ECAI 2016
G.A. Kaminka et al. (Eds.)

doi:10.3233/978-1-61499-672-9-1035

Online Auctions for Dynamic Assignment: Theory and Empirical Evaluation

Sujit Gujar[1] and **Boi Faltings** [2]

Abstract. Dynamic resource assignment is a common problem in multi-agent systems. We consider scenarios in which dynamic agents have preferences about assignments and the resources that can be assigned using online auctions. We study the trade-off between the following online auction properties: (i) truthfulness, (ii) expressiveness, (iii) efficiency, and (iv) average case performance. We theoretically and empirically compare four different online auctions: (i) Arrival Priority Serial Dictatorship, (ii) Split Dynamic VCG, (iii) e-Action, and (iv) Online Ranked Competition Auction. The latter is a novel design based on the competitive secretary problem. We show that, in addition to truthfulness and algorithmic efficiency, the degree of competition also plays an important role in selecting the best algorithm for a given context.

1 Introduction

Consider a scenario in which an Uber driver prefers customers who want to travel in a particular direction, e.g., the driver carries customers in a shared ride and hence prefers new passengers who have destinations close to those of the passengers already in the vehicle. In such situations, the driver might be willing to pay Uber a small amount (over the standard amount that Uber charges drivers for a fare) to carry a preferable customer. In expert crowdsourcing task assignment, expert agents have preferences about which tasks they would like to work on, and they may be willing to pay the platform a premium for obtaining preferable tasks [10].

As yet another example, consider a hotel booking platform. A hotel ranked lower down on the platform might be interested in being listed higher for a certain class of travellers with whom the hotel believes it has a higher chance of obtaining a booking. The hotel may be willing to pay a small fee to the platform to achieve this.

Motivated by such real-world examples, we consider dynamic assignment for crowds where the dynamic *agents* (Uber's drivers, the experts in the expert crowdsourcing example or the hotel owners on the hotel booking website) have preferences for different available *resources* (new Uber passengers, the tasks in the expert crowdsourcing example or the travellers on the hotel booking website) and assign certain valuations to matches (which can be zero, if an agent has no preference). Additionally, each agent has a deadline after which he no longer has use for the resource, known as his departure time. A *platform*'s goal (whether Uber, an expert crowdsourcing platform or a hotel booking website) is to improve the *quality of resource assignment*, which is also termed as *social welfare or efficiency* - the sum of agents' valuations for their assignments. To achieve this, the platform needs agents to report their valuations truthfully. This property is known as *incentive compatibility* [18]. Additional challenges are that agents are dynamic and that assignments must happen *online*, i.e., they must happen before the agents leave the system. Strategic agents may attempt to manipulate assignment mechanisms if it is beneficial to them. Moreover, strategic agents may manipulate their arrival-departure if it is part of their private information. There is thus a need to design appropriate game theoretic mechanisms to induce truthful reporting of private information.

Dynamic Mechanism Design Mechanism design theory [18] is useful for designing procedures to elicit *agent's* valuations of the resources they are matched with. Gujar and Faltings [10] proposed that agents should pay the *platform* a premium to obtain their preferred matches. When such monetary transfers are feasible, one can design dynamic mechanisms for assignments in such modern marketplaces.

In the literature, there are two well-studied approaches for designing dynamic mechanisms. The first approach, based on stochastic models of agents' private information, uses dynamic programming, Markov Decision Processes, etc.; e.g., the mechanisms proposed in [1, 4]. Adapting these dynamic mechanisms for crowds is challenging as it needs precise information about the probability distributions of agents' valuations of their assignments and this may not be available in a new marketplace. Furthermore, agents may have little knowledge of mechanism design theory, and understanding such complex, dynamic mechanisms may be demanding.

The second approach does not assume any probability distribution for agents' valuations. These model-free dynamic mechanisms are called *online auctions*. The theory developed to solve the classic *secretary problem* is useful for designing online auctions (Babaioff *et. al.*[2]). Mechanisms based on the secretary problem are often easy to understand and implement.[3] The present paper focuses on this second approach to online auctions for resource assignment.

A hypothetical optimal algorithm that has knowledge about all the agents' valuations in the beginning, is called *offline-optimal*. Online auction performance is evaluated using the *competitive ratio* (CR) metric, which indicates how far a given algorithm's solution is from the offline-optimal solution in the worst case. In real life, a worst case may not occur frequently. For repeated usage of an auction, the platform may prefer an online auction that performs well on average, rather than one that only performs well in a worst case.

The Problem This paper's goal is to determine which online auction mechanisms have the following desirable characteristics: (i) truthfulness, (ii) preference expressiveness (richness of preference elicitation), (iii) efficiency (social welfare) and, most importantly,

[1] IIIT Hyderabad, sujit.gujar@iiit.ac.in Most of the work in this paper was carried out when the first author was a post-doctoral researcher at the EPFL.
[2] EPFL, boi.faltings@epfl.ch

[3] Online auctions may be designed using completely different approaches from the secretary problem.

(iv) good performance on average.

Our Approach Typically, resources compete for the best possible assignment (agents). This is analogous to the competitive secretary problem [13, 15]. However, these two papers did not address agents' strategic behaviours. We adapt the techniques developed by Karlin *et. al.* [15] to design a new, truthful, online auction called, an *online ranked competition auction* (ORCA).

We hypothesize that online auctions, optimized for worst case guarantees, only perform well on average if there is a high level of competition, i.e., the degree of competition between agents for the resources affects the auction's average performance. We analyse auctions empirically by generating a large number of instances of the resource assignment problem for different stochastic models. To evaluate the performance of a given online auction, we introduce three metrics: (i) the *Empirical Competitive Ratio* (ECR), (ii) the *Sample Average Competitive Ratio* (SACR), and (iii) the *Empirical Expected Efficiency* (EEE). A given online auction's ECR is the worst performance observed among the instances generated. SACR measures how far, on average, a given online auction's solution is from an offline-optimal solution. EEE measures the average fraction of the expected valuations of all the agents in an offline-optimal solution, that can be achieved by a given online auction.

We study the resource assignment problem using the following online auctions: Arrival Priority Serial Dictatorship (APSD), proposed by Zou *et. al.*[22], the Split Dynamic VCG (SDV) [10], eAuction [10] and ORCA, proposed by the present paper.

Contributions We explore the application of truthful online auctions for resource assignment. We also propose a new, truthful, online, ranked competition auction (ORCA). We look for theoretical guarantees and average case performance. This paper's main contribution is its evaluation of online auctions for trade-offs between: (i) truthfulness, (ii) expressiveness, (iii) efficiency, and (iv) average case performance. We demonstrate empirically that the auctions designed for better worst-case guarantees often only perform well if there is a high degree of competition between agents. In less competitive settings, simpler auctions perform better.

Our empirical study considers APSD, SDV, eAuction and ORCA. Analysis validates our hypothesis, i.e., when compared with APSD and SDV, eAuction and ORCA only performed well in highly competitive settings (Figure 2 and Figure 3). ORCA also performs better when the agent arrival rate is lower, with moderate competition between agents (Figure 7). For all four online auctions, the empirical worst cases are not as bad as indicated by corresponding theoretical bounds (Figure 1, Figure 5) and worst cases are infrequent. We provide guidelines on how to select the most appropriate online auction mechanism for a platform's conditions.

Organisation In the next subsection we describe related work to ours. Section 2 explains the notation used in this paper and the secretary problem. Section 3 describes the online auctions studied. Section 4 formally defines the ECR, SACR and EEE, describes the experiments and analyses the empirical evaluation. Section 5 concludes the paper.

1.1 Related Work

Mechanism design theory is a rich field. Nisan *et. al.*[18] and the references cited therein provide pointers to it. Dynamic mechanism design has been addressed with regard to auction design when prior distribution of agents' arrival-departure and valuations are known [1, 4, 20]. However, we focus on a model-free design for online auctions.

Although the literature on online algorithms does not address agent's strategic behaviour, the techniques it has developed are very useful for designing online auctions. The notion of an online algorithm was popularised in a seminal paper by Karp *et. al.* [16]. For more details on online algorithms, readers are referred to [5, 8]. The classic secretary problem has been well studied in the literature [7, 13, 15, 17]. Solutions to it have also been used in the design of online auctions, e.g., [2, 3].

There is abundant literature on the task assignment problem in crowdsourcing [6, 11, 12, 14, 21]. However, there has not been much research on the use of online algorithms/auctions for task (resource) assignment, with exceptions being [9, 10]. The present paper addresses the resource assignment problem for new marketplaces using an online auction approach.

2 Preliminaries

Let $R = \{r_1, r_2, \ldots, r_k\}$ be the set of k available resources on a given platform. Let $N = \{1, 2, \ldots, n\}$ be the set of n agents interested in those resources. Each resource must be assigned to one agent only, and each agent is only interested in one resource.[4] Let $X_i \in R \cup \{\perp\}$ denote the resource assigned to agent i, where $\perp$ indicates no assigned resource and $\mathbb{1}_{X_i = r_j}$ denotes an indicator variable which is 1 if agent i obtains r_j and is 0 otherwise. Agent i gives a valuation v_{ij} to obtain resource r_j ($\forall i, v_{\mathrm{i}\perp} = 0$). Agent i arrives in the system at time period a_i and is available until time period d_i. The platform's goal is to maximise the sum of the agents' valuations of the resources assigned to them, as described in Problem (1):

$$\begin{aligned} \max \sum v_{\mathrm{iX_i}} \qquad \text{s.t.} \\ X_i \in R \cup \{\perp\} \; \forall i \in N \\ \sum_i \mathbb{1}_{X_i = r_j} \leq 1 \; \forall r_j \in R \end{aligned} \tag{1}$$

First, we assume that the agents are honest in reporting their valuations for the resources. If all the agents' valuations are known in advance, the platform can solve this optimization problem and efficiently assign resources. The hypothetical algorithm that solves Problem (1) in the presence of dynamic agents is called the *offline-optimal*. However, in dynamic environments, valuations only become known when agents arrive in the system and they are not all available simultaneously. Therefore, the platform cannot solve the above optimisation problem. Hence, the platform must look for mechanisms that are as close to the offline-optimal as possible. The secretary problem, and its analysis, is very useful for designing online auctions.

2.1 The Secretary Problem

In the secretary problem, a recruiter wishes to hire a secretary from among n candidates. The recruiter can only evaluate a candidate after interviewing him. However, the recruiter must either offer the job or reject the candidate before moving on to a new one. The decision is irrevocable. This problem was analysed by [7, 17]. An optimal strategy is for the recruiter to interview the first $\frac{n}{e}$ candidates and offer the job to the next candidate who is better than these first $\frac{n}{e}$ candidates [7, 17]. Here, e is the base of the natural logarithm.

[4] We focus on a time window in which each agent is typically only interested in one assignment.

In the resource assignment problem, with $k = 1$, the platform's goal is to assign the resource to the agent giving it the highest valuation which is known only after he arrives in the system. This resource assignment problem is exactly the same as the secretary problem. Thus, the platform should wait until the first $\frac{n}{e}$ agents have indicated their valuations and offer the resource to the next agent who provides a higher valuation than those first $\frac{n}{e}$ agents.

2.2 Competitive Secretary Problem

Consider a case where there are $k > 1$ resources and n agents competing for them. The platform prefers to assign each resource to the agent providing the highest valuation for that resource. The agents appear on the platform sequentially. Each agent can be offered zero, one or more resources while he is present in the system, but he can select only one of them. After his departure, the next agent arrives in the system. If an agent is offered multiple resources, he chooses to accept a single resource and rejects the remainder. The literature addresses two separate cases, depending on how the agent selects a resource from among multiple offers.
(i) Resources having equal rank: an agent with multiple offers has an equal probability of choosing any one of those resources.
(ii) Resources having ranked order: resources are ranked, and an agent with multiple offers accepts the highest ranked resource.

Let us assume that the platform waits until $\frac{n}{th_j}$ agents have arrived in the system and reported their valuation of r_j. The platform offers r_j to the first agent to give a valuation higher than that of the first $\frac{n}{th_j}$ agents. th_j is called the *stopping threshold* for r_j. Note that for the secretary problem ($k = 1$), $th_1 = e$ is the optimal stopping threshold. For the above two settings, the optimal stopping threshold for each resource should be different.

Resources Having Equal Rank Immorlica *et. al.* [13] addressed this case. However, a closed-form solution to optimal stopping thresholds for $k > 2$ is unknown. We leave the design of online auctions for such cases for future research.

Resources Having Ranked Order An agent can assign different valuations for different resources. If he receives multiple offers, he chooses the resource to which he assigned the highest valuation. Hence, the probability of accepting a particular offer is higher for a resource that is preferred by all the agents. This induces a natural ordering of resources and the higher-ranked resource will always be chosen by an agent with multiple offers. To take advantage of the possibility that the best possible agent arriving before the stopping threshold of a higher-ranked resource, the platform could use lower stopping thresholds for lower-ranked resources [15].

Note that the above approaches only work if agents report the platform valuations and availability truthfully. However, in real life, since agents are strategic, auction theory can be used instead, as explained in the next subsection.

2.3 Online Auctions

Strategic agents can boost their valuations to ensure they receive a resource. However, if they are made to pay an appropriate amount to the platform, truthful behaviour can be induced. Let p_i denote the payment that agent i makes to the platform for resource assignment, i.e., his utility for that resource assignment is $v_{iX_i} - p_i$. Note that the focus of the present paper is on the quality of assignment, hence we refer to agents' utility for assignments and *not external utility*, that an Uber driver may derive by serving a passenger, for example. Another possibility for manipulation is on arrival and departure. Thus for the agent i, the private information is: $\theta_i = (\mathbf{v}_i, a_i, d_i)$ where, $\mathbf{v}_i = (v_{i1}, v_{i2}, \dots, v_{ik}) \in \mathbb{R}_+^k$ [5]. This private information θ_i is called the *type* of agent i. Let Θ_i denote the space of possible types of agent i. Let $\theta = (\theta_1, \dots, \theta_n)$ denote the type profile of all the agents. θ is also indicated by (θ_i, θ_{-i}) where θ_{-i} is the type profile for all the agents except i.

For a given type profile, an online auction $\mathcal{A} = (X, p)$ selects a feasible resource assignment $X(\theta) = (X_1(\theta), X_2(\theta), \dots, X_n(\theta))$ and determines the payments $p(\theta) = (p_1(\theta), p_2(\theta) \dots, p_n(\theta))$. A feasible resource assignment is one in which each agent receives a resource, if any, before his departure time and this is independent of the types of the agents who are yet to arrive.

Let $C_i(\theta_i)$ denote the space of possible misreports available to agent i when his true type is θ_i. That is, he may report his type to be $\hat{\theta}_i \in C_i(\theta_i)$ if it is beneficial to him. Generally, agents cannot appear in the system before their true arrival time and cannot be present after their true departure time, though they can pretend to appear late or leave early or misreport their valuations. We call this domain of misreports *no-early-arrival-no-late-departure*. We restrict the domain of misreports to no-early-arrival-no-late-departure. That is, $C_{i(ex)}(\theta_i) = \{\hat{\theta}_i = (\hat{\mathbf{v}}_i, \hat{a_i}, \hat{d_i})\}$ with $\hat{\mathbf{v}}_i \in \mathbb{R}_+^k, a_i \leq \hat{a_i} \leq \hat{d_i} \leq d_i$. These settings are also called *exogenous* arrival-departure. In some cases, it may be possible to assume that agents cannot manipulate their arrival-departure times or that these are not part of their private information. We capture this setting as $C_{i(en)}(\theta_i) = \{\hat{\theta}_i = (\hat{\mathbf{v}}_i, a_i, d_i)\}$. This is also called *endogenous* arrival-departure.

Definition 1 (Truthfulness) *An online auction, $\mathcal{A}$, is dominant strategy-incentive compatible (or truthful) for domain of misreports $C_i(\theta_i)$s if for every agent i and for every $\theta_i \in \Theta_i$,*

$$v_{iX_i(\theta)} - p_i(\theta) \geq v_{iX_i(\theta_i', \theta_{-i})} - p_i(\theta_i', \theta_{-i}) \qquad (2)$$
$$\forall \theta_i' \in C_i(\theta_i), \forall \theta_{-i} \in \Theta_{-i}$$

The next subsection presents a necessary condition for a truthful online auction.

2.4 Necessary Condition for a Truthful Online Auction

In online auctions, the dynamics of the arrival-departure of different agents offers strategic agents more flexibility for manipulation. Hence, online auctions must be designed carefully.

Definition 2 (Arrival-Departure Priority) *Online auctions for resource assignment problems are said to have an arrival-departure priority if agent i's utility at a type θ_i having the same valuation as θ_i', but with either earlier arrival or later departure than θ_i', does not decrease. That is, $\forall \theta_i, \theta_i' \in \Theta_i$ such that $\mathbf{v}_i = \mathbf{v}_i'$, $a_i \leq a_i'$ and $d_i \geq d_i'$,*

$$v_{iX_i(\theta_i, \theta_{-i})} - p_i(\theta_i, \theta_{-i}) \geq v_{iX_i(\theta_i', \theta_{-i})} - p_i(\theta_i', \theta_{-i})$$

Lemma 1 *In exogenous settings, a truthful online auction must have an arrival-departure priority.*

[5] Recall, a_i, d_i are his arrival and departure times.

Proof: Suppose a truthful online auction $\mathcal{A}$ does not have an arrival-departure priority, i.e., for an agent, say i, there exist two types of $\theta_i : \theta_i'$ such that $\mathbf{v_i} = \mathbf{v_i}'$, and $a_i \leq a_i'$, and $d_i \geq d_i'$ and $u_i(\theta_i', \theta_{-i}) > u_i(\theta_i, \theta_{-i})$. If agent i has a true type θ_i and other agents have type θ_{-i}, agent i benefits from arriving late at a_i' and from reporting his type to be θ_i', contradicting the truthfulness of $\mathcal{A}$. □

The above lemma implies that, to design a truthful online auction for no-early-arrival-no-late-departure domains, one has to ensure that the auction satisfies the arrival-departure priority.

We now define the competitive ratio (CR) metric. Let $V^{\mathcal{A}}(\theta)$ denote the total valuation by all the agents for the resources in an online auction $\mathcal{A}$, and let $V^*(\theta)$ be the total valuation by all the agents in the offline-optimal solution. A CR of $\mathcal{A}$ is defined as:

Definition 3 (CR) *An online auction $\mathcal{A}$ is said to be α-competitive if*

$$min_{\{\theta:V^*(\theta)\neq 0\}}\mathbb{E}\frac{V^{\mathcal{A}}(\theta)}{V^*(\theta)} \geq \frac{1}{\alpha}$$

where expectation is taken with respect to random orderings of the agents.

A CR is a fair measure with which to evaluate different online auctions as online auctions are independent of stochastic models. A low CR is desirable because even in the worst case, the given online auction is close to the offline-optimal. In general, CRs are quite high as online auctions can perform poorly in worst cases.

Having provided background information on online auctions, the next section describes the auctions studied.

3 Online Auctions for Resource Assignment

We consider the following online auctions: (i) APSD, (ii) SDV, (iii) eAuction and (iv) ORCA. eAuction and ORCA are based on the secretary problem.

3.1 Arrival Priority Serial Dictatorship (APSD)

Zou *et. al.* [22] proposed arrival priority serial dictatorship (APSD) for assignment problems. In APSD, upon arrival, each agent selects the resource for which he has the highest valuation from the pool of available resources, but does not pay the platform. The authors proved that APSD is the only truthful mechanism for no-early-arrival-no-late-departure domains if monetary transfers are not allowed.

Note that as payments are absent in APSD, it is not an auction as we imagine a real-world auction. However, it is a very simple, yet truthful mechanism without asking the agents to report anything.

3.2 Split Dynamic VCG (SDV)

Gujar and Faltings [10] proposed a Vickrey-Clarke-Groves (VCG)-based mechanism for resource assignment in crowdsourcing. VCG mechanism for static settings (i.e., $\forall i,\ a_i = d_i = 1$) is as follows. It finds an assignment that maximizes the sum of the agents' valuations for the resources and the payments are based on the externalities they impose on the system [18]. In [10], the authors considered the partition of the agents such that the agents in each part of the partition are available simultaneously. The platform assigns the remaining resources to the agents by solving VCG mechanism for each part of the partition. SDV mechanism does not satisfy arrival-departure priority and hence, it is only truthful for endogenous settings.

3.3 eAuction

In [19], Parkes proposes an online auction for a single item using a solution to the secretary problem. Gujar and Faltings [10] adapted this for a k resources setting. The platform waits until $\frac{n}{e}$ agents have arrived and if the agent providing the highest valuation for any resource is available, he gets the resource by paying the second-highest reported valuation. Otherwise, for each resource, the highest valuation received in the first phase is set as a reserve price and whichever agent provides a higher valuation than that reserve price obtains the resource by paying the reserve price. If an agent is eligible for more than one resource (i.e., having provided valuations higher than the reserve prices), the platform assigns the agent the resource with the highest utility to him. This is referred to as an eAuction, and eAuctions are truthful for no-early-arrival-no-late-departure domains.

In the competitive secretary problem - when there are multiple resources - optimal stopping thresholds for different resources are different [13, 15]. The next subsection proposes a threshold-based online auction framework which enables the use of different thresholds for different resources using the online ranked competition auction.

3.4 Online Ranked Competition Auction (ORCA)

Here we propose a generic *threshold-based online auction* framework for k resources.

Definition 4 (Threshold-Based Online Auction) *Let $th_1, \ldots, th_k$ be the stopping thresholds for $r_1, \ldots r_k$, respectively. Let h_j and sh_j be the highest bid and second-highest bid for r_j from the first $\frac{n}{th_j}$ bids. Agent i has a priority higher than agent j if $a_i < a_j$ (ties are resolved randomly).*

At each time slot, an agent i locks in r_j if: (i) resource r_j is unassigned, (ii) it has not been locked in by another agent with a higher priority, and (iii) $d_i \geq a_{\frac{n}{th_j}}$. If an agent with a lower priority has already locked it in, the lower priority agent loses that lock in on r_j. At d_i, i is assigned the resource giving him the highest utility from among the resources he has locked in. If agent i receives resource r_j, then he pays the platform sh_j if $a_i \leq a_{\frac{n}{th_j}}$, otherwise he pays the platform h_j. All the other resources locked in for i are released at d_i.

Proposition 1 *A threshold-based online auction satisfies arrival-departure priority.*

Proof: Consider an agent i with two types, θ_i and θ_i', such that $v_i = v_i', a_i \leq a_i'$ and $d_i \geq d_i'$. Let us fix other agents' types as θ_{-i}. When agent i has type θ_i, he can lock in all the resources that he can lock in with type θ_i', but additionally, he may be able to lock in more resources in θ_i either because he now has a higher priority or because some resources were released after d_i' to which he may now get access. Agent i is offered the resource yielding him the highest utility and hence, $u_i(\theta_i, \theta_{-i}) \geq u_i(\theta_i', \theta_{-i})$. As this is true for all agents, the proposition follows. □

Theorem 1 *A threshold-based online auction is truthful for a no-early-arrival-no-late-departure misreports domain.*

Proof: The agent's payment is independent of his bid for a resource, hence no agent has an incentive to lie about his valuation for that resource. However, in dynamic settings, an agent may try to manipulate the online auction in order to get the resource with the highest utility to him. From Proposition 1, a threshold-based online auction satisfies the arrival-departure priority. With this property and the fact that the threshold-based online auction offers the agent the resource that has the highest utility to him throughout his availability, no agent has any incentive to misreport his type. □

As explained earlier, there are two approaches to determining stopping thresholds for each resource in the classic competitive secretary problem. We focus on the case where resources are ordered.

Let r_j be the j^{th} ranked resource. Let $Pr_j(l)$ be the probability that r_j cannot be matched with the best possible agent when l is used as the stopping threshold. Then Karlin *et. al.* [15] showed:

Theorem 1 *[15] Optimal stopping threshold th_j for j^{th} ranked resource (r_j) is given by*

$$th_j = \min\{l : Pr_j(l) \geq 1 - \frac{l}{n}\} - 1$$

Online Ranked Competition Auction (ORCA) Karlin *et. al.* provided a dynamic programme with which to compute the above thresholds. We use the solution to this programme and plug these thresholds into a threshold-based online auction referred to as an *Online Ranked Competition Auction* (ORCA).

3.5 Comparing APSD, SDV, eAuction and ORCA

The complexity of the implementing mechanism increases as we move from APSD to ORCA. However, each system is designed to achieve better CR and to provide more information to the auction. Table 1 summarizes[6] the theoretical properties of the online auctions discussed.

	APSD	SDV	eAuction	ORCA
Preference elicitation	No	Only v_i's	v_i's and a_i's	v_i's, a_i's and d_i's
Truthfulness	Exogenous	Endogenous	Exogenous	Exogenous
CR	n	n	e^2	$< e^2$

Table 1. Comparison of the theoretical properties of APSD, SDV, eAuction and ORCA

This section concludes by illustrating how all the mechanisms work using the following example.

$i =$	1	2	3	4	5	6	7	8	9	10
a_i	1	2	3	4	4	5	5	6	6	6
d_i	2	2	3	6	7	6	5	8	9	7
v_{i1}	6	8	7	15	12	16	4	17	2	5
v_{i2}	5	3	4	1	4	2	3	2	1	4

Table 2. Example: $n = 10, k = 2$

Example: Consider a market with $k = 2$ resources and $n = 10$ competing agents. Resource 1 is preferable to resource 2, i.e., all agents are more likely to value resource 1 more than resource 2. Each agent's arrival time, departure time and valuations are given in Table 2. The mechanisms described above yield the following outcomes:

- OFFLINE-OPTIMAL: Agent 8 gets resource 1 and agent 1 gets resource 2, with optimal social welfare $= 22$.
- APSD: Agent 1 gets resource 1 and agent 2 gets resource 2, with social welfare $= 9$.
- SDV: SDV executes VCG at $t = 2$. Agent 2 gets resource 1 and agent 1 gets resource 2, and their payments are 1 and 0, respectively, while social welfare $= 13$.
- e-Auction: e-Auction waits for $\lfloor \frac{10}{e} \rfloor = 3$ agents to submit their bids. Thus, e-Auction sets reserve prices for resource 1 at 8 and for resource 2 at 5. Agent 4 gets resource 1, however no agent gets resource 2, leading to social welfare $= 15$.
- ORCA: ORCA waits for $\lfloor \frac{10}{e} \rfloor = 3$ agents to submit their bids to for resource 1 and for 2 bids to arrive for resource 2. Thus, agent 4 obtains resource 1 by paying 8 and agent 1 obtains resource 2 by paying 3. Thus ORCA achieves social welfare of 20 in this instance.

In the next section, we empirically evaluate all the online auctions described above for average performance using different stochastic models for θ's.

4 Evaluating Online Auctions

This paper's goal is to study online auctions empirically to evaluate how they perform in practice using various stochastic models. To do this, we define performance measures for evaluating a given online auction.

4.1 Performance Measures for Online Auctions

Although CR captures an online auction's worst case performance, we believe that worst cases performances may not be recurrent in practice. To evaluate a given online auction, we generated N different instances of θ's according to a fixed stochastic model for θ. Let f_i be the probability density function (pdf) for θ_i, let S denote the set of N samples generated with $f_1, f_2, \ldots, f_n$ and let $f = f_1 \times \ldots \times f_n$ denote the joint probability distribution function. New measures are now defined as follows:

Definition 5 (ECR) *Online auction $\mathcal{A}$ is said to have an empirical β_f^N competitive ratio (ECR) if*

$$min_{\{\theta \in S: V^*(\theta) \neq 0\}} \left\{ mean_{\theta|v \text{ is fixed}} \frac{V^{\mathcal{A}}(\theta)}{V^*(\theta)} \right\} \geq \frac{1}{\beta_f^N}$$

The ECR measures how far online auction $\mathcal{A}$'s solution is away from the offline-optimal using generated samples. Even for large N, if the ECR is good, then worst cases are rare.

Definition 6 (SACR) *Online auction $\mathcal{A}$ has a sample average competitive ratio (SACR) γ_f^N if*

$$mean_{\{\theta \in S: V^*(\theta) \neq 0\}} \left\{ mean_{\theta|v \text{ is fixed}} \frac{V^{\mathcal{A}}(\theta)}{V^*(\theta)} \right\} \geq \frac{1}{\gamma_f^N}$$

An SACR measures, via an analysis of average cases, how far online auction $\mathcal{A}$'s solution is away from the offline-optimal, where the average is taken from generated samples. Even for large N, if the SACR is low, then, on average, the auction performs better than one with a higher SACR.

[6] CRs for APSD, SDV, eAuction are taken from [10]. We believe the CR for ORCA should be better than eAuction.

Definition 7 (EEE) *Online auction $\mathcal{A}$ is said to have an empirical expected efficiency (EEE) as Δ_f^N, where*

$$\Delta_f^N = \frac{mean_{\theta \in S} V^{\mathcal{A}}(\theta)}{mean_{\theta \in S} V^*(\theta)}$$

The EEE captures the average fraction of expected offline-optimal social welfare achieved by online auction $\mathcal{A}$. The closer EEE is to 1, the closer, on average, $\mathcal{A}$ is to the offline-optimal.

The next section describes this empirical analysis.

4.2 Experiments

For a fixed number of resources (k), the parameters that can vary are the size of the agent pool (n), the agent arrival rate (λ), agent waiting time and agents' preferences v_{ij}'s. First, we explain the different models of agents' preferences considered in these experiments.

4.2.1 Preference Models

The following agents' preference models were considered:
Low Competition

- Preference Model 1 (PM1): each agent's valuation for each resource is an independent and identically distributed (i.i.d.) random variable with a uniform distribution on [0,1].
- Preference Model 2 (PM2): each agent's valuation for each resource is an i.i.d. random variable with a triangular distribution on [0,1] with a peak at 0.5.

High Competition

- Preference Model 3 (PM3): each agent has the same valuation for every resource and these valuations have a uniform distribution on [0,1].
- Preference Model 4 (PM4): Resources are ranked. Any agent's valuation for resource r_j is uniformly drawn from $[\frac{k-j}{k}, \frac{k-j+1}{k}]$.

In the first two PMs, there is relatively less competition between agents for each resource. The latter two PMs induce higher competition between agents for resources.

The next subsection explains the study's different experimental setups.

4.2.2 Experimental Setups

The four following experimental variations were analysed:
Experiment 1 (Effect of n on ECR, SACR and EEE for fixed k): This experiment fixed the number of tasks $k = 5$, $\lambda = 0.5$ and varied $n = 8 \rightarrow 20$.
Experiment 2 (Effect of k on ECR, SACR and EEE for fixed n): This experiment fixed the number of agents $n = 20$, $\lambda = 0.5$ and varies $k = 2 \rightarrow 20$.
Experiment 3 (Effect of λ on ECR, SACR and EEE for fixed n, k): This experiment varied the agent arrive rate (λ) on the platform for $k = 5, n = 20$ and the waiting period was exponentially distributed with mean $\mu = 0.5$.
Experiment 4 (Effect of λ on ECR, SACR and EEE for fixed n, k): This experiment is the same as Experiment 3 except that the agents are impatient.

For each of the four PMs, we generated 8,000 valuation profiles for each of the four experiments described above. For each valuation profile, 120 random agent orderings were considered. First, the sample averages of the total valuation achieved by each auction mechanism for these 120 orderings were calculated. Second, ECR and SACR were measured for the 8,000 sample valuation profiles. Also, the sample averages of the total valuation of each auction mechanism and the offline-optimals were calculated over 8,000 x 120 instances, in order to measure EEE. As ECR and SACR are > 1, and indeed may take much larger values, we plotted $\frac{1}{\text{ECR}}$ and $\frac{1}{\text{SACR}}$ to view them in $[0, 1]$.

These experiments used $k \in [2, 20]$ and $n \in [5, 50]$, as we believe that typical online auctions for resource assignment in new marketplaces will be of a similar size. For example, although there may be a large number of Uber drivers and passengers at the same time, a driver may only be interested in a couple of customers and there may not be many drivers nearby interested in every single customer. If k and n are scaled proportionately, we still believe that similar results will hold true.

4.2.3 Experimental Results

The following observations were common to all the experiments.

- *O1: Correlations between PM1-PM2 and PM3-PM4.* Across all four experiments, the three metrics for PM2 demonstrated the same trend as under PM1, but at different scales. There was a similar correlation between PM3 and PM4. This is attributed to the fact that PM1 and PM2 encourage little competition, and PM3 and PM4 encourage high competition. Hence, below, we illustrate our results w.r.t. PMs 1 and 3 only.
- *O2: Correlation Across ECR, SACR, EEE.* In general, the graphs for ECR, SACR and EEE showed similar trends, though scales and rates of change could differ. (Figure 1, Figure 2 and Figure 3).
- *O3: ORCA under High Competition.* In general, ORCA performed better when preferences induced more competition for resources (i.e., in PM3, PM4).
- *O4: CR vs ECR.* Worst case competitiveness across the generated samples (ECR) was much better than the theoretical CR, thus worst cases did not occur frequently. For example, CR across all the auctions, considering all the settings, is always > 7.38 and as bad as 50 in some cases. However, empirically all the auctions in our experiments were better than 5-competitive (1/ECR> 0.2).

Some of the specific observations were as follows.

- **Experiment 1:** Figures 1 to 3 show how ECR, SACR and EEE change w.r.t. n for the different auction mechanisms when $k = 5$ and $\lambda = 0.5$ for agents following PM1. In Experiment 1 and with all PMs, ORCA performed better for a larger agent pool on measures ECR and SACR. However, for EEE, SDV was the best auction mechanism across all PMs. These experiments clearly show that as competition increases further, ORCA should outperform SDV in all the PMs. Figure 4 illustrates ORCA's superiority under PM3. Because of the correlations in O2, we drop other plots using PM3 to save space.
- **Experiment 2:** Figure 5 illustrates how ECR varies w.r.t. the number of resources, k when $n = 20, \lambda = 0.5$ under PM1. Similar behaviour was observed under PM3, where ORCA was superior to the other mechanisms until $k = 8$. As competition reduces, (i.e., k increases) SDV and APSD perform better (because of O1 and O2, not all measures under all PMs are displayed).
- **Experiment 3:** The arrival rate's effect on auction mechanism performance was also studied (Figures 6 and 7). Experiments demonstrated that for $\lambda \leq 1$, threshold-based online auctions performed

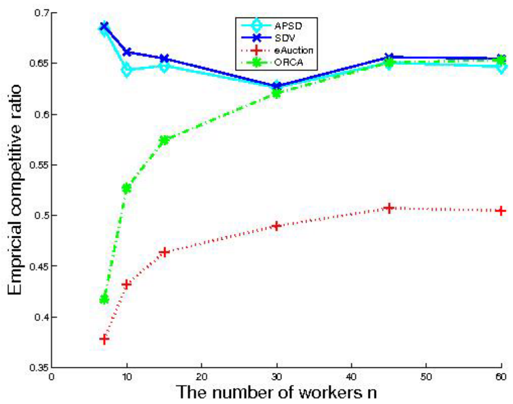

Figure 1. Experiment 1: ECR vs n for $k = 5, \lambda = 0.5$, PM1

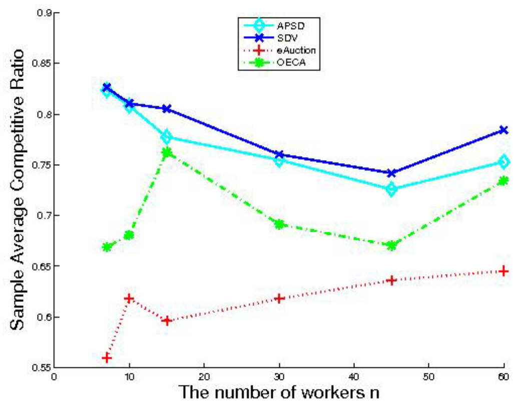

Figure 2. Experiment 1: SACR vs n for $k = 5, \lambda = 0.5$, PM1

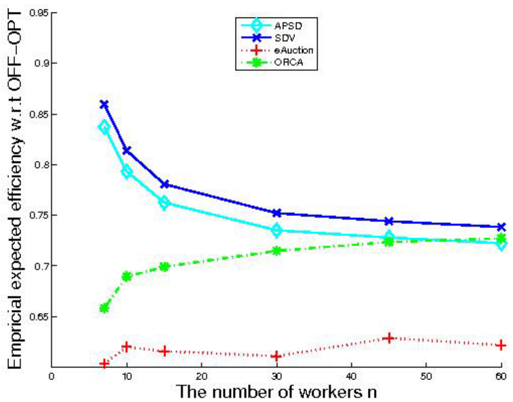

Figure 3. Experiment 1: EEE vs n for $k = 5, \lambda = 0.5$, PM1

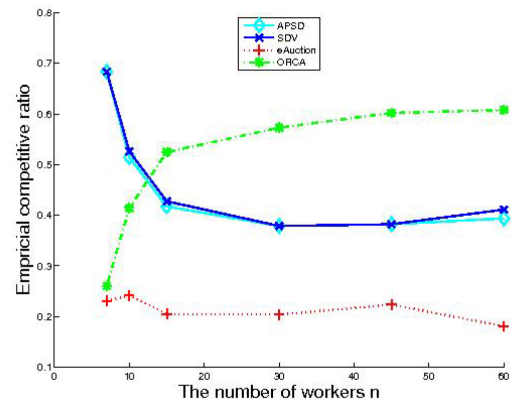

Figure 4. Experiment 1: ECR vs n for $k = 5, \lambda = 0.5$, PM3

better than APSD and SDV as measured by SACR under PM3. However, for higher λ, i.e., when agents arrive in large numbers at every time slot, SDV performed better, especially as measured using EEE or ECR. APSD performed better than threshold-based online auctions, but 5%-10% below SDV.

- **Experiment 4:** In Experiment 4, all the auctions showed a similar performance trend to Experiment 3. The performances of APSD and e-Auction are not supposed to change significantly in the presence of impatient agents. Experiments also showed that ORCA did not change much in the presence of impatient agents. SDV performed slightly less well in Experiment 4, but was still superior at higher λ, without changing any of our conclusions. Hence we do not display plots for Experiment 4.

4.3 Discussion

Based on these experiments, we consider two broad settings:
(1) Low competition for resources and/or a high arrival rate.
(2) High competition for resources and/or a low arrival rate.

Low Competition If many agents log in simultaneously (high λ), SDV is superior to all the other online auctions (Figures 6 and 7). As the number of resources increases, the performances of SDV and APSD improve and they become superior to threshold-based auctions (Experiment 2).

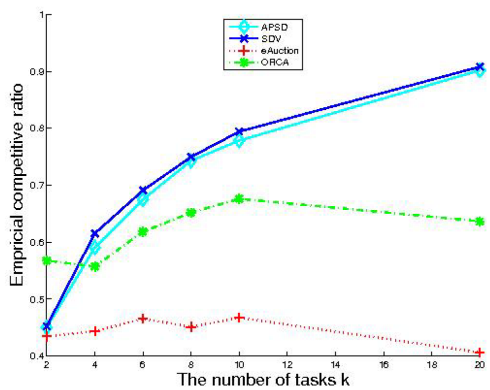

Figure 5. Experiment 2: ECR vs k for $n = 20, \lambda = 0.5$, PM1

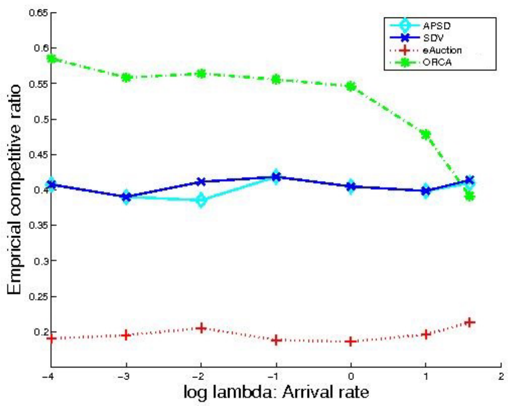

Figure 6. Experiment 3: ECR vs λ for $k = 5, n = 20$, PM3

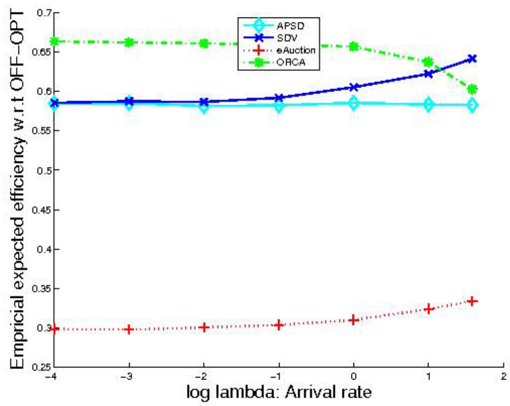

Figure 7. Experiment 3: EEE vs λ for $k = 5, n = 20$, PM3

Note that the empirical superiority of SDV might be attributable to the fact that it tries to match many agents simultaneously, leading to more efficient assignment. Threshold-based online auctions typically drop a certain fraction of agents in order to learn which asymptotically improves worst case guarantees. Hence, they perform best only in highly competitive settings, as explained below.

High Competition If there is strong competition between agents (that is either large n for fixed k, or $\frac{k}{n} < 0.1$), threshold-based online auctions (especially ORCA) perform better (Figures 1 to 5) for all types of PMs.

Overall, these experiments showed that ORCA outperforms the other auctions when (i) agents arrive sequentially (very low λ) and are impatient, and (ii) preferences models are of the PM3 or PM4 type.

From Table 1's ranking of auction mechanism CRs, APSD $\prec$ SDV $\prec$ e-Auction $\prec$ ORCA. However, the experiments presented here rank the four auction mechanisms, by measure, as shown in Table 3.

	APSD	SDV	eAuction	ORCA
CR	4	3	2	1
Low Competition and High λ				
ECR	2	1	4	3
SACR	2	1	4	3
EEE	2	1	4	3
High Competition and Low λ				
ECR	3	2	4	1
SACR	3	2	4	1
EEE	3	1	4	2

Table 3. Comparison of APSD, SDV, eAuction and ORCA: relative rankings using CR, ECR, SACR and EEE from empirical analysis

Recommendations If there are few resources and a large pool of agents, the auction platform should choose threshold-based online auctions. If the platform expects that (i) all agents put the same valuation on each different resource or, (ii) some resources are preferred over others, then the platform can implement ORCA. If the agents' valuations of resources are independent of each other, then the platform can use eAuction.

If there are large numbers of resources or large numbers of agents logging in to the system at every time period, then the platform can use SDV. However, SDV can be manipulated for arrival-departure. If the platform prefers not to charge the agents for resource assignment and/or prefers to work using no-early-arrival-no-late-departure domains, then it can use APSD. APSD is simple to implement but has a cost of a 5%-10% loss in performance compared to SDV. However, in many settings and preference models, it is better than threshold-based online auctions.

5 Summary

This paper addressed the resource assignment problem for dynamic agents and proposed a new Online Ranked Competition Auction (ORCA) mechanism to deal with this. We hypothesised that the auctions targeted for worst case guarantees perform better in practice only when there is strong competition for resources between agents, i.e., the degree of competition between agents plays an important role in the trade-off between properties such as truthfulness, expressiveness, efficiency and average case performance. Our experiments validated this hypothesis.

We studied the application of four different online auctions to the resource assignment problem, namely APSD, SDV, eAuction and ORCA. We compared their theoretical properties (Table 1). Instead of relying exclusively on the competitive ratio to evaluate average case online auctions, we proposed three new measures, namely ECR, SACR and EEE. Furthermore, experimental worst cases generated from samples were much better than theoretical worst cases (Table 3). In the last section, we provided suggestions as to how a platform should choose its online auction mechanism based on the size of the agent pool, the size of the resource pool and how frequently agents log in to the system.

The ORCA and eAuction mechanisms were only observed to give better average-case performances in specific preference models. Otherwise, overall, SDV is a very good online auction mechanism when compared to the others studied in this paper. Future research might attempt to design a better model-free resource assignment mechanism (online auction), one that is more efficient than SDV and is truthful in no-early-arrival-no-late-departure domains for a broad class of preference models.

REFERENCES

[1] Susan Athey and Ilya Segal. An efficient dynamic mechanism. *Econometrica*, 81(6):2463–2485, 2013.

[2] Moshe Babaioff, Nicole Immorlica, David Kempe, and Robert Kleinberg. Online auctions and generalized secretary problems. *ACM SIGecom Exchanges*, 7(2):7, 2008.

[3] Moshe Babaioff, Nicole Immorlica, and Robert Kleinberg. Matroids, secretary problems, and online mechanisms. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 434–443. Society for Industrial and Applied Mathematics, 2007.

[4] Dirk Bergemann and Juuso Välimäki. The dynamic pivot mechanism. *Econometrica*, 78(2):771–789, 2010.

[5] Niv Buchbinder and Joseph Naor. The design of competitive online algorithms via a primal: dual approach. *Foundations and Trends® in Theoretical Computer Science*, 3(2–3):93–263, 2009.

[6] Xi Chen, Qihang Lin, and Dengyong Zhou. Optimistic knowledge gradient policy for optimal budget allocation in crowdsourcing. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 64–72, 2013.

[7] Eugene B Dynkin. The optimum choice of the instant for stopping a markov process. In *Soviet Math. Dokl*, volume 4, 1963.

[8] Amos Fiat. Online algorithms: The state of the art (lecture notes in computer science). 1998.

[9] Gagan Goel, Afshin Nikzad, and Adish Singla. Allocating tasks to workers with matching constraints: truthful mechanisms for crowdsourcing markets. In *Proceedings of the companion publication of the 23rd international conference on World wide web companion*, pages 279–280, 2014.

[10] Sujit Guajr and Boi Faltings. Auction based mechanisms for dynamic task assignments in expert crowdsourcing. In *Proceedings of the International workshop on Agent Mediated E-Commerce and Trading Agent Design and Analysis (AMEC/TADA'15)*, 2015.

[11] Chien-Ju Ho, Shahin Jabbari, and Jennifer W Vaughan. Adaptive task assignment for crowdsourced classification. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 534–542, 2013.

[12] Chien-Ju Ho and Jennifer Wortman Vaughan. Online task assignment in crowdsourcing markets. In *AAAI*, 2012.

[13] Nicole Immorlica, Robert Kleinberg, and Mohammad Mahdian. Secretary problems with competing employers. In *Internet and Network Economics*, pages 389–400. Springer, 2006.

[14] David R Karger, Sewoong Oh, and Devavrat Shah. Budget-optimal crowdsourcing using low-rank matrix approximations. In *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*, pages 284–291. IEEE, 2011.

[15] Anna Karlin and Eric Lei. On a competitive secretary problem. In *AAAI*, pages 944–950, 2015.

[16] R. M. Karp, U. V. Vazirani, and V. V. Vazirani. An optimal algorithm for on-line bipartite matching. In *Proceedings of the Twenty-second Annual ACM Symposium on Theory of Computing*, STOC '90, pages 352–358, New York, NY, USA, 1990. ACM.

[17] Denis V Lindley. Dynamic programming and decision theory. *Applied Statistics*, pages 39–51, 1961.

[18] Noam Nisan. Introduction to mechanism design. In Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay Vazirani, editors, *Algorithmic game theory*, pages 209–242. Cambridge University Press, 2007.

[19] David Parkes. Online mechanisms. In Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay Vazirani, editors, *Algorithmic game theory*, pages 411–439. Cambridge University Press, 2007.

[20] David C. Parkes, Ruggiero Cavallo, Florin Constantin, and Satinder Singh. Dynamic Incentive Mechanisms. *Artificial Intelligence Magazine*, 31:79–94, 2010.

[21] Long Tran-Thanh, Sebastian Stein, Alex Rogers, and Nicholas R Jennings. Efficient crowdsourcing of unknown experts using multi-armed bandits. In *European Conference on Artificial Intelligence*, pages 768–773, 2012.

[22] James Y Zou, Sujit Gujar, and David C Parkes. Tolerable manipulability in dynamic assignment without money. In *AAAI*, 2010.