New Results for the k-Secretary Problem

Susanne Albers
Department of Informatics, Technical University of Munich, Germany
albers@in.tum.de

Leon Ladewig
Department of Informatics, Technical University of Munich, Germany
ladewig@in.tum.de

Abstract

Suppose that n numbers arrive online in random order and the goal is to select k of them such that the expected sum of the selected items is maximized. The decision for any item is irrevocable and must be made on arrival without knowing future items. This problem is known as the *k-secretary problem*, which includes the classical secretary problem with the special case $k = 1$. It is well-known that the latter problem can be solved by a simple algorithm of competitive ratio $1/e$ which is asymptotically optimal. When k is small, only for $k = 2$ does there exist an algorithm beating the threshold of $1/e$ [Chan et al. SODA 2015]. The algorithm relies on an involved selection policy. Moreover, there exist results when k is large [Kleinberg SODA 2005].

In this paper we present results for the k-secretary problem, considering the interesting and relevant case that k is small. We focus on simple selection algorithms, accompanied by combinatorial analyses. As a main contribution we propose a natural deterministic algorithm designed to have competitive ratios strictly greater than $1/e$ for small $k \geq 2$. This algorithm is hardly more complex than the elegant strategy for the classical secretary problem, optimal for $k = 1$, and works for all $k \geq 1$. We explicitly compute its competitive ratios for $2 \leq k \leq 100$, ranging from 0.41 for $k = 2$ to 0.75 for $k = 100$. Moreover, we show that an algorithm proposed by Babaioff et al. [APPROX 2007] has a competitive ratio of 0.4168 for $k = 2$, implying that the previous analysis was not tight. Our analysis reveals a surprising combinatorial property of this algorithm, which might be helpful for a tight analysis of this algorithm for general k.

2012 ACM Subject Classification Theory of computation → Online algorithms

Keywords and phrases Online algorithms, secretary problem, random order model

Digital Object Identifier 10.4230/LIPIcs.ISAAC.2019.18

Funding Work supported by the European Research Council, Grant Agreement No. 691672.

1 Introduction

The *secretary problem* is a well-known problem in the field of optimal stopping theory and is defined as follows: Given a sequence of n numbers which arrive online and in random order, select the maximum number. Thereby, upon arrival of an item, the decision to accept or reject it must be made immediately and irrevocably, especially without knowing future items. The statement of the problem dates back to the 1960s and its solution is due to Lindley [23] and Dynkin [10]. For discussions on the origin of the problem, we refer to the survey [13].

In the past years, generalizations of the secretary problem involving selection of multiple items have become very popular. We consider one of the most canonical generalizations known as the *k-secretary problem*: The algorithm is allowed to choose k elements and the goal is to maximize the expected sum of accepted elements. Other objective functions, such as maximizing the probability of accepting the k best [2, 14] or general submodular functions [20], have been studied as well. Maximizing the sum of accepted items is closely related to the *knapsack secretary problem* [3, 19]. If all items have unit weight and thus the knapsack capacity is a cardinality bound, the k-secretary problem arises. The *matroid*

30th International Symposium on Algorithms and Computation (ISAAC 2019).
Editors: Pinyan Lu and Guochuan Zhang; Article No. 18; pp. 18:1–18:19
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

secretary problem, introduced by Babaioff et al. [6], is a generalization where an algorithm must maintain a set of accepted items that form an independent set of a given matroid. We refer the reader to [11, 12, 22] for recent work. If the matroid is k-uniform, again, the k-secretary problem occurs. Another closely related problem was introduced by Buchbinder, Jain, and Singh [8]. In the (J, K)-*secretary problem*, an algorithm has J choices and the objective is to maximize the number of selected items among the K best. Assuming the ordinal model [17] and a monotonicity property of the algorithm, any c-competitive algorithm for the (k, k)-secretary problem is c-competitive for the k-secretary problem, and vice versa [8]. In the ordinal model [17], an algorithm decides based on the total order of items only, rather than on their numeric values. In fact, most known and elegant algorithms for the k-secretary problem assume the ordinal model [3, 10, 21, 23].

The large interest in generalizations of the classical secretary problem is motivated mainly by numerous applications in online market design [4, 6, 21]. Apart from these applications, the secretary problem is the prototype of an online problem analyzed in the random order model: An adversarial input order often rules out (good) competitive ratios when considering online optimization problems without further constraints. By contrast, the assumption that the input is ordered randomly improves the competitive ratios in many optimization problems. This includes packing problems [18, 19], scheduling problems [15], and graph problems [7, 24]. Therefore, developing new techniques for secretary problems may, more generally, yield relevant insights for the analysis of online problems in randomized input models as well.

1.1 Previous Work

The k-secretary problem was introduced by Kleinberg [21] in 2005. He presents a randomized algorithm attaining a competitive ratio of $1-5/\sqrt{k}$, which approaches 1 for $k \to \infty$. Moreover, Kleinberg gives in [21] a hardness result stating that any algorithm has a competitive ratio of $1 - \Omega(\sqrt{1/k})$. Therefore, from an asymptotic point of view, the k-secretary problem is solved by Kleinberg's result. However, the main drawback can be seen in the fact that the competitive ratio is not defined if $k \leq 24$ and breaks the barrier of $1/e$ only if $k \geq 63$ (see Figure 2, p. 11).

In 2007 the problem was revisited by Babaioff et al. [3]. The authors propose two algorithms called VIRTUAL and OPTIMISTIC and prove that both algorithms have a competitive ratio of at least $1/e$ for any k. While the analysis of VIRTUAL is simple and tight, it takes much more effort to analyze OPTIMISTIC [3, 4]. The authors believe that their analysis for OPTIMISTIC is not tight for $k \geq 2$.

Buchbinder, Jain, and Singh [8] developed a framework to analyze secretary problems and their optimal algorithms using linear programming techniques. By numerical simulations for the (k, k)-secretary problem with $n = 100$, Buchbinder et al. obtained competitive ratios of 0.474, 0.565, and 0.612, for $k = 2, 3$, and 4, respectively. However, obtaining an algorithm from their framework requires a formal analysis of the corresponding LP in the limit of $n \to \infty$, which is not provided in the article [8, p. 192].

Chan, Chen, and Jiang [9] revisited the (J, K)-secretary problem and obtained several fundamental results. Notably, they showed that optimal algorithms for the k-secretary problem require access to the numeric values of the items, which complements the previous line of research in the ordinal model. Chan et al. demonstrate this by providing a 0.4920-competitive algorithm for the 2-secretary problem which is based on a 0.4886-competitive algorithm for the $(2, 2)$-secretary problem. Still, an analysis for the general (J, K)-case is not known, even for $J = K$. Moreover, the resulting algorithms seem overly involved. This dims the prospect of elegant k-secretary algorithms for $k \geq 3$ obtained from this approach.

Table 1 Competitive ratios α of SINGLE-REF for $k \in [1..20]$.

k	1	2	3	4	5	6	7	8	9	10
α	$1/e$	0.4119	0.4449	0.4785	0.4999	0.5148	0.5308	0.5453	0.5567	0.5660
k	11	12	13	14	15	16	17	18	19	20
α	0.5740	0.5834	0.5914	0.5983	0.6043	0.6096	0.6155	0.6211	0.6261	0.6306

1.2 Our Contribution

We study the k-secretary problem, the most natural and immediate generalization of the classical secretary problem. While the extreme cases $k = 1$ and $k \to \infty$ are well studied, hardly any results for small values of $k \geq 2$ exist. We believe that simple selection algorithms, performing well for small k, are interesting both from a theoretical point of view and for practical settings. Moreover, the hope is that existing algorithms for related problems based on k-secretary algorithms can be improved this way [8, p. 191]. We study algorithms designed for the ordinal model, which guarantees robustness and plainer decision rules.

For this purpose, we propose a simple deterministic algorithm SINGLE-REF. This algorithm uses a single value as threshold for accepting items. Although similar approaches based on this natural idea have been used to solve related problems [1], to the best of our knowledge, this algorithm has not been explored for the k-secretary problem so far. As a strength of our algorithm we see its simplicity: It is of plain combinatorial nature and can be fine-tuned using only two parameters. In contrast, the optimal algorithms which follow theoretically from the (J, K)-secretary approach [9] would involve k^2 parameters and the same number of different decision rules.

An important insight for the analysis of SINGLE-REF is that items can be partitioned into two classes, which we will call *dominating* and *non-dominating*. Both have certain properties on which we base our fully parameterized analysis. In Table 1, we list the competitive ratios of SINGLE-REF for $k \leq 20$. While the competitive ratio for $k = 1$ is optimal, we obtain a value significantly greater than $1/e$ already for $k = 2$. Furthermore, the competitive ratios are monotonically increasing in the interval $k \in [1..20]$, already breaking the threshold of 0.5 at $k = 6$. Numerical computations suggest that this monotonicity holds for general k. See Figure 2 (p. 11) for the competitive ratios up to $k = 100$ and a comparison with Kleinberg's algorithm [21]. Providing a closed formula for the competitive ratio for any value of k is one direction of future work (see Section 5).

Moreover, we investigate the OPTIMISTIC algorithm by Babaioff et al. [3] for the case $k = 2$. Although Chan et al. [9] provide the optimal algorithm for $k = 2$, we think studying this elegant algorithm is interesting for two reasons: First, a tight analysis of OPTIMISTIC is stated as open problem in [3]. Article [3] does not provide the proof of the $(1/e)$-bound and a recent journal publication [5] (evolved from [3] and [6]) does not cover the OPTIMISTIC algorithm at all. We make progress in this problem by proving that for $k = 2$ its competitive ratio is exactly 0.4168 which significantly breaks the $(1/e)$-barrier. Second, our proof reveals an interesting property of this algorithm, which we show in Lemma 4.1: The probability that OPTIMISTIC accepts the second best item is exactly the probability that the optimal algorithm for $k = 1$ from [10, 23] accepts the best item. A similar property might hold for $k \geq 3$, which could be a key insight into the general case.

From a technical point of view, we derive the exact probabilities using basic combinatorial constructs exclusively. This is in contrast to previous approaches [8, 9] which can only be analyzed using heavyweight linear programming techniques. In addition, we always consider the asymptotic setting of $n \to \infty$ items, which gives more meaningful bounds on the competitive ratio. Throughout the analyses of both algorithms, we associate probabilities with sets of permutations (see Section 2.2). Hence, probability relations can be shown equivalently by set relations. This is a simple but powerful technique which may be useful in the analysis of other optimization problems with random arrival order as well.

2 Preliminaries

Let $v_1 > v_2 > \ldots > v_n$ be the *elements* (also called *items*) of the input. In the ordinal model, we can assume w.l.o.g. all items to be distinct. Therefore we say that i is the *rank* of element v_i. An *input sequence* is any permutation of the list $v_1, \ldots, v_n$. We denote the position of an element v given a specific input sequence π with $\mathrm{pos}_\pi(v) \in \{1, \ldots, n\}$ and write $\mathrm{pos}(v)$ whenever the input sequence is clear from the context.

Given any input sequence, an algorithm can accept up to k items, where the decision whether to accept or reject an item must be made immediately upon its arrival. Let ALG denote the sum of items accepted by the algorithm. The algorithm is *α-competitive* if $\mathbf{E}[\mathrm{ALG}] \geq \alpha \cdot \mathrm{OPT}$ holds for all item sets. Here the expectation is taken over the uniform distribution of all $n!$ input sequences and $\mathrm{OPT} = \sum_{i=1}^{k} v_i$.

Notation. For $a, b \in \mathbb{N}$ with $a \leq b$, we use the notation $[a..b]$ to denote the set of integers $\{a, a+1, \ldots, b\}$ and write $[a]$ for $[1..a]$. The (half-)open integer intervals $(a..b]$, $[a..b)$, and $(a..b)$ are defined accordingly. Further, we use the notation $n^{\underline{k}}$ for the falling factorial $\frac{n!}{(n-k)!}$.

2.1 Algorithms

In the following, we state the OPTIMISTIC algorithm proposed by Babaioff et al. (Algorithm 1) and our proposed algorithm SINGLE-REF (Algorithm 2) and compare both strategies.

Algorithm 1 OPTIMISTIC [3].

Parameters: $t \in (k..n-k]$ (sampling threshold)
Sampling phase: Reject the first $t-1$ items.
Let $s_1 > \ldots > s_k$ be the k best items from the sampling phase.
Selection phase: As j-th accepted item, choose the first item better than s_{k-j+1}.

Algorithm 2 SINGLE-REF.

Parameters: $t \in (k..n-k]$ (sampling threshold), $r \in [k]$ (reference rank)
Sampling phase: Reject the first $t-1$ items.
Let s_r be the r-th best item from the sampling phase.
Selection phase: Choose the first k items better than s_r.

While both algorithms consist of a sampling phase in which the first $t-1$ items are rejected, the main difference is the policy for accepting items: OPTIMISTIC uses the k best items from the sampling as reference elements. Right after the sampling phase, the first item better than s_k (the k-th best from the sampling) will be accepted. The following accepted items are chosen similarly, but with $s_{k-1}, s_{k-2}, \ldots, s_1$ as reference items. Note that this algorithm always sticks to this order of reference points, even if the first item already outperforms s_1. Hence, it is optimistic in the sense that it always expects that high-value items occur in the future.

SINGLE-REF has a simpler structure since it only uses a single item s_r from the sampling as reference point. Here, each item is compared to s_r (the r-th best from sampling), thus the first k elements better than s_r will be selected. Despite its simpler structure, the analysis of SINGLE-REF is involved due to the additional parameter r, as it is not clear how to choose this parameter optimally.

Note that in the case $k = 1$, OPTIMISTIC and SINGLE-REF (when setting $r = 1$) become the strategy known for the classical secretary problem [10, 23]: After rejecting the first $t-1$ items, choose the first one better than the best from sampling. A simple argument shows that this strategy selects the best item with probability $\frac{t-1}{n}\sum_{i=t}^{n}\frac{1}{i-1}$. If n tends to infinity and $t-1 \approx n/e$, this term approaches $1/e$ which is optimal.

The following lemma is used to bound the competitive ratios of both algorithms. It heavily relies on the monotonicity property of the algorithms, i.e., for any $v_i > v_j$, both algorithms select v_i with greater or equal probability than v_j.

▶ **Lemma 2.1.** *Let $\mathcal{A}$ be* OPTIMISTIC *or* SINGLE-REF *and for each $i \in [n]$ let p_i be the probability that $\mathcal{A}$ selects item v_i. The competitive ratio of $\mathcal{A}$ is $(1/k)\sum_{i=1}^{k} p_i$.*

Proof. First, we will argue that $p_i \geq p_{i+1}$ for all $i \in [n-1]$, i.e., $\mathcal{A}$ selects items of smaller rank with greater or equal probability. This follows if we can show that the number of permutations where v_{i+1} is accepted is not greater than the respective number of permutations for v_i (this concept is described more detailed in Section 2.2).

Consider any input sequence π in which v_{i+1} is accepted. Let $s_j < v_{i+1}$ be the sampling item to which v_{i+1} is compared (in case of SINGLE-REF we have $j = r$). Since v_{i+1} is accepted, we have $s_j \neq v_i$. By swapping v_i with v_{i+1}, we obtain a new permutation π' with the same reference element s_j. This is obvious if v_i is not in the sampling of π. Otherwise, note that in the ordered sequences of sampling items from π and π', both v_{i+1} and v_i have the same position. This implies that s_j is the j-th best sampling item in π'. Further, item v_i is at the former position of v_{i+1} in π', thus $\mathcal{A}$ accepts v_i at this position since $v_i > v_{i+1} > s_j$.

Thus, both sequences $p_1, \ldots, p_k$ and $v_1, \ldots, v_k$ are sorted decreasingly. Let $\mathrm{OPT}_k = \sum_{i=1}^{k} v_i$ and $\mathbf{E}[\mathcal{A}]$ be the expected sum of the items accepted by $\mathcal{A}$. Chebyshev's sum inequality [16] states that if $a_1 \geq a_2 \geq \ldots \geq a_n$ and $b_1 \geq b_2 \geq \ldots \geq b_n$, then $\sum_{i=1}^{n} a_i b_i \geq (1/n)\left(\sum_{i=1}^{n} a_i\right)\left(\sum_{i=1}^{n} b_i\right)$. Applying this inequality yields

$$\mathbf{E}[\mathcal{A}] = \sum_{i=1}^{n} p_i v_i \geq \sum_{i=1}^{k} p_i v_i \geq \frac{1}{k}\left(\sum_{i=1}^{k} v_i\right)\left(\sum_{i=1}^{k} p_i\right) = \left(\frac{1}{k}\sum_{i=1}^{k} p_i\right)\mathrm{OPT}_k\,.$$

Note that the above inequalities are tight: Assuming that the first k items are almost identical, i.e. $v_i = 1 - i\varepsilon$ for $i \in [1..k]$ and $\varepsilon \to 0$, and $v_i = 0$ for all remaining items of rank $i \in (k..n]$, the competitive ratio is exactly $(1/k)\sum_{i=1}^{k} p_i$. ◀

The same argument is used in [8] to show the equivalence of the k-secretary and the (k, k)-secretary problem for ordinal monotone algorithms.

2.2 Random Order Model

To analyze an algorithm given a random permutation, we often fix an order $u_1, u_2, \ldots, u_n$ of positions. Then, we draw the element for position u_1 uniformly from all n elements, next the element for position u_2 from the remaining $n-1$ elements, and so on. It is easy to see that by this process we obtain a permutation drawn uniformly at random.

Moreover, the uniform distribution allows us to prove probability relations using functions: Suppose that p_i is the probability that item v_i is accepted in a random permutation, then $p_i = |P_i|/n!$ where P_i is the set of all input sequences where v_i is accepted. Thus, we can

Table 2 Several identities involving binomial coefficients [16].

Rule	Equation	Parameters
(R1) Sum of products	$\sum_{k=0}^{l} \binom{l-k}{m}\binom{q+k}{n} = \binom{l+q+1}{m+n+1}$	$l, m, n, q \in \mathbb{Z}$ with $l, m \geq 0$ and $n \geq q \geq 0$
(R2) Symmetry	$\binom{n}{k} = \binom{n}{n-k}$	$n, k \in \mathbb{Z}$ with $n \geq 0$
(R3) Trinomial revision	$\binom{r}{m}\binom{m}{k} = \binom{r}{k}\binom{r-k}{m-k}$	$m, k \in \mathbb{Z}$ and $r \in \mathbb{R}$

prove $p_i \leq p_j$ by finding an injective function $f\colon P_i \to P_j$ and get $p_i = p_j$ if f is bijective. For example, this technique turns out to be highly useful in the proof of Lemma 4.1, where probabilities of different algorithms are related.

2.3 Combinatorics

We often need to analyze probabilities described by the following random experiment.

▶ **Fact 2.2.** *Suppose there are N balls in an urn from which M are blue and $N - M$ red. The probability of drawing K blue balls without replacement in a sequence of length K is* $h(N, M, K) := \binom{M}{K}/\binom{N}{K}$.

This fact follows from a special case of the hypergeometric distribution.

Furthermore, we make use of several identities involving binomial coefficients throughout the following sections. These equations, denoted by (R1), (R2), and (R3), are listed in Table 2.

3 Analysis of SINGLE-REF

In this section we analyze our proposed algorithm SINGLE-REF, which we denote by $\mathcal{A}$ throughout this section. Recall that this algorithm uses s_r, the r-th best sampling item, as the threshold for accepting items. As implied by the proof of Lemma 2.1, only the k largest items $v_1, \ldots, v_k$ contribute to the objective function. One essential idea of our approach is to separate the set of top-k items into two classes according to the following definition.

▶ **Definition 3.1.** *We say that item v_i is dominating if $i \leq r$, and non-dominating if $r + 1 \leq i \leq k$.*

The crucial property of dominating items becomes clear in the following scenario: Assume that any dominating item v occurs after the sampling phase. Since s_r is the r-th best item from the sampling phase, it follows that $v > s_r$. That is, each dominating item outside the sampling beats the reference item. Therefore there are only two situations when dominating items are rejected: Either they appear before position t, or after k accepted items.

3.1 Acceptance of Dominating Items

First we focus on dominating items. As we will show in Lemma 3.2, the algorithm cannot distinguish between them and thus each dominating item has equal acceptance probability.

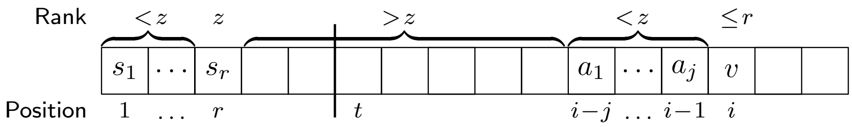

Figure 1 Event $\tilde{E}_j(z,i)$ considered in the proof of Lemma 3.2.

▶ **Lemma 3.2.** *Let v be a dominating item and $j \in [0..k)$. Let E_j be the event that $\mathcal{A}$ selects v as $(j+1)$-th item. It holds that* $\mathbf{Pr}\left[E_j\right] = \frac{\kappa\tau}{n}\sum_{i=t+j}^{n}\binom{i-t}{j}\frac{1}{(i-1)^{\underline{r+j}}}$, *where* $\tau = (t-1)^{\underline{r}}$ *and* $\kappa = (r-1+j)^{\underline{j}}$.

Proof. Let $E_j(z,i)$ be the event that $\mathcal{A}$ accepts v as $(j+1)$-th item at position $i = \text{pos}(v)$ and s_r has rank z (thus $s_r = v_z$). Note that there must be elements $s_1, \dots, s_{r-1}$ of rank smaller than z in the sampling (such that s_r is in fact the r-th best sampling element). Similarly, there must be j elements $a_1, \dots, a_j$ after the sampling but before v of rank smaller than z (which are accepted by $\mathcal{A}$).

The proof is in several steps. We first consider a stronger event $\tilde{E}_j(z,i)$. Later, we show how the probability of $E_j(z,i)$ can be obtained from $\tilde{E}_j(z,i)$. In the end, the law of total probability yields $\mathbf{Pr}\left[E_j\right]$.

Analysis of $\tilde{E}_j(z,i)$. Event $\tilde{E}_j(z,i)$ is defined as $E_j(z,i)$ with additional position constraints (see Figure 1): Elements $s_1, \dots, s_r$ are in this order at the first r positions and elements $a_1, \dots, a_j$ are in this order at the j positions immediately before v. Therefore, $\tilde{E}_j(z,i)$ occurs if and only if the following conditions hold:

(i) $\text{pos}(v) = i$, $\text{pos}(s_\ell) = \ell$ for $\ell \in [r]$, and $\text{pos}(a_m) = i - j + m - 1$ for $m \in [j]$.
(ii) Elements $s_1, \dots, s_{r-1}$ have rank smaller than z
(iii) Elements $a_1 \dots, a_j$ have rank smaller than z
(iv) All remaining items at positions $r+1, \dots, i-j-1$ have rank greater than z.

Using the concept described in Section 2.2, we think of sequentially drawing the elements for the positions $1, \dots, r, i-j, \dots, i$ and then $r+1, \dots, i-j-1$. The probability for (i) is $\prod_{\ell=0}^{j+r}\frac{1}{n-\ell} = 1/n^{\underline{j+r+1}} =: \beta$, since each item has the same probability to occur at each remaining position. In (ii), the $r-1$ elements can be chosen out of $z-2$ remaining items of rank smaller than z (since v is dominating and was already drawn). Therefore we get a factor of $\binom{z-2}{r-1}$. After this step, there remain $z-2-(r-1) = z-r-1$ elements of rank smaller than z, so we get factor $\binom{z-r-1}{j}$ for step (iii).
Finally, the probability of (iv) can be formulated using Fact 2.2. Note that at this point, there remain $n-(1+r+j)$ items and no item of rank greater than z has been drawn so far. In terms of the random experiment described in Fact 2.2, we draw $K = i-j-r-1$ balls (items) from an urn of size $N = n-(1+r+j)$ where $M = n-z$ balls are blue (rank greater than z). Hence, the probability for (iv) is $H := h(n-r-j-1, n-z, i-j-r-1)$.
Therefore we obtain

$$\mathbf{Pr}\left[\tilde{E}_j(z,i)\right] = \beta \cdot \binom{z-2}{r-1}\binom{z-r-1}{j} \cdot H\,. \tag{1}$$

This term can be simplified further by applying (R3) and (R2). Let $R = z-2$, $K = r-1$, and $M = j+r-1$. It holds that

$$\binom{z-2}{r-1}\binom{z-r-1}{j} \overset{(R3)}{=} \binom{R}{M}\binom{M}{K} \overset{(R2)}{=} \binom{R}{M}\binom{M}{M-K} = \binom{z-2}{j+r-1}\binom{j+r-1}{j}.$$

Let $\kappa = (j+r-1)^{\underline{j}}$, then $\binom{j+r-1}{j} = \kappa/j!$ and we get $\mathbf{Pr}\left[\tilde{E}_j(z,i)\right] = \frac{\beta\kappa}{j!} \cdot \binom{z-2}{j+r-1} \cdot H\,.$

Relating $\tilde{E}_j(z,i)$ to $E_j(z,i)$. In contrast to $\tilde{E}_j(z,i)$, in the event $E_j(z,i)$, the elements $s_1,\ldots,s_r$ can have any positions in $[t-1]$ and $a_1\ldots,a_j$ any positions in $[t..i)$. In the random order model, the probability of an event depends linearly on the number of permutations for which the event happens. Hence, we can multiply the probability with corresponding factors $(t-1)^{\underline{r}} =: \tau$ and $(i-t)^{\underline{j}} = \binom{i-t}{j}j!$ and get $\mathbf{Pr}\left[E_j(z,i)\right] = \binom{i-t}{j}\tau j!\cdot\mathbf{Pr}\left[\tilde{E}_j(z,i)\right]$.

Relating $E_j(z,i)$ to E_j. As the final step, we sum over all possible values for i and z to obtain $\mathbf{Pr}\left[E_j\right]$. The position i of item v ranges between $t+j$ and n, while the reference rank z is between $r+j+1$ (there are $r-1$ sampling elements and $j+1$ accepted elements of rank less than z) and n. Thus we get:

$$\begin{aligned}\mathbf{Pr}\left[E_j\right] &= \sum_{i=t+j}^{n}\sum_{z=r+j+1}^{n}\mathbf{Pr}\left[E_j(z,i)\right] = \tau j!\sum_{i=t+j}^{n}\binom{i-t}{j}\sum_{z=r+j+1}^{n}\mathbf{Pr}\left[\tilde{E}_j(z,i)\right]\\ &= \beta\kappa\tau\sum_{i=t+j}^{n}\binom{i-t}{j}\sum_{z=r+j+1}^{n}\binom{z-2}{j+r-1}\cdot H\\ &= \beta\kappa\tau\sum_{i=t+j}^{n}\binom{i-t}{j}\frac{1}{\binom{n-r-j-1}{i-j-r-1}}\sum_{z=r+j+1}^{n}\binom{z-2}{j+r-1}\binom{n-z}{i-j-r-1}, \qquad (2)\end{aligned}$$

where the last step follows from Fact 2.2. The sum over z in Equation (2) can be resolved using (R1). Let $L=n-r-j-1$, $N=Q=r+j-1$, and $M=i-j-r-1$. Then we have

$$\begin{aligned}&\sum_{z=r+j+1}^{n}\binom{z-2}{j+r-1}\binom{n-z}{i-j-r-1} = \sum_{z=0}^{n-r-j-1}\binom{r+j-1+z}{j+r-1}\binom{n-r-j-1-z}{i-j-r-1}\\ &= \sum_{z=0}^{L}\binom{Q+z}{N}\binom{L-z}{M} = \binom{L+Q+1}{M+N+1} = \binom{n-1}{i-1}. \qquad (3)\end{aligned}$$

Note that in order to apply (R1) we need to verify $L, M\geq 0$ and $N\geq Q\geq 0$. We can assume $k\leq n/2$, since for $k>n/2$, there exist a trivial (1/2)-competitive algorithm. Therefore, we have $L=n-r-j-1\geq n-k-(k-1)-1 = n-2k\geq 0$. Further, $i\geq t+j$, thus $i-j\geq t\geq k+1\geq r+1$ which implies $M\geq 0$. The condition $N\geq Q\geq 0$ holds trivially. By inserting Equation (3) into Equation (2), we obtain the quotient of binomial coefficients $\binom{n-1}{i-1}/\binom{n-r-j-1}{i-j-r-1}$. From (R3) we get

$$\binom{n-1}{i-1}\Big/\binom{n-1-(r+j)}{i-1-(r+j)} = \binom{n-1}{r+j}\Big/\binom{i-1}{r+j} = \frac{(n-1)^{\underline{r+j}}}{(i-1)^{\underline{r+j}}}.$$

Recall $\beta = 1/n^{\underline{j+r+1}}$, thus $(n-1)^{\underline{r+j}}\cdot\beta = 1/n$. Together with Equation (2) we get

$$\mathbf{Pr}\left[E_j\right] = \beta\kappa\tau\cdot(n-1)^{\underline{r+j}}\sum_{i=t+j}^{n}\binom{i-t}{j}\frac{1}{(i-1)^{\underline{r+j}}} = \frac{\kappa\tau}{n}\sum_{i=t+j}^{n}\binom{i-t}{j}\frac{1}{(i-1)^{\underline{r+j}}}, \qquad (4)$$

which concludes the proof. ◀

Lemma 3.2 provides the exact probability that a dominating item is accepted as $(j+1)$-th item. However, it is more meaningful to consider the asymptotic setting where $n\to\infty$. Here, we assume $t-1=cn$ for some constant $c\in(0,1)$. For this setting, we obtain the following lemma.

▶ **Lemma 3.3.** *Let E_j be defined as in Lemma 3.2. In the asymptotic setting described above,*

(A) *For $r = 1$ it holds that $\mathbf{Pr}[E_j] = c\left(\ln\frac{1}{c} + \sum_{\ell=1}^{j}\beta_\ell\frac{c^\ell-1}{\ell}\right)$, where $\beta_\ell = (-1)^{\ell+1}\binom{j}{\ell}$ for $\ell \in [j]$.*

(B) *For $r \geq 2$ it holds that $\mathbf{Pr}[E_j] = \frac{c}{r-1} - \frac{c^r(1-c)^j}{r-1}\sum_{\ell=0}^{j}\alpha_\ell\left(\frac{c}{1-c}\right)^\ell$, where $\alpha_\ell = \binom{j+r-1}{\ell+r-1}$ for $\ell \in [0..j]$.*

The proof of Lemma 3.3 relies on a sequence of technical lemmas and is given in Appendix A.

▶ Remark. As described in Section 2.1, SINGLE-REF generalizes the optimal strategy for the secretary problem ($k = 1$). Note that the combinatorial analysis from Lemma 3.2 as well as the asymptotic bound from Lemma 3.3 give exactly the respective terms from the secretary problem. To see this, we set $r = 1$ and consider the probability that the dominating item v_1 is accepted as first item. By Lemma 3.2 (with $j = 0$), the success probability is $\frac{t-1}{n}\sum_{i=t}^{n}\frac{1}{i-1}$. Moreover, Lemma 3.3(A) provides the asymptotic bound of $c\ln(1/c)$ for this case.

3.2 Non-Dominating Items

It remains to consider the acceptance probabilities of the non-dominating items $v_{r+1}, \ldots, v_k$. Fortunately, there exist some interesting connections to the probabilities for dominating items.

▶ **Lemma 3.4.** *Let $i \in [1..k-r]$ and $j \in [1..i]$. For the non-dominating item v_{r+i} it holds that $\mathbf{Pr}[v_{r+i}$ is j-th accept$] = \mathbf{Pr}[v_{r+i}$ is $(i+1)$-th accept$]$.*

Proof. First we argue that there are in total at least $i + 1$ accepts if v_{r+i} is accepted. Assuming that v_{r+i} is accepted, we have $s_r < v_{r+i}$. Let $\mathcal{S}$ be the set of elements which the algorithm may accept, i.e. $\mathcal{S} = \{v_1, \ldots, v_{r+i}\}$. Since s_r is the r-th best element in the sampling, at most $r - 1$ elements from $\mathcal{S}$ can be part of the sampling and thus at least $r + i - (r - 1) = i + 1$ elements from $\mathcal{S}$, including v_{r+i}, are accepted.

As described in Section 2.2, we construct a bijective function $f: P \to Q$ where P (resp. Q) is the set of permutations where v_{r+i} is the j-th (resp. $(i+1)$-th) accept. For each input sequence $\pi \in P$, let $a_1, \ldots, a_{i+1}$ with $a_j = v_{r+i}$ denote the first $i + 1$ accepts. The function f swaps the positions of $a_1, \ldots, a_{i+1}$ in a cyclic shift, such that $a_j = v_{r+i}$ is at the former position of a_{i+1}. In other words, the relative order of the first $i + 1$ accepted elements in $f(\pi)$ is changed in a way that v_{r+i} is the $(i + 1)$-th accept in $f(\pi)$. Note that the cyclic shift can be reversed, thus f is bijective. ◀

While Lemma 3.4 relates the acceptance probabilities of a single non-dominating item, the claim of Lemma 3.5 is in a way orthogonal by relating probabilities of non-dominating items to those for dominating items.

▶ **Lemma 3.5.** *Let $i \in [1..k-r]$ and $j \in [1..k-i]$. For the non-dominating item v_{r+i} and any dominating item v^+ it holds that $\mathbf{Pr}[v_{r+i}$ is $(i+j)$-th accept$] = \mathbf{Pr}[v^+$ is $(i+j)$-th accept$]$.*

Proof. Let P be the set of permutations where v_{r+i} is the $(i+j)$-th accept and let Q contain those where v^+ is the $(i + j)$-th accept. We prove the claim by defining a bijective function $f: P \to Q$. Let f be the function that swaps v_{r+i} with v^+ in the input sequence.

Consider any input sequence $\pi \in P$. As v_{r+i} is accepted, $s_r < v_{r+i}$. We can argue that in $f(\pi)$ element s_r is still the r-th best element of the sampling: This holds clearly if no item is moved out of or into the sampling. Otherwise, f moves v_{r+i} into the sampling and v^+ outside. But since $s_r < v_{r+i} < v^+$, this does not change the role of s_r as the r-th best sampling element. Thus f is injective.

To prove that f is surjective, let $\pi' \in Q$ be any input sequence where v^+ is the $(i+j)$-th accept. We next consider the rank z of $s_r = v_z$. As there must be sampling elements $s_1, \ldots, s_{r-1}$ and accepted elements $a_1, \ldots, a_{i+j-1}, v^+$ of rank smaller than z, we have $z > (r-1) + (i+j-1) + 1 \geq r+i$. Hence, $s_r < v_{r+i}$. The inverse function of f consists in swapping back v^+ with v_{r+i}. For the same reason as above, this maintains s_r. As $s_r < v_{r+i}$, element v_{r+i} gets accepted, thus $f^{-1}(\pi') \in P$. ◀

Using the previous results for dominating and non-dominating items we are now ready to state the main result of this section, namely the competitive ratio of SINGLE-REF. Due to the complex expressions from Lemma 3.3 we give numerical results for small values of k.

▶ **Theorem 3.6.** *In the asymptotic setting of $n \to \infty$ and assuming that $t - 1 = cn$ for a constant $c \in (0, 1)$,* SINGLE-REF *achieves the competitive ratios given in Table 1.*

Proof. For an item v_i, let $p_i^{(j)}$ be the probability that v_i is the j-th accept (with $1 \leq j \leq k$). The total acceptance probability of v_i is denoted by $p_i = \sum_{j=1}^{k} p_i^{(j)}$. According to Lemma 3.2, each dominating item has the same acceptance probability for a fixed acceptance position. Therefore, in the following we simply write p_1 (resp. $p_1^{(j)}$) for the acceptance probability of any dominating item.

By Lemma 2.1 the competitive ratio can be obtained by summing over the acceptance probabilities of all items divided by k. Clearly, $\sum_{i=1}^{r} p_i = rp_1$. Now consider any non-dominating item v_{r+i}. According to Lemmas 3.4 and 3.5, p_{r+i} can be related to respective probabilities $p_1^{(j)}$: It holds that $p_{r+i}^{(j)} = p_1^{(z)}$ with $z = \max\{j, i+1\}$. Therefore $p_{r+i} = \sum_{j=1}^{k} p_{r+i}^{(j)} = \sum_{j=1}^{i} p_1^{(i+1)} + \sum_{j=i+1}^{k} p_1^{(j)} = ip_1^{(i+1)} + \sum_{j=i+1}^{k} p_1^{(j)}$. Hence, we obtain the competitive ratio

$$\frac{1}{k} \sum_{i=1}^{k} p_i = \frac{1}{k} \left(rp_1 + \sum_{i=1}^{k-r} \left(ip_1^{(i+1)} + \sum_{j=i+1}^{k} p_1^{(j)} \right) \right) \tag{5}$$

with $p_1^{(j)} = \mathbf{Pr}\left[E_{j-1}\right]$ for the event E_j considered in Lemmas 3.2 and 3.3. To evaluate the performance of our algorithm, we maximized Equation (5) over the parameters r and c using a computer algebra system. This yields the competitive ratios shown in Table 1 (p. 3). ◀

For completeness, we evaluated the competitive ratio of SINGLE-REF in the interval $k \in [1..100]$ using the optimization procedure mentioned in the previous proof. Figure 2 shows the performance of SINGLE-REF in comparison with Kleinberg's result [21]; our algorithm reaches competitive ratios of up to 0.75 and outperforms the algorithm from [21] on this interval. In Appendix A, we provide the full list of optimal parameters for $k \in [1..100]$ (see Table 3, p. 19).

4 Analysis of OPTIMISTIC for $k = 2$

In this section we sketch the analysis of OPTIMISTIC for $k = 2$. Due to space constraints, for some proofs we refer to the full version of this paper. Let $\mathcal{A}_2$ denote OPTIMISTIC algorithm with $k = 2$ in the following. As implied by Lemma 2.1 the competitive ratio is determined by p_1 and p_2, the probabilities that $\mathcal{A}_2$ accepts v_1 and v_2, respectively. To find these probabilities, we make use of the relation between probabilities and sets (see Section 2.2). Let P_i be the set of permutations in which $\mathcal{A}_2$ accepts v_i.

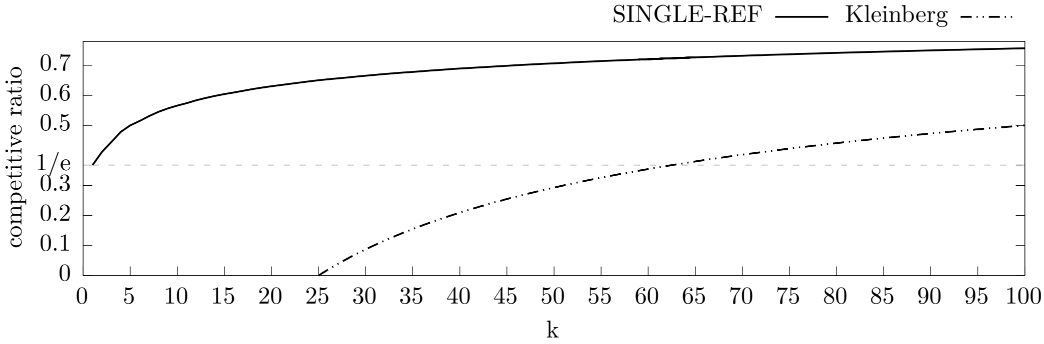

Figure 2 Comparison of our algorithm SINGLE-REF and the algorithm by Kleinberg [21], for $k \in [1..100]$. The parameters r and c for SINGLE-REF are chosen optimally.

Probability p_2. In the next lemma, we show a surprising relation between OPTIMISTIC for $k = 2$ and the algorithm for the classical (1-)secretary problem (see Section 2.1). The proof of Lemma 4.1 uses a sophistically tailored bijection between two respective sets of permutations. We sketch the proof method here and give the entire proof in the full version of this paper.

▶ **Lemma 4.1.** *Let $\mathcal{A}_1$ be the algorithm for the classical secretary problem. Assuming that both algorithms $\mathcal{A}_1$, $\mathcal{A}_2$ are parameterized with the same t, we have that $p_2 = \mathbf{Pr}[\mathcal{A}_2$ accepts $v_2] = \mathbf{Pr}[\mathcal{A}_1$ accepts $v_1]$.*

Sketch of proof. Equivalently, we prove that the corresponding complementary events happen with the same probability. For this purpose, we define for each permutation π where $\mathcal{A}_2$ does not accept v_2 a unique permutation $f(\pi)$ where $\mathcal{A}_1$ does not accept v_1. Different situations where $\mathcal{A}_2$ does not accept v_2 lead to a total number of five cases. If v_2 is in the sampling of π, we define $f(\pi)$ such that the positions of v_1 and v_2 are swapped. Then, $\mathcal{A}_1$ clearly does not accept v_1 in $f(\pi)$. Another case is when v_2 comes behind two accepted elements a_1, a_2 in π and $v_1 = a_1$ is the first accept. Note that since a_2 is accepted, $a_2 > s_1$. In this case, $f(\pi)$ can be defined by swapping the positions of both accepts v_1 and a_2. Recall that $\mathcal{A}_1$ accepts the first item better than s_1 following the sampling phase which is a_2 in $f(\pi)$, thus v_1 is not selected.

In the full proof, we consider all five cases according to π. In each case it is enough to define f such that the positions of at most three elements are swapped. Finally, we have to argue that the function f is indeed bijective. ◀

Probability p_1. In this part, we argue that $p_1 = p_2 + \delta$ holds for some $\delta > 0$. To obtain δ, we again consider cardinalities of sets instead of probabilities. First, we observe that P_2 can be related to a set $P_1' \subset P_1$ such that P_2 and P_1' have equal size.

▶ **Lemma 4.2.** *Let $P_1' = \{\pi \in P_1 \mid \mathrm{pos}_\pi(v_2) < t \Rightarrow \mathcal{A}_2$ accepts v_1 as first item$\}$. It holds that $|P_1'| = |P_2|$.*

Proof. Let $f\colon P_2 \to P_1'$ be the function that swaps v_1 with v_2 in the given sequence. We first have to argue that in fact $f\colon P_2 \to P_1'$, therefore let $\pi \in P_2$ be given. Then, v_1 gets accepted by $\mathcal{A}_2$ in $f(\pi)$ at the position $\mathrm{pos}_{f(\pi)}(v_1) = \mathrm{pos}_\pi(v_2)$, as v_1 is an item of higher value. So far we have $f(\pi) \in P_1$. If $\mathrm{pos}_{f(\pi)}(v_2) \geq t$, there is nothing to show. Assuming that $\mathrm{pos}_{f(\pi)}(v_2) < t$, it follows $\mathrm{pos}_\pi(v_1) < t$, i.e. v_1 was the best element in the sampling of π. Since no item (particularly not v_2) can beat v_1, but v_2 was accepted by $\mathcal{A}_2$ in π, we get that v_2 was the first accept in π. Hence v_1 is the first accept in $f(\pi)$.

Clearly, f is injective. For surjectivity, let $\pi' \in P_1'$ and let π the permutation obtained from π' by swapping (back) v_1 with v_2. If $\mathrm{pos}_{\pi'}(v_2) < t$, by definition of P_1' we know that v_1 is the first accept in π', implying that no item before $\mathrm{pos}_{\pi'}(v_1) = \mathrm{pos}_{\pi}(v_2)$ is chosen by $\mathcal{A}_2$. In the case $\mathrm{pos}_{\pi'}(v_2) \geq t$, since $\mathrm{pos}_{\pi'}(v_1) \geq t$, the smallest rank in the sampling of π' is 3 or greater. Therefore, v_2 gets accepted if not more than one item before v_2 gets accepted. This is the case in π, as $\mathrm{pos}_{\pi}(v_2) = \mathrm{pos}_{\pi'}(v_1)$. ◀

Since $|P_1| = |P_1'| + |P_1 \setminus P_1'| = |P_2| + |P_1 \setminus P_1'|$, we therefore get $\delta = |P_1 \setminus P_1'|/n!$, i.e., δ is the probability that a random permutation is in the set $|P_1 \setminus P_1'|$. This probability is considered in Lemma 4.3.

▶ **Lemma 4.3.** *Let $\delta = \mathbf{Pr}[\pi \in P_1 \setminus P_1']$ where π is drawn uniformly from the set of all permutations and P_1' is defined like in Lemma 4.2. It holds that $\delta = \frac{t-1}{n}\frac{t-2}{n-1}\sum_{i=t}^{n-1}\frac{n-i}{(i-2)(i-1)}$.*

The proof of Lemma 4.3 relies on a counting argument similar to the proof of Lemma 3.2. We prove Lemma 4.3 in the full version of this paper.

Competitive ratio. From Lemmas 4.1 and 4.3, we know the exact probabilities p_2 and p_1. For particular n, the term $(p_1 + p_2)/2$ can be optimized over t to find the optimal sampling size. In the following theorem we consider the asymptotic setting $n \to \infty$. Here, we assume that the sampling size is a constant fraction of the input size, i.e., $t - 1 = cn$ for some constant $c \in (0, 1)$.

▶ **Theorem 4.4.** *For $k = 2$, the algorithm* OPTIMISTIC *is 0.4168-competitive in the limit $n \to \infty$ and assuming that the sampling size is $t - 1 = cn$ for $c = 0.3521$.*

Proof. According to Lemma 4.1, p_2 is the probability that the classical secretary algorithm accepts the best item, i.e., $p_2 = \frac{t-1}{n}\sum_{i=t}^{n}\frac{1}{i-1}$. This term approaches $c\ln(1/c)$ asymptotically. From Lemma 4.3 we know $p_1 = p_2 + \delta$, where $\delta = \frac{t-1}{n}\frac{t-2}{n-1}\sum_{i=t}^{n-1}\frac{n-i}{(i-2)(i-1)}$. For $n \to \infty$, the sum $\sum_{i=t}^{n-1}\frac{n-i}{(i-2)(i-1)}$ is bounded from above and below by $\frac{1}{c} - \ln\frac{1}{c} - 1$. This can be seen by bounding the sum by two corresponding integrals. Further, $\lim_{n\to\infty}\frac{t-1}{n}\frac{t-2}{n-1} = c^2$. Therefore, $\delta = c^2\left(\frac{1}{c} - \ln\frac{1}{c} - 1\right)$ for large n. According to Lemma 2.1, $\mathcal{A}_2$ is $\alpha(c)$-competitive with

$$\alpha(c) = \frac{1}{2}(p_1 + p_2) = \frac{1}{2}(p_2 + \delta + p_2) = c\ln\frac{1}{c} + \frac{c^2}{2}\left(\frac{1}{c} - \ln\frac{1}{c} - 1\right).$$

Setting $c = 1/e$, we obtain a competitive ratio of $\alpha(1/e) = \frac{3e-2}{2e^2} \approx 0.4164$. However, the optimal choice for c is around $c^* = 0.3521 < 1/e$, improving the competitive ratio slightly to $\alpha(c^*) \approx 0.4168$. ◀

5 Conclusion and Future Work

We investigated two algorithms for the k-secretary problem with a focus on small values for $k \geq 2$. Aside from a tight analysis of the OPTIMISTIC algorithm [3] for $k = 2$, we introduced and analyzed the algorithm SINGLE-REF. For any value of k, the competitive ratio of SINGLE-REF can be obtained by numerical optimization.

We see various directions of future work. For SINGLE-REF, it remains to find the right dependency between the parameters r, c, and k in general and to find a closed formula for the competitive ratio for any value of k. OPTIMISTIC seems a promising and elegant algorithm, however no tight analysis for general $k \geq 3$ is known so far. For $k = 2$, we identified a key property in Lemma 4.1. Similar properties may hold in the general case. Lastly, to the best of our knowledge, no hardness results for the k-secretary problem are known (apart from the cases $k \leq 2$).

References

1 S. Agrawal, Z. Wang, and Y. Ye. A Dynamic Near-Optimal Algorithm for Online Linear Programming. *Operations Research*, 62(4):876–890, 2014.
2 M. Ajtai, N. Megiddo, and O. Waarts. Improved algorithms and analysis for secretary problems and generalizations. *SIAM Journal on Discrete Mathematics*, 14(1):1–27, 2001.
3 M. Babaioff, N. Immorlica, D. Kempe, and R. Kleinberg. A Knapsack Secretary Problem with Applications. In *Proc. 10th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems and 11th International Workshop on Randomization and Computation (APPROX/RANDOM)*, pages 16–28, 2007.
4 M. Babaioff, N. Immorlica, D. Kempe, and R. Kleinberg. Online auctions and generalized secretary problems. *SIGecom Exchanges*, 7(2), 2008.
5 M. Babaioff, N. Immorlica, D. Kempe, and R. Kleinberg. Matroid Secretary Problems. *Journal of the ACM (JACM)*, 65(6):35:1–35:26, 2018.
6 M. Babaioff, N. Immorlica, and R. Kleinberg. Matroids, secretary problems, and online mechanisms. In *Proc. 18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 434–443, 2007.
7 B. Bahmani, A. Mehta, and R. Motwani. A 1.43-Competitive Online Graph Edge Coloring Algorithm in the Random Order Arrival Model. In *Proc. 21st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 31–39, 2010.
8 N. Buchbinder, K. Jain, and M. Singh. Secretary Problems via Linear Programming. *Mathematics of Operations Research*, 39(1):190–206, 2014.
9 T.-H. H. Chan, F. Chen, and S. H.-C. Jiang. Revealing Optimal Thresholds for Generalized Secretary Problem via Continuous LP: Impacts on Online K-Item Auction and Bipartite K-Matching with Random Arrival Order. In *Proc. 26th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1169–1188, 2015.
10 E. B Dynkin. The optimum choice of the instant for stopping a Markov process. *Soviet Mathematics*, 4:627–629, 1963.
11 M. Feldman, O. Svensson, and R. Zenklusen. A Simple $O(\log\log(\text{rank}))$-Competitive Algorithm for the Matroid Secretary Problem. In *Proc. 26th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1189–1201, 2015.
12 M. Feldman, O. Svensson, and R. Zenklusen. A Framework for the Secretary Problem on the Intersection of Matroids. In *Proc. 29th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 735–752, 2018.
13 T. S. Ferguson. Who Solved the Secretary Problem? *Statistical Science*, 4(3):282–289, 1989.
14 P.R. Freeman. The secretary problem and its extensions: A review. *International Statistical Review/Revue Internationale de Statistique*, pages 189–206, 1983.
15 O. Göbel, T. Kesselheim, and A. Tönnis. Online Appointment Scheduling in the Random Order Model. In *Proc. 23rd Annual European Symposium on Algorithms (ESA)*, pages 680–692, 2015.
16 R. L. Graham, D. E. Knuth, and O. Patashnik. *Concrete mathematics - a foundation for computer science (2. ed.)*. Addison-Wesley, 1994.
17 M. Hoefer and B. Kodric. Combinatorial Secretary Problems with Ordinal Information. In *44th International Colloquium on Automata, Languages, and Programming (ICALP)*, pages 133:1–133:14, 2017.
18 C. Kenyon. Best-Fit Bin-Packing with Random Order. In *Proc. 7th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 359–364, 1996.
19 T. Kesselheim, K. Radke, A. Tönnis, and B. Vöcking. Primal beats dual on online packing LPs in the random-order model. In *Proc. 46th Annual ACM Symposium on Theory of Computing (STOC)*, pages 303–312, 2014.
20 T. Kesselheim and A. Tönnis. Submodular Secretary Problems: Cardinality, Matching, and Linear Constraints. In *Proc. 20th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems and 21st International Workshop on Randomization and Computation (APPROX/RANDOM)*, pages 16:1–16:22, 2017.

21 R. D. Kleinberg. A multiple-choice secretary algorithm with applications to online auctions. In *Proc. 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 630–631, 2005.

22 O. Lachish. O(log log Rank) Competitive Ratio for the Matroid Secretary Problem. In *Proc. 55th IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, pages 326–335, 2014.

23 D. V Lindley. Dynamic programming and decision theory. *Applied Statistics*, pages 39–51, 1961.

24 M. Mahdian and Q. Yan. Online bipartite matching with random arrivals: an approach based on strongly factor-revealing LPs. In *Proc. 43rd ACM Symposium on Theory of Computing (STOC)*, pages 597–606, 2011.

A Technical Proofs for SINGLE-REF

In several lemmas we need to find closed expressions for sums over values of a certain function. If the function is monotone, such sums can be bounded by corresponding integrals:

▶ **Fact A.1.** *Let $f\colon \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ and $a, b \in \mathbb{N}$.*

(A) *If f is monotonically decreasing, then $\int_a^{b+1} f(i)\ \mathrm{d}i \ \leq \sum_{i=a}^{b} f(i) \leq \int_{a-1}^{b} f(i)\ \mathrm{d}i$.*

(B) *If f is monotonically increasing, then $\int_{a-1}^{b} f(i)\ \mathrm{d}i \ \leq \sum_{i=a}^{b} f(i) \leq \int_{a}^{b+1} f(i)\ \mathrm{d}i$.*

In Lemma 3.3 we consider the acceptance probabilities of dominating items in the asymptotic setting $n \to \infty$ with $t-1 = cn$ for $c \in (0,1)$. We can assume further that $j, r \leq k = o(n)$. In the following, we prove Lemma 3.3 using some technical lemmas, stated and proven below the main proof.

Proof of Lemma 3.3. We first consider the sum $S := \sum_{i=t+j}^{n} \binom{i-t}{j} \frac{1}{(i-1)^{\underline{r+j}}}$ from Equation (4) and obtain the following lower bound:

$$S = \sum_{i=t+j}^{n} \binom{i-t}{j} \frac{1}{(i-1)^{\underline{r+j}}} = \frac{1}{j!} \sum_{i=t+j}^{n} \frac{(i-t)^{\underline{j}}}{(i-1)^{\underline{r+j}}} \geq \frac{1}{j!} \sum_{i=t+j}^{n} \frac{(i-t-j+1)^{j}}{(i-1)^{r+j}}$$
$$= \frac{1}{j!} \sum_{i=1}^{n-t-j+1} \frac{i^{j}}{(i+t+j-2)^{r+j}} .$$

Let $f(i) = i^j/(i+y)^{r+j}$ for $y = t+j-2$. Note that y can be seen as a constant independent from i. Let $m = n-t-j+1$, now the above inequality reads as $S \geq (1/j!) \sum_{i=1}^{m} f(i)$. In the following we investigate the function f.

Unfortunately, f is in general not monotone, hence we can not apply Fact A.1A or Fact A.1B directly in order to bound the sum by an integral. However, we can split the sum into two monotone parts. Let d be defined like in Lemma A.2 (following this proof). Now we can apply Fact A.1 as follows:

$$\sum_{i=1}^{m} f(i) = \sum_{i=1}^{d} f(i) + \sum_{i=d+1}^{m} f(i) \geq \int_0^{d} f(i)\ \mathrm{d}i \ + \int_{d+1}^{m+1} f(i)\ \mathrm{d}i$$
$$= \int_0^{m+1} f(i)\ \mathrm{d}i \ - \int_{d}^{d+1} f(i)\ \mathrm{d}i \ . \tag{6}$$

Finding the indefinite integral $\int f(i)\ \mathrm{d}i$ turns out to be a technical task and is therefore moved to separate lemmas (see Lemmas A.3 and A.4). If $F(i)$ is a function with $F'(i) = f(i)$, we have for κ, τ defined like in Equation (4)

$$\mathbf{Pr}\left[E_j\right] = \frac{\kappa\tau}{n} S \geq \frac{\kappa\tau}{nj!}\left(F(m+1) - F(0) - F(d+1) + F(d)\right) . \tag{7}$$

In the remainder of the proof we consider the two cases $r = 1$ and $r \geq 2$ separately.

Case A: $r = 1$**.** Let $F(i)$ and β_ℓ be defined like in Lemma A.3. In Equation (7), the factor $\frac{\kappa\tau}{nj!}$ resolves to c as $\kappa = (j+r-1)^{\underline{j}} = j^{\underline{j}} = j!$ and $\tau = (t-1)^{\underline{r}} = (t-1)^{\underline{1}} = t-1$. Further it holds that

$$\begin{aligned}\lim_{n\to\infty} F(m+1) &= \lim_{n\to\infty}\left(\ln((m+1)+y) + \sum_{\ell=1}^{j} \beta_\ell \frac{y^\ell}{\ell((m+1)+y)^\ell}\right) \\ &= \lim_{n\to\infty}\left(\ln n + \sum_{\ell=1}^{j}\beta_\ell \frac{(t+j-2)^\ell}{\ell n^\ell}\right) = \lim_{n\to\infty}\left(\ln n + \sum_{\ell=1}^{j}\beta_\ell\frac{c^\ell}{\ell}\right)\end{aligned}$$

and moreover

$$\begin{aligned}\lim_{n\to\infty} F(0) &= \lim_{n\to\infty}\left(\ln y + \sum_{\ell=1}^{j}\beta_\ell \frac{y^\ell}{\ell y^\ell}\right) = \lim_{n\to\infty}\left(\ln(t+j-2) + \sum_{\ell=1}^{j}\beta_\ell\frac{1}{\ell}\right) \\ &= \lim_{n\to\infty}\left(\ln t + \sum_{\ell=1}^{j}\beta_\ell \frac{1}{\ell}\right) .\end{aligned}$$

Hence, $\lim\limits_{n\to\infty}\left(F(m+1) - F(0)\right) = \ln\frac{1}{c} + \sum_{\ell=1}^{j}\beta_\ell\frac{c^\ell-1}{\ell}$. It remains to consider $F(d) - F(d+1)$ in the limit of $n \to \infty$. It holds that

$$\begin{aligned}F(d) - F(d+1) &= \ln(d+y) + \sum_{\ell=1}^{j}\beta_\ell\frac{y^\ell}{\ell(d+y)^\ell} - \ln(d+1+y) - \sum_{\ell=1}^{j}\beta_\ell\frac{y^\ell}{\ell(d+1+y)^\ell} \\ &= \ln\left(\frac{d+y}{d+1+y}\right) + \sum_{\ell=1}^{j}\frac{\beta_\ell}{\ell}\left(\left(\frac{y}{d+y}\right)^\ell - \left(\frac{y}{d+1+y}\right)^\ell\right)\end{aligned}$$

and since $y = t+j-2 = \Theta(n)$ and $d = (j/r)y = \Theta(y)$, we get that $\lim\limits_{n\to\infty}\left(F(d) - F(d+1)\right) = 0$.

Case B: $r \geq 2$**.** In this case let $F(i)$ and α_ℓ be defined according to Lemma A.4. Further, let $G(i) = -\alpha_0(r-1)F(i)$. Using Equation (7) we obtain

$$\begin{aligned}&= \frac{\kappa\tau}{nj!\alpha_0(r-1)}\left(G(0) - G(m+1) + G(d+1) - G(d)\right) \\ &= \frac{\tau}{n(r-1)}\left(G(0) - G(m+1) + G(d+1) - G(d)\right) ,\end{aligned} \tag{8}$$

where the last equality follows from the definition of $\alpha_0 = \binom{j+r-1}{r-1} = \kappa/j!$. We first notice

$$\lim_{n\to\infty}\frac{\tau}{n(r-1)} = \frac{1}{r-1}\lim_{n\to\infty}\frac{(t-1)^{\underline{r}}}{n} = \frac{1}{r-1}\lim_{n\to\infty}\frac{(t-1)^r}{n} = \frac{1}{r-1}c^r\lim_{n\to\infty}n^{r-1} .$$

Further it holds that

$$G(m+1) = \frac{\sum_{\ell=0}^{j}\alpha_\ell(m+1)^{j-\ell}(t+j-2)^\ell}{(m+1+t+j-2)^{r+j-1}} = \frac{\sum_{\ell=0}^{j}\alpha_\ell(m+1)^{j-\ell}(t+j-2)^\ell}{n^{r+j-1}} .$$

Note that $\lim_{n\to\infty}(m+1) = \lim_{n\to\infty}(n-(t-1)) = \lim_{n\to\infty}(n-cn) = \lim_{n\to\infty}(1-c)n$ and similarly $\lim_{n\to\infty}(t+j-2) = \lim_{n\to\infty}(t-1) = \lim_{n\to\infty} cn$. Hence we get

$$\lim_{n\to\infty} G(m+1) = \lim_{n\to\infty} \frac{\sum_{\ell=0}^{j} \alpha_\ell (1-c)^{j-\ell} n^{j-\ell} c^\ell n^\ell}{n^{r+j-1}} = \lim_{n\to\infty} \frac{\sum_{\ell=0}^{j} \alpha_\ell (1-c)^{j-\ell} c^\ell}{n^{r-1}} .$$

For the term $G(0)$ we obtain

$$G(0) = \frac{\sum_{\ell=0}^{j} \alpha_\ell 0^{j-\ell} y^\ell}{y^{r+j-1}} = \frac{\alpha_j y^j}{y^{r+j-1}} = \frac{1}{y^{r-1}}$$

and thus $\lim_{n\to\infty} G(0) = \lim_{n\to\infty} \frac{1}{y^{r-1}} = \lim_{n\to\infty} \frac{1}{(t-1)^{r-1}} = \frac{1}{c^{r-1}} \lim_{n\to\infty} \frac{1}{n^{r-1}}$.
In Equation (8) it remains to consider $G(d+1) - G(d)$. Similarly to case A we can show that this term approaches 0 for $n \to \infty$:

$$\begin{aligned} G(d+1) - G(d) &= \frac{\sum_{\ell=0}^{j} \alpha_\ell (d+1)^{j-\ell} y^\ell}{(d+1+y)^{r+j-1}} - \frac{\sum_{\ell=0}^{j} \alpha_\ell d^{j-\ell} y^\ell}{(d+y)^{r+j-1}} \\ &\le \frac{\sum_{\ell=0}^{j} \alpha_\ell y^\ell \left((d+1)^{j-\ell} - d^{j-\ell}\right)}{(d+y)^{r+j-1}} \end{aligned}$$

where the numerator approaches 0 since $d = \Theta(y) = \Theta(n)$. Using Equation (8) and all limits stated above, we get finally

$$\begin{aligned} \lim_{n\to\infty} \mathbf{Pr}\left[E_j\right] &= \lim_{n\to\infty} \frac{1}{r-1} c^r n^{r-1} \left(\frac{1}{c^{r-1}} \frac{1}{n^{r-1}} - \frac{\sum_{\ell=0}^{j} \alpha_\ell (1-c)^{j-\ell} c^\ell}{n^{r-1}} \right) \\ &= \frac{1}{r-1} \left(c - \sum_{\ell=0}^{j} \alpha_\ell c^{r+\ell} (1-c)^{j-\ell} \right) \\ &= \frac{c}{r-1} - \frac{c^r (1-c)^j}{r-1} \sum_{\ell=0}^{j} \alpha_\ell \left(\frac{c}{1-c} \right)^\ell . \end{aligned}$$

This concludes the proof. ◀

▶ **Lemma A.2.** *Let $f\colon \mathbb{R} \to \mathbb{R}$ with $f(i) = i^j/(i+y)^{r+j}$ and $j \ge 0$, $r \ge 1$, and $y > 0$ does not depend on i. The function f is monotonically increasing for $i \le d$ and monotonically decreasing for $i > d$ where $d = (jy)/r$.*

Proof. Let $g(i) = i^j$ and $h(i) = (i+y)^{r+j}$. We consider the first derivative $f'(i) = \frac{g'(i)h(i) - g(i)h'(i)}{h(i)^2}$. Since $h(i)^2$ is nonnegative, f grows monotonically if

$$g'(i)h(i) \ge g(i)h'(i) \;\Leftrightarrow\; j i^{j-1} (i+y)^{r+j} \ge i^j (r+j)(i+y)^{r+j-1} \;\Leftrightarrow\; j(i+y) \ge i(r+j) .$$

It is easy to see that the last inequality is equivalent to $i \le \frac{jy}{r} = d$. ◀

▶ **Lemma A.3.** *Let $f\colon \mathbb{R} \to \mathbb{R}$ with $f(i) = i^j/(i+y)^{r+j}$ and $r = 1$, $j \ge 0$, and $y > 0$ does not depend on i. The following function F fulfills $F'(i) = f(i)$:*

$$F(i) = \ln(i+y) + \sum_{\ell=1}^{j} \beta_\ell \frac{y^\ell}{\ell (i+y)^\ell}$$

where $\beta_\ell = (-1)^{\ell+1} \binom{j}{\ell}$ for $1 \le \ell \le j$.

Proof. We need to show $F'(i) = f(i)$ and observe first that

$$F'(i) = \frac{1}{i+y} + \sum_{\ell=1}^{j} \beta_\ell \frac{-\ell y^\ell}{\ell(i+y)^{\ell+1}} = \frac{1}{i+y} + \sum_{\ell=1}^{j} \beta_\ell y^\ell \frac{-(i+y)^{j-\ell}}{(i+y)^{j+1}}$$
$$= \frac{1}{(i+y)^{j+1}} \left((i+y)^j + \sum_{\ell=1}^{j} \beta_\ell y^\ell (-(i+y)^{j-\ell}) \right)$$

and since $\beta_0 = (-1)^{0+1}\binom{j}{0} = -1$ we get further

$$F'(i) = \frac{1}{(i+y)^{j+1}} \sum_{\ell=0}^{j} \beta_\ell y^\ell (-(i+y)^{j-\ell}) = \frac{1}{(i+y)^{j+1}} \sum_{\ell=0}^{j} (-1)^{\ell+2} \binom{j}{\ell} y^\ell (i+y)^{j-\ell}\,.$$

Finally, note that $(-1)^{\ell+2} y^\ell = (-y)^\ell$, thus by the binomial theorem the last sum evaluates to $((i+y)+(-y))^j = i^j$ which concludes the proof. ◀

▶ **Lemma A.4.** *Let $f\colon \mathbb{R} \to \mathbb{R}$ with $f(i) = i^j/(i+y)^{r+j}$ and $j \geq 0$, $r \geq 2$, and $y > 0$ does not depend on i. The following function F fulfills $F'(i) = f(i)$:*

$$F(i) = -\frac{\sum_{\ell=0}^{j} \alpha_\ell i^{j-\ell} y^\ell}{\alpha_0 (r-1)(i+y)^{r+j-1}}\,,$$

where $\alpha_\ell = \binom{j+r-1}{\ell+r-1}$ for $0 \leq \ell \leq j$.

Proof. Let $G(i)$ and $H(i)$ be the numerator and denominator of $F(i)$. It holds that $G'(i) = -\sum_{\ell=0}^{j} \alpha_\ell (j-\ell) i^{j-\ell-1} y^\ell$ and $H'(i) = \alpha_0(r-1)(r+j-1)(i+y)^{r+j-2} = H(i)r(i)$ where $r(i) = \frac{r+j-1}{i+y}$. In order to prove the claim, we show

$$G'(i)(i+y) - G(i)(r+j-1) = i^j \alpha_0 (r-1) \tag{9}$$

since then we have

$$F'(i) = \frac{G'(i)H(i) - G(i)H'(i)}{H(i)^2} = \frac{G'(i) - G(i)r(i)}{H(i)} = \frac{G'(i) - G(i)r(i)}{\frac{\alpha_0(r-1)}{i+y}(i+y)^{r+j}}$$
$$= \frac{(i+y)(G'(i) - G(i)r(i))}{\alpha_0(r-1)(i+y)^{r+j}} = \frac{(i+y)G'(i) - (r+j-1)G(i)}{\alpha_0(r-1)(i+y)^{r+j}}$$

With Equation (9), the last term resolves to $= \frac{i^j \alpha_0 (r-1)}{\alpha_0(r-1)(i+y)^{r+j}} = f(i)$. It remains to show Equation (9):

$$G'(i)(i+y) - G(i)(r+j-1)$$
$$= -\left(\sum_{\ell=0}^{j} \alpha_\ell (j-\ell) i^{j-\ell-1} y^\ell\right)(i+y) + \left(\sum_{\ell=0}^{j} \alpha_\ell i^{j-\ell} y^\ell\right)(r+j-1)$$
$$= -\left(\sum_{\ell=0}^{j} \alpha_\ell (j-\ell) i^{j-\ell} y^\ell\right) - \left(\sum_{\ell=0}^{j} \alpha_\ell (j-\ell) i^{j-\ell-1} y^{\ell+1}\right)$$
$$+ \left(\sum_{\ell=0}^{j} \alpha_\ell i^{j-\ell} y^\ell\right)(r+j-1)$$
$$= \left(\sum_{\ell=0}^{j} \alpha_\ell i^{j-\ell} y^\ell (r-1+\ell)\right) - \left(\sum_{\ell=0}^{j-1} \alpha_\ell (j-\ell) i^{j-\ell-1} y^{\ell+1}\right)\,.$$

Note that the first sum contains all powers of i from i^0 to i^j, while the latter sum only powers from i^0 to i^{j-1}. Therefore, we can split up the part for i^j from the first sum and group equal powers of i to obtain

$$\alpha_0(r-1)i^j + \sum_{\ell=1}^{j} \left(\alpha_\ell(r-1+\ell) - \alpha_{\ell-1}(j-\ell+1)\right) i^{j-\ell} y^\ell \,.$$

The claim follows if we can show that the last sum evaluates to zero. This is true, since by definition of α_ℓ it holds that

$$\alpha_\ell(r-1+\ell) = \binom{j+r-1}{\ell+r-1}(r-1+\ell) = \frac{(j+r-1)!}{(\ell+r-1)!(j-\ell)!}(r-1+\ell)$$
$$= \frac{(j+r-1)!}{(\ell+r-2)!(j-\ell+1)!}\frac{(j-\ell+1)}{(j-\ell)!} = \binom{j+r-1}{(\ell-1)+r-1}(j-\ell+1) = \alpha_{\ell-1}(j-\ell+1)\,. \quad ◀$$

Table 3 Optimal parameters and corresponding competitive ratios of SINGLE-REF for $k \in [1..100]$. For readibility, the numeric values are truncated after the fourth decimal place.

k	r	c	competitive ratio	k	r	c	competitive ratio
1	1	0.3678	0.3678	51	10	0.1662	0.7082
2	1	0.2545	0.4119	52	10	0.1635	0.7098
3	2	0.3475	0.4449	53	10	0.1608	0.7113
4	2	0.2928	0.4785	54	10	0.1582	0.7127
5	2	0.2525	0.4999	55	10	0.1557	0.7141
6	2	0.2217	0.5148	56	10	0.1532	0.7155
7	3	0.2800	0.5308	57	10	0.1509	0.7168
8	3	0.2549	0.5453	58	10	0.1486	0.7180
9	3	0.2338	0.5567	59	11	0.1597	0.7193
10	3	0.2159	0.5660	60	11	0.1574	0.7206
11	4	0.2570	0.5740	61	11	0.1551	0.7219
12	4	0.2410	0.5834	62	11	0.1529	0.7231
13	4	0.2267	0.5914	63	11	0.1508	0.7243
14	4	0.2140	0.5983	64	11	0.1487	0.7255
15	4	0.2026	0.6043	65	11	0.1467	0.7266
16	4	0.1924	0.6096	66	11	0.1447	0.7277
17	5	0.2231	0.6155	67	11	0.1428	0.7287
18	5	0.2133	0.6211	68	12	0.1527	0.7298
19	5	0.2042	0.6261	69	12	0.1508	0.7309
20	5	0.1959	0.6306	70	12	0.1489	0.7320
21	5	0.1882	0.6347	71	12	0.1470	0.7330
22	5	0.1811	0.6384	72	12	0.1452	0.7340
23	6	0.2054	0.6426	73	12	0.1434	0.7350
24	6	0.1985	0.6465	74	12	0.1417	0.7360
25	6	0.1919	0.6502	75	12	0.1400	0.7369
26	6	0.1858	0.6535	76	12	0.1384	0.7378
27	6	0.1800	0.6566	77	13	0.1473	0.7387
28	6	0.1746	0.6595	78	13	0.1456	0.7397
29	7	0.1947	0.6625	79	13	0.1440	0.7406
30	7	0.1893	0.6655	80	13	0.1424	0.7415
31	7	0.1842	0.6684	81	13	0.1408	0.7424
32	7	0.1793	0.6711	82	13	0.1393	0.7433
33	7	0.1747	0.6736	83	13	0.1378	0.7441
34	7	0.1703	0.6760	84	13	0.1363	0.7449
35	7	0.1662	0.6782	85	13	0.1349	0.7457
36	8	0.1830	0.6805	86	14	0.1429	0.7465
37	8	0.1788	0.6829	87	14	0.1415	0.7473
38	8	0.1748	0.6851	88	14	0.1400	0.7482
39	8	0.1710	0.6873	89	14	0.1386	0.7490
40	8	0.1673	0.6893	90	14	0.1372	0.7497
41	8	0.1638	0.6912	91	14	0.1359	0.7505
42	8	0.1605	0.6930	92	14	0.1346	0.7512
43	9	0.1750	0.6948	93	14	0.1333	0.7520
44	9	0.1716	0.6968	94	14	0.1320	0.7527
45	9	0.1683	0.6986	95	14	0.1307	0.7534
46	9	0.1651	0.7004	96	15	0.1381	0.7541
47	9	0.1621	0.7021	97	15	0.1368	0.7548
48	9	0.1592	0.7037	98	15	0.1356	0.7555
49	9	0.1563	0.7052	99	15	0.1343	0.7562
50	9	0.1536	0.7067	100	15	0.1331	0.7569