Beating Competitive Ratio 4 for Graphic Matroid Secretary

Kiarash Banihashem
University of Maryland, College Park, MD, USA

MohammadTaghi Hajiaghayi
University of Maryland, College Park, MD, USA

Dariusz R. Kowalski
Augusta University, GA, USA

Piotr Krysta
Augusta University, GA, USA
University of Liverpool, Liverpool, UK

Danny Mittal
University of Maryland, College Park, MD, USA

Jan Olkowski
University of Maryland, College Park, MD, USA

Abstract

One of the classic problems in online decision-making is the *secretary problem*, where the goal is to hire the best secretary out of n rankable applicants or, in a natural extension, to maximize the probability of selecting the largest number from a sequence arriving in random order. Many works have considered generalizations of this problem where one can accept multiple values subject to a combinatorial constraint. The seminal work of Babaioff, Immorlica, Kempe, and Kleinberg (SODA'07, JACM'18) proposed the *matroid secretary conjecture*, suggesting that there exists an $O(1)$-competitive algorithm for the matroid constraint, and many works since have attempted to obtain algorithms for both general matroids and specific classes of matroids. The ultimate goal of these results is to obtain an e-competitive algorithm, and the *strong matroid secretary conjecture* states that this is possible for general matroids.

One of the most important classes of matroids is the *graphic matroid*, where a set of edges in a graph is deemed independent if it contains no cycle. Given the rich combinatorial structure of graphs, obtaining algorithms for these matroids is often seen as a good first step towards solving the problem for general matroids. For matroid secretary, Babaioff et al. (SODA'07, JACM'18) first studied graphic matroid case and obtained a 16-competitive algorithm. Subsequent works have improved the competitive ratio, most recently to 4 by Soto, Turkieltaub, and Verdugo (SODA'18).

In this paper, we break the 4-competitive barrier for the problem, obtaining a new algorithm with a competitive ratio of 3.95. For the special case of simple graphs (i.e., graphs that do not contain parallel edges) we further improve this to 3.77. Intuitively, solving the problem for simple graphs is easier as they do not contain cycles of length two. A natural question that arises is whether we can obtain a ratio arbitrarily close to e by assuming the graph has a large enough girth.

We answer this question affirmatively, proving that one can obtain a competitive ratio arbitrarily close to e even for constant values of girth, providing further evidence for the strong matroid secretary conjecture. We further show that this bound is tight: for any constant g, one cannot obtain a competitive ratio better than e even if we assume that the input graph has girth at least g. To our knowledge, such a bound was not previously known even for simple graphs.

2012 ACM Subject Classification Theory of computation → Online algorithms

Keywords and phrases online algorithms, graphic matroids, secretary problem

Digital Object Identifier 10.4230/LIPIcs.ESA.2025.52

Related Version *Full Version*: https://arxiv.org/abs/2501.08846

33rd Annual European Symposium on Algorithms (ESA 2025).
Editors: Anne Benoit, Haim Kaplan, Sebastian Wild, and Grzegorz Herman; Article No. 52; pp. 52:1–52:16
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Funding The work is partially supported by DARPA QuICC, ONR MURI 2024 award on Algorithms, Learning, and Game Theory, Army-Research Laboratory (ARL) grant W911NF2410052, NSF AF:Small grants 2218678, 2114269, 2347322, and Royal Society grant IES\R2\222170.

1 Introduction

In the past two decades there has been a renewed interest in online item selection problems where a sequence of items arrive one by one, revealing their *weight*, and a decision maker needs to irrevocably decide whether or not to accept each item as it arrives. The goal is to maximize the total accepted weight, subject to a feasibility constraint on the chosen items. An algorithm's performance is typically measured by its *competitive ratio*, which compares the algorithm's total weight with the offline optimum – the total weight achievable if all item weights were known in advance. These problems are appealing both from a mathematical perspective, as they are concise models for online decision making, and from an economical perspective, as they have close connection to pricing and auction theory [30, 29, 8, 12].

In the absence of any information about future items, the problem is essentially hopeless as any single item could be significantly heavier than all others, and the decision maker has no way of deciding which it is. As such, most works make distributional assumptions on either the weight of the arriving items, or their arrival order. The former class of problems are generally referred to as *prophet inequalities* while the latter are known as *secretary problems*. Numerous works have studied secretary problems for a large class of combinatorial constraints [38, 19, 34, 33, 20, 44, 46, 23, 31] and objective functions [10, 25], and considered close variants of these problems such as the prophet-secretary problem [22].

Perhaps the most important open question in the area of online decision making is the *matroid secretary problem* posed by the seminal work of Babaioff, Immorlica, Kempe, and Kleinberg (SODA'07,JACM'18) [8, 7]. In the matroid secretary problem, items arrive in a random order, and each item corresponds to an element in a matroid $M = (E, I)$,where E is the set of elements and I is the collection of independent sets in matroid M. Upon arrival, the weight of each item is revealed, and the decision maker must immediately decide whether to accept or reject it. The goal is to maximize the total weight of accepted items, with the constraint that the selected items form an independent set in the matroid. The matroid secretary conjecture [8, 7] states that there exists a constant-competitive ratio algorithm for this problem, and the *strong matroid secretary conjecture* (e.g., see [46]) states that there exists an e-competitive algorithm. Many works have studied the problem for both general matroids and specific cases.

In this paper, we focus on the specific case of graphic matroids. In this case, the arriving items correspond to the edges of a graph and the goal is to accept a set of edges that do not contain a cycle. Solving problems for graphic matroids is often viewed as a promising first step toward addressing arbitrary matroids, as graphs possess rich combinatorial structures and graphic matroids, along with linear matroids, are among the most intuitive examples of non-trivial matroids. Many counterexamples for candidate matroid secretary algorithms are, in fact, graphic matroids [8, 9]. It is common to illustrate the main ideas behind general algorithms by showing their behavior on this specific case [47]. Recent work has also explored whether techniques for graphic matroids can be extended to general matroids [1].

The seminal paper of Babaioff et al. originally studied the graphic matroid secretary problem, obtaining a 16-competitive algorithm for the problem. Babaioff et. al. [6] designed a $3e$-competitive algorithm. The competitive ratio was later improved to $2e$ by Korula and Pal [38] and later to 4 by Soto, Turkieltaub, and Verdugo [46]. Whether or not the ratio can be improved has been open at least since 2018.[1]

1.1 Our results and techniques

In this paper we obtain an algorithm with competitive ratio 3.95 for the problem, breaking the 4-competitive barrier. We further improve this result for the specific case of *simple graphs*, i.e., graphs that do not have parallel edges. Intuitively, simple graphs represent an easier special case as they don't have any cycles of length 2; since the algorithm is forbidden from accepting edges that form a cycle, the lack of 2-cycles gives the algorithm more freedom to accept edges. Our main result is the following theorem.

▶ **Theorem 1.** *There exists a* 3.95*-competitive algorithm for the graphic matroid secretary problem. Furthermore, if the input graph is assumed to be simple, there exists an algorithm with competitive ratio* 3.77*.*

The basic approach for this result is to, at each step, compute a set of outgoing edges such that each node has exactly one outgoing edge. More exactly, at each step we compute a maximum spanning forest, then direct the edges in this forest toward an arbitrary root; each node then has a unique outgoing edge. We then argue that if the algorithm only accepts an edge for whose endpoints it has not previously accepted an outgoing edge, then the algorithm's set of taken edges will always be independent.

Given this, the key aspect of Algorithm 1 that allows it to obtain an improved competitive ratio is a slightly stronger condition used to determine whether an edge is taken. Specifically, while for one endpoint we only demand that we have not taken an outgoing edge, for the other endpoint we demand that we have not even seen an outgoing edge. We can then show an increased probability for the former condition being satisfied for a currently considered edge by using the fact that the latter condition may not be satisfied by a previously seen outgoing edge, causing said edge to not be taken. Algorithm 1 furthermore makes use of random choice in determining which endpoint to apply the stronger condition to, which is crucial for handling the case of duplicate edges. When the input graph is guaranteed to be simple, this random choice is unnecessary – removing this random choice gives our algorithm for simple graphs (see the full version of the paper) which allows us to obtain an even lower competitive ratio.

Motivated by the improvement for simple graphs, we additionally study, for the first time, the landscape of the graphic matroid secretary problem for graphs of high *girth*, where we recall that the girth of a graph is the length of its shortest cycle. We note that the fact that a graph is simple follows from the assumption that girth is at least 3. We show that when the graph has large (but constant) girth, one can obtain a competitive ratio arbitrarily close to e. Formally, we prove the following theorem.

▶ **Theorem 2.** *For any graph G with girth at least g, there exists an $(e + o_g(1))$-competitive algorithm.*

[1] This paper was previously submitted to STOC'25 on November 4, 2024. Independently of our work, on November 18, 2024, Bérczi et al. [11] posted a preprint on arXiv, obtaining a 3.99-competitive algorithm and a 3.71-competitive algorithm for general graphs and simple graphs, respectively.

The result builds on yet another property of the combinatorial structure of the set of outgoing edges introduced in the proof of Theorem 1. We first observe that the fact that every vertex has at most one outgoing edge implies that, in a graph induced by the outgoing edges, each edge belongs to at most one cycle. On the other hand, we prove that if Algorithm 1 were to accept outgoing edges without respecting the graphic matroid independence condition, it would yield a higher acceptance probability of $\frac{1}{e}$ for an edge from the maximum independent set. These observations lead to a natural approach: accepting an outgoing edge only with a certain probability such that, for every cycle of length g, the probabilities of taking each edge of this cycle are equal. Intuitively, as the length of the shortest cycle increases, this probability tends towards $\frac{1}{e}$. We refer the reader to the full version of the paper for more details.

The best competitive ratios we attain over the three algorithms we introduce are listed in Table 1 for girths less than 10.

Table 1 The best competitive ratio we obtain for graphic matroid secretary when the input graph is restricted to have girth at least g for $g < 10$. $g = 2$ corresponds to multigraphs, while $g = 3$ corresponds to simple graphs. As g approaches infinity, the competitive ratio approaches e. See the full version of the paper for the algorithms for simple graphs and high girth graphs.

Girth (g)	Competitive Ratio	Algorithm
2	3.95	Algorithm 1
3	3.77	Simple graph algorithm
4	3.77	Simple graph algorithm
5	3.76	High girth algorithm
6	3.61	High girth algorithm
7	3.50	High girth algorithm
8	3.42	High girth algorithm
9	3.35	High girth algorithm

Perhaps more surprisingly, we show that this is tight: no algorithm can achieve a competitive ratio better than e, even for graphs with high girth. To our knowledge, this lower bound was not previously known even for simple graphs, and our techniques may be of independent interest for related online arrival problems.

▶ **Theorem 3.** *For any $g \in \mathbb{N}$, there does not exist an algorithm for the graphic matroid secretary problem on graphs of girth at least g that obtains competitive ratio less than e.*

The two results fully characterize the landscape of the secretary problem in the high-girth setting, essentially showing that the problem becomes as easy as the single secretary problem in the limit. To prove the lower bound in Theorem 3, we first show that weighted single secretary, also known as a cardinal secretary, is hard "on average" in the following sense. We demonstrate the existence of a finite distribution over the instances of this problem such that even if the adversary samples instances, as opposed to choosing them in a worst-case manner, from that distribution, it is still hard for any algorithm to choose the maximum weight element.

We then construct a high girth bipartite graph based on Ramanujan graphs, where vertices of one bipartition have same degree d. The hard distribution of instances of weighted single secretary of size d can now be sampled independently and embedded on the edges incident to each degree-d vertex on one side of the constructed bipartite graph. Given an algorithm that performs well on this graph, we obtain an algorithm performing well on the original distribution of weighted single secretary by simulating the first algorithm and mimicking the choices that it makes on a single degree-d vertex.

Our construction of the input graph in addition to high-girth Ramanujan graphs uses the probabilistic method. Our construction of the sets of hard weights is based on an extension of a similar argument from an infinite to a finite version of Ramsey theorem. We employ zero-sum games duality argument to show an existence of the final probabilistic distribution over the instances.

2 Related work

Many other derivations and specific cases of the general matroid secretary problem have been given attention over the years. Hajiaghayi, Kleinberg and Parkes [30] had first introduced the *multiple-choice value version* of the problem, aka the uniform matroid secretary problem, in which the goal is to maximize the expected sum of the at most k selected numbers. Kleinberg [36] later presented a tight $(1-O(\sqrt{1/k}))^{-1}$-competitive algorithm for the k uniform secretary resolving an open problem of [30]. Transversal matroids were first considered in [8] who gave $4d$-competitive algorithm, where d is the degree of the transversal matroid. This was improved by Dimitrov and Plaxton [19] who showed a ratio of 16 for all transversal matroids. The optimal ratio of e follows from the e-competitive algorithm for secretarial online weighted bipartite matching problem [35], which generalizes the problem on transversal matroids. For the laminar matroids, a long line of work led to $3\sqrt{3}e$ competitive ratio [32, 33, 41], which was improved to 9.6 in [41], $3\sqrt{3}$ in [46], 4.75 in [31], and 3.26 in [11]. The challenging general class of regular matroids was proven to admit $9e$-competitive algorithm [20].

Other generalizations of the secretary problem such as the submodular variant have been initially studied by the Bateni, Hajiaghayi, and ZadiMoghaddam [10] and Gupta, Roth, Schoenebeck, and Talwar [28]. The connection between the secretary problem and online auction mechanisms has been explored by Kesselheim et al. [34], who give a e-competitive solution to the online bipartite weighted matching problem.

The *prophet secretary* problem is another well-studied variant of the secretary problem, closely related to prophet inequalities. In the *prophet inequality* setting, introduced by Krengel and Sucheston [39, 40], we know the distributions of n arriving items and aim to maximize the ratio of the expectation of the selected item's value to the expectation of the sequence maximum, with a tight competitive ratio of 2. Research connecting prophet inequalities and online auctions, initiated by Hajiaghayi, Kleinberg, and Sandholm [29], led to follow-up studies such as Alaei, Hajiaghayi, and Liaghat's work [3] on the bipartite matching variant of prophet inequality, also achieving a competitive ratio of 2 [3]. Feldman et al. [24] expanded the problem to combinatorial auctions with multiple buyers, achieving the same bound through a posted pricing scheme, and Kleinberg and Weinberg [37] extended this result to matroids with a 2-competitive algorithm. A recent paper explores matroid structures that allow breaking the tight competitive ratio of 2 [4]. Contrary to results presented in this paper, they prove that in the general matroid setting, a high girth is not a sufficient condition while considering k-fold matroid unions suffices.

The prophet secretary model, introduced by Esfandiari, Hajiaghayi, Liaghat, and Monemizadeh [22], assumes a random arrival order and known item distributions. They designed an algorithm achieving a competitive factor of $(\frac{e}{e-1})$, which has proven challenging to improve. However, Azar et al. [5] and Correa et al. [16, 17] improved this bound to $(1-1/e+1/30)^{-1} \approx 1.502$. For the single-item i.i.d. case, Abolhasani et al. [2] achieved a 1.37-competitive ratio, later improved to 1.342 by Correa et al. [14]. Recently, Peng and Tang [43] achieved a 1.379-competitive algorithm for the *free order* case, however finding the tight competitive bound for the general prophet secretary problem remains open.

Last but not least, extensions beyond matroids for the secretary problem have also been studied. For example, Kleinberg and Weinberg [37] provided an $O(p)$-competitive algorithm for the intersection of p matroids, later generalized to polymatroids by Dütting and Kleinberg [21]. Rubinstein [44] and Rubinstein and Singla [45] considered prophet inequalities and secretary problems in arbitrary downward-closed set systems. For such settings, Babaioff et al. [8] proved a lower bound of $\Omega(\log n \log\log n)$, and further studies have explored combinatorial optimization applications [18, 26, 27, 42].

3 Preliminaries

Graph Notation

In this paper, we assume all graphs are undirected, weighted, and may contain multiple edges between the same pair of vertices. Specifically, a graph is defined as a triple $G = (V, E, w : E \to \mathbb{R})$, where V, with $|V| = n$, is the set of vertices; E, with $|E| = m$, is the multiset of edges; and w assigns weights to the edges. We assume graphs do not contain loops (i.e., edges connecting a vertex to itself[2]). Given any vertex $u \in V$, we denote the degree of u by $deg_G(u)$. A graph is called *simple* if there is at most one edge connecting any pair of vertices. For a graph G, we denote g as the *girth* of G, representing the length of its shortest cycle. In graphs with multiple edges, a cycle is defined as any multiset of edges $C = \{(a_1, b_1), \ldots, (a_k, b_k)\}$, for $k \geq 2$, such that $\forall_{1 \leq i \leq k-1}\, b_i = a_{i+1}$ and $b_k = a_1$. For an integer $g \geq 2$, let $\mathcal{G}_g$ denote the set of all graphs with girth at least g. We also denote $[n] = \{1, 2, \ldots, n\}$.

Matroids

A *matroid* $M = (E, I)$ is a combinatorial structure that generalizes independence. It consists of a finite set E and a collection I of independent subsets of E satisfying: (1) the empty set is independent, (2) any subset of an independent set is also independent, and (3) if one independent set is larger than another, an element from the larger set can extend the smaller one while preserving independence. These properties capture the concept of independence, making matroids useful for modeling optimization problems where we seek a maximum-weight or maximum-cardinality independent subset of elements.

A *graphic matroid* $\mathcal{M}(G)$ associated with a graph G has as its elements the edges in the multiset E, and defines independent sets as acyclic subsets of edges (i.e., subgraphs without cycles). The weight of an independent set is the sum of its edge weights.

Problems

Consider a graphic matroid $\mathcal{M}(G)$ associated with a *multigraph* $G = (V, E, w : E \to \mathbb{R})$. In the online secretary problem on graphic matroids, the elements of E are presented to the algorithm in a random order, chosen uniformly from all possible permutations of the multiset E. Elements arrive one at a time, effectively creating m time steps during the algorithm's execution. Upon the arrival of an element, the algorithm must decide whether to accept or reject it, with the constraint that an element can only be accepted if it forms an independent set with the already accepted elements. Decisions are irrevocable.

[2] As explained in the following paragraph, such edges are irrelevant in the context of graphic matroids.

The objective is to design an algorithm that maximizes the expected sum of the weights of the accepted elements, referred to as the algorithm's gain. For an algorithm ALG, we denote its (random) gain by ALG and its expected gain by $\mathbb{E}(ALG)$.

Let OPT denote an independent set with maximum weight in the matroid $\mathcal{M}(G)$. We say an algorithm ALG is *α-competitive* (or has an α competitive ratio, for some $\alpha \geq 1$) for a family of matroids if $\alpha \cdot \mathbb{E}(ALG) \geq W(OPT)$ for all matroids in that family, where $W(OPT)$ is the weight of OPT.

4 Improved bounds for graphic matroid secretary: proof of Theorem 1

In this section, we prove the following theorem, which contains the result from Theorem 1 for general graphs.

▶ **Theorem 4.** *There exists a* 3.95*-competitive algorithm for graphic matroid secretary.*

To prove the theorem, we present an algorithm that attains a competitive ratio less than 3.95 for the graphic matroid secretary problem, surpassing a result of Soto, Turkieltaub, and Verdugo [46] attaining a competitive ratio of 4 that previously stood for six years as the best result achieved for graphic matroid secretary; this algorithm is described in Section 4.1.

To prove Theorem 1 for simple graphs, we consider a slight modification of our algorithm. See the full version of the paper for details.

4.1 Improvement for general graphs

Algorithm 1 New algorithm for graphic matroid secretary.

```
Let $E'$ be the first $m'$ edges                          // set of observed edges
$A \leftarrow \emptyset$                                  // set of accepted edges
$\forall v \in V$: $seen\_outgoing(v) \leftarrow False$
$\forall v \in V$: $taken\_outgoing(v) \leftarrow False$
for $t \in \{ m'+1, \dots, m \}$ do
    Let $e_t$ be the edge arriving in time $t$
    Add $e_t$ to $E'$
    $T_t^{\text{opt}} \leftarrow$ maximum weighted directed forest on $G[E']$
    $\forall v \in V : outgoing(v) \leftarrow$ the edge directed away from $v$ in $T_t^{\text{opt}}$ if it exists
    $\forall v \in V$ such that $outgoing(v) = null : outgoing(v) \leftarrow$ a unique edge in $E'$ not in
     $T_t^{\text{opt}}$
    if $e_t \in T_t^{\text{opt}}$ then
        $e_t \leftarrow (u, v)$                  // where $e_t$ is directed from $u$ to $v$ in $T_t^{\text{opt}}$
        $(a, b) \leftarrow$ either $(u, v)$ or $(v, u)$ with equal probability
        if $seen\_outgoing(a) = False$ and $taken\_outgoing(b) = False$ then
            $taken\_outgoing(u) \leftarrow True$
            add $e_t$ to $A$
        end
    end
    if $outgoing(w) = e_t$ for some $w \in V$ then
        $seen\_outgoing(w) \leftarrow True$
    end
end
```

The pseudocode of the algorithm is provided in Figure 1. The algorithm consists of two phases. During the first m' steps (where m' depends only on m), edges are only observed without being taken. In later steps, let e be the presented edge. The algorithm computes a maximum spanning forest T_t^{opt} of all edges seen so far. We assume the forest is rooted, where the root for each tree is chosen lexicographically for simplicity; how we choose the root does not affect the algorithm as long as it depends only on the set of the edges already seen and not their order. This forest is directed towards the root, so that every edge is directed and each node has at most one outgoing edge. We additionally compute an array of *outgoing* edges: for each node v, if v has an outgoing edge in T_t^{opt}, then that is its outgoing edge; if it does not, then we choose an arbitrary edge not already assigned to be the outgoing edge of another vertex [3] and declare that to be the outgoing edge from v. Note that said edge does not even have to have v as an endpoint. Also note that crucially, the choice of edges declared to be outgoing for the vertices that need it depends on the set of edges observed so far but not their order.

We only consider e if it is in T_t^{opt}. In this case, it points from some u to some v, and so is the outgoing edge for u. In order to take e, we randomly order u, v as a, b and impose the following constraints on e:

- No edge e' has been taken such that e' was the outgoing edge for b when e' was presented.
- No edge e' has appeared such that e' was the outgoing edge for a when e' was presented.

If e satisfies both constraints then it is taken. Otherwise, the algorithm still notes that e could have been selected for u.

The distinct constraints used for a, b are key – the stronger condition applied to endpoint a allows us to lower bound the probability that an edge is blocked from being taken, which can then be used to show an increased probability that the weaker condition for endpoint b is satisfied for an edge that we would like to be taken. The randomness in ordering u, v into a, b is also essential. To see why, consider the specific example of an edge $e_t \in OPT$ with endpoints u, v that is presented at step t, and consider some step $j < t$. Suppose that T_j^{opt} contains an edge e^u outgoing from u to a third node x and an edge e^v outgoing from v to a fourth node y. We would like to argue that in the case that either e^u or e^v is the edge e_j presented at step j, there is a possibility that some outgoing edge from x or y respectively had already been presented prior to step j, meaning that e^u or e^v could not have been taken.

This means that, for example, if e^u were the edge presented at step j, we would desire that, in the case of edge e_t, $b = u$, and in the case of edge $e_j = e^u$, $a = x$ (and so $b = u$), so that e^u could be blocked by an edge outgoing from x simply being presented, while e_t is then not blocked because e^u was not actually taken. This suggests that we could simply always take $b = u$ and $a = v$, applying the stronger condition to the endpoint which the edge is directed towards.

However, as in general graphs it is possible for multiple edge to have the same endpoints, it is possible that we in fact have $x = v$, in which case an edge outgoing from x would block e_t itself. This is illustrated in the top-left quadrant of Figure 1. Therefore, in order to guarantee an improvement, we crucially allow for the possibility that in the case of edge e_t, $b = v$, and in the case of edge $e_j = e^v$, $a = y$ – if e^u has the same endpoints as e_t, then e^v cannot as T_j^{opt} is a spanning tree, and so y will necessarily be distinct from u, meaning that there is a possibility for e^v to be blocked by an edge outgoing from y while still allowing e_t to be taken. This is illustrated in the bottom-left quadrant of Figure 1.

[3] Note that if m is too small (specifically, if $m' + 1 < n$) then there may not exist sufficiently many edges for this to be possible. However, if this is the case, we can simply mix in “dummy edges” that the algorithm can treat the same as real edges in order to cause $m' + 1$ to be at least n. The details of this mixing in are described in the proof of Theorem 4.

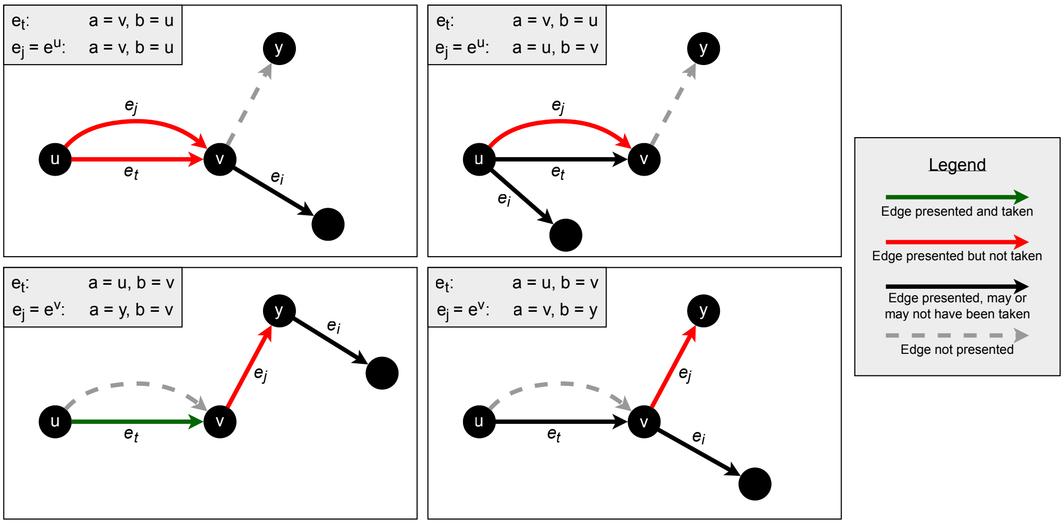

Figure 1 The described "worst-case" example for Algorithm 1. At step t, we are presented with an edge e_t in the optimum solution. e_t may be blocked from being taken due to an earlier step j. We suppose that the outgoing edge e^u from u in T_j^{opt} goes to v while the outgoing edge e^v from v in T_j^{opt} goes to a third vertex y; these are the only possible values of e_j with a potential to block e_t. All of e_t, e^u, e^v are depicted. Additionally depicted is an edge e_i outgoing from the endpoint of e_j selected as b – this edge being presented at step i would block e_j from being taken. Each of the four images depicts one equally likely possibility for the random choice of (a, b) in the case of e_t as well as in the case of e_j, where e_j is assumed to be the edge outgoing from b in T_j^{opt} (as this is the only case relevant for an increased probability arising from e_j not being taken). Only in one case is e_t guaranteed to be taken (assuming steps other than those depicted do not block e_t from being taken).

Analysis

We now proceed with a formal analysis. Due to space constraints, parts of the proof are deferred to the full version of the paper. We first show that any set of edges accepted by Algorithm 1 is independent, meaning that the algorithm is a valid graphic matroid secretary algorithm.

▶ **Lemma 5.** *The set A of accepted edges in Algorithm 1 is always an independent set in the graphic matroid.*

We now proceed to demonstrate that Algorithm 1 has the desired approximation factor for an appropriate choice of m'. We first define quantities f, g, h below that will be crucial for the analysis:

▶ **Definition 6.** *Define $f_{u,S}(i)$ to be the probability that after step i, seen_outgoing(u) = False, given that S is the set of edges that are presented in steps $1, \ldots, i$.*

▶ **Definition 7.** *Define $g_{u,v,S}(i)$ to be the probability that after step i, both seen_outgoing(u) = False and seen_outgoing(v) = False, given that S is the set of edges that are presented in steps $1, \ldots, i$.*

▶ **Definition 8.** *Define $h_{u,v,S}(i)$ to be the probability that after step i, if we take (a, b) to be either (u, v) or (v, u) with equal probability, then seen_outgoing(a) = False and taken_outgoing(b) = False, given that S is the set of edges that are presented in steps $1, \ldots, i$.*

The condition used to define h is the same condition checked to see whether an edge between u and v can be taken.

A key tool for our analysis is the following. Note that for any node u, *seen_outgoing*(u) is set to $True$ when at some step j we observe an edge e such that e is in T_j^{opt} and is directed from u. Importantly, T_j^{opt} depends only on the set of edges presented up to step j and not their order. Therefore, conditioned on the set of edges that appears in steps $1, \ldots, j$, the probability that at step j we are presented with the outgoing edge from u is at most $\frac{1}{j}$ because there is exactly one such edge, and this edge, if it exists, is presented at step i with probability $\frac{1}{i}$.

The above idea is applied in the following lemma:

▶ **Lemma 9.** *For all vertices u, all i from m' to m, and all S with $|S| = i$, $f_{u,S}(i) = \frac{m'}{i}$.*

The next lemma uses the same idea to derive a similar expression for g:

▶ **Lemma 10.** *For all $u, v \in V$ and $i \in [m', m]$, and all S with $|S| = i$, $g_{u,v,S}(i) \geq \frac{m'(m'-1)}{i(i-1)}$.*

We now proceed to lower bound h; we again use the same idea but now additionally take advantage of the stronger condition applied to a in the algorithm to derive a bound for h superior to the one we derived for g by lower bounding the probability that an edge that would have been selected for one of u, v was not actually taken. As the lower bound for h is more complicated, we define its lower bound as an additional function h_δ.

▶ **Definition 11.** *Define $h_\delta(i)$ recursively by $h_\delta(m') = 1$ and $h_\delta(i) = (1 - \frac{2}{i})h_\delta(i-1) + \frac{\delta}{i} \cdot \frac{m'(m'-1)}{(i-1)(i-2)} \left[1 - \frac{m'}{i-1}\right]$.*

We parameterize h by δ as h will reused later with a different value of δ.

▶ **Lemma 12.** *For all vertices u, v, all i from m' to m, and all S with $|S| = i$, $h_{u,v,S}(i) \geq h_{\frac{1}{4}}(i)$.*

As mentioned before, the conditions defining $h_{u,v,S}(i)$ are identical to the conditions for taking an edge given that it is in T_i^{opt}. As any edge in the overall optimum is in T_i^{opt}, lower bounding the probability of such an edge being taken allows us to lower bound the overall approximation factor. This is expressed in the following lemma.

▶ **Lemma 13.** *Let $W(S)$ be the sum of the weights of edges in S, and recall that A is the set of edges accepted by Algorithm 1. Then,*

$$\frac{\mathbb{E}[W(A)]}{W(OPT)} \geq \frac{1}{m} \sum_{t=m'+1}^{m} h_{\frac{1}{4}}(t-1).$$

It now only remains to give a concrete lower bound for $\frac{1}{m}\sum_{t=m'+1}^{m} h_{\frac{1}{4}}(t-1)$. We proceed to derive such a bound in the following lemma.

▶ **Lemma 14.** *If we let $m' = \lfloor \alpha m \rfloor$, then*

$$\lim_{m \to \infty} \frac{1}{m} \sum_{i=m'+1}^{m} h_\delta(i-1) \geq \alpha - \alpha^2 + \delta\alpha^2 \ln \alpha + \frac{\delta}{2}\left(\alpha - \alpha^3\right).$$

Setting $\alpha = 0.4914$, we can show that the above lemma implies Theorem 4. We refer to the full version of the paper for the full proof.

5 Impossibility result for high girth graphs: proof of Theorem 3

In this section, we prove a hardness result for high-girth graphs. We begin by showing that the weighted single secretary problem remains hard even when the instance is drawn from a known distribution, using the Finite Ramsey Theorem and zero-sum game duality. We then reduce this to the weighted graphic matroid secretary problem on large-girth graphs via a construction based on Ramanujan graphs. Specifically, by carefully removing edges using the probabilistic method, we construct bipartite graphs with certain properties and embed hard instances of single secretary into them to obtain similarly hard graphic secretary instances. Due to space constraints, parts of the proofs are deferred to the full version of the paper.

We note that it is crucial that the hardness holds even when the distribution is known in advance, unlike deterministic instances that are only hard when the instance is hidden. This allows us to present multiple instances from the distribution without the algorithm gaining any advantage from earlier ones – since it already knows the distribution fully.

5.1 Hard inputs for single secretary

We first prove that no algorithm for the secretary problem can achieve a competitive ratio better than $1/e$, even if the weights are drawn from a known finite set.

▶ **Theorem 15.** *For any $\varepsilon > 0$ and for any $\rho > 1$, there exists an integer n and a finite set of numbers $W \subset \{1, \rho, \rho^2, \dots\}$ over which no (randomized) algorithm for the single secretary problem with n items can choose the maximum with probability $\frac{1}{e} + \varepsilon$.*

See the full version of the paper for the proof. The result essentially serves as the starting point of our proof as it provides "hard instances" of the secretary problem which we then carefully embed in graphs of large girth. Correa et al. [15, 13] previously proved this theorem for infinite sets; our proof modifies their technique using the finite version of Ramsey's theorem.

While the generalization to finite sets may seem like a small technical issue, it has important consequences. As we will see, the finite assumption is necessary as it allows us to invoke the Minimax theorem for finite games. This in turn allows us to guarantee that there is a hard distribution over inputs, rather than just a single instance. As we will need to simultaneously embed multiple "independent" hard instances of the single secretary problem to obtain a hard instance of the graphical matroid secretary, this generalization is necessary.

We will now use the Minimax principle and Theorem 15 to prove the following.

▶ **Theorem 16.** *For any $\varepsilon > 0$ and for any $\rho > 1$, there exists an integer n, a finite set $W \subset \{1, \rho, \rho^2, \dots\}$, and a finite distribution D over all subsets of size n from W with the following property. If a random instance of the secretary problem is drawn from D, for any algorithm, the probability of choosing the maximum element from the random instance is at most $\frac{1}{e} + \varepsilon$, where the probability is over the randomness of the instance and the algorithm.*

5.2 Embedding single secretary in simple graphs

This subsection presents the proof of Theorem 3. We first describe some preliminaries which are used in our analysis.

Preliminaries

It is straightforward to reduce the single secretary problem to the graphic matroid secretary problem when the graph allows for multiple edges. In that case, given the single secretary problem with n adversarial weights, we create a multi-graph with just two vertices u and v

and a set of n parallel edges between u and v, with each edge assigned exactly one of these n adversarial weights. Note, that this multi-graph has girth 2. We will need to work harder to reduce the single secretary problem to the graphic matroid problem on simple graphs, where we will even assume that the simple graph has an arbitrarily large girth.

For said reduction, we define below three different measures of performance of a secretarial algorithm. Given an algorithm $a \in \mathcal{A}$ and an instance I of the matroid secretary problem, we abuse the notation from the previous section and denote by $a(I)$ the set of elements of the matroid output by a on instance I. Let also $\mathit{opt}(I)$ be the optimal solution to the problem on the instance I. Given any subset X of the matroid's elements, we denote by $w(X)$ the sum of the weights of all elements in X.

Let Π be a matroid secretary problem on a matroid which has $n \in \mathbb{N}$ elements. Given a finite set of numbers (weights) $W \subset \mathbb{R}_+$, let D be a probability distribution over the set W^n. We will later on instantiate Π with the single secretary problem and with the graphic matroid secretary problem.

Let $a \in \mathcal{A}$ be a (randomized) algorithm for problem Π. Given an instance I of Π, let $S_a(I)$ be a random variable such that $S_a(I) = 1$ if a outputs the optimal solution on instance I, that is, when $a(I) = \mathit{opt}(I)$; and $S_a(I) = 0$ if $a(I) \neq \mathit{opt}(I)$. Then we have that $\mathbb{E}[S_a(I)] = \mathbb{P}\mathrm{r}[a(I) = \mathit{opt}(I)]$; note that the probability is over the internal randomness of algorithm a and over the randomness in the arrival order of the matroid elements.

For the convenience of this subsection, it will be useful to define an algorithm a as having a performance guarantee $\alpha \leq 1$ of

- type 1 if $\mathbb{E}_{I\sim D}[\mathbb{E}[S_a(I)]] \geq \alpha$.
  That is, the algorithm chooses the optimum with probability at least α in expectation over instances.
- type 2 if $\mathbb{E}_{I\sim D}[w(a(I))/w(\mathit{opt}(I))] \geq \alpha$.
  That is, the expected ratio of the algorithm's weight to the optimum's weight is at least α, where the expectation is over instances, algorithm randomness, and element order.
- type 3 if $\mathbb{E}_{I\sim D}[w(a(I))] \geq \mathbb{E}_{I\sim D}[w(\mathit{opt}(I))] \cdot \alpha$.
  That is, the algorithm's expected weight is at least an α fraction of the expected optimum weight. The first expectation includes all randomness; the second expectation is over instances only.

We next reformulate Theorem 16 in terms of the type 1 performance guarantee.

▶ **Theorem 17.** *For any $\epsilon > 0$, $\rho > 1$, there exists a finite distribution D of instances of weighted single secretary such that no algorithm's type 1 performance guarantee is $\geq \frac{1}{e} + \epsilon$. For any instance I of single secretary in the support of D, any two weights in I differ by at least a factor of ρ.*

We now extend the result of Theorem 17 to the type 2 performance guarantee. This proof will make use of the guarantee depending on ρ provided by Theorem 17 to bound the weight attained by an algorithm when it fails to select the maximum.

▶ **Lemma 18.** *For any $\epsilon > 0$, there exists a finite distribution D of instances of weighted single secretary such that no algorithm's type 2 performance guarantee is $\geq \frac{1}{e} + \epsilon$.*

The following lemma then shows an equivalence in hardness between the type 2 and type 3 performance guarantees. The idea of its proof is to reweight the distribution D by making the probability of each instance being chosen inversely proportional to its optimum weight.

▶ **Lemma 19.** *For any matroid secretary problem Π, there exists a bijection B between finite distributions over instances of Π such that if $D' = B(D)$, then there exists an algorithm with a type 2 performance guarantee of r on D iff there exists one with a type 3 performance guarantee of r on D'.*

We now move to the portion of the proof that extends the hardness of the single secretary problem to the graph setting. As part of this reduction, we first show the following lemma, demonstrating the existence of bipartite graphs of high girth with the key property that one part is both much larger than the other and has high degree. This construction proceeds by applying the probabilistic method to derive a graph with our desired properties by removing edges from a high-girth Ramanujan graph.

▶ **Lemma 20.** *For any $d, g, t \in \mathbb{N}$ with $d \geq 2$, $t \geq 4$, there exist $m, n \in \mathbb{N}$ such that $\frac{n}{m} \geq t$ and there exists a graph G' of girth at least g, whose vertices can be partitioned into sets A and B, such that $|A| = m$, $|B| = n$, all edges are between A and B, and each vertex in B has degree at least d.*

We now apply the graph provided by Lemma 20 to extend the hardness of a distribution of instances of the single secretary problem to hardness of matroid secretary on a high-girth graph. The proof, which is quite technical, is provided in the full version of the paper.

▶ **Lemma 21.** *If there exists a finite distribution D of instances of weighted single secretary such that no algorithm has a type 3 performance guarantee $\geq r$ on D, then for any $\epsilon > 0, g \in \mathbb{N}$ there exists a graph G with girth $\geq g$ and a finite distribution D_G of instances of weighted graph secretary on G such that no algorithm has a type 3 performance guarantee $\geq r + \epsilon$ on D'.*

Combining the above results, we obtain Theorem 3; see the full version of the paper for more details.

References

1 Dorna Abdolazimi, Anna R. Karlin, Nathan Klein, and Shayan Oveis Gharan. Matroid partition property and the secretary problem. In Yael Tauman Kalai, editor, *14th Innovations in Theoretical Computer Science Conference, ITCS 2023, January 10-13, 2023, MIT, Cambridge, Massachusetts, USA*, volume 251 of *LIPIcs*, pages 2:1–2:9. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023. doi:10.4230/LIPICS.ITCS.2023.2.

2 Melika Abolhassani, Soheil Ehsani, Hossein Esfandiari, MohammadTaghi HajiAghayi, Robert Kleinberg, and Brendan Lucier. Beating 1-1/e for ordered prophets. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 61–71. ACM, 2017. doi:10.1145/3055399.3055479.

3 Saeed Alaei, MohammadTaghi Hajiaghayi, and Vahid Liaghat. The online stochastic generalized assignment problem. In *Proceedings of the 16th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX)*, pages 11–25, 2013. doi:10.1007/978-3-642-40328-6_2.

4 Noga Alon, Nick Gravin, Tristan Pollner, Aviad Rubinstein, Hongao Wang, S Matthew Weinberg, and Qianfan Zhang. A bicriterion concentration inequality and prophet inequalities for k-fold matroid unions. *arXiv preprint arXiv:2411.11741*, 2024. doi:10.48550/arXiv.2411.11741.

5 Yossi Azar, Ashish Chiplunkar, and Haim Kaplan. Prophet secretary: Surpassing the 1-1/e barrier. In *Proceedings of the 2018 ACM Conference on Economics and Computation, Ithaca, NY, USA, June 18-22, 2018*, pages 303–318, 2018. doi:10.1145/3219166.3219182.

6 Moshe Babaioff, Michael Dinitz, Anupam Gupta, Nicole Immorlica, and Kunal Talwar. Secretary problems: weights and discounts. In *Proceedings of the twentieth annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1245–1254. SIAM, 2009. doi:10.1137/1.9781611973068.135.

7 Moshe Babaioff, Nicole Immorlica, David Kempe, and Robert Kleinberg. Matroid secretary problems. *J. ACM*, 65(6):35:1–35:26, 2018. doi:10.1145/3212512.

8 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. URL: http://dl.acm.org/citation.cfm?id=1283383.1283429.

9 Maryam Bahrani, Hedyeh Beyhaghi, Sahil Singla, and S. Matthew Weinberg. Formal barriers to simple algorithms for the matroid secretary problem. In Michal Feldman, Hu Fu, and Inbal Talgam-Cohen, editors, *Web and Internet Economics - 17th International Conference, WINE 2021, Potsdam, Germany, December 14-17, 2021, Proceedings*, volume 13112 of *Lecture Notes in Computer Science*, pages 280–298. Springer, 2021. doi:10.1007/978-3-030-94676-0_16.

10 Mohammadhossein Bateni, Mohammadtaghi Hajiaghayi, and Morteza Zadimoghaddam. Submodular secretary problem and extensions. *ACM Trans. Algorithms*, 9(4):Art. 32, 23, 2013.

11 Kristóf Bérczi, Vasilis Livanos, José Soto, and Victor Verdugo. Matroid secretary via labeling schemes. *arXiv preprint arXiv:2411.12069*, 2024. doi:10.48550/arXiv.2411.12069.

12 Shuchi Chawla, Jason D. Hartline, David L. Malec, and Balasubramanian Sivan. Multi-parameter mechanism design and sequential posted pricing. In *Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC 2010, Cambridge, Massachusetts, USA, 5-8 June 2010*, pages 311–320, 2010. doi:10.1145/1806689.1806733.

13 José Correa, Paul Dütting, Felix A. Fischer, and Kevin Schewior. Prophet inequalities for I.I.D. random variables from an unknown distribution. In Anna R. Karlin, Nicole Immorlica, and Ramesh Johari, editors, *Proceedings of the 2019 ACM Conference on Economics and Computation, EC 2019, Phoenix, AZ, USA, June 24-28, 2019*, pages 3–17. ACM, 2019. doi:10.1145/3328526.3329627.

14 José Correa, Patricio Foncea, Ruben Hoeksma, Tim Oosterwijk, and Tjark Vredeveld. Posted price mechanisms for a random stream of customers. In *Proceedings of the 2017 ACM Conference on Economics and Computation*, pages 169–186. ACM, 2017. doi:10.1145/3033274.3085137.

15 José R. Correa, Paul Dütting, Felix A. Fischer, and Kevin Schewior. Prophet inequalities for independent random variables from an unknown distribution. *CoRR*, abs/1811.06114v2, 2021. URL: http://arxiv.org/abs/1811.06114v2.

16 José R. Correa, Raimundo Saona, and Bruno Ziliotto. Prophet secretary through blind strategies. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, pages 1946–1961. SIAM, 2019. doi:10.1137/1.9781611975482.118.

17 José R. Correa, Raimundo Saona, and Bruno Ziliotto. Prophet secretary through blind strategies. *Math. Program.*, 190(1):483–521, 2021. doi:10.1007/S10107-020-01544-8.

18 Sina Dehghani, Soheil Ehsani, MohammadTaghi Hajiaghayi, Vahid Liaghat, and Saeed Seddighin. Stochastic k-server: How should uber work? In *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10-14, 2017, Warsaw, Poland*, pages 126:1–126:14, 2017. doi:10.4230/LIPICS.ICALP.2017.126.

19 Nedialko B Dimitrov and C Greg Plaxton. Competitive weighted matching in transversal matroids. In *Automata, Languages and Programming: 35th International Colloquium, ICALP 2008, Reykjavik, Iceland, July 7-11, 2008, Proceedings, Part I 35*, pages 397–408. Springer, 2008. doi:10.1007/978-3-540-70575-8_33.

20 Michael Dinitz and Guy Kortsarz. Matroid secretary for regular and decomposable matroids. *SIAM Journal on Computing*, 43(5):1807–1830, 2014. doi:10.1137/13094030X.

21 Paul Dütting and Robert Kleinberg. Polymatroid prophet inequalities. In *Algorithms-ESA 2015*, pages 437–449. Springer, 2015. doi:10.1007/978-3-662-48350-3_37.

22 Hossein Esfandiari, MohammadTaghi Hajiaghayi, Vahid Liaghat, and Morteza Monemizadeh. Prophet secretary. *SIAM J. Discret. Math.*, 31(3):1685–1701, 2017. doi:10.1137/15M1029394.

23 Tomer Ezra, Michal Feldman, Nick Gravin, and Zhihao Gavin Tang. General graphs are easier than bipartite graphs: Tight bounds for secretary matching. In David M. Pennock, Ilya Segal, and Sven Seuken, editors, *EC '22: The 23rd ACM Conference on Economics and Computation, Boulder, CO, USA, July 11 - 15, 2022*, pages 1148–1177. ACM, 2022. doi:10.1145/3490486.3538290.

24 Michal Feldman, Nick Gravin, and Brendan Lucier. Combinatorial auctions via posted prices. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 123–135. SIAM, 2015. doi:10.1137/1.9781611973730.10.

25 Moran Feldman and Rico Zenklusen. The submodular secretary problem goes linear. *SIAM J. Comput.*, 47(2):330–366, 2018. doi:10.1137/16M1105220.

26 Naveen Garg, Anupam Gupta, Stefano Leonardi, and Piotr Sankowski. Stochastic analyses for online combinatorial optimization problems. In *SODA*, pages 942–951. Society for Industrial and Applied Mathematics, 2008. URL: http://dl.acm.org/citation.cfm?id=1347082.1347185.

27 Oliver Göbel, Martin Hoefer, Thomas Kesselheim, Thomas Schleiden, and Berthold Vöcking. Online independent set beyond the worst-case: Secretaries, prophets, and periods. In *International Colloquium on Automata, Languages, and Programming*, pages 508–519. Springer, 2014. doi:10.1007/978-3-662-43951-7_43.

28 Anupam Gupta, Aaron Roth, Grant Schoenebeck, and Kunal Talwar. Constrained non-monotone submodular maximization: Offline and secretary algorithms. In *Proceedings of the 6th International Workshop on Internet and Network Economics (WINE'10)*, 2010.

29 Mohammad Taghi Hajiaghayi, Robert Kleinberg, and Tuomas Sandholm. Automated online mechanism design and prophet inequalities. In *AAAI*, volume 7, pages 58–65, 2007. URL: http://www.aaai.org/Library/AAAI/2007/aaai07-009.php.

30 Mohammad Taghi Hajiaghayi, Robert D. Kleinberg, and David C. Parkes. Adaptive limited-supply online auctions. In *Proceedings 5th ACM Conference on Electronic Commerce (EC-2004), New York, NY, USA, May 17-20, 2004*, pages 71–80. ACM, 2004. doi:10.1145/988772.988784.

31 Zhiyi Huang, Zahra Parsaeian, and Zixuan Zhu. Laminar matroid secretary: Greedy strikes back. In Timothy M. Chan, Johannes Fischer, John Iacono, and Grzegorz Herman, editors, *32nd Annual European Symposium on Algorithms, ESA 2024, September 2-4, 2024, Royal Holloway, London, United Kingdom*, volume 308 of *LIPIcs*, pages 73:1–73:8. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPICS.ESA.2024.73.

32 Sungjin Im and Yajun Wang. Secretary problems: Laminar matroid and interval scheduling. In *Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete Algorithms*, pages 1265–1274. SIAM, 2011. doi:10.1137/1.9781611973082.96.

33 Patrick Jaillet, José A Soto, and Rico Zenklusen. Advances on matroid secretary problems: Free order model and laminar case. In *Integer Programming and Combinatorial Optimization: 16th International Conference, IPCO 2013, Valparaíso, Chile, March 18-20, 2013. Proceedings 16*, pages 254–265. Springer, 2013. doi:10.1007/978-3-642-36694-9_22.

34 Thomas Kesselheim, Klaus Radke, Andreas Tönnis, and Berthold Vöcking. An optimal online algorithm for weighted bipartite matching and extensions to combinatorial auctions. In Hans L. Bodlaender and Giuseppe F. Italiano, editors, *Algorithms - ESA 2013 - 21st Annual European Symposium, Sophia Antipolis, France, September 2-4, 2013. Proceedings*, volume 8125 of *Lecture Notes in Computer Science*, pages 589–600. Springer, 2013. doi:10.1007/978-3-642-40450-4_50.

35 Thomas Kesselheim, Klaus Radke, Andreas Tönnis, and Berthold Vöcking. An optimal online algorithm for weighted bipartite matching and extensions to combinatorial auctions. In Hans L. Bodlaender and Giuseppe F. Italiano, editors, *Algorithms - ESA 2013 - 21st Annual European Symposium, Sophia Antipolis, France, September 2-4, 2013. Proceedings*, volume 8125 of *Lecture Notes in Computer Science*, pages 589–600. Springer, 2013. doi:10.1007/978-3-642-40450-4_50.

36 Robert Kleinberg. A multiple-choice secretary algorithm with applications to online auctions. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 630–631. Society for Industrial and Applied Mathematics, 2005. URL: http://dl.acm.org/citation.cfm?id=1070432.1070519.

37 Robert Kleinberg and S. Matthew Weinberg. Matroid prophet inequalities. In *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, New York, NY, USA, May 19 - 22, 2012*, pages 123–136, 2012. doi:10.1145/2213977.2213991.

38 Nitish Korula and Martin Pál. Algorithms for secretary problems on graphs and hypergraphs. In Susanne Albers, Alberto Marchetti-Spaccamela, Yossi Matias, Sotiris E. Nikoletseas, and Wolfgang Thomas, editors, *Automata, Languages and Programming, 36th Internatilonal Colloquium, ICALP 2009, Rhodes, Greece, July 5-12, 2009, Proceedings, Part II*, volume 5556 of *Lecture Notes in Computer Science*, pages 508–520. Springer, 2009. doi:10.1007/978-3-642-02930-1_42.

39 Ulrich Krengel and Louis Sucheston. Semiamarts and finite values. *Bull. Am. Math. Soc*, 1977.

40 Ulrich Krengel and Louis Sucheston. On semiamarts, amarts, and processes with finite value. *Advances in Prob*, 4:197–266, 1978.

41 Tengyu Ma, Bo Tang, and Yajun Wang. The simulated greedy algorithm for several submodular matroid secretary problems. *Theory of Computing Systems*, 58:681–706, 2016. doi:10.1007/S00224-015-9642-4.

42 Adam Meyerson. Online facility location. In *Foundations of Computer Science, 2001. Proceedings. 42nd IEEE Symposium on*, pages 426–431. IEEE, 2001. doi:10.1109/SFCS.2001.959917.

43 Bo Peng and Zhihao Gavin Tang. Order selection prophet inequality: From threshold optimization to arrival time design. In *Proceedings of the 63rd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2022.

44 Aviad Rubinstein. Beyond matroids: Secretary problem and prophet inequality with general constraints. *arXiv preprint arXiv:1604.00357*, 2016. arXiv:1604.00357.

45 Aviad Rubinstein and Sahil Singla. Combinatorial prophet inequalities. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1671–1687, 2017. doi:10.1137/1.9781611974782.110.

46 José A. Soto, Abner Turkieltaub, and Victor Verdugo. Strong algorithms for the ordinal matroid secretary problem. In Artur Czumaj, editor, *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018*, pages 715–734. SIAM, 2018. doi:10.1137/1.9781611975031.47.

47 Ola Svensson. A simple o(loglog(rank))-competitive algorithm for the matroid secretary problem, 2016. Presentation at Microsoft Research available on Youtube; timestamp 35:42. URL: https://www.youtube.com/watch?v=hHBGKbq79yk&t=1776s.