in Fig. 1, where we break bidding ties in favor of Player 1. In this example, we identify the first few thresholds. In Thm. 15, we show that the thresholds in this game are $T_{v_1}(B_2) = \lfloor B_2/\phi \rfloor$ and $T_{v_2}(B_2) = \lfloor B_2 \cdot \phi \rfloor$, where $\phi \approx 1.618$ is the golden ratio.[1] First, when both budgets are 0, all biddings result in ties, which Player 1 wins and forces the game to t. Second, we argue that Player 1 wins from $\langle v_1, 0, 1 \rangle$. Indeed, Player 1 bids 0. In order to avoid losing, Player 2 must bid 1, wining the

[1] We encourage the reader to read more about these two sequences in https://oeis.org/A000201 and https://oeis.org/A005206. See also Remark 16.

bidding and pays the bid to the bank. Thus, the next configuration is $\langle v_2, 0, 0\rangle$, from which Player 1 wins. Third, we show that $T_{v_2}(1) = 1$. Indeed, Player 2 wins from $\langle v_2, 0, 1\rangle$ by bidding 1. On the other hand, from $\langle v_2, 1, 1\rangle$ Player 1 wins since by bidding 1, he forces the game to $\langle v_1, 0, 1\rangle$, from which he wins. Finally, $T_{v_1}(2) > 0$ since Player 2 can force two consecutive wins when the budgets are $\langle 0, 2\rangle$, and $T_{v_1}(2) = 1$ since by bidding 1, Player 1 forces Player 2 to pay at least 2 in order not to lose immediately, and he wins from $\langle v_2, 1, 0\rangle$.

Applications. In *sequential first-price auctions* m items are sold sequentially in independent first-price auctions (e.g., [18, 14]). The popularity of these auctions stems from their simplicity. Indeed, in each round of the auction, a user is only asked to bid for the current item on sale, whereas in *combinatorial auctions*, users need to provide an exponential input: a valuation for each subset of items. Two-player sequential auctions are a special case of bidding games played on DAGs. Each vertex v represents an auction for an item. A path from the root to v represents the outcomes of previous rounds, i.e., a subset of items that Player 1 has purchased so far. For a target bundle T of items, this modeling allows us to obtain a bidding strategy that is guaranteed to purchase at least the bundle T no matter how the opponent bids. Indeed, we solve the corresponding bidding game with the Player 1 objective of reaching a vertex in which T is purchased. We can also capture a quantitative setting in which Player 1 associates a value with each bundle of items. Given a target value k, we set Player 1's target to be vertices that represent a purchased bundle of value at least k. We can then either find the threshold budget for obtaining value k or fix the initial budgets and optimize over k.

Next, we describe two important classes of continuous poorman-bidding games that are technically challenging, and we argue that it is appealing to bypass this challenge by considering their discrete-bidding variants. Our study lays the basis for these extensions. First, all-pay poorman bidding games constitute a dynamic version of the well-known Colonel Blotto games [12]: we think of budgets as resources with no inherent value (e.g., time or energy) and a strategy invests the resources in order to achieve a goal. In fact, many applications of Colonel Blotto games are dynamic, thus all-pay bidding games are arguably a more accurate model [6]. All-pay poorman bidding games are surprisingly technically complex, e.g., already in extremely simple games, optimal strategies rely on infinite-support distributions, and have never been studied under discrete bidding. Second, the study of partial-observation bidding games was initiated recently [8]. Poorman bidding is both appealing from a theoretical and practical standpoint but is technically complex. Again, it is appealing to consider partial-information in combination with discrete bidding.

Finally, poorman discrete bidding are amenable to extensions such as multi-player games or non-zero-sum games [20].

Our Contribution

Existence of thresholds. In discrete-bidding games, one needs to explicitly state how bidding ties are resolved [1]. Throughout the paper, we always break ties in favor of Player 1. We start by showing existence of thresholds in every game, including games that are not DAGs. Our techniques are adapted from [1] for Richman discrete-bidding games. We note that existence of thresholds coincides with *determinacy*: from every configuration, one of the players has a *pure* winning strategy. We point out that while determinacy holds in turn-based games for a wide range of objectives [19], determinacy of bidding games is not immediate due to the *concurrent* choice of bids.

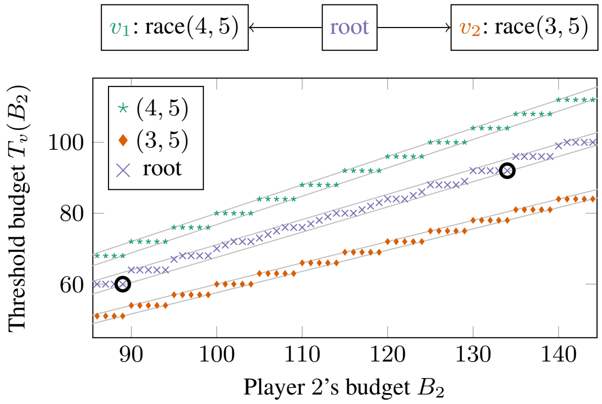


Figure 2: The thresholds in three vertices: a root vertex whose two children are roots of race games race$(3, 5)$ and race$(4, 5)$. For visibility, the x-axis starts at 85. We also depict the lower and upper bounds we obtain from our *pipe theorem* (indicated by solid lines) and highlight two points indicating the periodicity in the root vertex.

For example, *matching pennies* is a very simple concurrent game that is not determined: neither player can ensure winning.

Threshold budgets in DAGs. In continuous bidding, each vertex v is associated with a *threshold ratio* which is a value $t \geq 0$ such that when the ratio between the two players' budgets is $t + \epsilon$, Player 1 wins, and when the ratio is $t - \epsilon$, Player 2 wins [16].

First, we bound the discrete thresholds based on continuous ratios as follows. Let t_v denote the continuous ratio at a vertex v. Then, for every $B_2 \in \mathbb{N}$, we show that $T_v(B_2)$ lies in the *pipe*: $(B_2 - n) \cdot t_v \leq T_v(B_2) \leq B_2 \cdot t_v$, where n is the number of vertices in the game. We point out that the width of the pipe is fixed, so for large budgets B_2 the value $T_v(B_2)/B_2$ tends to the threshold ratio t_v.

Second, we show that threshold budgets in DAGs exhibit a periodic behavior. While we view this as a positive result, it has a negative angle: The periods are surprisingly complex even for fairly simple games, so even though we identify a compact representation for the thresholds in Example 1, we do not expect a compact representation in general games.

Third, in continuous-bidding games, the compact representation of the thresholds (i.e., each vertex being associated with a ratio) is the key to obtaining a linear-time backwards-inductive algorithm to compute thresholds in DAGs. Under discrete bidding, given a Player 2 budget B_2, we present a pseudo-linear algorithm to find $T(B_2)$, namely its running time is linear in the size of the game and in B_2.

Fourth, we obtain closed-form solutions for a class of games called *race games*: for $a, b \in \mathbb{N}$, the race game race(a, b) ends within $a + b$ turns, Player 1 wins the game if he wins a biddings before Player 2 wins b biddings. For example, a "best of 7" tournament (as in the NBA playoffs) is race$(4, 4)$.

Example 2. We illustrate some of our main results. In Fig. 2, we depict the threshold budgets in three vertices of a game as a function of Player 2's budget. First, the discrete thresholds reside in a "pipe" with slope equal to the corresponding continuous ratio (Thm. 10). Second, v_1 and v_2 are roots of race games, thus their thresholds are simple step functions (Thm. 13). Moreover, they lie exactly on the boundary of the pipe infinitely often, i.e. the pipe bound is tight (Cor. 14). Third, the threshold budgets are periodic (Thm. 12), we have $T_r(B_2+45) = T_r(B_2)+32$. We find it surprising that in such a simple game both the periodicity in the root node and the irregularity within this period are comparatively large.

Implementation and Experiments. We provide a pseudo-polynomial algorithm to find the threshold budget given the initial budget of Player 2 in general games together with a specialized, faster variant for DAGs. We implement the algorithm, experiment with it, and develop conjectures based on our findings. Beyond the theoretical interest, the running time we observed is extremely fast, illustrating the practicality of finding exact thresholds.

2 Preliminaries

A reachability bidding game is $\mathcal{G} = \langle V, E, t, s\rangle$, where V is the set of vertices, $E \subseteq V \times V$ is the set of edges, Player 1's *target* is $t \in V$, a *sink* $s \in V$ has no path to t and we think of s as Player 2's target, we assume that all other vertices have a path to both t and s. We write $N(v) = \{u \mid (v, u) \in E\}$ to denote the *neighbours* of v.

A *configuration* of $\mathcal{G}$ is of the form $c = \langle v, B_1, B_2\rangle$, where $v \in V$ is the vertex on which the token is placed and B_i is the budget of Player i, for $i \in \{1, 2\}$. At c, both players simultaneously choose *actions*, and the pair of actions determines the next configuration. For $i \in \{1, 2\}$, Player i's action is a pair $\langle b_i, u_i\rangle$, where $b_i \leq B_i$ is an *integer* bid that does not exceed the available budget and $u_i \in N(v)$ is a neighbor of v to move to upon winning the bidding. If $b_1 \geq b_2$, then Player 1 moves the token and pays "the bank", thus the next configuration is $\langle u_1, B_1 - b_1, B_2\rangle$. Dually, when $b_2 > b_1$, the next configuration is $\langle u_2, B_1, B_2 - b_2\rangle$.

A *strategy* is a function that maps each configuration to an action.[2] A pair of strategies σ_1, σ_2, and an initial configuration c_0 gives rise to a unique *play* denoted by $\text{play}(c_0, \sigma_1, \sigma_2)$, which is defined inductively. The inductive step, namely the definition of how a configuration is updated given two actions from the strategies, is described above. Let $\text{play}(c_0, \sigma_1, \sigma_2) = c_0, c_1, \ldots$, where $c_i = \langle v_i, B_1^i, B_2^i\rangle$. The *path* that corresponds to $\text{play}(c_0, \sigma_1, \sigma_2)$ is $v_0, v_1, \ldots$

Definition 3 (Winning Strategies). *A Player 1 strategy σ_1 is called a* winning strategy *from configuration c_0 iff for any Player 2 strategy σ_2, play$(c_0, \sigma_1, \sigma_2)$ visits the target t. On the other hand, a Player 2 strategy σ_2 is a winning strategy from c_0 iff for any Player 1 strategy σ_1, play$(c_0, \sigma_1, \sigma_2)$* does not visit *the target t. For $i \in \{1, 2\}$, we say that Player i* wins *from c_0 if he has a winning strategy from c_0.*

Throughout the paper, we focus on the necessary and sufficient budget that Player 1 needs for winning, given a Player 2 budget, defined formally as follows.

Definition 4 (Threshold budgets). *Consider a vertex $v \in V$. The* threshold budget *at v is a function $T_v : \mathbb{N} \to \mathbb{N}$ such that for every $B_2 \in \mathbb{N}$ Player 1 wins from $\langle v, T_v(B_2), B_2\rangle$, and Player 2 wins from $\langle v, T_v(B_2) - 1, B_2\rangle$.*

3 Existence of Thresholds

In this section we show the existence of threshold budgets in games played on general graphs.

Definition 5. (Determinacy). *A game is* determined *if from every configuration, one of the players has a pure winning strategy.*

We claim that determinacy is equivalent to existence of thresholds. It is not hard to deduce both implications from the following observation. An additional budget cannot harm a player; namely, if Player 1 wins from a configuration $\langle v, B_1, B_2\rangle$, he also wins from $\langle v, B_1', B_2\rangle$, for $B_1' > B_1$, and dually for Player 2.

In the rest of this section, we prove determinacy of poorman discrete-bidding games. Our proof is based on a technique that was developed in [1] to show determinacy of *Richman* discrete-bidding. We illustrate the key ideas. Consider a reachability bidding game $\mathcal{G} = \langle V, E, t, s\rangle$ and a configuration $c = \langle v, B_1, B_2\rangle$. We define a *bidding matrix* M_c that corresponds to c. For $\langle b_1, b_2\rangle \in \{0, \ldots, B_1\} \times \{0, \ldots, B_2\}$, the $(b_1, b_2)^{\text{th}}$ entry in M_c is associated with Player i bidding b_i, for $i \in \{1, 2\}$. We label entries in M_c by 1 or 2 as follows. Let $\mathcal{G}_1$ denote a turn-based game that is the same as $\mathcal{G}$ only that in each turn, Player 1 reveals his bid first and Player 2 responds. Technically, once both players reveal their bids, the game proceeds to an intermediate vertex $i_{b_1,b_2} = \langle b_1, b_2, c\rangle$. Since G_1 is turn-based, it is determined, thus one of the players has a winning strategy from i_{b_1,b_2}. We label the $(b_1, b_2)^{\text{th}}$ entry in M_c by $i \in \{1, 2\}$ iff Player i wins from i_{b_1,b_2}. For $i \in \{1, 2\}$, we call a row or a column of M_c a *i-row* or *i-column*, respectively, if all its entries are labeled i.

Definition 6. (Local Determinacy) *A bidding game $\mathcal{G}$ is called* locally determined *if for every configuration c, the bidding matrix M_c either has a 1-row or a 2-column.*

Local determinacy is used as follows. It can be shown that if Player 1 wins from c, then M_c has a 1-row. More importantly, if Player 1 does not win in c, local determinacy implies that if Player 2 bids b_2 in $\mathcal{G}$, corresponding to a 2-column, the game proceeds to a configuration c' from which Player 1 does not win. In reachability games, since Player 2's goal is to avoid the target, traversing non-losing configurations for Player 2 is in fact winning.[3]

Lemma 7. ([1, Theorem 3.5]) *If a reachability bidding game $\mathcal{G}$ is locally determined, then $\mathcal{G}$ is determined.*

Local determinacy of poorman discrete-bidding games follows from the following observations on bidding matrices whose proof can be found in the full version [9].

Lemma 8. *Consider a poorman discrete-bidding game $\mathcal{G}$ where Player 1 always wins tie, and consider a configuration $c = \langle v, B_1, B_2\rangle$. (1) Entries in M_c in a column above the top-left to bottom-right diagonal are equal: for bids $b_2 > b_1 > b_1'$, we have $M_c[b_1, b_2] = M_c[b_1', b_2]$. (2) Entries on a row, left of the diagonal are equal: for bids $b_1 > b_2 > b_2'$, we have $M_c[b_1, b_2] = M_c[b_1, b_2']$. (3) The entry immediately under the diagonal equals the entry on the diagonal: For a bid b, we have $M_c[b, b] = M_c[b, b-1]$.*

The proof of [1, Theorem 4.5] shows that a game whose bidding matrices have the properties of Lem. 8 is locally determined, irrespective of whether Richman or poorman bidding is employed. Combining with Lem. 7, we obtain the following.

Theorem 9. *Reachability poorman discrete-bidding games are determined.*

4 Threshold Budgets for Games on DAGs

In this section, we focus on games played on directed acyclic graphs (DAGs). We present two main results: First, the *Pipe theorem* that relates the threshold budgets to the threshold ratio in the continuous-bidding game; and, second, the *Periodicity theorem* which shows that the threshold budgets eventually exhibit a periodic behavior. Throughout this section, let $\mathcal{G} = \langle V, E, t, s\rangle$ be a game with $\langle V, E\rangle$ a DAG.

[2] In full generality, strategies map *histories* of configurations to actions. However, *positional* strategies suffice for reachability games.

[3] The theorem is stated for reachability objectives and it is extended in [1] to richer objectives.

4.1 Relating Discrete and Continuous Thresholds

We call the following theorem the *Pipe theorem* since it shows that the threshold budgets $T_v(B_2)$ lie in a "pipe" below a line whose slope is the threshold ratio t_v (see Example 2). We note that threshold ratios can be computed in DAGs in time polynomial in the size of the game (a fact we also exploit later on in our algorithm on DAGs), thus an immediate corollary of the Pipe theorem is an efficient approximation algorithm to computing the threshold budgets. In Corollary 14, we show that the lower bound is tight. For a vertex v, let max-path(v) denote the length of the longest path from v to either t or s. Note that max-path$(v) \leq |V| - 1$.

Theorem 10 (Pipe theorem). *Given $v \in V$, denote by t_v the threshold ratio in the continuous-bidding game at v. Then, for every initial budget $B_2 \in \mathbb{N}$ of Player 2, we have*

$$t_v \cdot (1 - \text{max-path}(v)/B_2) \leq T_v(B_2)/B_2 \leq t_v.$$

The right-hand side inequality holds even when $\mathcal{G}$ is not a DAG.

Proof sketch. (See the full version [9] for the detailed proof.) To prove the right-hand side inequality, we show that if Player 1 has initial budget of at least $t_v \cdot B_2$ then Player 1 can win by following a winning strategy in the *continuous-bidding* game and *rounding down* the bids. More formally, let σ_{cont} be a winning strategy for Player 1 under continuous-bidding when the game starts in v, Player 1's initial budget is at least $t_v \cdot B_2$, and Player 2's initial budget is B_2. We define a Player 1 strategy σ_{disc} as follows. Whenever σ_{cont} prescribes a pair $\langle b, u\rangle$, where b is a bid and $u \in V$ is the vertex to move to upon winning, σ_{disc} prescribes $\langle \lfloor b \rfloor, u\rangle$.

To prove the left-hand side inequality, we show that if Player 1 has initial budget strictly less than $t_v \cdot (B_2 - \text{max-path}(v))$ and Player 2 has initial budget B_2, then Player 2 can win by following the winning strategy in a *continuous-bidding* game and rounding the bids up. More formally, let σ_{cont} be a winning strategy for Player 2 under continuous-bidding when the game starts in v, Player 1's initial budget is at most $t_v \cdot (B_2 - \text{max-path}(v))$ and Player 2's initial budget is $B_2 - \text{max-path}(v)$. Suppose that σ_{cont} prescribes $\langle b, u\rangle$, then σ_{disc} for Player 2 prescribes $\langle \lceil b \rceil, u\rangle$. The fact that Player 2 always has enough budget to bid $\lceil b \rceil$ follows from the fact that the game necessarily ends within max-path(v) turns. □

An immediate corollary of Thm. 10 is that the ratio $T_v(B_2)/B_2$ tends to t_v.

Corollary 11 (Convergence to continuous ratios). *For every $v \in V$ we have $\lim_{B_2 \to \infty} T_v(B_2)/B_2 = t_v$.*

4.2 Periodicity of Threshold Budgets

The following theorem shows that for any fixed $v \in V$ the function $T_v(\cdot)$ that yields the threshold budgets exhibits an eventually periodic behavior, as seen in Example 2.

Theorem 12 (Periodicity theorem). *For any vertex $v \in V$ there exist values $B, u_x, u_y \in \mathbb{N}$ such that for all $B_2 \geq B$ we have $T_v(B_2 + u_x) = T_v(B_2) + u_y$. Moreover, the values B, u_x, u_y can be computed in polynomial time.*

Proof sketch. (See the full version [9] for the detailed proof.) The proof is by induction with respect to the topological order of the graph. If v is a leaf, then the claim is obvious. Consider v that is not a leaf. The proof is based on three ingredients. First, intuitively, when the children of v have different threshold ratios then their pipes diverge. Let v^- and v^+ respectively denote the children whose pipe is lowest and highest. By Thm. 10, under discrete-bidding, for large budgets, Player 1 and Player 2 will respectively proceed to v^- and v^+ upon winning the bidding in v.

Second, we show that if v^- satisfies $T_{v^-}(B_2 + u_x^-) = T_{v^-}(B_2) + u_y^-$ and v^+ satisfies $T_{v^+}(B_2 + u_x^+) = T_{v^+}(B_2) + u_y^+$ (both for large enough B_2), then v satisfies the same equality with $u_x = u_x^- \cdot (u_x^+ + u_y^+)$ and $u_y = u_y^+ \cdot (u_x^- + u_y^-)$. We illustrate the idea using Fig. 3, which depicts a configuration $c = \langle v, B_1, B_2\rangle$ as a point $[B_2, B_1]$ in the plane. Consider first the left image. Suppose that Player 1 bids b from $\langle v, B_1, B_2\rangle$ (see point P). The case that Player 1 wins the bidding corresponds to "stepping down" from $[B_2, B_1]$ to $[B_2, B_1 - b]$. Note that the token moves to v^-. Thus, a necessary condition for $B_1 \geq T_v(B_2)$ is $B_2 - b \geq T_{v^-}(B_2)$. The second case is when Player 2 bids $b + 1$ and wins the bidding, which corresponds to "stepping left" to $[B_2 - (b+1), B_1]$, the token moves to v^+, and we obtain a second necessary condition $B_1 \geq T_{v^+}(B_2 - (b+1))$. Then, given configurations on the thresholds of v^- and v^+ (depicted as Q and R), the "lowest" point that satisfies both conditions is a point on the threshold of v. The right part of Fig. 3 shows how the period of T_v is determined by the periods of T_{v^+} and T_{v^-}.

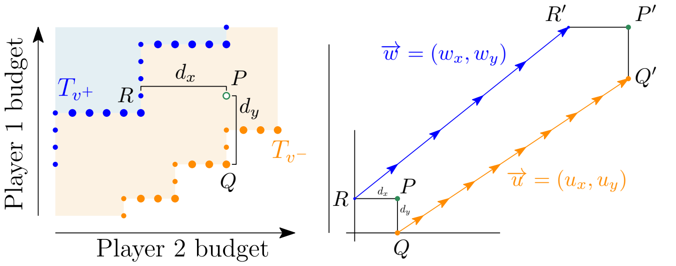


Figure 3: Left: Point P lies on or above T_v if and only if $d_x \leq d_y + 1$. Right: Chaining $v_x + v_y$ copies of u and $u_x + u_y$ copies of v, the situation repeats.

Third, if multiple children have the same threshold ratio, we reduce to the previous case by using the fact that a minimum of two periodic functions over integers is itself periodic. □

This result implies that for each $v \in V$, the function $T_v(\cdot)$ can be finitely represented: let B be Player 2's budget when the period "kicks in", then for all $B' \leq B$, the value $T_v(B')$ is stored explicitly and these values can be extrapolated to find $T_v(B'')$ for $B'' > B$.

We point out that periodicity may indeed appear only "eventually", as illustrated by Fig. 4; namely, only at $B = 7$ state $(2, 2)$ continuously is an optimal choice and the periodic behaviour is observed. Replacing race$(5, 4)$ with race$(2x + 1, 2x)$ leads to quickly growing periodicity thresholds B. Finally, we note that on non-DAGs, the behaviour is not necessarily periodic, as illustrated by Thm. 15 below.

5 Closed-form Solutions

In this section, we show closed-form solutions for threshold budgets in two special classes of games.

5.1 Race Games

Race games are a class of games played on DAGs. For $a, b \in \mathbb{N}$, the race game race(a, b) ends within $a + b$ turns, Player 1 wins the game if he wins a biddings before Player 2 wins b biddings. The key property of race games that we employ is that for each vertex v

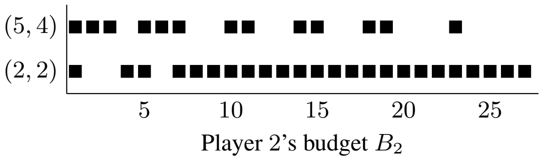


Figure 4: We consider a game comprising a root node v with two children, which are roots to race(5,4) and race(2,2). We depict Player 1's *winning moves*: for each Player 2's budget B_2, we depict the vertex (or vertices) that Player 1 may proceed to upon winning the bidding at configuration $\langle v, T_v(B_2), B_2\rangle$.

independent of the budgets, there is a neighbor v_i such that Player i proceeds to v_i upon winning the bidding at v, for $i \in \{1, 2\}$. The proof of the following theorem is obtained by induction. See the full version [9] for details and examples.

Theorem 13. *Let v be the root of a race game race(a, b). Then $T_v(B_2) = a \cdot \lfloor B_2/b\rfloor$.*

With the exact closed-form of threshold budgets for race games, we now show that the bounds in Thm. 10 are tight.

Corollary 14. *For every rational number $q = n/m$, there exist infinitely many games $\mathcal{G}$ with vertex v such that $t_v = q$ and for infinitely many B the lower and upper bound of Thm. 10 actually is an equality for some $B_2 > B$.*

Proof. Choose $\mathcal{G} = \text{race}(n, m)$ (or any multiple thereof) and insert the closed form of Thm. 13. Note that in a race game max-path(v) of the root vertex v clearly is $\max(n, m)$. □

5.2 Tug-of-War games

Given an integer $n \geq 1$, a *tug-of-war* game $\text{TOW}(n)$ is a game played on a chain with $n + 2$ nodes, namely n interior nodes and two endpoints s and t. We develop closed-form representations of thresholds in $\text{TOW}(2)$ (depicted in Fig. 1) and $\text{TOW}(3)$ (depicted in the full version [9]). For integers $k \in [1, n]$ and $b \geq 0$, we denote by $\text{tow}(n, k, b)$ the smallest budget that Player 1 needs to win the tug-of-war game $\text{TOW}(n)$ at the vertex that is k steps from his target t, when the opponent has budget b.

Theorem 15. *For $b \geq 0$, we have $\text{tow}(2, 1, b) = \lfloor b/\phi\rfloor$ and $\text{tow}(2, 2, b) = \lfloor b \cdot \phi\rfloor$, where $\phi = (\sqrt{5} + 1)/2 \approx 1.618$ is the golden ratio.*

Proof. To simplify notation, we use the same vertex names as in Fig. 1 and, for a Player 2 budget b, we denote by $t_b = \text{tow}(2, 1, b)$ and $u_b = \text{tow}(2, 2, b)$, the thresholds in v_1 and v_2, respectively. The core of the proof follows from the following properties of t_b and u_b:

1. $t_0 = u_0 = 0$
2. $u_b = t_b + b$ for any $b \geq 1$
3. $t_b = \min_x\{\max(x, u_{b-1-x}) \mid 0 \leq x \leq b\}$ for any $b \geq 1$

Item 1 is trivial: both players bid 0, Player 1 wins ties, thus he wins all biddings (see Example 1). For Item 2, consider the configuration $\langle v_2, u_b, b\rangle$. Since v_2 neighbors s, it is dominant for Player 2 to bid all her budget b. In order to avoid losing, Player 1 must bid b, and the game proceeds to $\langle v_1, u_b - b, b\rangle$, thus $t_b = u_b - b$. For Item 3, consider a configuration $\langle v_1, x, b\rangle$ from which Player 1 wins, i.e., $x \geq t_b$. Note that it is dominant for Player 1 to bid his whole budget x. In order to avoid losing, Player 2 must bid $x + 1$, and proceed to $\langle v_2, x, b - (x + 1)\rangle$ from which Player 1 wins, thus $x \geq u_{b-(x+1)}$, and t_b is obtained from the minimal such x.

This gives us the system of three equations with three unknowns (for a fixed b), thus existence of an unique solution, if any. In the full version [9], we verify that the expressions $t_b = \lfloor \frac{b}{\phi}\rfloor$ and $u_b = \lfloor b \cdot \phi\rfloor$ satisfy the equations. □

Remark 16. The closed-form solution in Thm. 15 has a striking similarity to a classic result in Combinatorial Game Theory. *Wythoff Nim* is played by two players who alternate turns in removing chips from two stacks. A configuration of the game is $\langle s_1, s_2\rangle$, for integers $s_1 \geq s_2 \geq 0$, representing the number of chips placed on each stack. A player has two types of actions: (1) choose a stack and remove any $k > 0$ chips from that stack, i.e., proceed to $\langle s_1 - k, s_2\rangle$ or $\langle s_1, s_2 - k\rangle$, or (2) remove any $k > 0$ chips from both stacks, i.e., proceed to $\langle s_1 - k, s_2 - k\rangle$. The player who cannot move loses. Wythoff [24] identified the configurations from which the first player to move loses. Trivially, $\langle 0, 0\rangle$ is losing, followed by $\langle 1, 2\rangle$, $\langle 3, 5\rangle, \ldots$. In general, the n-th losing configuration is $\langle\lfloor n\cdot\phi\rfloor, \lfloor n\cdot\phi\rfloor+n\rangle$. Note the similarity to the thresholds in v_2 and v_1, which can be written respectively as $\langle\lfloor b \cdot \phi\rfloor, \lfloor b \cdot \phi\rfloor - b\rangle$, for $b \geq 0$.

Theorem 17. *For $b \geq 1$ we have $\text{tow}(3, 1, b) = \lfloor \frac{b-1}{2}\rfloor$, $\text{tow}(3, 2, b) = b - 1$, and $\text{tow}(3, 3, b) = 2b - 1$.*

Proof. We proceed similarly to the proof of Thm. 15. This time, we need to check that the expressions

$$t_b = \lfloor(b-1)/2\rfloor, \quad u_b = b-1, \quad \text{and} \quad v_b = 2b-1$$

satisfy the relations

1. $t_1 = u_1 = 0, v_1 = 1$,
2. $v_b = u_b + b$ for any $b \geq 2$,
3. $u_b = \min_x\{\max\{t_b + x, v_{b-1-x}\} \mid 0 \leq x \leq b\}$ for any $b \geq 2$.
4. $t_b = \min_x\{\max\{x, u_{b-1-x}\} \mid 0 \leq x \leq b\}$ for any $b \geq 2$.

This time, both Item 1 and Item 2 follow by direct substitution.

Regarding Item 3, we need to show that

$$b - 1 = \min_x\{\max\{\lfloor(b-1)/2\rfloor + x, 2b - 3 - 2x\} \mid 0 \leq x \leq b\}$$

To that end, we distinguish two cases based on the parity of b. This analysis can be found in the full version [9].

Finally, regarding Item 4 we have $u_{b-1-x} = b - 2 - x$, hence the two numbers inside the $\max(\cdot)$ function always sum up to $b-2$. Here too, we analyse by distinguishing the parity of b, and the detailed argument can be found in the full version [9]. □

We note that for $n \geq 4$ the situation gets surprisingly more complicated. For $n = 5$ the threshold budgets do eventually converge to a simple pattern, but only from around $b = 4 \cdot 10^3$ on. In contrast, for $n \in \{4, 6\}$ the threshold budgets exhibit no clear pattern up until $b = 10^6$. Moreover, while the pipe theorem Thm. 10 seems to hold for $n \leq 5$ (experimentally validated up to $b = 10^7$), it is (quickly) violated for $n \geq 6$. This suggests that a simple closed form solution for general games is unlikely, given that these structurally similar games behave so differently.

6 Algorithms for Threshold Budgets

In this section, we discuss an algorithmic approach to compute threshold budgets. We point out that the Pipe theorem (Thm. 10) only provides an approximation for the thresholds, and periodicity (Thm. 12) only holds eventually, thus, in order to use it, exact thresholds need to be computed until periodicity "kicks in". We study the following problem: Given a game $\mathcal{G}$, a vertex v in $\mathcal{G}$, and a budget B_2 of Player 2, determine $T_v(B_2)$. We develop an algorithm for general games, running in time pseudo-polynomial in B_2 and polynomial in $|\mathcal{G}|$, and then a specialized variant for DAGs which is pseudo-*linear* in B_2. In the following, we write B for an "arbitrary" Player 2 budget and B_2 for the particular budget for which we want to compute $T_v(B_2)$.

As a first step, we show that poorman discrete-bidding games end after a finite number of steps. Consider a vertex v. We define the *maximal step count*, denoted $\mathsf{Steps}_{\mathcal{G}}(B)$, to be the maximal number of steps Player 2 can delay reaching t when the initial budgets are B and $T_v(B)$ for Player 2 and Player 1, respectively, and Player 1 follows some winning strategy. Let $\mathsf{Steps}_{\mathcal{G}}(B) = \max_v T_v(B)$. The following lemma bounds $\mathsf{Steps}_{\mathcal{G}}(B)$.

Lemma 18. *Given a budget of $T_v(B_2)$, Player 1 can ensure winning after at most $\mathcal{O}(|V| \cdot B_2)$ steps.*

Proof. If Player 2 does not win a bid for $|V|$ steps, then Player 1 can surely move to the target t. Otherwise, Player 2 has to win at least one bid, decreasing the budget by at least 1 every $|V|$ steps. □

We note that this is a very crude approximation, we conjecture that actually $\mathsf{Steps}_{\mathcal{G}}(B) \in \mathcal{O}(\log B)$, as we explain later. However, the existence of such a bound already motivates us to consider the step-bounded variant of the game: Let $T_v^i(B)$ equal the minimal budget that Player 1 needs to ensure winning from v against a budget of B *in at most i steps* (or ∞ if this is not possible). By Lem. 18, $T_v^i(B) = T_v(B)$ for some large enough i. Thus, we are interested in computing $T_v^i(B)$ for increasing i until convergence. Let us briefly discuss simple cases. For the target vertex, clearly $T_t(B) = T_t^i(B) = 0$ for any Player 2 budget B and any i. For the sink, $T_s(B) = T_s^i(B) = \infty$, as well as $T_v^0(B) = \infty$ for all non-target vertices. As it turns out, we can compute all other values by a dynamic programming approach.

We first describe a recursive characterization of $T_v^i(B)$, which then immediately yields our algorithm. To this end, we consider the *step operator* $\mathrm{step}_B(v, f, b)$, which given a threshold function f (such as $T_v^i(B)$) and vertex v yields the outcome of placing bid b as Player 1 against a Player 2 budget B. The intuition is as follows: Suppose f is the actual threshold required to win in every vertex. There are two distinct cases. If Player 1 bids B, i.e. all of Player 2's budget, a win of the auction is guaranteed. Player 1 pays B and then naturally moves to the "cheapest" successor, i.e. one with minimal threshold as given by f. Otherwise, with a bid of $b < B$ by Player 1, Player 2 could either bid 0, again leaving Player 1 to pay b and choose the best option, or bid $b + 1$, i.e. Player 2 wins instead, paying the bid and choosing the most expensive successor. The overall best choice for Player 1 then directly is given as minimum over all sensible bids.

Definition 19. *Let B a budget for Player 2 and a function $f : V \times \{0, \dots, B\} \to \mathbb{N}$ yielding a threshold for each budget (e.g. $T_v^i(B)$). We define $\mathrm{step}_B(v, f, B) = B + \min_{v' \in N(v)} f(v', B)$ and, for any other bid $0 \le b < B$, let*

$$\mathrm{step}_B(v, f, b) = \max \begin{cases} b + \min_{v' \in N(v)} f(v', B) \\ \max_{v' \in N(v)} f(v', B - (b+1)) \end{cases}$$

Finally, $\mathrm{step}_B(v, f) = \min_{0 \le b \le B} \mathrm{step}_B(v, f, b)$.

Indeed, step allows us to iteratively compute T_v^i as follows:

Lemma 20. *For all $i > 0$, we have $T_v^i(B) = \mathrm{step}_B(v, T_\circ^{i-1})$.*

Proof. We proceed by induction over i. The correctness of the base cases follows immediately. To go from step $i - 1$ to i, observe that Player 1 surely never wants to bid more than B, since this bid suffices to guarantee winning. Moreover, for any fixed bid $b < B$, the opponent Player 2 either wants to bid 0, letting Player 1 win, or $b + 1$, claiming the win at minimal potential cost: Bidding anything between 0 and b as Player 2 does not change the outcome, and bidding more than $b + 1$ certainly is wasteful. By this observation, we can immediately see that for each potential bid b between 0 and B, $\mathrm{step}_B(v, T_\circ^{i-1}, b)$ yields the best possible outcome against an optimal opponent. In particular, if Player 1 bids b but the available budget is one smaller than $\mathrm{step}_B(v, T_\circ^{i-1}, b)$, then there exists a response of Player 2 where Player 1 is left with less budget than $T_{v'}^{i-1}(B')$ in some vertex v' against Player 2 budget B', which by induction hypothesis is not sufficient. □

This naturally gives rise to an iterative algorithm: Given budget B_2, we compute $T_v^i(B)$ for all vertices v and budgets $0 \le B \le B_2$ for increasing i until a fixpoint is reached. We briefly outline the algorithm in the full version [9, Algorithm 1].

At first glance, evaluating $\mathrm{step}_B(v, f)$ requires $\mathcal{O}(B \cdot |N(v)|)$ time – we need to consider all possible bids and go over all successors. Thus, to compute $T_v^i(B)$ for all $B \le B_2$ and vertices v takes $\mathcal{O}(B_2^2 \cdot |E|)$. (By our assumption, every vertex has at least one outgoing edge, meaning $|V| \in \mathcal{O}(|E|)$.) While the graphs (and thus $|E|$) we consider typically are small, quadratic dependence on B_2 is undesirable, since we may want to compute optimal solutions for considerably large budgets. It turns out that we can exploit some properties of $T_v^i(B)$ to obtain speed-ups.

Theorem 21. *For budget B_2 of Player 2, the threshold budget can be determined in $\mathcal{O}(\mathsf{Steps}_{\mathcal{G}}(B_2) \cdot B_2 \cdot \log(B_2) \cdot |E|)$.*

Proof. Observe that $T_v^i(B)$ is monotone in B: Winning against a larger budget of Player 2 certainly requires the same or more resources. Thus, the first expression of the maximum in Definition 19 is a (strictly) monotonically increasing function, while the second is decreasing. Together, the step function intuitively is convex in b: There is a "sweet spot", bidding too much is not worth it and bidding too little lets Player 2 gain too much. Consequently, we can determine $T_v^i(B)$ by a binary search between 0 and B. This yields a running time of $\mathcal{O}(\log B \cdot |N(v)|)$ for a fixed vertex v and budget B. In turn, to compute a complete step, i.e. for all vertices determine $T_v^i(B)$ for all budgets $B \le B_2$, we get $\mathcal{O}(B_2 \cdot \log(B_2) \cdot |E|)$. (Note that $\sum_{v \in V} |N(v)| = |V|$.) □

6.1 A Pseudo-Linear Algorithm for DAGs

Using insights of the previous section together with further observations, we can obtain tighter bounds in the case of DAGs. In particular, by exploiting both the given topological ordering as well as the bounds given by Thm. 10, we obtain an algorithm linear in the numerical value of B_2.

Theorem 22. *For a DAG game and any budget B_2 of Player 2, the threshold budget $T_v(B_2)$ can be determined in $\mathcal{O}(B_2 \cdot \log(|V|) \cdot |E|)$ steps for all vertices.*

Proof sketch. (See the full version [9] for the detailed proof.)

In essence, we use four observations. First, since the game is a DAG, we can fully compute $T_v(B)$ for each vertex and budget $0 \le B \le B_2$ at once by evaluating vertices in reverse topological order. Intuitively, each vertex only occurs at most once along any play in a DAG game.

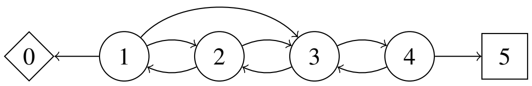


Figure 5: A small game where Thm. 10 is violated.

Thus, we only need to consider each vertex once. Second, we can exploit the budget bounds given by Thm. 10 to obtain lower and upper bounds on an optimal bid. The size of this interval as given by Thm. 10 depends on the magnitude of the continuous thresholds. Thirdly, we rely on actually knowing these thresholds. Thus, we give a bound on the size and computational complexity of determining them. Finally, applying binary search to this interval, using the insights of Thm. 21, yields the result. □

7 Experiments and Conjectures

In this section, we present several experimental results which in turn motivate conjectures for general games.

The Pipe Theorem In our experiments, we observed that Thm. 10 does not hold for all general graphs. We depict the smallest bidding game we found where Thm. 10 is violated in Fig. 5. We note that this game has an interesting structure: It is a "normal" tug of war game, with a single edge added. Moreover, whenever this "gadget" is a part of a game, the same problem arises. However, this structure is not the only potential cause: While the pipe theorem even seems to hold for tug of war games of up to 5 interior states (validated up to $B_2 = 10^7$), we observed that it is violated for 6 or more.

Conjectures on General Graphs Despite this apparently chaotic behaviour, we observed that a variant of Thm. 10 seems to be satisfied in general.

Conjecture 23. *In any game and vertex v, we have that*

$$t_v \cdot B_2 - \mathcal{O}(\log B_2) \leq T_v(B_2).$$

Consider Fig. 6, where we plot the difference $d(B) = t_v \cdot B - T_v(B)$ for a tug-of-war game with 21 states. The x-axis, i.e. Player 2's budget B, is scaled logarithmically. If the conjecture holds, then $d(B) \in \mathcal{O}(\log B)$, which would appear as a line on such a graph. And indeed, we clearly see a linear "pipe". We observed similar graphs for all investigated games.

Based on experimental evidence, we believe that the underlying reason is similar to the proof idea of Thm. 10, namely that for large budgets, the actual bids do not differ too much from the continuous behaviour.

Conjecture 24. *Winning bids are proportional to the current budget in play, i.e. for each vertex there is a ratio r_v such that all winning bis are $b = r_v \cdot B_2 + \mathcal{O}(1)$.*

In Fig. 6 we also display optimal bids for Player 1 in relation to Player 2's budget. A clear linear dependence with a ratio of approximately $r_v \approx \frac{1}{3}$ is visible.

This implies our "pipe conjecture" as follows: When bids are proportional to the budget, then the total budget in play decays exponentially. Thus, the length of the game is logarithmic in the available budget, i.e. $\mathsf{Steps}_{\mathcal{G}}(B_2) \in \mathcal{O}(\log B_2)$ for a fixed game. Recall that in Thm. 10 we prove the lower bound by arguing that Player 2 needs a "+1" at most $|V|$ times to compensate for rounding. With this general step bound, we can similarly argue that this is required at most logarithmic number of times. In other words, Player 1 can exploit the "rounding advantage" only logarithmically often. We also mention that this would then put the complexity of our general algorithm at $\mathcal{O}(B_2 \cdot \log(B_2) \cdot |E|)$.

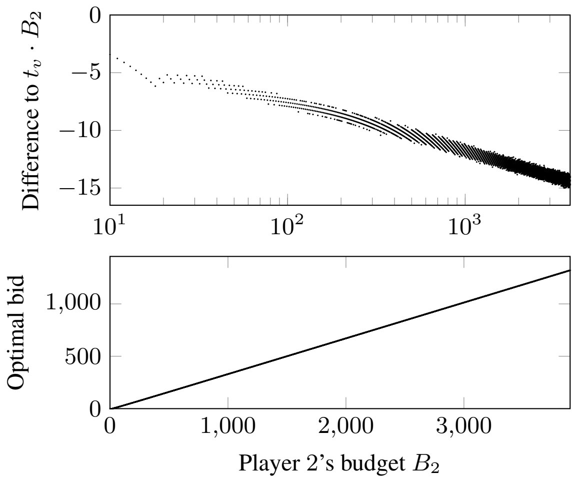


Figure 6: Plot of $t_v \cdot B_2 - T_v(B_2)$ (logarithmic) and Player 1's optimal bids in state 1 of a tug-of-war with 21 states.

Implementation and Performance We implemented our algorithm in Java (executed with OpenJDK 17) and ran it on consumer hardware (AMD Ryzen 3600). Generation of games and visualization of results was done using Python scripts.

While not the focus of our evaluation, we observed that our implementation can easily handle large graphs and budgets. For example, solving a tug of war game with 20 states and $B_2 = 10^6$ took around 1 minute (483 steps).

8 Conclusion

We study, for the first time, bidding games that combine poorman with discrete bidding. On the negative side, threshold budgets in poorman discrete-bidding games exhibit complex behavior already in simple games, in particular in games with cycles. On the positive side, we identify interesting structure: we prove determinacy, in DAGs, we relate the threshold budgets with continuous ratios, and prove that thresholds are periodic. Additionally, our implementation efficiently computes exact solutions to non-trivial games. We particularly invite the interested reader to explore bidding games using it, the code will be available on demand.

Our work opens several venues for future work:

Theoretically, we left several open problems and conjectures. Beyond that, poorman discrete-bidding is more amendable to extensions when compared with poorman continuous-bidding, which quickly becomes technically challenging, or Richman discrete-bidding, which is a rigid mechanism. For example, it is interesting to introduce into the basic model, multi-players or complex objectives, e.g., that take into account left over budgets [15].

Practically, poorman is more popular than Richman bidding since it coincides with the popular first-price auction and discrete- is more popular than continuous-bidding since most if not all practical applications employ some granularity constraints on bids. It is interesting to develop applications based on these games. For example, to analyze and develop bidding strategies in sequential auctions or fair allocation of goods [11]. Further, it is interesting to study *mechanism design*: synthesize an arena so that the game has guarantees (e.g., [21]).

Acknowledgements

This research was supported in part by ISF grant no. 1679/21, ERC CoG 863818 (FoRM-SMArt) and the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie Grant Agreement No. 665385.

References

[1] M. Aghajohari, G. Avni, and T. A. Henzinger, 'Determinacy in discrete-bidding infinite-duration games', *Log. Methods Comput. Sci.*, **17**(1), (2021).
[2] R. Alur, T. A. Henzinger, and O. Kupferman, 'Alternating-time temporal logic', *J. ACM*, **49**(5), 672–713, (2002).
[3] G. Avni, T. A. Henzinger, and V. Chonev, 'Infinite-duration bidding games', *J. ACM*, **66**(4), 31:1–31:29, (2019).
[4] G. Avni, T. A. Henzinger, and R. Ibsen-Jensen, 'Infinite-duration poorman-bidding games', in *Proc. 14th WINE*, volume 11316 of *LNCS*, pp. 21–36. Springer, (2018).
[5] G. Avni, T. A. Henzinger, and D. Zikelic, 'Bidding mechanisms in graph games', *J. Comput. Syst. Sci.*, **119**, 133–144, (2021).
[6] G. Avni, R. Ibsen-Jensen, and J. Tkadlec, 'All-pay bidding games on graphs', in *Proc. 34th AAAI*, pp. 1798–1805. AAAI Press, (2020).
[7] G. Avni, I. Jecker, and Đ. Žikelić, 'Infinite-duration all-pay bidding games', in *Proc. 32nd SODA*, pp. 617–636, (2021).
[8] G. Avni, I. Jecker, and D. Zikelic, 'Bidding graph games with partially-observable budgets', in *Proc. 37th AAAI*, (2023).
[9] G. Avni, T. Meggendorfer, S. Sadhukhan, J. Tkadlec, and Đ. Žikelić, 'Reachability poorman discrete-bidding games', *CoRR*, (2023). https://arxiv.org/abs/2307.15218.
[10] G. Avni and S. Sadhukhan, 'Computing threshold budgets in discrete bidding games', in *Proc. 42nd FSTTCS*, (2022).
[11] M. Babaioff, T. Ezra, and U. Feige, 'On best-of-both-worlds fair-share allocations', in *Proc. 18th WINE*, volume 13778 of *LNCS*, pp. 237–255. Springer, (2022).
[12] E. Borel, 'La théorie du jeu les équations intégrales á noyau symétrique', *Comptes Rendus de l'Académie*, **173**(1304–1308), 58, (1921).
[13] M. Develin and S. Payne, 'Discrete bidding games', *The Electronic Journal of Combinatorics*, **17**(1), R85, (2010).
[14] I. L. Gale and M. Stegeman, 'Sequential auctions of endogenously valued objects', *Games and Economic Behavior*, **36**(1), 74–103, (2001).
[15] Z. Huang, N. R Devanur, and D. Malec, 'Sequential auctions of identical items with budget-constrained bidders', *CoRR*, **abs/1209.1698**, (2012).
[16] A. J. Lazarus, D. E. Loeb, J. G. Propp, W. R. Stromquist, and D. H. Ullman, 'Combinatorial games under auction play', *Games and Economic Behavior*, **27**(2), 229–264, (1999).
[17] A. J. Lazarus, D. E. Loeb, J. G. Propp, and D. Ullman, 'Richman games', *Games of No Chance*, **29**, 439–449, (1996).
[18] R. Paes Leme, V. Syrgkanis, and É. Tardos, 'Sequential auctions and externalities', in *Proc. 23rd SODA*, pp. 869–886, (2012).
[19] D.A. Martin, 'Borel determinacy', *Annals of Mathematics*, **65**, 363–371, (1975).
[20] R. Meir, G. Kalai, and M. Tennenholtz, 'Bidding games and efficient allocations', *Games and Economic Behavior*, **112**, 166–193, (2018).
[21] M. Mittelmann, B. Maubert, A. Murano, and L. Perrussel, 'Automated synthesis of mechanisms', in *Proc. 31st IJCAI*, pp. 426–432. ijcai.org, (2022).
[22] Y. Peres, O. Schramm, S. Sheffield, and D. B. Wilson, 'Tug-of-war and the infinity laplacian', *J. Amer. Math. Soc.*, **22**, 167–210, (2009).
[23] A. Pnueli and R. Rosner, 'On the synthesis of a reactive module', in *Proc. 16th POPL*, pp. 179–190, (1989).
[24] W.A. Wythoff, 'A modification of the game of nim', *Nieuw Archief voor Wiskunde*, **7**(2), 199–202, (1907).