Delegated Online Search

Pirmin Braun[1] , **Niklas Hahn**[2] , **Martin Hoefer**[1] and **Conrad Schecker**[1]
[1] Goethe University Frankfurt, Germany
[2] Sorbonne Université, CNRS, LIP6 UMR 7606, Paris, France
pirminbraun16@gmail.com, niklas.hahn@lip6.fr, {mhoefer,schecker}@em.uni-frankfurt.de

Abstract

In a delegation problem, a *principal* $\mathcal{P}$ with commitment power tries to pick one out of n options. Each option is drawn independently from a known distribution. Instead of inspecting the options herself, $\mathcal{P}$ delegates the information acquisition to a rational and self-interested *agent* $\mathcal{A}$. After inspection, $\mathcal{A}$ proposes one of the options, and $\mathcal{P}$ can accept or reject. In this paper, we study a natural *online* variant of delegation, in which the agent searches through the options in an online fashion. How can we design algorithms for $\mathcal{P}$ that approximate the utility of her best option in hindsight?

We show that $\mathcal{P}$ can obtain a $\Theta(1/n)$-approximation and provide more fine-grained bounds independent of n based on two parameters. If the ratio of maximum and minimum utility for $\mathcal{A}$ is bounded by a factor α, we obtain an $\Omega(\log\log\alpha/\log\alpha)$-approximation algorithm, and we show that this is best possible. If $\mathcal{P}$ cannot distinguish options with the same value for herself, we show that ratios polynomial in $1/\alpha$ cannot be avoided. If the utilities of $\mathcal{P}$ and $\mathcal{A}$ for each option are related by a factor β, we obtain an $\Omega(1/\log\beta)$-approximation, and $O(\log\log\beta/\log\beta)$ is best possible.

1 Introduction

The study of delegation problems is a prominent area with numerous applications. There are two parties – a decision maker (called *principal*) $\mathcal{P}$ and an *agent* $\mathcal{A}$. n actions or *options* are available to $\mathcal{P}$. Each option has a utility for $\mathcal{P}$ and a (possibly different) utility for $\mathcal{A}$, which are drawn from a known distribution $\mathcal{D}$. Instead of inspecting options herself, $\mathcal{P}$ delegates the search for a good option to $\mathcal{A}$. $\mathcal{A}$ sees all realized utility values and sends a signal to $\mathcal{P}$. Based on this signal (and $\mathcal{D}$), $\mathcal{P}$ chooses an option. Both parties play this game in order to maximize their respective utility from the chosen option.

Many interesting applications can be captured within this framework. For example, consider a company that is trying to hire an expert in a critical area. Instead of searching the market, the company delegates the search to a head-hunting agency that searches the market for suitable candidates. Alternatively, consider an investor, who hires a financial consultant to seek out suitable investment opportunities. Clearly, principal and agent might not always have aligned preferences. While the investor might prefer investments with high interest rates, the financial consultant prefers selling the products for which he gets a provision.

In applications such as searching for job candidates or financial investments, availability of options often changes over time, and the pair of agents needs to solve a stopping problem. For example, many lucrative financial investment opportunities arise only within short notice and expire quickly. Therefore, a consultant has to decide whether or not to recommend an investment without exactly knowing what future investment options might become available. Here $\mathcal{A}$ faces an online search problem, in which the n options are realized in a sequential fashion. After seeing the realization of option i, he has to decide whether to propose the option to $\mathcal{P}$ or discard it. If the option is proposed, $\mathcal{P}$ decides to accept or reject this option and the process ends. Otherwise, the process continues with option $i+1$.

In the study of delegation problems, $\mathcal{P}$ usually has commitment power, i.e., $\mathcal{P}$ specifies in advance her decision for each possible signal, taking into account the subsequent best response of $\mathcal{A}$. This is reasonable in many applications (e.g., an investor can initially restrict the investment options she is interested in, or the company fixes in advance the required qualifications for the new employee). Interestingly, although $\mathcal{P}$ commits and restricts herself in advance, this behavior is usually in her favor. The induced best response of $\mathcal{A}$ can lead to better utility for $\mathcal{P}$ than in any equilibrium, where both parties mutually best respond. Using a revelation-principle style argument, the communication between $\mathcal{P}$ and $\mathcal{A}$ can be reduced to $\mathcal{A}$ revealing the utilities of a single option and $\mathcal{P}$ deciding to accept or reject that option (for a discussion, see, e.g. [Kleinberg and Kleinberg, 2018]).

The combination of online search and delegation has been examined before, albeit from a purely technical angle. Kleinberg and Kleinberg [2018] designed approximation algorithms for delegation, showing that $\mathcal{P}$ can obtain a constant-factor approximation to the expected utility of her best option in hindsight. Their algorithms heavily rely on techniques and tools developed in the domain of prophet inequalities. However, they are applied to an *offline* delegation problem. Instead, we consider the natural extension of [Kleinberg and

Kleinberg, 2018] to online search. Interestingly, we exhibit a notable contrast – in online delegation a constant-factor approximation might be impossible to achieve. In fact, the approximation ratio can be as low as $\mathcal{O}(1/n)$, and this can always be achieved. Motivated by this sharp contrast, we provide a fine-grained analysis based on two natural problem parameters: (1) the discrepancy of utility for the agent, and (2) the misalignment of agent and principal utilities.

1.1 Model

We study *online delegation* between principal $\mathcal{P}$ and agent $\mathcal{A}$ in (up to) n rounds. In every round i, an option is drawn independently from a known distribution $\mathcal{D}_i$ with finite support Ω_i of size s_i. We denote the options of $\mathcal{D}_i$ by $\Omega_i = \{\omega_{i1}, \ldots, \omega_{i,s_i}\}$ and the random variable of the draw by O_i. For every $i \in [n]$ and $j \in [s_i]$, the option ω_{ij} has probability p_{ij} to be drawn from $\mathcal{D}_i$. If this option is proposed by $\mathcal{A}$ and chosen by $\mathcal{P}$, then $\mathcal{A}$ has utility $a_{ij} \geq 0$ and $\mathcal{P}$ utility $b_{ij} \geq 0$.

We assume that $\mathcal{P}$ has commitment power. Before the start of the game, she commits to an *action scheme* φ with a value $\varphi_{ij} \in [0, 1]$ for each option ω_{ij}. φ_{ij} is the probability that $\mathcal{P}$ accepts option ω_{ij} when it is proposed by $\mathcal{A}$ in round i. For a *deterministic* scheme with all $\varphi_{ij} \in \{0, 1\}$, we define the sets $E_i = \{\omega_{ij} \mid \varphi_{ij} = 1\}$ of *acceptable options* in each round i.

In contrast to $\mathcal{P}$, $\mathcal{A}$ gets to see the n random draws from the distributions in an online fashion. He decides after each round whether he proposes the current option O_i to $\mathcal{P}$ or not. If he decides to propose it, then $\mathcal{P}$ sees the option and decides based on φ whether or not she accepts it. If $\mathcal{P}$ accepts, the utility values of the option are realized; if not, both players get utility 0. In either case, the game ends after $\mathcal{P}$ decides. Both players strive to maximize their expected utility.

Initially, both players know the distribution $\mathcal{D}_i$ for every round $i \in [n]$. The sequence of actions then is as follows: (1) $\mathcal{P}$ decides φ and communicates this to $\mathcal{A}$; (2) in each round i, $\mathcal{A}$ sees $O_i \sim \mathcal{D}_i$ and irrevocably decides to propose or discard O_i; (3) when $\mathcal{A}$ decides to propose some option $O_i = \omega_{ij}$, then $\mathcal{P}$ accepts it with probability φ_{ij}, and the game ends.[1]

$\mathcal{A}$ knows the distributions and the action scheme φ of upcoming rounds which determines his expected utility from proposed options. Hence, $\mathcal{A}$ essentially faces an online stopping problem that can be solved via backwards induction. We can assume without loss of generality that all decisions (not) to propose an option by $\mathcal{A}$ are deterministic: If the expected utility from the realization in the current round is greater than the expected utility obtained in the remaining rounds, propose the current option (otherwise do not). To avoid technicalities, we assume that $\mathcal{A}$ breaks ties in favor of $\mathcal{P}$.

Our goal is to design action schemes φ with high expected utility for $\mathcal{P}$. We compare the expected utility to the one in the non-delegated (online) problem, where $\mathcal{P}$ searches through the n realized options herself. The latter is a classic stopping problem, for which the expected utilities of optimal online and offline search differ by at most a factor of 2 (due to the basic prophet inequality [Krengel and Sucheston, 1977; Krengel and Sucheston, 1978]).

[1] In the full version [Braun *et al.*, 2022], we also consider a variant with $k \geq 1$ proposals. Here the game continues until $\mathcal{P}$ accepts or after $\mathcal{P}$ has rejected k different options. Action schemes become more complicated and must rely on the history of rejected options.

	Round 1		Round 2	
option ω_{ij}	ω_{11}	ω_{12}	ω_{21}	ω_{22}
value-pair (a_{ij}, b_{ij})	(3,1)	(3,8)	(2,4)	(16,4)
probability p_{ij}	0.75	0.25	0.75	0.25

Table 1: An example instance

We also analyze scenarios with *oblivious* and *semi-oblivious proposals*. In both these scenarios, $\mathcal{A}$ reveals only the utility value b_{ij} for $\mathcal{P}$ when proposing an option (but not his own value a_{ij}). In contrast, when $\mathcal{P}$ gets to see the utility values of both agents, we term this *conscious proposals*. The difference between semi-oblivious and (fully) oblivious scenarios lies in the prior knowledge of $\mathcal{P}$. In the semi-oblivious scenario, $\mathcal{P}$ is fully aware of the distributions, including all potential utility values a_{ij} for $\mathcal{A}$. In the oblivious scenario, $\mathcal{P}$ initially observes the probabilities of all options along with her utility values b_{ij}, but the values a_{ij} of $\mathcal{A}$ remain unknown to $\mathcal{P}$ throughout. In the scenarios with restricted discrepancy (in Section 3), we assume $\mathcal{P}$ is aware of the bound $\alpha = \max_{i,j} a_{ij} / \min_{i,j} a_{ij}$.

Example 1. We discuss a simple example to illustrate the definitions. We consider deterministic strategies by $\mathcal{P}$ and conscious proposals. There are two rounds with the options distributed according to Table 1. For the benchmark, we assume that $\mathcal{P}$ can see and choose the options herself. The best option is ω_{12}. If this is not realized in round 1, the option realized in round 2 is the best choice. Note that this optimal choice for $\mathcal{P}$ can be executed even in an online scenario, where she first sees round 1 and gets to see round 2 only after deciding about round 1. The expected utility of this best (online) choice for $\mathcal{P}$ is 5.

Now in the delegated scenario, suppose $\mathcal{P}$ accepts option ω_{22}. Then $\mathcal{A}$ would always wait for round 2 and hope for a realization of ω_{22}, even if ω_{21} would not be accepted by $\mathcal{P}$. Hence, accepting ω_{22} leads to an expected utility for $\mathcal{P}$ of at most 4. In contrast, the optimal decision scheme for $\mathcal{P}$ is to accept only ω_{12} and ω_{21} with an expected utility of 4.25. For the (semi-)oblivious scenario, $\mathcal{P}$ cannot distinguish the options in round 2, and her expected utility is at most 4.

Clearly, $\mathcal{P}$ has to strike a careful balance between (1) accepting a sufficient number of high-profit options to obtain a high expected utility overall and (2) rejecting options to motivate $\mathcal{A}$ to propose better options for $\mathcal{P}$ in earlier rounds. ♦

1.2 Contribution and Outline

In Section 2 we show that the worst-case approximation ratio for online delegation is $\Theta(1/n)$ and this is tight. Intuitively, $\mathcal{A}$ waits too long and forgoes many profitable options for $\mathcal{P}$. $\mathcal{P}$ can only force $\mathcal{A}$ to propose earlier if she refuses to accept later options – this, however, also hurts the utility of $\mathcal{P}$. The instances require a ratio of maximum and minimum utility values for $\mathcal{A}$ that is in the order of $n^{\Theta(n)}$. We further show that the bounds extend to more general variants in which (1) $\mathcal{A}$ has a lookahead of k rounds, or (2) $\mathcal{A}$ can propose up to

k options, resulting in tight approximation ratios of $\Theta(k/n)$. Note that all impossibility results in this paper apply already for IID instances, while all algorithmic results apply for general instances.

In Section 3, we examine the effect of the discrepancy of utility for $\mathcal{A}$ using the ratio α of maximum and minimum utility values. We obtain an $\Omega(\log\log\alpha/\log\alpha)$-approximation of the optimal (online) search for $\mathcal{P}$. This result is tight. The algorithm limits the acceptable options of $\mathcal{P}$, partitions them into different bins, and then restricts $\mathcal{A}$'s search space to the best possible bin for $\mathcal{P}$. The challenge is to design a profitable set of options inside a bin that should be accepted by $\mathcal{P}$ without giving $\mathcal{A}$ an incentive to forgo proposing many of them. Our algorithm shows that $\mathcal{P}$ can obtain a good approximation even if differences in utility of $\mathcal{A}$ are polynomial in n.

Additionally, we consider the more challenging *oblivious* scenario in which $\mathcal{P}$ does not get to see the agent's utility of the proposed option. Our algorithm for this scenario achieves an $\Omega(1/\alpha)$-approximation. This is contrasted with a set of instances for which no action scheme can extract more than an $O(1/\alpha)$-approximation. We study the *semi-oblivious scenario* in the full version [Braun *et al.*, 2022]. In this scenario, $\mathcal{P}$ has a priori knowledge of the prior, but still does not see the agent's utility for proposed options. For this setting, our algorithm achieves an $\Omega(1/(\sqrt{\alpha}\log\alpha))$-approximation, and in general $O(1/\sqrt{\alpha})$ is best-possible. The results highlight the effect of hiding $\mathcal{A}$'s utilities from $\mathcal{P}$ (in the proposal, or in the proposal and the prior) – the achievable approximation ratios increase from logarithmic to polynomial ratios in α.

In Section 4, we consider the misalignment of agent and principal utilities via a parameter $\beta \geq 1$, which is the smallest value such that all utilities of $\mathcal{P}$ and $\mathcal{A}$ are related by a factor in $[1/\beta, \beta]$. Limited misalignment also leads to improved approximation results for $\mathcal{P}$. We show an $\Omega(1/\log\beta)$-approximation of the optimal (online) search for $\mathcal{P}$. Moreover, every algorithm must have a ratio in $O(\log\log\beta/\log\beta)$. For the agent-oblivious variant, we obtain an $\Omega(1/\beta)$-approximation, whereas every algorithm must have a ratio in $O(1/\sqrt{\beta})$.

Additional descriptions, pseudocode for our algorithms as well as all missing proofs can be found in the full version of this paper [Braun *et al.*, 2022].

1.3 Related Work

Holmstrom [1977; 1984] initiated the study of delegation as a bilevel optimization between an uninformed principal and a privately informed agent. The principal delegates the decision to the agent who himself has an interest in the choice of decision. Since then, there have been numerous works on various aspects of delegation. For example, [Melumad and Shibano, 1991; Alonso and Matouschek, 2008] studied the impact of (mis)alignment of the agent's and the principal's interests on the optimal delegation sets. Armstrong and Vickers [2010] studied the delegation problem over discrete sets of random cardinality with elements drawn from some distribution. They identify sufficient conditions for the search problem to have an optimal solution.

A similar model to ours was considered in computer science by Kleinberg and Kleinberg [2018], where the option set searched by the agent consists of n IID draws from a known distribution. They show constant-factor approximations of the optimal expected principal utility when performing the search herself rather than delegating it to the agent. For their analysis, they rely on tools from online stopping theory. The key difference between their work and our paper is that – albeit using tools from online optimization – they study an *offline* problem while we focus on an *online* version.

Bechtel and Dughmi [2021] recently extended this line of research by combining delegation with stochastic probing. Here a subset of elements can be observed by the agent (subject to some constraints), and several options can be chosen (subject to a different set of constraints). Similarly, Bechtel et al. [2022] study connections between delegation and a generalized Pandora's Box problem.

A related but different area is contract theory, which considers principal-agent settings with uncertainty and monetary transfers. An early formalization was introduced by Grossmann and Hart [1983]. Computational aspects of this problem are recently starting to attract interest, see, e.g., Duetting et al. [2019; 2020] and earlier work (on a slightly different model) [Babaioff *et al.*, 2012; Babaioff and Winter, 2014].

The study of persuasion, another model of strategic communication, has gained a lot of traction at the intersection between economics and computation in recent years. Here, the informed party (the "sender") is the one with commitment power, trying to influence the behavior of the uninformed agent (the "receiver"). Persuasion has proven to be a popular topic in AI. Castiglioni et al. [2022] studied Bayesian posted price auctions where buyers arrive sequentially and receive signals from the revenue maximizing seller. Moreover, signaling may be used in other settings, e.g., persuading voters [Castiglioni *et al.*, 2020a; Castiglioni and Gatti, 2021], or for reducing social cost in congestion games with uncertain delays [Castiglioni *et al.*, 2021a; Griesbach *et al.*, 2022]. Closer to our paper is the study of persuasion in the context of stopping problems [Hahn *et al.*, 2020b; Hahn *et al.*, 2020a]. These works study persuasion problems in a prophet inequality [Hahn *et al.*, 2020a] as well as in a secretary setting [Hahn *et al.*, 2020b].

Other notable algorithmic results on persuasion problems concern optimal policies, hardness, and approximation algorithms in the general case [Dughmi and Xu, 2016] as well as in different variations, e.g., with multiple receivers [Babichenko and Barman, 2017; Badanidiyuru *et al.*, 2018; Dughmi and Xu, 2017; Rubinstein, 2017; Xu, 2020], with limited communication complexity [Dughmi *et al.*, 2016; Gradwohl *et al.*, 2021], or relations to online learning [Castiglioni *et al.*, 2020b; Castiglioni *et al.*, 2021b; Zu *et al.*, 2021]

2 Impossibility

2.1 A Tight Bound

As a first simple observation, note that $\mathcal{P}$ can always achieve an n-approximation with a deterministic action scheme, even in with oblivious proposals. $\mathcal{P}$ accepts exactly all options in a single round i^* with optimal expected utility, i.e., $E_{i^*} = \{\omega_{i^*,j} \mid j \in [s_{i^*}]\}$ for $i^* = \arg\max_{i \in [n]} \mathbb{E}[b_{ij}]$, and $E_j = \emptyset$ otherwise. This motivates $\mathcal{A}$ to always propose the option from round i^*, and $\mathcal{P}$ gets expected utility $\mathbb{E}[b_{i^*,j}]$. By a union bound, the optimal utility from searching through all options herself is upper bounded by $\mathbb{E}\left[\sum_i b_{ij}\right] \le n \cdot \mathbb{E}[b_{i^*,j}]$.

Proposition 1. *For online delegation there is a deterministic action scheme φ such that $\mathcal{P}$ obtains at least a $1/n$-approximation of the expected utility for optimal (online) search.*

We show a matching impossibility result, even in the IID setting with $\mathcal{D}_i = \mathcal{D}$ for all rounds $i \in [n]$, and when $\mathcal{P}$ gets to see the full utility pair of any proposed option. There are instances in which $\mathcal{P}$ suffers a deterioration in the order of $\Theta(n)$ over the expected utility achieved by searching through the options herself.

For the proof, consider the following class of instances. The distribution $\mathcal{D}$ can be cast as an independent composition, i.e., we independently draw the utility values for $\mathcal{P}$ and $\mathcal{A}$. For $\mathcal{P}$ there are two possibilities, either utility 1 with probability $1/n$, or utility 0 with probability $1-1/n$. For $\mathcal{A}$, there are n possibilities with agent utility of $n^{4\ell}$, for $\ell = 1, \ldots, n$, where each one has probability $1/n$. In combination, we can view $\mathcal{D}$ as a distribution over $j = 1, \ldots, 2n$ options. Options ω_j for $j = 1, \ldots, n$ have probability $1/n^2$ and utilities $(b_j, a_j) = (1, n^{4j})$, for $j = n+1, \ldots, 2n$ they have probability $1/n - 1/n^2$ and utilities $(b_j, a_j) = (0, n^{4(j-n)})$.

Theorem 1. *There is a class of instances of online delegation in the IID setting, in which every action scheme φ obtains at most an $O(1/n)$-approximation of the expected utility for optimal (online) search.*

Proof. For simplicity, we first show the result for schemes φ with $\varphi_{ij} = 0$ for all rounds $i \in [n]$ and all $j = n+1, \ldots, 2n$. In the end of the proof we discuss why this can be assumed for an optimal scheme.

Since all options $j \in [n]$ have the same utility for $\mathcal{P}$, she wants to accept one of them as soon as it appears. If she searches through the options herself, the probability that there is an option of value 1 is $1 - (1 - 1/n)^n \ge 1 - 1/e$. Her expected utility is a constant. In contrast, when delegating the search to $\mathcal{A}$, the drastic utility increase motivates him to wait for the latest round in which a better option is still acceptable by $\mathcal{P}$. As a result, $\mathcal{A}$ waits too long, and removing acceptable options in later rounds does not remedy this problem for $\mathcal{P}$.

More formally, interpret an optimal scheme φ as an $n \times n$ matrix, for rounds $i \in [n]$ and options $j \in [n]$. We outline some adjustments that preserve the optimality of matrix φ.

Consider the set S of all entries with $\varphi_{ij} \le 1/n$. For each $(i,j) \in S$, the probability that option j is realized in round i is $1/n^2$. When it gets proposed by $\mathcal{A}$, then it is accepted by $\mathcal{P}$ with probability at most $1/n$. By a union bound, the utility obtained from all these options is at most $|S|/n^2 \cdot 1/n \le 1/n$.

Suppose we change the scheme by decreasing φ_{ij} to 0 for each $(i,j) \in S$. Then each entry in φ is either 0 or at least $1/n$. If $\mathcal{A}$ makes the same proposals as before, the change decreases the utility of $\mathcal{P}$ by at most $1/n$. Then again, in the new scheme $\mathcal{A}$ can have an incentive to propose other options in earlier rounds. Since all options with $\varphi_{ij} \neq 0$ have utility 1 for $\mathcal{P}$, this only leads to an increase of utility for $\mathcal{P}$. Moreover, in round 1 we increase all acceptance probabilities to $\varphi_{1j} = 1$ for $j \in [n]$. Then, upon arrival of such an option ω_j in round 1, the change can incentivize $\mathcal{A}$ to propose this option – which is clearly optimal for $\mathcal{P}$, since this is an optimal option for her. Since the change is in round 1, it introduces no incentive to wait for $\mathcal{A}$. As such, it can only increase the utility for $\mathcal{P}$. Now consider any entry $\varphi_{ij} \ge 1/n$. We observe two properties:

1. Suppose $\varphi_{i'j'} \ge 1/n$ for $i' < i$ and $j' < j$. Then $\mathcal{P}$ accepts realization $\omega_{j'}$ in round i' with positive probability, but she will also accept the better (for $\mathcal{A}$) realization ω_j in a later round i with positive probability. $\mathcal{A}$ will not propose $\omega_{j'}$ in round i' but wait for round i, since the expected utility in the later round i is at least $n^{4j} \cdot 1/n^2 \cdot \varphi_{ij} \ge n^{4j-3} > n^{4(j-1)} \ge n^{4j'} \cdot \varphi_{i'j'}$, the utility in round i'. As such, w.l.o.g. we set $\varphi_{i'j'} = 0$ for all $i' < i$ and $j' < j$.
2. Suppose $\varphi_{i'j} < \varphi_{ij}$ for $i' < i$. By property 1., all realizations $\omega_{j'}$ with $j' < j$ are not accepted in rounds $1, \ldots, i-1$. Hence, setting $\varphi_{i'j} = \varphi_{ij}$ does not change the incentives for $\mathcal{A}$ w.r.t. other options, and thus only (weakly) increases the expected utility of $\mathcal{P}$. By the same arguments, we set $\varphi_{ij'} = \max\{\varphi_{ij'}, \varphi_{ij}\}$ for all inferior options $j' < j$ in the same round i.

We apply the adjustments implied by the two properties repeatedly, starting for the entries φ_{in} in the n-th column for option ω_n, then in column $n-1$, etc. By 1., every positive entry $\varphi_{ij} \ge 1/n$ leads to entries of 0 in all "dominated" entries $\varphi_{i'j'}$ with $i' < i$ and $j' < j$. As a consequence, the remaining positive entries form a Manhattan path in the matrix φ. The path starts at φ_{1n}, ends at φ_{n1}, and for each $\varphi_{ij} \ge 1/n$ it continues either at $\varphi_{i+1,j} \ge 1/n$ or $\varphi_{i,j-1} \ge 1/n$. See Table 2 for an example.

We can upper bound the expected utility of $\mathcal{P}$ by assuming that all $2n-1$ entries on the Manhattan path are 1 (i.e., φ is deterministic) and $\mathcal{A}$ proposes an acceptable option whenever possible. The probability that this happens is at most $(2n-1)/n^2 = O(1/n)$ by a union bound. This is an upper bound on the expected utility of $\mathcal{P}$ and proves the theorem for schemes with $\varphi_{ij} = 0$ for all $i \in [n]$ and $j \ge n+1$.

Finally, suppose $\varphi_{ij} > 0$ for some $j \ge n+1$. Clearly, option ω_j adds no value to the expected utility of $\mathcal{P}$. Moreover, the fact that ω_j has positive probability to be accepted in round i can only motivate $\mathcal{A}$ to refrain from proposing inferior options in earlier rounds. As such, setting $\varphi_{ij} = 0$ only (weakly) increases the utility of $\mathcal{P}$. □

The lower bound in Theorem 1 remains robust in several extensions of the model. In the full version of this paper [Braun *et al.*, 2022], we discuss two generalizations (when $\mathcal{A}$ has a lookahead of k rounds, and when $\mathcal{A}$ is allowed to make k proposals). For both scenarios, we show the following lower bounds (along with simple algorithms providing matching upper bounds).

Corollary 1. *Consider online delegation in the IID setting with either (1) lookahead k, or (2) k proposals. For each*

Rnd.	ω_1	ω_2	ω_3	ω_4		Rnd.	ω_1	ω_2	ω_3	ω_4		Rnd.	ω_1	ω_2	ω_3	ω_4
1	0.3	**0.2**	0.6	0.9		1	*0.3*	*0*	*0.6*	0.9		1	0	0	0	**1**
2	0.4	0.3	0.8	1	$\rightarrow$	2	*0.4*	*0.3*	0.8	**1**	$\rightarrow$	2	0	0	**1**	1
3	**0.2**	0.4	0.6	**0.2**		3	*0*	*0.4*	0.6	0		3	0	0	**0.7**	0
4	0.5	**0.1**	0.7	0		4	0.5	0	**0.7**	0		4	**0.7**	**0.7**	0.7	0

Table 2: Adjustments from the proof of Theorem 1 for an example scheme φ for $n = 4$. Left: Entries $\varphi_{ij} < 1/n$ (bold) get set to 0. The expected utility for $\mathcal{P}$ decreases by at most $1/n$. Middle: Italic entries show options that never get proposed due to options with bold entries (cf. property 1.). Italic entries can be set to 0. Right: Bold entries have been raised according to property 2. A Manhattan path of $2n - 1$ entries evolves.

scenario, there are classes of instances such that every action scheme φ obtains at most an $O(k/n)$-approximation of the expected utility for optimal (offline) search.

3 Discrepancy of Agent Utilities

3.1 Conscious Proposals

The lower bound instances in Theorem 1 have agent utilities between 1 and $n^{O(n)}$. Is such a drastic discrepancy necessary to show a substantial lower bound? Can we obtain better approximation results for instances with a smaller ratio of the maximum and minimum utility values for $\mathcal{A}$?

Here, we restrict to $a_{ij} > 0$ for all options and study the cases with $a_{ij} = 0$ in the full version [Braun *et al.*, 2022]. Let $\alpha = \max\{a_{ij} \mid i \in [n], j \in [s_i]\} / \min\{a_{ij} \mid i \in [n], j \in [s_i]\}$. W.l.o.g. we scale all utility values to $a_{ij} \in [1, \alpha]$, where both boundaries α and 1 are attained by at least one option. Then $\mathcal{A}$ has *α-bounded utilities*.

We show how to compute a good action scheme with respect to parameter α. Intuitively, we partition the best options for $\mathcal{P}$ that add up a total probability mass of roughly $1/2$ into $O(\log\alpha / \log\log\alpha)$ many bins. Each bin is constructed in a way such that $\mathcal{A}$ is incentivized to propose the first option he encounters from that particular bin, assuming that $\mathcal{P}$ only accepts options from that bin. The algorithm determines the best bin for $\mathcal{P}$ and deterministically restricts the acceptable options to the ones from this bin.

Let us discuss the algorithm in more detail (pseudocode can be found in the full version of this paper [Braun *et al.*, 2022]). As a first step, the algorithm uses a procedure RestrictOptions($\mathcal{D}_1, \dots, \mathcal{D}_n, m$) with parameter $m = 1/2$ as subroutine. The procedure considers all options in descending order of principal utility until a total probability mass m is reached. Starting out with $\hat{Q} = \emptyset$, options are added to $\hat{Q} = \{(i,j) \mid b_{ij} \geq b_{i',j'} \ \forall (i',j') \notin \hat{Q}\}$ as long as $\sum_{(i,j)\in\hat{Q}} p_{ij} < m$. The first option (i^*, j^*) that would reach or surpass the combined mass of m (i.e., such that $\sum_{(i,j)\in\hat{Q}\cup\{(i^*,j^*)\}} p_{ij} \geq m$) is considered separately. The procedure RestrictOptions then returns either $Q = \hat{Q}$ or $Q = \{(i^*, j^*)\}$, whichever set provides a better expected utility for $\mathcal{P}$. As a consequence, $\sum_{(i,j)\in Q} p_{ij} \cdot b_{ij} \geq m/2 \cdot \mathbb{E}_{\omega_{ij}\sim\mathcal{D}_i}[\max_{i\in[n]} b_{ij}]$. Lemma 1 summarizes the claim. The proof can be found in the full version [Braun *et al.*, 2022].

Lemma 1. *The subroutine RestrictOptions($\mathcal{D}_1, \dots, \mathcal{D}_n, m$) with distributions $\mathcal{D}_1, \dots, \mathcal{D}_n$ and a parameter $0 < m \leq 1$ as input identifies Q, the best set of options for $\mathcal{P}$, such that*

$$\sum_{(i,j)\in Q} p_{ij} \cdot b_{ij} \geq m/2 \cdot \mathbb{E}_{\omega_{ij}\sim\mathcal{D}_i}[\max_{i\in[n]} b_{ij}]$$

and either (1) the combined mass $\sum_{(i,j)\in Q} p_{ij} < m$ or (2) all options in Q arrive in the same round.

If the set Q returned by RestrictOptions only spans a single round i, the agent will always be incentivized to propose an acceptable option in round i. For this scenario, the algorithm only creates a single bin B_1. Otherwise, it continues with the second and third step described in the following.

In the second step of the algorithm, the options identified by RestrictOptions are classified by their utility for $\mathcal{A}$. The algorithm divides Q into $c = \lceil \log_2 \alpha \rceil$ classes depending on the agent utility such that the lowest and highest agent utilities in any given class differ by at most a factor of 2. More precisely, classes $\mathcal{C}_1, \dots, \mathcal{C}_c$ are constructed such that $\mathcal{C}_k = \{(i,j) \in Q \mid a_{ij} \in [2^{k-1}, 2^k)\}$ for $k = 1, \dots, c-1$ and $\mathcal{C}_c = \{(i,j) \in Q \mid a_{ij} \in [2^{c-1}, 2^c]\}$.

For the third step, subsequent classes (by their agent utility value) are combined into bins such that (1) the bins are as big as possible and (2) $\mathcal{A}$ optimizes his own expected utility by proposing the first option he encounters from any bin – assuming that only options from this bin are allowed. Classes are either added to a bin completely or not at all. Let s be the index of the class with the highest agent utilities currently considered, i.e., the first class to be added to the current bin B_b. We consider the classes by decreasing agent utility values, i.e., with indices $k = s, s-1, \dots$. While $2^{k-1} \geq 2^s \cdot \sum_{(i,j)\in B_b\cup\mathcal{C}_k} p_{ij}$, a rational $\mathcal{A}$ will always propose the first option available from the current bin if that is the only allowed bin as it has a higher utility than what $\mathcal{A}$ can expect from later rounds. Hence, the class currently under consideration can be added to the current bin.

Finally, having constructed all bins, the algorithm simply chooses the best one for $\mathcal{P}$.

Lemma 2 shows that our algorithm achieves an approximation ratio which is linear in the number of bins opened. The subsequent Lemma 3 bounds the number of bins opened, showing that it is in the order of $O(\log\alpha / \log\log\alpha)$. Together, the lemmas prove Theorem 2, our main result of the section.

Lemma 2. *Let ℓ be the number of bins opened by the algorithm. Then the scheme computed by the algorithm obtains* *at least an $1/(8\ell)$-approximation of the expected utility of the best option for $\mathcal{P}$ in hindsight.*

Proof. We know that Q satisfies $4 \cdot \sum_{(i,j)\in Q} p_{ij} \cdot b_{ij} \geq \mathbb{E}_{\omega_{ij}\sim\mathcal{D}_i}[\max_{i\in[n]} b_{ij}] = \mathrm{OPT}$ by Lemma 1.

Now consider the construction of the bins. Suppose we split Q into ℓ bins $B_1, B_2, \ldots, B_\ell$. We pick the best one B_{b^*} from the ℓ bins $B_1, \ldots, B_\ell$, so

$$\sum_{(i,j)\in B_{b^*}} p_{ij} b_{ij} \geq \frac{1}{\ell} \sum_{(i,j)\in Q} p_{ij} b_{ij} \geq \frac{1}{4\ell} \cdot \mathrm{OPT} \quad .$$

The action scheme restricts attention to B_{b^*} and accepts each proposed option ω_{ij} from the bin with probability 1. Let $k^- = \min\{k \mid \mathcal{C}_k \subseteq B_{b^*}\}$ be the class of smallest index in B_{b^*}, and k^+ the one with largest index, respectively. Now suppose the agent learns in round i that an option ω_{ij} with $(i,j) \in B_{b^*}$ arrives in this round. We claim that $\mathcal{A}$ will then decide to propose this option. This is obvious if all options in B_{b^*} are only realized in round i. Otherwise, the agent might want to wait for an option in a later round. If $\mathcal{A}$ proposes, then his utility is a_{ij}. Otherwise, if he waits for another option from B_{b^*} in a later round, then a union bound shows that the expected utility is at most

$$\sum_{\substack{(i',j')\in B_{b^*}\\ i'>i}} p_{i'j'} \cdot a_{i'j'} \leq \sum_{\substack{(i',j')\in B_{b^*}\\ i'>i}} p_{i'j'} \cdot 2^{k^+}$$
$$< 2^{k^+} \cdot \sum_{(i',j')\in B_{b^*}} p_{i'j'} \leq 2^{k^- -1} \leq a_{ij} \quad ,$$

where the second-to-last inequality is a consequence from the construction of the bin. Hence, the first option from the bin that is realized also gets proposed by $\mathcal{A}$ and accepted by $\mathcal{P}$.

Now for each option $(i,j) \in B_{b^*}$, the probability that this option is proposed and accepted is the combination of two independent events: (1) no other option from B_{b^*} was realized in any of the rounds $i' < i$, (2) option ω_{ij} is realized in round i. The probability for event (2) is p_{ij}. For event (1), we define $m_i = \sum_{(i,j)\in B_{b^*}} p_{ij}$. With probability $\prod_{i'<i}(1 - m_{i'})$, no option from B_{b^*} is realized in rounds $i' < i$. Note that $\sum_{i=1}^n m_i \leq 1/2$. The term $\prod_{i=1}^n (1-m_i)$ is minimized for $m_1 = 1/2$ and $m_{i'} = 0$ for $1 < i' < i$. Thus $\prod_{i=1}^n (1 - m_i) \geq 1/2$, i.e., the probability of event (1) is at least 1/2.

Overall, by linearity of expectation, the expected utility of $\mathcal{P}$ when using φ is at least

$$\sum_{(i,j)\in B_{b^*}} \frac{1}{2} \cdot p_{ij} \cdot b_{ij} \geq \frac{1}{2\ell} \cdot \sum_{(i,j)\in Q} p_{ij} \cdot b_{ij} \geq \frac{1}{8\ell} \cdot \mathrm{OPT}\,.$$

□

Lemma 3. *Let ℓ be the number of bins opened by the algorithm. It holds that $\ell = O(\log\alpha/\log\log\alpha)$.*

Proof. Consider a bin B and its mass $p_B = \sum_{(i,j)\in B} p_{ij}$. We want to argue that at most $O(c/\log c)$ bins are opened. To do so, we first condition on having ℓ open bins and strive to lower bound the number of classes in these ℓ bins.

Consider a bin B starting at $\mathcal{C}_s$. The algorithm adds classes to B until $2^{k-1} < 2^s p_B$. Thus, $s - k + 1 > \log_2(1/p_B)$, i.e., the number of classes in B_i is lower bounded by $\log_2(1/p_B)$.

Now consider two bins B_i and B_{i+1} and condition on $q = p_{B_i} + p_{B_{i+1}}$. Together the bins contain at least $\log_2(1/p_{B_i}) + \log_2(1/(q - p_{B_i}))$ classes. Taking the derivative for p_{B_i}, we see that this lower bound is minimized when $p_{B_i} = q/2 = p_{B_{i+1}}$. Applying this balancing step repeatedly, the lower bound on the number of classes in all bins is minimized when $p_{B_i} = p_{B_j}$ for all bins B_i, B_j. Thus, when opening ℓ bins, we obtain the smallest lower bound on the number of classes in these bins by setting $p_{B_i} = 1/\ell \cdot \sum_{(i,j)\in Q} p_{ij} < 1/(2\ell)$ for all bins B_i. Conversely, when opening ℓ bins, we need to have at least $\ell \log_2(2\ell)$ classes in these bins.

Now, since we need to put c classes into the bins, we need to ensure that for the number ℓ of open bins we have $\ell(\log_2 \ell + 1) \leq c$, since otherwise the ℓ bins would require more than c classes in total. This implies that $c = \Omega(\ell \log_2 \ell)$ and, hence, $\ell = O(c/\log c) = O(\log\alpha/\log\log\alpha)$. □

Theorem 2. *If the agent has α-bounded utilities, there is a deterministic action scheme such that $\mathcal{P}$ obtains an $\Omega(\log\log\alpha/\log\alpha)$-approximation of the expected utility for optimal (online) search.*

Observe that the approximation ratio of this algorithm is tight in general. Consider the instances in Theorem 1 with $\alpha = n^{O(n)}$. The theorem shows that every scheme can obtain at most a ratio of $O(1/n) = O(\log\log\alpha/\log\alpha)$.

3.2 Oblivious Proposals

In the previous section, we considered algorithms for $\mathcal{P}$ when she learns the utility pair for the proposed option. In this section, we show that (fully) oblivious proposals can be a substantial drawback for $\mathcal{P}$. Obviously, the lower bound in Theorem 1 remains intact even for oblivious proposals, when $\mathcal{P}$ does not learn the utility value of the proposed option for $\mathcal{A}$. For oblivious proposals and α-bounded agent utilities, we can significantly strengthen the lower bound. In contrast to the logarithmic approximation guarantee above, we provide a linear lower bound in α for oblivious proposals.

Theorem 3. *There is a class of instances of online delegation with α-bounded utilities for the agent and oblivious proposals, in which every action scheme φ obtains at most an $O(1/\alpha)$-approximation of the expected utility for optimal (online) search.*

Proof. Consider the following class of instances. In $\mathcal{D}_i$, there are two options with the following probabilities and utilities: ω_{i1} with $p_{i1} = 1 - 1/n$ and $(b_{i1}, a_{i1}) = (0, 1)$, as well as ω_{i2} with $p_{i2} = 1/n$ and $(b_{i2}, a_{i2}) = (1, x_i)$, where $x_i \in \{1, \alpha\}$ and $\alpha \in [1, n]$. In the first rounds $i = 1, \ldots, i^* - 1$ we have $x_i = 1$, then $x_i = \alpha$ for rounds $i = i^*, \ldots, n$. The expected utility when $\mathcal{P}$ performs (undelegated) online search is $1 - (1 - 1/n)^n \geq 1 - 1/e$.

$\mathcal{P}$ wants that $\mathcal{A}$ proposes any profitable option ω_{i2} as soon as possible. As in the proof of Theorem 1, we can assume that all $\varphi_{i1} = 0$ in an optimal scheme – this option has no value for $\mathcal{P}$ and can only raise the incentive to wait for $\mathcal{A}$.

Due to oblivious proposals, $\mathcal{P}$ has to choose φ without being aware of the value of i^*. For our impossibility result, we adjust i^* to the scheme φ chosen by $\mathcal{P}$: Set $i^* \in \{1, \ldots, n\}$ to the largest number such that $\sum_{i=i^*}^{n} \varphi_{i2} \geq e \cdot n/\alpha$, or $i^* = 1$ if no such number exists.

First, suppose that $i^* = 1$. Then, even if we force $\mathcal{A}$ to propose *every* option ω_{i2} as soon as it arises, a union bound shows that the expected utility of $\mathcal{P}$ is upper bounded by $\sum_{i=1}^{n} \frac{1}{n} \cdot \varphi_{i2} \leq \frac{e}{\alpha} + \frac{1}{n}$. Hence, $\mathcal{P}$ obtains only an $O(1/\alpha)$-approximation, for any $\alpha \in [1, n]$.

Now suppose that $i^* > 1$. Consider an optimal scheme φ for $\mathcal{P}$. If ω_{i2} arises in round i, $\mathcal{A}$ decides if it is more profitable to propose i or wait for a later round. Indeed, we show that $\mathcal{A}$ never proposes ω_{i2} in any round $i < i^*$. Consider the expected utility from proposing the first option ω_{k2} arising in rounds $k = i^*, \ldots, n$. This is

$$\begin{aligned}
&\alpha \cdot \left(\sum_{k=i^*}^{n} \frac{1}{n} \left(1 - \frac{1}{n}\right)^{k-i^*} \varphi_{k2} \right) \\
&= \alpha \cdot \frac{1}{n} \cdot \sum_{k=i^*}^{n} \left(1 - \frac{1}{n}\right)^{k-i^*} \varphi_{k2} \\
&> \frac{\alpha}{n} \cdot \frac{1}{e} \cdot \sum_{k=i^*}^{n} \varphi_{k2} \geq \frac{\alpha}{en} \cdot \frac{en}{\alpha} = 1 \geq \varphi_{i2} \ ,
\end{aligned}$$

i.e., strictly larger than the expected utility φ_{i2} from proposing ω_{i2} in round $i < i^*$. Hence, $\mathcal{A}$ only proposes in rounds $k = i^*, \ldots, n$. Even if $\mathcal{A}$ would be able to propose *every* option ω_{k2} in rounds $k = i^*, \ldots, n$, a union bound implies that the expected utility of $\mathcal{P}$ from these rounds is upper bounded by $\sum_{k=i^*}^{n} \frac{1}{n} \cdot \varphi_{k2} \leq \frac{e}{\alpha} + \frac{1}{n}$. For any $\alpha \in [1, n]$, $\mathcal{P}$ obtains an $O(1/\alpha)$-approximation. □

Theorem 4. *If the agent has α-bounded utilities and makes oblivious proposals, there is a deterministic action scheme such that $\mathcal{P}$ obtains an $\Omega(1/\alpha)$-approximation of the expected utility for optimal (online) search.*

For the scheme, $\mathcal{P}$ simply sets $\varphi_{ij} = 1$ for all $(i, j) \in Q$, where Q is the set returned by RestrictOptions$(\mathcal{D}_1, \ldots, \mathcal{D}_n, 1/(2\alpha))$. The key insight is that by bounding the combined mass of acceptable options by $1/(2\alpha)$, $\mathcal{A}$ is incentivized to propose the first acceptable option he encounters. Pseudocode as well as the proof of Theorem 4 can be found in the full version [Braun *et al.*, 2022].

4 Misalignment of Principal and Agent Utility

In this section, we consider performance guarantees based on the amount of misalignment of principal and agent utility. For most of the section, we assume that all utility values are strictly positive. Consider the smallest $\beta \geq 1$ such that

$$\frac{1}{\beta} \cdot \frac{a_{ij}}{a_{i'j'}} \leq \frac{b_{ij}}{b_{i'j'}} \leq \beta \cdot \frac{a_{ij}}{a_{i'j'}}$$

for any two options ω_{ij} and $\omega_{i'j'}$ in the instance. Then the preference of $\mathcal{P}$ between any pair $\omega_{ij}, \omega_{i'j'}$ of options is shared by $\mathcal{A}$ – up to a factor of at most β. We term this *β-bounded utilities*. Alternatively, consider $\gamma \geq 1$ as a bound on the utility ratio $1/\gamma \cdot a_{ij} \leq b_{ij} \leq \gamma \cdot a_{ij}$ for every single ω_{ij}. Then the utilities are γ^2-bounded in the sense defined above, and we obtain asymptotically the same bounds.

Suppose we choose an arbitrary realization $\omega_{i'j'}$. Divide all utility values of $\mathcal{P}$ for all realizations by $b_{i'j'}$, and all utility values of $\mathcal{A}$ by $a_{i'j'}$. Note that this adjustment neither affects the incentives of the players nor the approximation ratios of our algorithms. Considering ω_{ij} with the adjusted utilities, we see that $1/\beta \cdot b_{ij}/a_{ij} \leq 1 \leq \beta \cdot b_{ij}/a_{ij}$, and thus $1/\beta \leq b_{ij}/a_{ij} \leq \beta$ for all $\omega_{i'j'}$. This condition turns out to be convenient for our analysis.

Our main idea is to use $O(\log \beta)$ clusters $\mathcal{C}_k$ to group all the options that have a utility ratio between 2^k and 2^{k+1}, i.e.,

$$\mathcal{C}_k = \{\omega_{ij} \in \Omega \mid 2^k \leq b_{ij}/a_{ij} < 2^{k+1}\}$$

for $k = \lfloor \log 1/\beta \rfloor, \ldots, \lceil \log \beta \rceil$. Our deterministic scheme restricts the acceptable options to a single cluster $\mathcal{C}_{k^*}$. Note that here $\mathcal{P}$ is assumed to see a_{ij} upon a proposal. The principal determines the cluster k^*, such that the best response by $\mathcal{A}$ (i.e., his optimal online algorithm applied with the options from that cluster) delivers the largest expected utility for $\mathcal{P}$.

Theorem 5. *If principal and agent have β-bounded utilities, there is a deterministic action scheme such that $\mathcal{P}$ obtains an $\Omega(1/\log \beta)$-approximation of the expected utility for optimal (online) search.*

Proof. Consider any cluster $\mathcal{C}_k$. We denote by $b(\mathcal{A}, k)$ and $a(\mathcal{A}, k)$ the expected utility for $\mathcal{P}$ and $\mathcal{A}$ when $\mathcal{P}$ uses $\mathcal{C}_k$ to determine φ. Now consider a hypothetical algorithm for $\mathcal{P}$ that observes all realizations and chooses the best option from $\mathcal{C}_k$ for $\mathcal{P}$ if possible. If there is no such option, it obtains a utility of 0. Let $b(\mathcal{P}, k)$ and $a(\mathcal{P}, k)$ be the expected utility of the hypothetical algorithm for $\mathcal{P}$ and $\mathcal{A}$, respectively. Clearly, $b(\mathcal{P}, k) \geq b(\mathcal{A}, k)$ and $a(\mathcal{A}, k) \geq a(\mathcal{P}, k)$, but also, by definition of $\mathcal{C}_k$,

$$b(\mathcal{A}, k) \geq a(\mathcal{A}, k) \cdot 2^k \geq a(\mathcal{P}, k) \cdot 2^k \geq b(\mathcal{P}, k)/2 \ .$$

Now consider the best option for $\mathcal{P}$ in hindsight. The best-option-algorithm for cluster $\mathcal{C}_k$ picks the best option in hindsight if it comes from cluster $\mathcal{C}_k$. Otherwise, it returns a value of 0. Let b_k^* be the expected utility of this algorithm for $\mathcal{P}$, and let OPT be the expected utility of the best option in hindsight for $\mathcal{P}$. Then

$$\begin{aligned}
OPT &= \sum_{k=\lfloor \log 1/\beta \rfloor}^{\lceil \log \beta \rceil} b_k^* \leq \sum_{k=\lfloor \log 1/\beta \rfloor}^{\lceil \log \beta \rceil} b(\mathcal{P}, k) \\
&\leq \sum_{k=\lfloor \log 1/\beta \rfloor}^{\lceil \log \beta \rceil} b(\mathcal{A}, k) \cdot 2 \ .
\end{aligned}$$

Hence, since the scheme chooses the cluster k^* that maximizes $b(\mathcal{A}, k^*)$, we obtain an $\Omega(1/\log \beta)$-approximation. □

Acknowledgments

Hahn gratefully acknowledges the support of GIF grant I-1419-118.4/2017. Hoefer gratefully acknowledges the support of GIF grant I-1419-118.4/2017 and DFG grants Ho 3831/6-1, 7-1 and 9-1.

References

[Alonso and Matouschek, 2008] Ricardo Alonso and Niko Matouschek. Optimal delegation. *Rev. Econ. Stud.*, 75(1):259–293, 2008.

[Armstrong and Vickers, 2010] Mark Armstrong and John Vickers. A model of delegated project choice. *Econometrica*, 78(1):213–244, 2010.

[Babaioff and Winter, 2014] Moshe Babaioff and Eyal Winter. Contract complexity. In *Proc. 15th Conf. Econ. Comput. (EC)*, page 911, 2014.

[Babaioff *et al.*, 2012] Moshe Babaioff, Michal Feldman, Noam Nisan, and Eyal Winter. Combinatorial agency. *J. Econ. Theory*, 147(3):999–1034, 2012.

[Babichenko and Barman, 2017] Yakov Babichenko and Siddharth Barman. Algorithmic aspects of private Bayesian persuasion. In *Proc. 8th Symp. Innov. Theoret. Comput. Sci. (ITCS)*, pages 34:1–34:16, 2017.

[Badanidiyuru *et al.*, 2018] Ashwinkumar Badanidiyuru, Kshipra Bhawalkar, and Haifeng Xu. Targeting and signaling in ad auctions. In *Proc. 29th Symp. Discret. Algorithms (SODA)*, pages 2545–2563, 2018.

[Bechtel and Dughmi, 2021] Curtis Bechtel and Shaddin Dughmi. Delegated stochastic probing. In James R. Lee, editor, *Proc. 12th Symp. Innov. Theoret. Comput. Sci. (ITCS)*, pages 37:1–37:19, 2021.

[Bechtel *et al.*, 2022] Curtis Bechtel, Shaddin Dughmi, and Neel Patel. Delegated Pandora's box. In *Proc. 23rd Conf. Econ. Comput. (EC)*, 2022.

[Braun *et al.*, 2022] Pirmin Braun, Niklas Hahn, Martin Hoefer, and Conrad Schecker. Delegated online search. https://arxiv.org/abs/2203.01084, 2022.

[Castiglioni and Gatti, 2021] Matteo Castiglioni and Nicola Gatti. Persuading voters in district-based elections. In *Proc. 35th Conf. Artif. Intell. (AAAI)*, pages 5244–5251, 2021.

[Castiglioni *et al.*, 2020a] Matteo Castiglioni, Andrea Celli, and Nicola Gatti. Persuading voters: It's easy to whisper, it's hard to speak loud. In *Proc. 34th Conf. Artif. Intell. (AAAI)*, pages 1870–1877, 2020.

[Castiglioni *et al.*, 2020b] Matteo Castiglioni, Andrea Celli, Alberto Marchesi, and Nicola Gatti. Online bayesian persuasion. In *Proc. 34th Conf. Adv. Neural Inf. Processing Syst. (NIPS)*, 2020.

[Castiglioni *et al.*, 2021a] Matteo Castiglioni, Andrea Celli, Alberto Marchesi, and Nicola Gatti. Signaling in bayesian network congestion games: the subtle power of symmetry. In *Proc. 35th Conf. Artif. Intell. (AAAI)*, pages 5252–5259, 2021.

[Castiglioni *et al.*, 2021b] Matteo Castiglioni, Alberto Marchesi, Andrea Celli, and Nicola Gatti. Multi-receiver online bayesian persuasion. In *Proc. 38th Int. Conf. Machine Learning (ICML)*, volume 139 of *Proceedings of Machine Learning Research*, pages 1314–1323, 2021.

[Castiglioni *et al.*, 2022] Matteo Castiglioni, Giulia Romano, Alberto Marchesi, and Nicola Gatti. Signaling in posted price auctions. In *Proc. 36th Conf. Artif. Intell. (AAAI)*, pages 4941–4948, 2022.

[Dughmi and Xu, 2016] Shaddin Dughmi and Haifeng Xu. Algorithmic Bayesian persuasion. In *Proc. 48th Symp. Theory Comput. (STOC)*, pages 412–425, 2016.

[Dughmi and Xu, 2017] Shaddin Dughmi and Haifeng Xu. Algorithmic persuasion with no externalities. In *Proc. 18th Conf. Econ. Comput. (EC)*, pages 351–368, 2017.

[Dughmi *et al.*, 2016] Shaddin Dughmi, David Kempe, and Ruixin Qiang. Persuasion with limited communication. In *Proc. 17th Conf. Econ. Comput. (EC)*, pages 663–680, 2016.

[Dütting *et al.*, 2019] Paul Dütting, Tim Roughgarden, and Inbal Talgam-Cohen. Simple versus optimal contracts. In *Proc. 20th Conf. Econ. Comput. (EC)*, page 369–387, 2019.

[Dütting *et al.*, 2020] Paul Dütting, Tim Roughgarden, and Inbal Talgam-Cohen. The complexity of contracts. In *Proc. 31st Symp. Discret. Algorithms (SODA)*, pages 2688–2707, 2020.

[Gradwohl *et al.*, 2021] Ronen Gradwohl, Niklas Hahn, Martin Hoefer, and Rann Smorodinsky. Algorithms for persuasion with limited communication. In *Proc. 32nd Symp. Discret. Algorithms (SODA)*, pages 637–652, 2021.

[Griesbach *et al.*, 2022] Svenja Griesbach, Martin Hoefer, Max Klimm, and Tim Koglin. Public signals in network congestion games. In *Proc. 23rd Conf. Econ. Comput. (EC)*, page 736, 2022.

[Grossman and Hart, 1983] Sanford J. Grossman and Oliver D. Hart. An analysis of the principal-agent problem. *Econometrica*, 51(1):7–45, 1983.

[Hahn *et al.*, 2020a] Niklas Hahn, Martin Hoefer, and Rann Smorodinsky. Prophet inequalities for Bayesian persuasion. In *Proc. 29th Int. Joint Conf. Artif. Intell. (IJCAI)*, pages 175–181, 2020.

[Hahn *et al.*, 2020b] Niklas Hahn, Martin Hoefer, and Rann Smorodinsky. The secretary recommendation problem. In *Proc. 21st Conf. Econ. Comput. (EC)*, page 189, 2020.

[Holmstrom, 1977] Bengt Robert Holmstrom. *On Incentives and Control in Organizations*. PhD thesis, Stanford University, 1977.

[Holmstrom, 1984] Bengt Robert Holmstrom. On the theory of delegation. In Marcel Boyer and Richard Kihlstrom, editors, *Bayesian Models in Economic Theory*, pages 115–141. Elsevier, 1984.

[Kleinberg and Kleinberg, 2018] Jon Kleinberg and Robert Kleinberg. Delegated search approximates efficient search. In *Proc. 19th Conf. Econ. Comput. (EC)*, pages 287–302, 2018.

[Krengel and Sucheston, 1977] Ulrich Krengel and Louis Sucheston. Semiamarts and finite values. *Bull. Amer. Math. Soc*, 83:745–747, 1977.

[Krengel and Sucheston, 1978] Ulrich Krengel and Louis Sucheston. On semiamarts, amarts and processes with finite value. *Adv. Prob.*, 4:197–266, 1978.

[Melumad and Shibano, 1991] Nahum D. Melumad and Toshiyuki Shibano. Communication in settings with no transfers. *RAND J. Econ.*, 22(2):173–198, 1991.

[Rubinstein, 2017] Aviad Rubinstein. Honest Signaling in Zero-Sum Games Is Hard, and Lying Is Even Harder. In *Proc. 44th Int. Colloq. Autom. Lang. Programming (ICALP)*, pages 77:1–77:13, 2017.

[Xu, 2020] Haifeng Xu. On the tractability of public persuasion with no externalities. In *Proc. 31st Symp. Discret. Algorithms (SODA)*, pages 2708–2727, 2020.

[Zu *et al.*, 2021] You Zu, Krishnamurthy Iyer, and Haifeng Xu. Learning to persuade on the fly: Robustness against ignorance. In *Proc. 22nd Conf. Econ. Comput. (EC)*, pages 927–928, 2021.