Monte Carlo Tree Search Bidding Strategy for Simultaneous Ascending Auctions

Alexandre Pacaud[1,2], Marceau Coupechoux[1], and Aurelien Bechler[2]

[1]LTCI, Telecom Paris, Institut Polytechnique de Paris, France
[2]Orange Labs, France

***Abstract*—We tackle in this work the problem for a player to efficiently bid in Simultaneous Ascending Auctions (SAA). Although the success of SAA partially comes from its relative simplicity, bidding efficiently in such an auction is complicated as it presents a number of complex strategical problems. No generic algorithm or analytical solution has yet been able to compute the optimal bidding strategy in face of such complexities. By modelling the auction as a turn-based deterministic game with complete information, we propose the first algorithm which tackles simultaneously two of its main issues: *exposure* and *own price effect*. Our bidding strategy is computed by *Monte Carlo Tree Search* (MCTS) which relies on a new method for the prediction of closing prices. We show that our algorithm significantly outperforms state-of-the-art existing bidding methods. More precisely, our algorithm achieves a higher expected utility by taking lower risks than existing strategies.**

***Index Terms*—MCTS, Ascending Auctions, Exposure, Own Price Effect, Budget Constraints**

I. Introduction

In 2019, telecommunications companies began to deploy the latest technology standard for broadband cellular networks (5G). Most countries have decided to allocate the spectrum to mobile companies through auctions. One of the most widespread auction designs is the Simultaneous Ascending Auction (SAA) [6], also known as Simultaneous Multi Round Auction (SMRA), which has been used by many countries like Portugal, Germany or Italy to sell the 5G spectrum licences. This auction has the particularity of having a dynamic multi-round format in which bids are submitted simultaneously on all items. It ends when no new bids have been submitted during a round. The SAA was first introduced in 1994 by the US Federal Communications Commission (FCC) for the allocation of wireless spectrum rights. It was developed by Paul Milgrom and Robert Wilson who both received the 2020 Sveriges Riksbank Prize in Economic Sciences in Memory of Alfred Nobel, mainly for their contributions to the SAA. One of the reasons for SAA success is that bidders can freely adjust their bids throughout the auction while taking into account the latest information about the likelihood of acquiring different subsets of objects. Hence, bidders can progressively adapt their strategies during the whole course of the auction. This leaves room for a wide range of bidding strategies.

However, auction theory or exact game resolution methods are unable to compute the optimal bidding strategies due to the high complexity of the game. Indeed, SAA induces a n-player simultaneous move game with incomplete information and large state space for the solution of which there is no known generic algorithm [18]. In addition to the above difficulties, a number of strategical complexities adds up due to the rules and mechanism of the SAA. The main strategical challenges are known as *exposure problem*, *own price effect*, *budget constraints* and *activity rules*. In this work, we focus on the two first ones. The *exposure problem* refers to the possibility that, by bidding on a set of complementary goods, a bidder ends up paying more than its valuation for the subset it actually wins as the goods have become too expensive. The *own price effect* corresponds to the fact that bidding on an object systemically raises the price of this object and, thus, diminishes the utility of all players willing to acquire it.

A. Related works

To the best of our knowledge, no work has yet been proposed to bid efficiently in SAA by tackling simultaneously the *exposure problem* and *own price effect*. Until now, these issues have only been studied separately, generally in simplified versions of SAA.

For instance, from the regulator's point of view, the reduction of allocational efficiency and revenue caused by the *exposure problem* has been studied in toy examples of SAA with complete information by Milgrom [15] and Bykowsky et al. [4]. From the bidder's point of view, Goeree et al [8] compute the bidder's optimal drop-out level through a Bayesian framework in homogeneous-good environments. To do so, they consider a clock version of the SAA where participants are either *local bidders* or *global bidders* with super-additive value functions. They extend their work to the case of two global bidders with regional complementarities. Zheng [25] proposes a modified clock version of the SAA in which a continuation equilibrium can be built to fully eliminate the exposure problem using *jump bidding* in a two object auction with one global bidder. One of the very few works which has tried to tackle the exposure problem in the original version of SAA [6] is from Wellman et al. [24]. They propose an algorithm based on probabilistic predictions of final prices to tackle this issue. Their algorithm obtained promising results but experiments were only undertaken with players having specific super-additive value functions. Thus, it is difficult to conclude on the effectiveness of the algorithm in more generic settings.

ISBN 978-3-903176-49-2 © 2022 IFIP

Regarding *own price effect*, Milgrom [15] describes a *collusive equilibrium* in an example of a 2-object SAA with complete information between two players having the same value functions. Following this work, Brusco and Lopomo [3] construct a collusive equilibrium in a 2-object SAA between two players in the case of additive and super-additive value functions through signalling of the most valuable item. They show that the scope of collusion narrows when the ratio between the number of bidders and the number of object increases. By probabilistically predicting final prices, Wellman et al. [24] build a simple algorithm in a homogeneous-good environment where all players have subadditive value function but obtained results are unsatisfactory.

B. Contributions

In this paper, we make the following contributions:

- We introduce a simplified version of SAA which is turn-based, deterministic, with complete and perfect information. We call it d-SAA.
- We present a bidding strategy that tackles efficiently and simultaneously the *exposure problem* and *own price effect* in d-SAA using Monte Carlo Tree Search (MCTS)[1] [2]. To the best of our knowledge, it is the first algorithm that tackles both of these issues in any version of SAA. Compared to the literature, we see our MCTS for d-SAA as a step forward for the design of efficient generic bidding strategies for SAA.
- We propose a simple method to compute the prediction of closing prices through the convergence of a specific sequence. This is used to enhance our MCTS rollout strategy.
- Extensive numerical experiments on typical examples from the literature and a large number of random instances show that our MCTS bidding strategy significantly outperforms state-of-the-art algorithms and achieves higher expected utility by taking less risks.

II. d-SAA DESCRIPTION AND STRATEGICAL COMPLEXITIES

A. Deterministic model with complete and perfect information of SAA

In our work, we focus on a deterministic turn-based model with perfect information of SAA [6], [15], [24] which we refer to as d-SAA. In this specific auction, m indivisible goods are sold via separate and concurrent English auctions between n players. Bidding occurs in multiple rounds. Players take turns bidding such that, at each round, only the designated player is allowed to bid. The *bid price* of item j, denoted P_j, corresponds to the highest bid obtained so far for item j and, thus, is its current selling price. At the beginning of each round, the bid price and the current winner of each item is announced to each player. Hence, players are aware of all past actions taken by their opponents. All bid prices are initialised to zero. For each item j, players can only submit a bid P_j^{new} equal to the current bid price P_j plus the fixed bid increment ε: $P_j^{new} = P_j + \varepsilon$. This simplification of the bidding space is common in the literature on ascending auctions [8], [24]. As in the original SAA [15], the auction ends when none of the players have submitted an admissible bid during their last round. This is equivalent in our model to saying that the bid price of each item has not increased during n consecutive rounds. We make the classical assumption that a player will not continue bidding on an item that it has temporarily won [24]. Hence, none of the players can suffer from the winner's curse as the players obtain the goods at a price ε above the highest bid of their opponents [14].

The only difference between d-SAA and SAA is that players take turns bidding. By this change, we eliminate stochasticity and simultaneity from our problem. Indeed, ties between players having bid the same amount on a given item can not occur in d-SAA. Hence, the temporal winner is no longer selected randomly amongst these players. This facilitates future studies and the conception of a simpler tree-search algorithm with no chance nodes [22] and with a closed-loop implementation [17]. From the bidder's point of view, both mechanisms are strategically equivalent if the bid increment is small.

We assume that d-SAA game is a complete information game [19], [20]. Therefore, the value function v_i of each player i is common knowledge. Such assumption is relevant in spectrum auctions as generally bidders have a relatively precise estimation of their opponents' valuations.

B. Utility and Value functions

Each player i participating in the d-SAA game is defined by its value function v_i and its utility function σ_i. At the end of the auction, if player i wins the set of goods X and the bid price vector is P, then its utility is:

$$\sigma_i(X, P) = v_i(X) - \sum_{j \in X} P_j \tag{1}$$

We assume that value functions are normalised ($v_i(\emptyset) = 0$), finite and verify the free disposal condition, i.e., for any two subsets X and Y, $X \subset Y$ implies $v(X) \leq v(Y)$ [12], [15]. A set of goods X is said to exhibit complementarities with a disjoint set of goods Y if $v(X+Y) > v(X) + v(Y)$. Such preferences induce the strategical issue of exposure.

C. Game and strategical complexities

1) Game complexities: To compute the complexity of the d-SAA game, we consider its extensive form, i.e. a game tree representation, defined by nodes corresponding to different states of the game and by edges representing the different feasible bids in this state. A state is defined by the player to bid, the current bid price of each item and their respective temporary winner. Two metrics are often used: *state space complexity* and *game tree complexity* [21]. The first refers to the number of different possible states which can be reached legally from the initial state of the game while the second

[1] MCTS has already been used for bidding in other contexts such as Periodic Double Auctions [5]. However, it is the first time that a bidding strategy is computed by MCTS in any version of SAA.

specifies the number of possible different paths in its extensive form. With the assumption of finite valuations, the auction finishes after a finite number of rounds. Thus, we restrict our analysis to a given number of rounds R. It is relatively straightforward to prove that the state space complexity of an instance Γ of a d-SAA game with n bidders, m distinguishable objects and R rounds is equal to

$$\sum_{i'=0}^{n-1}(1+\sum_{i=0}^{n-1}(R-i-i')^+)^m \mathbb{1}_{\{R\geq i'\}} \tag{2}$$

where $\mathbb{1}_{\{z\geq w\}}$ is the indicator function, equal to 1 if $z \geq w$ and 0 otherwise.

It can also be shown that a lower bound of the game tree complexity of Γ is $2^{m(n-1)\lfloor \frac{R}{n} \rfloor}$.

Example. The SAA, which took place in Italy in 2018, had 12 5G spectrum licences sold between 5 telecommunication companies after 171 rounds [7]. For such values, the state space and game tree complexities of the corresponding d-SAA game are respectively greater than 10^{35} and 10^{491}.

2) Strategical complexities: To the difficulties generated by the high state space and game tree complexities of the d-SAA game, a number of strategical issues adds up due to the rules and mechanism of this specific auction.

- **Exposure problem [8]:** This occurs when a bidder tries to acquire a set of complementary goods but ends up paying too much for only a subset of these goods as the competition was tougher than expected. Thus, the exposed bidder incurs a loss. A famous situation of exposition [24] is a 2-item auction sale ($\varepsilon = 1$) between a player who considers both goods as perfect complements and a player who considers both goods as perfect substitutes. In this particular auction presented in Table I (referred to as Example 1 hereafter), a rational strategy for player 1 is to bid on the cheapest item if the bid price of this item is lower than $12 - \varepsilon$ and player 1 is temporarily winning no items or to pass its turn otherwise. Given the fact that player 1 plays rationally, if player 2 bids on an item, player 2 will end up exposed as it will not be able to acquire both items for a price inferior to 22.

	$v(\{1\})$	$v(\{2\})$	$v(\{1,2\})$
Player 1	12	12	12
Player 2	0	0	20

TABLE I: Example of exposition.

- **Own price effect problem [24]:** Each time a player bids on a item, its bid price rises which results in the decrease of utility of all players wishing to acquire this item. Therefore, as players have a mutual interest of keeping prices low, a new strategic behaviour emerges named *demand reduction* [1], [23]. This strategy consists in conceding items to your opponents so that they do not raise the bid price of the goods you are temporarily winning. The fact that players divide goods among themselves to keep prices low is called collusion [3]. It is important to note than there is no explicit communication between players as it is illegal. Nevertheless, players can still intuit how their opponents are going to bid based on their valuation estimates and, from this prior, coordinate themselves to form a collusion.

D. Performance indicators

The *expected utility* is a usual metric for evaluating the efficiency of a bidding algorithm. However, a spectrum auction is generally played only once by an operator and the amount of money involved is huge. It is thus important for a bidder to minimise the risk of exposure. We introduce two metrics related to the exposure problem. The *exposure frequency* is defined as the number of times a strategy ends up with a negative payoff divided by the number of plays. This estimates the probability of ending up exposed. The *cumulative loss* is defined as the sum of negative payoffs obtained in all plays. It measures the magnitude of the losses due to exposure. To analyse the own price effect, we consider the *average price payed per item won*. However, acquiring only undesired items at a reasonably low price could potentially explain low values of this average price. We thus complement this metric with the *average number of items won*.

III. MCTS BIDDING STRATEGY

Monte Carlo Tree search (MCTS) is a breadth-first search method which builds iteratively a search tree and runs Monte-Carlo simulations at its leaf nodes. Each node represents a possible future state of the game. The directed links between nodes and their children correspond to the actions leading to the next states. MCTS repeats a process named search iteration generally divided into four phases until some predefined computational budget (time, memory, iteration constraint) is reached. The four different phases are named selection, expansion, rollout and backpropagation. We implement a variant of MCTS-max^n [16]. Each node x stores the following statistics: the sum of rewards r_x found in the corresponding subtree, the number of visits n_x, the estimated lower bound a_x and the estimated higher bound b_x of the reward support. The four different phases of the search iteration are described below.

A. Selection

The selection phase consists in selecting a path from the root to a leaf node of the search tree. We propose a new version of selection index UCT [10] based on Hoeffding inequality [9] and an online estimation of the size of the reward support. It is adapted to any scale and underlying distribution of rewards. From a selected node y, the selection strategy chooses the child x with the highest score q_x:

$$\begin{aligned} q_x = \frac{r_x}{n_x} + \max(b_x - a_x, \varepsilon)\sqrt{\frac{2\log(n_y)}{n_x}} \\ -no\_obj(x) - risky\_move(x) \end{aligned} \tag{3}$$

where r_x is the sum of rewards found in the subtree with root x, n_x is the number of visits of child node x, n_y is the number of visits of parent node y, ε is the bid increment, a_x is the estimated lower bound and b_x is the estimated higher bound of the reward support found in the subtree with root x.

The two first terms come directly from *Hoeffding's Inequality* and give an upper confidence bound on the average reward $\frac{r_x}{n_x}$ found in the corresponding subtree using $\max(b_x - a_x, \varepsilon)$ as an estimation of the size of the reward support.

The third term $no\_obj(x)$ is a penalty term which has been introduced to avoid players from passing their turn if they have got nothing to lose by bidding on an additional item and, thus, emulate rational behaviour. For instance, if a player is currently winning a set of undesired goods and can increase its utility by directly acquiring another item, it should continue to bid even though its chances of obtaining any other item might be slim. We define $no\_obj(x)$ as the maximum surplus a player could obtain by just acquiring another item under the assumption that the auction ends after its turn. More formally, let S_{-j} be the sets of goods which do not contain item j. A player i' will no longer bid on item j if $\forall X \in S_{-j}, P_j + \varepsilon \geq v_{i'}(X + \{j\}) - v_{i'}(X)$. Thus, $\Pi_j^i = \max_{i' \in \{1,...,n\} \setminus \{i\}} \max_{X \in S_{-j}} v_{i'}(X+\{j\}) - v_{i'}(X) - \varepsilon$ is the minimal bid price from which item j is considered as undesired by all opponents of i. If P^x is the price vector at child node x, i is the player bidding at parent node y and X_x^i the set of goods temporarily won by player i at x, the penalty term is defined as

$$no\_obj(x) = \begin{cases} \max_{j \in \{1,...,m\} \setminus X_x^i} (v_i(X_x^i + \{j\}) - v_i(X_x^i) \\ -P_j - \varepsilon)^+ \quad \text{if } \{j' \in X_x^i, P_x^{j'} < \Pi_{j'}^i\} = \emptyset \\ 0 \quad \text{otherwise} \end{cases} \tag{4}$$

The fourth term $risky\_move(x)$ is a penalty term which has been introduced to deter players from bidding on sets of goods which might lead to exposure. These are sets that contain a subset of goods which, if acquired by the player i bidding at parent node y, will result in a negative utility. The actions leading to such sets are penalised by $\lambda_i v_i(\{1, ..., m\})$ with $\lambda_i \in [0, 1]$ a risk-aversion hyperparameter. Depending on whether i is the root player r or not, λ_i either takes the value λ^r or λ^o. The term $v_i(\{1, ..., m\})$ acts as a scaling factor. More formally, if P^x is the price vector at child node x, r is the root player, i is the player bidding at parent node y and X_x^i the set of goods temporarily won by player i at node x, then X_x^i is said to lead to exposure if $\exists Y \subseteq X_x^i, \sigma_i(Y, P^x) < 0$ and the penalty term is defined as

$$risky\_move(x) = \begin{cases} \lambda^r v_i(\{1, ..., m\}) \quad \text{if } X_x^i \text{ can lead to} \\ \text{exposure for } i = r \text{ at price } P^x \\ \lambda^o v_i(\{1, ..., m\}) \quad \text{if } X_x^i \text{ can lead to} \\ \text{exposure for } i \neq r \text{ at price } P^x \\ 0 \quad \text{otherwise} \end{cases} \tag{5}$$

B. Expansion

The expansion phase consists in choosing which children of the leaf node obtained in the selection phase are expanded to the search tree. In our MCTS, a node x is chosen randomly amongst the non-expanded children of the selected leaf node and added to the search tree. The statistics of node x are set the following way: $r_x = 0$, $n_x = 0$, $a_x = +\infty$ and $b_x = -\infty$.

C. Rollout phase

In the rollout phase, moves are played starting from the newly added node using a rollout strategy until the game ends in order to simulate the outcome of the game from this particular node. As the game considered is a n-player game, the outcome is a vector of size n where each index corresponds to the utility obtained by each player at the end of the simulation. To simulate a game, the default rollout strategy used in MCTS is usually to play randomly. However, in the considered game, this leads, in expectation, to absurd outcomes with extremely high prices as the probability that everybody passes their turn consecutively is inferior to $\frac{1}{2^{(n-1)m}}$. Therefore, to guarantee good sampling, we propose a new rollout strategy which we describe in Section IV.

D. Backpropagation

The backpropagation phase consists in propagating backwards the outcome of the game obtained in the rollout phase from the newly added node to the root node to update the statistics stored in each selected node. Let V be the vector of utility obtained in the rollout phase. Let x be a selected node which has y as parent node. Let i be the player playing at node y. The statistics at node x are updated as follows: $r_x \leftarrow r_x + V_i$, $n_x \leftarrow n_x + 1$, $a_x \leftarrow \min(a_x, V_i)$ and $b_x \leftarrow \max(b_x, V_i)$.

E. Final move selection

The four above steps run until a computational criteria (time, memory, iteration constraint) is reached. A final move selection is then performed to choose which action to play. We decide to select the move leading to the child with the highest penalised average. More formally, our final move selection chooses to play the action which leads to child x with the highest quantity q_x defined as

$$q_x = \frac{r_x}{n_x} - no\_obj(x) - risky\_move(x) \tag{6}$$

IV. Rollout Strategy with Domain Knowledge

In this section, we present our MCTS rollout strategy which relies on a new method of prediction of closing prices.

A. Point-price prediction bidding

Definition 4.1. [24] A point-price prediction bidder computes the subset of goods

$$X^* = \operatorname*{argmax}_X \sigma(X, \rho(\mathcal{B})) \tag{7}$$

breaking ties in favour of smaller subsets and lower-numbered goods. Every subset of goods X contains all the items the bidder is temporarily winning. The bidder then bids $P_j + \varepsilon$ on all items j that it is not currently winning in X^*. The function $\rho : \mathcal{B} \rightarrow \mathbf{R}_+{}^m$ maps the bidder's information state $\mathcal{B}$ to an estimation of the final prices of all objects $\rho(\mathcal{B})$ using only the initial estimation of the final prices of the items $\rho(\mathcal{B}_0)$ and follows the below update rule:

$$\rho_j(\mathcal{B}) = \begin{cases} \max(\rho_j(\mathcal{B}_0), P_j) & \text{if winning good j} \\ \max(\rho_j(\mathcal{B}_0), P_j + \varepsilon) & \text{otherwise} \end{cases} \tag{8}$$

where $\mathcal{B}_0$ is the null information state of the game. We refer to this bidding strategy as *PP*.

In d-SAA, $\mathcal{B}$ can be described by the current winner and bid price of each item. If so, $\mathcal{B}_0$ is defined by the fact that each item is allocated to the auctioneer at bid price 0.

If a player knows the final price of all items and these prices are independent of its bidding strategy, then playing according to PP with $\rho(\mathcal{B}_0)$ equal to the actual final prices is optimal [24]. However, such conditions rarely happen in practice. An extreme case is the well-studied straightforward bidding strategy (*SB*) [15] which corresponds to a PP with null initial estimation of final prices ($\rho(\mathcal{B}_0) = 0$). The quality of PP fully depends on the initial prediction of final prices. Indeed, if the initial prediction considerably underestimates the final price of each item, then when the bid prices exceed the initial prediction the bidder plays exactly SB. Conversely, if one's initial prediction overestimates too much the final prices of an auction, the player might stop bidding prematurely.

B. Initial estimation of final prices

There exist many different ways in literature to compute an initial prediction of final prices $\rho(\mathcal{B}_0)$. However, all existing methods seem to present some deficiencies in d-SAA. For instance, concepts such as Walrasian price equilibrium or self-confirming price prediction [24] do not always exist when preferences exhibit complementarities as in Example 1 [24]. Tâtonnement processes such as the one presented by Wellman et al. to compute the *expected price equilibrium* [24] do not take into account the auction's mechanism and, thus, return the same price vector regardless of the auction's specificities such as the bid increment. Predictions based on the closing price of a specific strategy profile seem relevant only if each bidder eventually decides to play the corresponding strategy of this particular strategy profile. Thus, we propose below a new method of prediction of final prices fixing these flaws.

Definition 4.2. For any instance Γ of d-SAA, let $f_\Gamma : \mathbb{R}_+^m \mapsto \varepsilon\mathbb{N}^m$ be the function mapping any vector prices $p \in \mathbb{R}_+^m$ to the final prices $f_\Gamma(p)$ obtained in Γ when all players play PP with initial prediction p.

Theoretically, a closed-form expression of f_Γ can be computed for any instance Γ of d-SAA. However, in practice, this can only be done for small instances. Thus, $f_\Gamma(p)$ is usually computed numerically by simulating a d-SAA where all players play PP with initial prediction p. In Example 1, for $p = (11, 11)$, $f_\Gamma(p) = (1, 0)$ as player 1 bids on the item of lowest index and player 2 does not bid as it predicts that the price for both items is 22.

Conjecture 4.1. *Let Γ be an instance of d-SAA. The sequence $P_{t+1} = \frac{1}{t+1} f_\Gamma(P_t) + (1 - \frac{1}{t+1}) P_t$ with P_0 the null vector of prices converges to an element P^*.*

It can be shown that the above conjectured limit P^* corresponds either to a fixed point of f_Γ known as self-confirming price prediction [24] or to a jump discontinuity of f_Γ reflecting a change in strategies by one or more players. In the latter case, an initial prediction greater than P^* will generally deter point-price prediction bidders from bidding, notably to avoid exposure, while a smaller initial prediction will lead to high final prices. The proof of the conjecture is left for future work.

Computing our initial prediction as above fixes the flaws of existing methods. Indeed, (i) we observe that our sequence converges in all undertaken numerical instances, (ii) our approach takes into account the auction's specificities through f_Γ and (iii) it does not rely on the results of a single specific strategy profile by construction.

C. Rollout strategy

Before running our MCTS, we compute the conjectured limit P^* as above. Then, at the beginning of each rollout phase, we set $\rho(\mathcal{B}_0) = P^* + \eta$ where η is a random variable which follows a bounded uniform distribution $U([-\varepsilon, \varepsilon]^m)$. During the simulation, each bidder plays PP with initial prediction $\rho(\mathcal{B}_0)$. Ties are still broken in favour of smaller subsets but no longer by selecting the goods with the lowest indices. Instead, they are chosen randomly. By applying noise to the initial prediction and by breaking ties randomly, players bidding behaviours are diversified at each new simulation which improves the quality of sampling of our MCTS.

V. Main results

We now analyse the performance of our MCTS bidding strategy in a variety of d-SAA games by comparing it to five other strategies: an MCTS which is similar to ours but without the penalties (*MCTS^{np}*), a *UCB* algorithm [11] using the same simulations and selection index as our MCTS but without the penalties, a PP using the expected price equilibrium (*EPE*) as initial price prediction [24], a distribution-price bidding strategy using self-confirming price distribution (*SCPD*) as initial price prediction [24] and SB [15]. It is important to note that bidders are unaware of their opponents' strategy.

The hyperparameters used for our MCTS bidding strategy are $\lambda^r = 0.07$ and $\lambda^o = 0.07$. They are obtained by grid-search. The experiments were run on a server consisting of Intel®Xeon®E5-2699 v4 2.2GHz processors. All algorithms were given a maximum of 30 seconds CPU thinking time.

A. Test cases

One of the biggest advantages of our MCTS compared to other existing methods is that it is able to judge pertinently whether it is more beneficial to adopt a demand reduction strategy to keep prices low or to bid straightforwardly on a set of goods. To highlight this capacity, we use an experiment from [3] in which two players participate in a 2-item auction with additive value functions. The first player values each

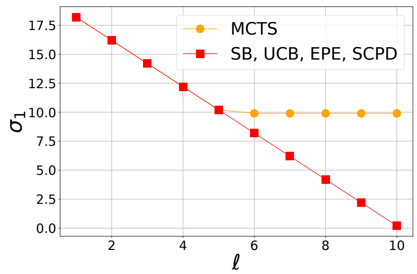


Fig. 1: Evolution of player 1's utility σ_1 depending on strategy versus player 2's valuation ℓ in Test experiment [3] ($h = 10$, $\varepsilon = 0.1$) given that player 2 plays optimally

object as h and the second values each one as $0 < \ell \leq h$. For an infinitesimal bid increment ε, it is more worthwhile for the first player to bid competitively on both goods if $h < 2(h-\ell)$. However, if $h > 2(h-\ell)$, then the first player will be better off by forming a collusion and conceding an item to its opponent. The four algorithms UCB, EPE, SCPD and SB always suggest to bid straightforwardly even if $h > 2(h-\ell)$ and, thus, never propose demand reduction even in situations where it is highly beneficial. However, our MCTS always adopts the appropriate strategy. We plot in Figure 1 the payoff σ_1 obtained by player 1 for each strategy given that player 2 plays optimally, i.e., continues to bid on the cheapest item while its bid price is inferior to ℓ and player 1 has not conceded an item to player 2. This example shows that our algorithm chooses the most beneficial strategy, at least in simple environments.

Moreover, our MCTS is capable of avoiding obvious exposure. For instance, in Example 1, our MCTS suggests to player 2 not to bid. EPE and UCB also prevent player 2 from bidding. However, this is not the case for SCPD and SB which expose player 2 by inciting player 2 to bid aggressively.

B. Extensive experiments

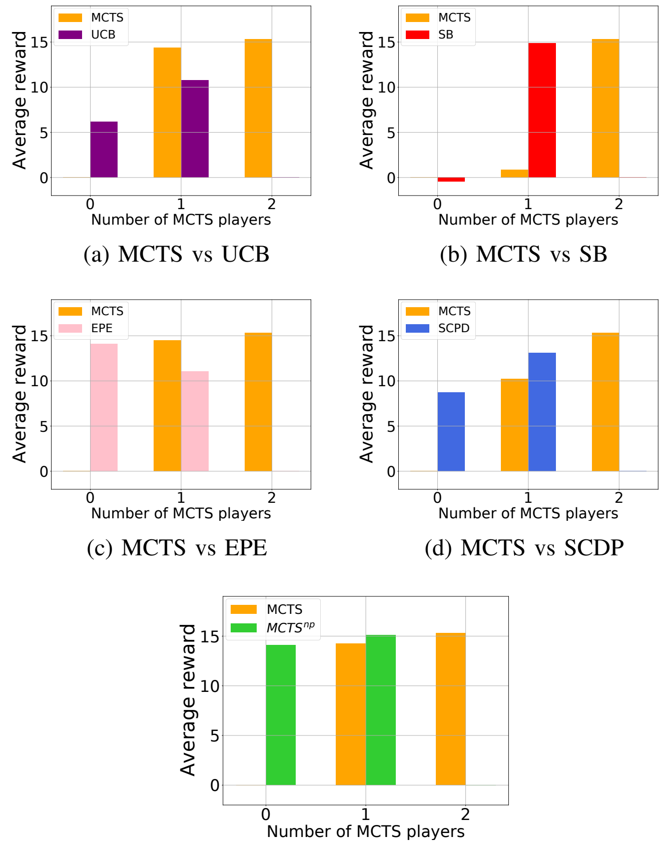
We study the performance of each algorithm mainly through our performance indicators. Each experimental result has been run on 1000 different d-SAA instances. We will focus on d-SAA with $n = 2$, $m = 7$ and $\varepsilon = 1$. Value functions are chosen according to the following setting with $V = 5$.

Setting. *Let Γ be an instance of d-SAA with $n \geq 2$ players, $m \geq 1$ goods, bid increment ε and maximum stand-alone value $V > 0$. Each player i has a general value function v_i such that $v_i(\emptyset) = 0$ and, for any set of goods X, we have*

$$v_i(X) \sim U([\max_{j \in X} v_i(X \backslash \{j\}), V + \max_{j \in X} v_i(X \backslash \{j\}) + v_i(\{j\})]) \tag{9}$$

where U is the uniform distribution. The lower-bound ensures that the free disposal [15] condition is respected while the upper-bound caps the maximum surplus of complementarity possibly gained by adding an item j to the set of goods $X \backslash \{j\}$ by V.

As value functions are generated randomly and are associated to a bidder's position, it might be advantageous to play first on average for the 1000 different d-SAA instances. Indeed, as players do not have the opportunity to play at the same bid price, the order in which players submit their bids may have an impact on the auction's outcome. To eliminate such variance and guarantee a fair comparison between two strategies A and B, for each d-SAA instance, a game will be run with the first bidder playing A and the second playing B and another with the first bidder playing B and the second playing A.

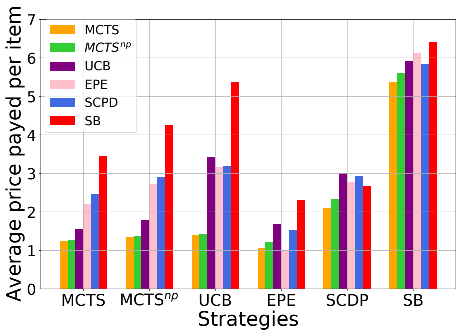


(a) MCTS vs UCB
(b) MCTS vs SB
(c) MCTS vs EPE
(d) MCTS vs SCDP
(e) MCTS vs MCTS^{np}

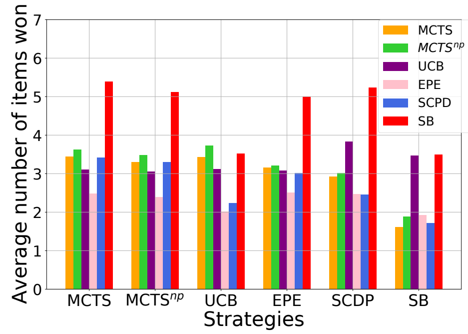
Fig. 2: Normal-form payoffs for a d-SAA game with six strategies

1) Expected utility: We first analyse our MCTS performance through the same empirical game analysis approach as Wellman et al. [24] which maps strategy profiles to the average payoff obtained in the 1000 different d-SAA instances for each player. More precisely, we study the symmetric normal form game in expected payoff where each player has the choice between playing our MCTS bidding strategy or another specified strategy A. The resulting empirical games for each possible strategy A are given in Figure 2.

It is clear that, in each empirical game, the deviation from UCB, EPE, SCPD, SB or MCTS^{np} to our MCTS bidding strategy is always profitable. Thus, the strategy profile (MCTS, MCTS) is the only Nash equilibrium of the normal-form d-SAA game in expected payoffs with strategy set {MCTS,



(a) Average price payed per item



(b) Average number of items won per auction

Fig. 3: Own price effect analysis of d-SAA with six strategies

UCB, EPE, SCPD, SB, MCTS^{np}}. It is also important to note the significant increase in average payoffs between the strategy profile (MCTS,MCTS) and the other strategy profiles where all bidders play the same strategy. For instance, in Figure 2, the profile (MCTS,MCTS) has an increase of 108%, 247%, 108% and 175% compared respectively to (MCTS^{np},MCTS^{np}), (UCB,UCB), (EPE,EPE) and (SCPD,SCPD) in average payoffs. Complete information of d-SAA enables EPE bidders to share the same expected competitive equilibrium if their tâtonnement process uses the same initial price vector and adjustment parameter. This explains that (EPE,EPE) obtains an expected utility nearly as high as (MCTS^{np},MCTS^{np}). Due to exposure, the profile (SB,SB) obtains a negative expected utility. Hence, both players would have preferred not to participate in the d-SAA initially.

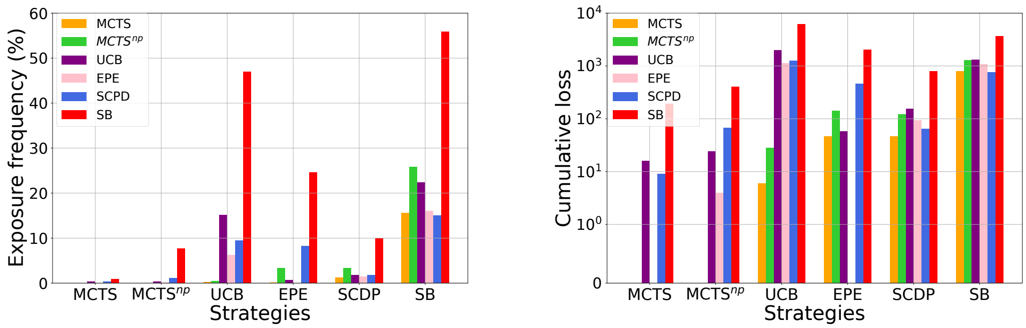
The relative high performance of our MCTS bidding strategy can be explained by three factors: (i) its ability to judge in which situations it is more beneficial to perform demand reduction or to bid competitively as seen in Section V-A; (ii) its ability to perform demand reduction to keep prices low and (iii) its ability to stop bidding on specific sets of goods early on to avoid exposure.

2) Demand reduction: Our algorithm has the capacity of conceding items to its opponents in order to keep prices low. We highlight this feature in Figure 3(a) where we plot the average price payed per item won by each strategy against every strategy displayed on the x-axis. Our MCTS bidding strategy obtains the lowest average price payed per item won against every strategy except against EPE. Indeed, each item won is bought at ε when both bidders play EPE as they share the same expected competitive equilibrium. This explains the slight underperformance of our algorithm. Nevertheless, our MCTS bidding strategy spends 43.2%, 50%, 55.9%, 24.5% and 12.1% less per item won than EPE against MCTS, MCTS^{np}, UCB, SCPD and SB respectively. Moreover, our MCTS strategy is competitive and does not just purchase undesired items at relatively low prices. This is highlighted in Figure 3(b) where we plot the average number of items won by each strategy against every strategy displayed on the x-axis. Our MCTS strategy wins at least 3 items out of 7 against every strategy except SB. In comparison, EPE never wins more than 2.5 items on average against any strategy. Regarding strategy profiles where all bidders play the same strategy, 98.3% of all items are allocated for MCTS whereas only 71.4% and 70.3% respectively for EPE and SCPD which partially explains their underperformance. The fact that nearly all goods are allocated and are acquired at a small price explains the high performance of the strategy profile (MCTS,MCTS) compared to the other strategy profiles. Through smart usage of demand reduction, our MCTS bidding strategy tackles the own price effect and still remains fairly competitive.

3) Reduction of exposure: As already explained in Section II-D, minimising exposure is extremely important in d-SAA. In Figure 4(a), we have plotted the exposure frequency of each strategy against every strategy displayed on the x-axis. Our MCTS bidding strategy has at most 1.2% of chance of getting exposed against every strategy except SB against which it obtains a similar exposure frequency to SCPD. Moreover, the strategy profile (MCTS,MCTS) has the remarkable property of never suffering from exposure. It is important to highlight the significant enhancement due to our selection penalties as it is exposed 95.6%, 62.7% and 39.8% less than MCTS^{np} against respectively EPE, SCPD and SB. In Figure 4(b), we have plotted the cumulative loss of each strategy against every strategy displayed on the x-axis. MCTS cumulative loss is at least one order of magnitude below the corresponding SCPD one against MCTS, MCTS^{np}, UCB and EPE. They are about the same order against SCPD and SB. Moreover, our MCTS bidding strategy generates 78.6%, 66.7%, 61.2% and 37.3% less losses than MCTS^{np} against respectively UCB, EPE, SCPD and SB. Thus, in addition to being profitable against all strategies, our MCTS bidding strategy considerably minimises exposure notably through its selection penalties.

VI. Discussions

This paper introduces the first algorithm that efficiently tackles the *exposure problem* and *own price effect* in a simplified version of SAA (d-SAA). Our MCTS bidding strategy relies on a new method for the prediction of closing prices. It is based on a specific sequence that converges in practice



(a) Exposure frequency   (b) Cumulative loss due to exposure

Fig. 4: Exposure analysis of d-SAA with six strategies

in all d-SAA instances. This method has the advantage of taking into account the auctions' particularities and computing a prediction of final prices which is independent of a single specific strategy profile. Experimental results support the fact that our MCTS bidding strategy largely outperforms state-of-the-art algorithms in d-SAA by obtaining greater expected utility and taking less risks against all other strategies.

We believe that the biggest field of improvement for our MCTS is related to its selection penalties. Indeed, although they have greatly contributed to our reduction of exposure, the introduction of more formal statistics may be beneficial. However, using quantities such as mean-variance [13] did not enhance our algorithm's performance.

Our algorithm is easily extended to environments with budget constraints and complementary experiments (not included in this article due to a lack of space) show that our MCTS bidding strategy remains efficient and robust to significant errors in the valuation estimates. Hence, a possibility for relaxing the complete information hypothesis and dealing with valuation distributions is to take their expectation and then apply our MCTS. Future work should be guided towards an increase in the number of players as well as introducing simultaneity and incomplete information in our SAA model.

References

[1] Lawrence M Ausubel, Peter Cramton, Marek Pycia, Marzena Rostek, and Marek Weretka. Demand reduction and inefficiency in multi-unit auctions. *The Review of Economic Studies*, 81(4):1366–1400, 2014.

[2] Cameron B Browne, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter I Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in games*, 4(1):1–43, 2012.

[3] Sandro Brusco and Giuseppe Lopomo. Collusion via signalling in simultaneous ascending bid auctions with heterogeneous objects, with and without complementarities. *The Review of Economic Studies*, 69(2):407–436, 2002.

[4] Mark M Bykowsky, Robert J Cull, and John O Ledyard. Mutually destructive bidding: The fcc auction design problem. *Journal of Regulatory Economics*, 17(3):205–228, 2000.

[5] Moinul Morshed Porag Chowdhury, Christopher Kiekintveld, Son Tran, and William Yeoh. Bidding in periodic double auctions using heuristics and dynamic monte carlo tree search. In *IJCAI*, pages 166–172, 2018.

[6] Peter Cramton et al. Simultaneous ascending auctions. *Combinatorial auctions*, pages 99–114, 2006.

[7] European 5G Observatory. Italian 5g spectrum auction. https://5gobservatory.eu/italian-5g-spectrum-auction-2/, 2018.

[8] Jacob K Goeree and Yuanchuan Lien. An equilibrium analysis of the simultaneous ascending auction. *Journal of Economic Theory*, 153:506–533, 2014.

[9] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. In *The collected works of Wassily Hoeffding*, pages 409–426. Springer, 1994.

[10] Levente Kocsis and Csaba Szepesvári. Bandit based monte-carlo planning. In *European conference on machine learning*, pages 282–293. Springer, 2006.

[11] Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

[12] Benny Lehmann, Daniel Lehmann, and Noam Nisan. Combinatorial auctions with decreasing marginal utilities. *Games and Economic Behavior*, 55(2):270–296, 2006.

[13] Xingchi Liu, Mahsa Derakhshani, Sangarapillai Lambotharan, and Mihaela Van der Schaar. Risk-aware multi-armed bandits with refined upper confidence bounds. *IEEE Signal Processing Letters*, 28:269–273, 2020.

[14] R Preston McAfee and John McMillan. Auctions and bidding. *Journal of economic literature*, 25(2):699–738, 1987.

[15] Paul Milgrom. Putting auction theory to work: The simultaneous ascending auction. *Journal of political economy*, 108(2):245–272, 2000.

[16] Joseph Antonius Maria Nijssen. *Monte-Carlo tree search for multi-player games*. PhD thesis, Maastricht University, 2013.

[17] Diego Perez Liebana, Jens Dieskau, Martin Hunermund, Sanaz Mostaghim, and Simon Lucas. Open loop search for general video game playing. In *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation*, pages 337–344, 2015.

[18] Stuart Russell and Peter Norvig. *Artificial intelligence: a modern approach*. Pearson Education, 2021.

[19] Balazs Szentes and Robert W Rosenthal. Beyond chopsticks: Symmetric equilibria in majority auction games. *Games and Economic Behavior*, 45(2):278–295, 2003.

[20] Balázs Szentes and Robert W Rosenthal. Three-object two-bidder simultaneous auctions: chopsticks and tetrahedra. *Games and Economic Behavior*, 44(1):114–133, 2003.

[21] H Jaap Van Den Herik, Jos WHM Uiterwijk, and Jack Van Rijswijck. Games solved: Now and in the future. *Artificial Intelligence*, 134(1-2):277–311, 2002.

[22] Joel Veness, Kee Siong Ng, Marcus Hutter, William Uther, and David Silver. A monte-carlo aixi approximation. *Journal of Artificial Intelligence Research*, 40:95–142, 2011.

[23] Robert J Weber. Making more from less: Strategic demand reduction in the fcc spectrum auctions. *Journal of Economics & Management Strategy*, 6(3):529–548, 1997.

[24] Michael P Wellman, Anna Osepayshvilli, Jeffrey K MacKie-Mason, and Daniel Reeves. *Bidding strategies for simultaneous ascending auctions*. B.E. J. Theoret. Econom, 2008.

[25] Charles Z Zheng. Jump bidding and overconcentration in decentralized simultaneous ascending auctions. *Games and Economic Behavior*, 76(2):648–664, 2012.