On Variants of the Matroid Secretary Problem

Shayan Oveis Gharan[1] * and Jan Vondrák[2]

[1] Stanford University, Stanford, CA shayan@stanford.edu
[2] IBM Almaden Research Center, San Jose, CA jvondrak@us.ibm.com

Abstract. We present a number of positive and negative results for variants of the matroid secretary problem. Most notably, we design a constant-factor competitive algorithm for the "random assignment" model where the weights are assigned randomly to the elements of a matroid, and then the elements arrive on-line in an adversarial order (extending a result of Soto [20]). This is under the assumption that the matroid is known in advance. If the matroid is unknown in advance, we present an $O(\log r \log n)$-approximation, and prove that a better than $O(\log n/ \log\log n)$ approximation is impossible. This resolves an open question posed by Babaioff et al. [3].
As a natural special case, we also consider the classical secretary problem where the number of candidates n is unknown in advance. If n is chosen by an adversary from $\{1, \ldots, N\}$, we provide a nearly tight answer, by providing an algorithm that chooses the best candidate with probability at least $1/(H_{N-1} + 1)$ and prove that a probability better than $1/H_N$ cannot be achieved (where H_N is the N-th harmonic number).

1 Introduction

The secretary problem is a classical problem in probability theory, with obscure origins in the 1950's and early 60's ([11,17,8]; see also [10]). Here, the goal is to select the best candidate out of a sequence revealed one-by-one, where the ranking is uniformly random. A classical solution finds the best candidate with probability at least $1/e$ [10]. Over the years a number of variants have been studied, starting with [12] where multiple choices and various measures of success were considered for the first time.

Recent interest in variants of the secretary problem has been motivated by applications in on-line mechanism design [14,18,3], where items are being sold to agents arriving on-line, and there are certain constraints on which agents can be simultaneously satisfied. Equivalently, one can consider a setting where we want to hire several candidates under certain constraints. Babaioff, Immorlica and Kleinberg [3] formalized this problem and presented constant-factor competitive algorithms for several interesting cases. The general problem formulated in [3] is the following.

* This work was done while the author was an intern at IBM Almaden Research Center, San Jose, CA

Matroid secretary problem. Given a matroid $\mathcal{M} = (E, \mathcal{I})$ with non-negative weights assigned to E; the only information known up-front is the number of elements $n := |E|$. The elements of E arrive in a random order, with their weights revealed as they arrive. When an element arrives, it can be selected or rejected. The selected elements must always form an independent set in $\mathcal{M}$, and a rejected element cannot be considered again. The goal is to maximize the expected weight of the selected elements.

Additional variants of the matroid secretary problem have been proposed and studied, depending on how the input ordering is generated, how the weights are assigned and what is known in advance. In all variants, elements with their weights arrive in an on-line fashion and an algorithm must decide irrevocably whether to accept or reject an element once it has arrived. We attempt to bring some order to the multitude of models and we classify the various proposed variants as follows.

Ordering of matroid elements on the input:
- AO = Adversarial Order: the ordering of elements of the matroid on the input is chosen by an adversary.
- RO = Random Order: the elements of the matroid arrive in a random order.

Assignment of weights:
- AA = Adversarial Assignment: weights are assigned to elements of the matroid by an adversary.
- RA = Random Assignment: the weights are assigned to elements by a random permutation of an adversarial set of weights (independent of the input order, if that is also random).

Prior information:
- MK = Matroid Known: the matroid is known beforehand (by means of an independence oracle).
- MN = Matroid - n known: the matroid is unknown but the cardinality of the ground set is known beforehand.
- MU = Matroid - Unknown: nothing about the matroid is known in advance; only subsets of the elements that arrived already can be queried for independence.

For example, the original variant of the matroid secretary problem [3], where the only information known beforehand is the total number of elements, can be described as RO-AA-MN in this classification. We view this as the primary variant of the matroid secretary problem.

We also consider variants of the classical secretary problem; here, only 1 element should be chosen and the goal is to maximize the probability of selecting the best element.

Classical secretary problems:
- CK = Classical - Known n: the classical secretary problem where the number of elements in known in advance.

– CN = Classical - known upper bound N: the classical secretary problem where the number of elements is chosen adversarially from $\{1, \ldots, N\}$, and N is known in advance.
– CU = Classical - Unknown n: the classical secretary problem where no information on the number of elements is known in advance.

Since the independence sets of the underlying matroid in this model are independent of the particular labeling of the ground set, we just use the weight assignment function to characterize different variants of this model. The classical variant of the secretary problem which allows a $1/e$-approximation would be described as RA-CK. The variant where the number of elements n is not known in advance is very natural — and has been considered under different stochastic models where n is drawn from a particular distribution [21,1] — but the worst-case scenario does not seem to have received attention. We denote this model RA-CU, or RA-CN if an upper bound on the number of candidates is given. In the model where the input ordering of weights is adversarial (AA-CK), it is easy to see that no algorithm achieves probability better than $1/n$ [5]. We remark that variants of the secretary problem with other objective functions have been also proposed, such as discounted profits [2], and submodular objective functions [4,13]. We do not discuss these variants here.

1.1 Recent related work

The primary variant of matroid secretary problem (RO-AA-MN model) was introduced in [3]. In the following, let n denote the total number of elements and r the rank of the matroid. An $O(\log r)$-approximation for the RO-AA-MN model was given in [3]. It was also conjectured that a constant-factor approximation should exist for this problem and this question is still open. Constant-factor approximations were given in [3] for some special cases such as partition matroids and graphic matroids with a given explicit representation. Further, constant-factor approximations were given for transversal matroids [7,19] and laminar matroids [16]. However, even for graphic matroids in the RO-AA-MK model when the graphic matroid is given by an oracle, no constant factor is known.

Babaioff et al. in [3] also posed as an open problem whether there is a constant-factor approximation algorithm for the following two models: Assume that a set of n numerical values are assigned to the matroid elements using a random one-to-one correspondence but that the elements are presented in an adversarial order (AO-RA in our notation). Or, assume that both the assignment of values and the ordering of the elements in the input are random (RO-RA in our notation). The issue of whether the matroid is known beforehand is left somewhat ambiguous in [3].

In a recent work [20], José Soto partially answered the second question, by designing a constant-factor approximation algorithm in the RO-RA-MK model: An adversary chooses a list of non-negative weights, which are then assigned to the elements using a random permutation, which is independent of the random order at which the elements are revealed. The matroid is known in advance here.

1.2 Our results

Matroid secretary. We resolve the question from [3] concerning adversarial order and random assignment, by providing a constant-factor approximation algorithm in the AO-RA-MK model, and showing that no constant-factor approximation exists in the AO-RA-MN model. More precisely, we prove that there is a $40/(1-1/e)$-approximation in the AO-RA-MK model, i.e. in the model where weights are assigned to the elements of a matroid randomly, the elements arrive in an adversarial order, and the matroid is known in advance. We provide a simple thresholding algorithm, which gives a constant-factor approximation for the AO-RA-MK model when the matroid $\mathcal{M}$ is uniformly dense. Then we use the principal sequence of a matroid to design a constant-factor approximation for any matroid using the machinery developed by Soto [20].

On the other hand, if the matroid is not known in advance (AO-RA-MN model), we prove that the problem cannot be approximated better than within $\Omega(\log n / \log\log n)$. This holds even in the special case of rank 1 matroids; see below. On the positive side, we show an $O(\log r \log n)$-approximation for this model. We achieve this by providing an $O(\log r)$-approximation thresholding algorithm for the AO-AA-MU model (when both the input ordering and the assignment of weights to the elements the matroid are adversarial), when an estimate on the weight of the largest non-loop element is given. Here, the novel technique is to employ a dynamic threshold depending on the rank of the elements seen so far.

Classical secretary with unknown n. A very natural question that arises in this context is the following. Consider the classical secretary problem, where we want to select 1 candidate out of n. The classical solution relies on the fact that n is known in advance. However, what if we do not know n in advance, which would be the case in many practical situations? We show that if an upper bound N on the possible number of candidates n is given (RA-CN model: i.e., n is chosen by an adversary from $\{1, \ldots, N\}$), the best candidate can be found with probability $1/(H_{N-1}+1)$, while there is no algorithm which achieves probability better than $1/H_N$ (where $H_N = \sum_{i=1}^{N} \frac{1}{i}$ is the N-th harmonic number).

In the model where we maximize the expected value of the selected candidate, and n is chosen adversarially from $\{1, \ldots, N\}$, we prove we cannot achieve approximation better than $\Omega(\log N / \log\log N)$. On the positive side, even if no upper bound on n is given, the maximum-weight element can be found with probability $\epsilon / \log^{1+\epsilon} n$ for any fixed $\epsilon > 0$. We remark that similar results follow from [15] and [9] where an equivalent problem was considered in the context of online auctions. More generally, for the matroid secretary problem where no information at all is given in advance (RO-AA-MU), we achieve an $O(\frac{1}{\epsilon} \log r \log^{1+\epsilon} n)$ approximation for any $\epsilon > 0$. See Table 1 for an overview of our results.

Organization. In section 2 we provide a $40/(1-1/e)$ approximation algorithm for the AO-RA-MK model. In section 3 we provide an $O(\log n \log r)$ approximation algorithm for the AO-RA-MN model, and an $O(\frac{1}{\epsilon} \log r \log^{1+\epsilon} n)$ approximation for the RO-AA-MU model. Finally, in section 4 we provide a $(H_{N-1}+1)$-approximation and H_N-hardness for the RA-CN model.

Problem	New approximation	New hardness
RA-CN	$H_{N-1}+1$	H_N
RA-CU	$O(\frac{1}{\epsilon}\log^{1+\epsilon} n)$	$\Omega(\log n)$
AO-RA-MK	$40/(1-1/e)$	-
AO-RA-MN	$O(\log r \log n)$	$\Omega(\log n/\log\log n)$
AO-RA-MU	$O(\frac{1}{\epsilon}\log r \log^{1+\epsilon} n)$	$\Omega(\log n/\log\log n)$
RO-AA-MU	$O(\frac{1}{\epsilon}\log r \log^{1+\epsilon} n)$	$\Omega(\log n/\log\log n)$

Table 1. Summary of results

2 Approximation for adversarial order and random assignment

In this section, we derive a constant-factor approximation algorithm for the AO-RA-MK model, i.e. assuming that the ordering of the elements of the matroid is adversarial but weights are assigned to the elements by a random permutation, and the matroid is known in advance. We build on Soto's algorithm [20], in particular on his use of the *principal sequence of a matroid* which effectively reduces the problem to the case of a uniformly-dense matroid while losing only a constant factor $(1-1/e)$. Interestingly, his reduction only requires the randomness in the assignment of weights to the elements but not a random ordering of the matroid on the input. Due to limited space here we do not include the details of the reduction, and we defer it to the full version of the paper.

Recall that the density of a set in a matroid $\mathcal{M}=(E,\mathcal{I})$ is the quantity $\gamma(S)=\frac{|S|}{rank(S)}$. A matroid is uniformly dense, if $\gamma(S)\le\gamma(E)$ for all $S\subseteq E$. We present a simple thresholding algorithm which works in the AO-RA-MK model (i.e. even for an adversarial ordering of the elements) for any uniformly dense matroid.

Throughout this section we use the following notation. Let $\mathcal{M}=(E,I)$ be a uniformly dense matroid of rank r. This also means that $\mathcal{M}$ contains no loops. Let $|E|=n$ and let $e_1,e_2,\ldots,e_n$ denote the ordering of the elements on the input, which is chosen by an adversary (i.e. we consider the worst case). Furthermore, the adversary also chooses $W=\{w_1>w_2>\ldots>w_n\}$, a set of non-negative weights. The weights are assigned to the elements of $\mathcal{M}$ via a random bijection $\omega: E\to W$. For a weight assignment ω, we denote by $w(S)=\sum_{e\in S}\omega(e)$ the weight of a set S, and by $\omega(S)=\{\omega(e): e\in S\}$ the set of weights assigned to S. We also let $\mathrm{OPT}(\omega)$ be the maximum-weight independent set in $\mathcal{M}$.

Recall that r denotes the rank of the matroid. We show that there is a simple thresholding algorithm which includes each of the topmost $\lfloor r/4\rfloor$ weights (i.e. $w_1,\ldots,w_{\lfloor r/4\rfloor}$) with a constant probability. This will give us a constant factor approximation algorithm, as $w(\mathrm{OPT}(\omega))\le\sum_{i=1}^{r} w_i$, where $w_1>w_2>\ldots>w_r$ are the r largest weights in W. It is actually important that we compare our algorithm to the quantity $\sum_{i=1}^{r} w_i$, because this is needed in the reduction to the uniformly dense case.

The main idea is that the randomization of the weight assignment makes it very likely that the optimum solution contains many of the top weights in W. Therefore, instead of trying to compute the optimal solution with respect to ω, we can just focus on catching a constant fraction of the top weights in W. Let $A = \{e_1, \ldots, e_{n/2}\}$ denote the first half of the input and $B = \{e_{n/2+1}, \ldots, e_n\}$ the second half of the input. Note that the partition into A and B is determined by the adversary and not random. Our solution is to use the $\lfloor r/4 \rfloor + 1$-st topmost weight in the "sampling stage" A as a threshold and then include every element in B that is above the threshold and independent of the previously selected elements. Details are described in Algorithm 1.

Algorithm 1 Thresholding algorithm for uniformly dense matroids in AO-RA-MK model

Input: A uniformly dense matroid $\mathcal{M} = (E, \mathcal{I})$ of rank r.
Output: An independent set $\mathrm{ALG} \subseteq E$.

1: **if** $r < 12$ **then**
2:    run the optimal algorithm for the classical secretary problem, and return the resulting singleton.
3: **end if**
4: $\mathrm{ALG} \leftarrow \emptyset$
5: Observe a half of the input (elements of A) and let w^* be the $(\lfloor r/4 \rfloor + 1)^{st}$ largest weight among them.
6: **for** each element $e \in B$ arriving afterwards **do**
7:    **if** $\omega(e) > w^*$ and $ALG \cup \{e\}$ is independent **then**
8:       $\mathrm{ALG} \leftarrow \mathrm{ALG} \cup \{e\}$
9:    **end if**
10: **end for**
11: **return** ALG

Theorem 2.1. *Let $\mathcal{M}$ be a uniformly dense matroid of rank r, and $\mathrm{ALG}(\omega)$ be the set returned by Algorithm 1 when the weights are defined by a uniformly random bijection $\omega : E \to W$. Then*

$$\mathbf{E}_\omega \left[w(\mathrm{ALG}(\omega)) \right] \geq \frac{1}{40} \sum_{i=1}^{r} w_i$$

where $\{w_1 > w_2 > \ldots > w_r\}$ are the r largest weights in W.

If $r < 12$, the algorithm finds and returns the largest weight w_1 with probability $1/e$ (step 2; the optimal algorithm for the classical secretary problem). Therefore, for $r < 12$, we have $\mathbf{E}_\omega \left[w(\mathrm{ALG}(\omega)) \right] \geq \frac{1}{11e} \sum_{i=1}^{r} w_i > \frac{1}{40} \sum_{i=1}^{r} w_i$.

For $r \geq 12$, we prove that each of the topmost $\lfloor r/4 \rfloor$ weights will be included in $\mathrm{ALG}(\omega)$ with probability at least $1/8$. Hence, we will obtain

$$\mathbf{E}_\omega \left[w(\mathrm{ALG}(\omega)) \right] \geq \frac{1}{8} \sum_{i=1}^{\lfloor r/4 \rfloor} w_i \geq \frac{1}{40} \sum_{i=1}^{r} w_i. \tag{1}$$

Let $t = 2\lfloor r/4 \rfloor + 2$. Define $C'(\omega) = \{e_j : \omega(e_j) \geq w_t\}$ to be the set of elements of $\mathcal{M}$ which get one of the top t weights. Also let $A'(\omega) = C'(\omega) \cap A$ and $B'(\omega) = C'(\omega) \cap B$. Moreover, for each $1 \leq i \leq t$ we define $C'_i(\omega) = \{e_j : \omega(e_j) \geq w_t \ \& \ \omega(e_j) \neq w_i\}$, $A'_i(\omega) = C'_i(\omega) \cap A$ and $B'_i(\omega) = C'_i(\omega) \cap B$, i.e. the same sets with the element of weight w_i removed.

First, we fix $i \leq \lfloor r/4 \rfloor$ and argue that the size of $B'_i(\omega)$ is smaller than $A'_i(\omega)$ with probability 1/2. Then we will use the uniformly dense property of $\mathcal{M}$ to show that the span of $B'_i(\omega)$ is also quite small with probability 1/2 and consequently w_i has a good chance of being included in $\mathrm{ALG}(\omega)$.

Claim 2.2 *Let $\mathcal{M}$ be a uniformly dense matroid of rank r, $t = 2\lfloor r/4 \rfloor + 2$, $1 \leq i \leq \lfloor r/4 \rfloor$, and $B'_i(\omega)$ defined as above. Then we have*

$$\mathbf{P}_\omega \left[|B'_i(\omega)| \leq \lfloor r/4 \rfloor \right] = 1/2. \tag{2}$$

Proof: Consider $C'_i(\omega)$, the set of elements receiving the top t weights except for w_i. This is a uniformly random set of odd size $t - 1 = 2\lfloor r/4 \rfloor + 1$. By symmetry, with probability exactly 1/2, a majority of these elements are in A, and hence at most $\lfloor r/4 \rfloor$ of these elements are in B, i.e. $|B'_i(\omega)| \leq \lfloor r/4 \rfloor$. □

Now we consider the element receiving weight w_i. We claim that this element will be included in $\mathrm{ALG}(\omega)$ with a constant probability.

Claim 2.3 *Let $\mathcal{M}$ be a uniformly dense matroid of rank r, and $i \leq \lfloor r/4 \rfloor$. Then*

$$\mathbf{P}_\omega \left[\omega^{-1}(w_i) \in \mathrm{ALG}(\omega) \right] \geq 1/8.$$

Proof: Condition on $C'_i(\omega) = S$ for some particular set S of size $t-1$ such that $|B'_i(\omega)| = |S \cap B| \leq \lfloor r/4 \rfloor$. This fixes the assignment of the top t weights except for w_i. Under this conditioning, weight w_i is still assigned uniformly to one of the remaining $n - t + 1$ elements.

Since we have $|A'_i(\omega)| = |S \cap A| \geq \lfloor r/4 \rfloor + 1$, the threshold w^* in this case is one of the top t weights and the algorithm will never include any weight outside of the top t. Therefore, we have $\mathrm{ALG}(\omega) \subseteq B'(\omega)$. The weight w_i is certainly above w^* because it is one of the top $\lfloor r/4 \rfloor$ weights. It will be added to $\mathrm{ALG}(\omega)$ whenever it appears in B and it is not in the span of previously selected elements. Since all the previously included elements must be in $B'_i(\omega) = S \cap B$, it is sufficient to avoid being in the span of $S \cap B$. To summarize, we have

$$\omega^{-1}(w_i) \in B \setminus \mathit{span}(S \cap B) \Rightarrow \omega^{-1}(w_i) \in \mathrm{ALG}(\omega).$$

What is the probability that this happens? Similar to the proof of [20, Lemma 3.1], since $\mathcal{M}$ is uniformly dense, we have

$$\frac{|\mathit{span}(S \cap B)|}{|S \cap B|} \leq \frac{|\mathit{span}(S \cap B)|}{\mathit{rank}(\mathit{span}(S \cap B))} \leq \frac{n}{r} \Longrightarrow |\mathit{span}(S \cap B)| \leq \frac{n}{r}|S \cap B| \leq \frac{n}{4}$$

using $|S \cap B| \leq \lfloor r/4 \rfloor$. Therefore, there are at least $n/4$ elements in $B \setminus span(S \cap B)$. Given that the weight w_i is assigned uniformly at random among $n - t$ possible elements, we get

$$\mathbf{P}_\omega\left[\omega^{-1}(w_i) \in B \setminus span(S \cap B) \mid C'_i(\omega) = S\right] \geq \frac{n/4}{n-t} \geq \frac{1}{4}.$$

Since this holds for any S such that $|S \cap B| \leq \lfloor r/4 \rfloor$, and $S \cap B = C'_i \cap B = B'_i(\omega)$, it also holds that

$$\mathbf{P}_\omega\left[\omega^{-1}(w_i) \in B \setminus span(B'_i(\omega)) \mid |B'_i(\omega)| \leq \lfloor r/4 \rfloor\right] \geq \frac{1}{4}.$$

Using Claim 2.2, we get $\mathbf{P}_\omega\left[\omega^{-1}(w_i) \in B \setminus span(B'_i(\omega))\right] \geq 1/8$. □

This finishes the proof of Theorem 2.1.

Combining our algorithm with Soto's reduction [20, Lemma 4.4], we obtain a constant-factor approximation algorithm for the matroid secretary problem in AO-RA-MK model.

Corollary 2.4. *There exists a $\frac{40}{1-1/e}$-approximation algorithm in the AO-RA-MK model.*

3 Approximation algorithms for unknown matroids

In this section we will be focusing mainly on the AO-RA-MN model. i.e. assuming that the ordering of the elements of the matroid is adversarial, weights are assigned randomly, but the matroid is unknown, and the algorithm only knows n in advance. We present an $O(\log n \log r)$ approximation algorithm for the AO-RA-MN model, where n is the number of elements in the ground set and r is the rank of the matroid. At the end of this section we also give a general framework that can turn any α approximation algorithm for the RO-AA-MN model, (i.e. the primary variant of the matroid secretary problem) into an $O(\alpha \log^{1+\epsilon} n/\epsilon)$ approximation algorithm in the RO-AA-MU model.

It is worth noting that in these models the adversary may set some of the elements of the matroid to be loops, and the algorithm does not know the number of loops in advance. For example it might be the case that after observing the first 10 elements, the rest are all loops and thus the algorithm should select at least one of the first 10 elements with some non-zero probability. This is the idea of the counterexample in section 4 (Corollary 4.4), where we reduce AO-RA-MN, AO-RA-MU models to RA-CN, RA-CU models respectively, and thus we show that there is no constant-factor approximation for either of the models. In fact, no algorithm can do better than $\Omega(\log n/\log\log n)$. Therefore our algorithms are tight within a factor of $O(\log r \log\log n)$ or $O(\log r \log^\epsilon n)$.

We use the same notation as section 2: $\mathcal{M} = (E, I)$ is a matroid of rank r (which is not known to the algorithm), and $e_1, e_2, \ldots, e_n$ is the the adversarial ordering of the elements of $\mathcal{M}$, and $W = \{w_1 > w_2 > \ldots > w_n\}$ is the set of hidden weights chosen by the adversary that are assigned to the elements of $\mathcal{M}$ via a random bijection $\omega : E \to W$.

We start by designing an algorithm for AO-RA-MN model. Our algorithm basically tries to ignore the the loops and only focuses on the non-loop elements. We design our algorithm in two phases. In the first phase we design a randomized algorithm that works even in the AO-AA-MU model assuming that it has a good estimate on the weight of the largest non-loop element. In particular, fix bijection $\omega : W \to E$, and let e_1^* be the largest non-loop element with respect ω, e_2^* be the second largest one. We assume that the algorithm knows a bound $\omega(e_2^*) < L < \omega(e_1^*)$ on the largest non-loop element in advance. We show there is a thresholding algorithm, with a *non-fixed* threshold, that achieves an $O(\log r)$ fraction of the optimum.

Algorithm 2 for AO-AA-MU model with an estimate of the largest non-loop element

Input: The bound L such that $\omega(e_2^*) < L < \omega(e_1^*)$.
Output: An independent set $\mathrm{ALG} \subseteq E$.

with probability 1/2, pick a non-loop element with weight above L and return it.
$\mathrm{ALG} \leftarrow \emptyset$ and $r^* \leftarrow 2$; set threshold $w^* \leftarrow L/2$.
for each arriving element e_i **do**
  if $\omega(e_i) > w^*$ and $ALG \cup \{e_i\}$ is independent **then**
    $\mathrm{ALG} \leftarrow \mathrm{ALG} \cup \{e_i\}$
  end if
  if $rank(\{e_1, \ldots, e_i\}) \geq r^*$ **then**
    with probability $\frac{1}{\log 2r^*}$ set $w^* \leftarrow L/2r^*$.
    $r^* \leftarrow 2r^*$.
  end if
end for
return ALG

Theorem 3.1. *For any matroid $\mathcal{M} = (E, \mathcal{I})$ of rank r, and any bijection $\omega : E \to W$, given the bound $\omega(e_2^*) < L < \omega(e_1^*)$, Algorithm 2 is a $16 \log r$ approximation in the AO-AA-MU model. i.e.*

$$\mathbf{E}\left[w(\mathrm{ALG}(\omega))\right] \geq \frac{1}{16 \log r} w(\mathrm{OPT}(\omega)),$$

where the expectation is over all of the randomization in the algorithm.

In order to solve the original problem, in the second phase we divide the non-loop elements into a set of blocks $B_1, B_2, \ldots, B_{\log n}$, and we use the previous algorithm as a module to get an $O(\log r)$ of optimum within each block.

Theorem 3.2. *For any matroid $\mathcal{M} = (E, \mathcal{I})$ of rank r, there is a polynomial time algorithm with an approximation factor $O(\log r \log n)$ in the AO-RA-MN model.*

Finally, we show how we can use essentially the same technique (decomposing the input into blocks of exponential size) to obtain an algorithm for AO-RA-MU model:

Theorem 3.3. *Let $\mathcal{M}$ be a matroid of rank r on n elements. If there is an α approximation algorithm for the matroid secretary problem on $\mathcal{M}$ in the RO-AA-MN model, then for any fixed $\epsilon > 0$, there is also an $O(\frac{\alpha}{\epsilon} \log^{1+\epsilon} n)$-approximation for the matroid secretary problem on $\mathcal{M}$ with no information given in advance (i.e., the RO-AA-MU model).*

Due to limited space all of the proofs of this section is deferred to the full version of the paper.

4 Classical secretary with unknown n

In this section, we consider a variant of the classical secretary problem where we want to select exactly one element (i.e. in matroid language, we consider a uniform matroid of rank 1). However, here we assume that the total number of elements n (which is crucial in the classical $1/e$-competitive algorithm) is not known in advance - it is chosen by an adversary who can effectively terminate the input at any point.

First, let us consider the following scenario: an upper bound N is given such that the actual number of elements on the input is guaranteed to be $n \in \{1, 2, \ldots, N\}$. The adversary can choose any n in this range and we do not learn n until we process the n-th element. (E.g., we are interviewing candidates for a position and we know that the total number of candidates is certainly not going to be more than 1000. However, we might run out of candidates at any point.) The goal is the select the highest-ranking element with a certain probability. Assuming the *comparison model* (i.e., where only the relative ranks of elements are known to the algorithm), we show that there is no algorithm achieving a constant probability of success in this case.

Theorem 4.1. *Given that the number of elements is chosen by an adversary in $\{1, \ldots, N\}$ and N is given in advance, there is a randomized algorithm which selects the best element out of the first n with probability at least $1/(H_{N-1}+1)$.*

On the other hand, there is no algorithm in this setting which returns the best element with probability more than $1/H_N$. Here, $H_N = \sum_{i=1}^{n} \frac{1}{i}$ is the N-th harmonic number.

Due to limited space, we just sketch the main ideas of the proof. Our proof is based on the method of Buchbinder et al. [6] which bounds the optimal achievable probability by a linear program. In fact the optimum of the linear program is *exactly* the optimal probability that can be achieved.

Lemma 4.2. *Given the classical secretary problem where the number of elements is chosen by an adversary from $\{1, 2, \ldots, N\}$ and N is known in advance,* *the best possible probability with which an algorithm can find the optimal element is given by*

$$\begin{aligned} &\max \qquad\qquad \alpha : \\ \forall n \le N;\ & \tfrac{1}{n}\textstyle\sum_{i=1}^{n} i p_i \ge \alpha, & (3)\\ \forall i \le N;\ & \textstyle\sum_{j=1}^{i-1} p_j + i p_i \le 1, & (4)\\ \forall i \le N;\ & \qquad p_i \ge 0. \end{aligned}$$

The only difference between this LP and the one in [6] is that we have multiple constraints (3) instead of what is the objective function in [6]. We use essentially the same proof to argue that this LP captures *exactly* the optimal probability of success α that an algorithm can achieve. For a given N, an algorithm can explicitly solve the LP given by Lemma 4.2 and thus achieve the optimal probability. Theorem 4.1 can be proved by estimating the value of this LP.

A slightly different model arises when elements arrive with (random) weights and we want to maximize the expected weight of the selected element. This model is somewhat easier for an algorithm; any algorithm that selects the best element with probability at least α certainly achieves an α-approximation in this model, but not the other way around. Given an upper bound N on the number of elements (and under a more stringent assumption that weights are chosen i.i.d. from a known distribution), by a careful choice of a probability distribution for the weights, we prove that still no algorithm can achieve an approximation factor better than an $\Omega(\log N/\log\log N)$-approximation.

Theorem 4.3. *For the classical secretary problem with random nonnegative weights drawn i.i.d. from a known distribution and the number of candidates chosen adversarially in the range* $\{1,\dots,N\}$*, no algorithm achieves a better than* $\frac{\log N}{32\log\log N}$*-approximation in expectation.*

The hard examples are constructed based on a particular exponentially distributed probability distribution. Similar constructions have been used in related contexts [15,9]. The proof is deferred to the full version of the paper. Consequently, we obtain that no algorithm can achieve an approximation factor better than $\Omega(\log N/\log\log N)$ in the AO-RA-MN model.

Corollary 4.4. *For the matroid secretary problem in the AO-RA-MN (and AO-RA-MU, RO-AA-MU) models, no algorithm can achieve a better than* $\Omega(\frac{\log N}{\log\log N})$*-approximation in expectation.*

References

1. A. R. Abdel-Hamid, J.A. Bather, G. B. Trustrum. The secretary problem with an unknown number of candidates. *J. Appl. Prob.* 19, 619–630, 1982.
2. M. Babaioff, M. Dinitz, A. Gupta, N. Immorlica and K. Talwar. Secretary problems: weights and discounts. In *SODA 2009*, 1245–1254.

3. M. Babaioff, N. Immorlica and R. Kleinberg. Matroids, secretary problems, and online mechanisms. In *SODA 2007*, 434–443.
4. M. H. Bateni, M. T. Hajiaghayi and M. Zadimoghaddam. Submodular secretary problem and extensions. In *APPROX 2010*, 39–52.
5. Peter Borosan and Mudassir Shabbir A survey of secretary problem and its extensions. Preprint, 2009, available at http://paul.rutgers.edu/~mudassir/Secretary/paper.pdf.
6. N. Buchbinder, K. Jain and M. Singh. Secretary problems via linear programming. In *IPCO 2010*, 163–176.
7. N. B. Dimitrov and C. G. Plaxton. Competitive weighted matching in transversal matroids. In *ICALP 2008*, 397–408.
8. E. B. Dynkin. The optimum choice of the instant for stopping a markov process. *Soviet Mathematics, Doklady* 4, 1963.
9. J. Feldman, M. Henzinger, N. Korula, V. S. Mirrokni and C. Stein. Online stochastic packing applied to display ad allocation. In *ESA 2010*, 182–194.
10. T.S. Ferguson. Who solved the secretary problem? *Statistical Science*, 4:3, 282–289, 1989.
11. M. Gardner. Mathematical Games column, *Scientific American*, February 1960.
12. J. Gilbert and F. Mosteller. Recognizing the maximum of a sequence. *J. Amer. Statist. Assoc.* 61:35–73, 1966.
13. A. Gupta, A. Roth, G. Schoenebeck and K. Talwar. Constrained non-monotone submodular maximization: Offline and secretary algorithms. In *WINE 2010*, 246–257.
14. M. T. Hajiaghayi, R. Kleinberg and D. Parkes. Adaptive limited-supply online auctions. In *EC 2004*, 71–80.
15. M. T. Hajiaghayi, R. Kleinberg and T. Sandholm. Automated online mechanism design and prophet inequalities. In *International Conference on Artificial Intelligence 2007*, 58–65, 2007.
16. S. Im and Y. Wang. Secretary problems: Laminar matroid and interval scheduling. In *SODA 2011*, 1265-1274.
17. D. V. Lindley. Dynamic programming and decision theory. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 10:1, 39–51, 1961.
18. R. Kleinberg. A multiple-choice secretary algorithm with applications to online auctions. In *SODA 2005*, 630–631.
19. N. Korula and M. Pál. Algorithms for secretary problems on graphs and hypergraphs. In *ICALP 2009*, 508–520.
20. J. A. Soto. Matroid secretary problem in the random assignment model. In *SODA 2011*, 1275-1284.
21. T. J. Stewart. The secretary problem with an unknown number of options. *Operations Research* 29:1, 130–145, 1981.