

Online Sampling and Decision Making with Low Entropy

Mohammad Taghi Hajiaghayi¹, Dariusz R. Kowalski², Piotr Krysta^{2,3} and Jan Olkowski¹

¹Department of Computer Science, University of Maryland, College Park, Maryland, USA

²School of Computer and Cyber Sciences, Augusta University, Augusta, Georgia, USA

³Department of Computer Science, University of Liverpool, Liverpool, UK

hajiagha@umd.edu, {dkowalski, pkrysta}@augusta.edu, olkowski@umd.edu

Abstract

Suppose we are given an integer $k \leq n$ and n boxes, labeled $1, 2, \dots, n$ by an adversary, each containing a single number chosen from an unknown distribution; the n distributions are not necessarily identical. We have to choose an order to sequentially open the boxes, and each time we open the next box in this order, we learn the number inside. If we reject a number in a box, the box cannot be recalled. Our goal is to accept k of these numbers, without necessarily opening all boxes, such that the accepted numbers are the k largest numbers in the boxes (thus their sum is maximized). This problem, sometimes called a free order multiple-choice secretary problem, is one of the classic examples of online decision making problems.

A natural approach to solve such problems is to sample elements in random order; however, as indicated in several sources, e.g., Turan et al. NIST'15 [35], Bierhorst et al. Nature'18 [10], pure randomness is hard to get in reality. Thus, pseudo-randomness has to be used, with a small entropy.

We show that with a very small $O(\log \log n)$ entropy an almost-optimal approximation of the value of k largest numbers can be selected, with only a polynomially small additive error, for $k < \log n / \log \log n$. Our solution works for exponentially larger range of parameter k compared to previously known algorithms (STOC'15 [22]). We also prove a corresponding lower bound on the entropy of optimal (and even close-to-optimal, with respect to competitive ratio) solutions for this problem of choosing k largest numbers, matching the entropy of our algorithm. No previous lower bound on entropy was known for this problem if $k > 1$.

1 Introduction

Online decision making problems and their important part – sampling, heavily depend on randomness. Efficient sampling has a long history, since Knuth, Vitter and others [Knuth, 1981; Vitter, 1987], who studied how to generate a randomly ordered sample close to uniformly random. Since then, effi-

cient and accurate sampling has become a fundamental problem in data science, see the recent survey by Mahmud et al. [Mahmud et al., 2020]. However, as indicated in several prominent sources, such as National Institute of Standards and Technology (NIST) [Turan et al., 2015] or Nature [Bierhorst et al., 2018], pure randomness is hard to get, thus in practice pseudo-randomness with a small entropy has to be used. Recently, the problem has brought more awareness and detailed analysis also in the online community [Buchbinder et al., 2023; Kesselheim et al., 2015a].

Prominent examples of the online decision problems are secretary and prophet-types problems. The most classical version of the secretary problems, introduced by statisticians in the 60s, asks for irrevocably hiring the best secretary among n rankable applicants and was analyzed in [Lindley, 1961; Dynkin, 1963; Chow et al., 1964; Gilbert and Mosteller, 1966; Bateni et al., 2013; Rubinstein, 2016; Rubinstein and Singla, 2017]. In the simplest problem's version, the goal is to find the best strategy when choosing from a sequence of randomly ordered applicants. A multiple-choice refers to the fact that the strategy is allowed to choose up to k applicants. The secretary problem has also played a fundamental role in advancing stopping theory, online algorithms, and various other fields [Babaioff et al., 2007; Babaioff et al., 2008; Hajiaghayi et al., 2004; Esfandiari et al., 2017; Hajiaghayi et al., 2007]. Recently, the problem has attracted attention from the learning communities who studied modifications in which the algorithm is given a prediction about the best among applicants [Antoniadis et al., 2020], a prediction interval for evaluation of each applicant [Jiang et al., 2021], or the objective is to rank all applicants rather than choosing best subset of them [Assadi et al., 2019].

Free-order secretary. We focus on a variant of the secretary problem called *the free-order secretary problem*. The free-order assumption reflects the fact that there is a third party that can decide the order of appearing applicants. This is in contrast to the original secretary problem where the order is uniformly random, but also generalizes the model as the third party can always randomly permute the applicants. On the other hand, the hardness of the problem remains in the fact that evaluations of candidates are still unknown to both the third party and an algorithm. For this model, different metrics for creating a fair order cheaper (i.e., an order for which there exist competitive algorithms) have been pro-

posed. Kesselheim, Kleinberg, and Niazadeh [Kesselheim *et al.*, 2015a] were the first to ask this question for secretary problems, showing how to construct in polynomial-time, a probability distribution on orders (permutations of size n), with entropy $O(\log \log n)$ such that when the classic secretary algorithm ($k = 1$) is executed on this distribution, it is successful with probability close to the optimal probability $\frac{1}{e}$. They also prove that if this distribution has entropy $o(\log \log n)$ then no 1-secretary algorithm can achieve constant success probability. When the evaluations of applicants are not deterministic, but rather follow some probabilistic distributions, Arsenis, Drosis, and Kleinberg [Arsenis *et al.*, 2021] prove that a small set of orders can be pre-computed, such that for any collection of n distributions of evaluations, the classic secretary algorithm achieves a constant competitive ratio when sampled from the pre-computed distribution.

Despite efforts, previous work does not provide exhaustive answers to the following important questions:

What is the minimum entropy of a random order distribution that allows to sample and choose k elements in polynomial time with nearly-optimal competitive ratio? How large k could be?

We make a significant step to answer both questions. We show that in the free-order model, one can sample from a polynomially constructible distribution of orders with entropy $O(\log \log n)$, achieving a competitive ratio $1 - \epsilon(k)$, where $\epsilon(k)$ part is polynomially smaller than that in [Kesselheim *et al.*, 2015a] and only $\sqrt{\log k}$ factor from the absolute bound $\Theta(\frac{1}{\sqrt{k}})$ for any entropy [Kleinberg, 2005]. We prove that no smaller entropy could yield good competitiveness, for *any* k . Our construction lets us select up to $k = O(\frac{\log n}{\log \log n})$ candidates, a doubly exponential improvement over $k = O(\log \log \log n)$ of [Kesselheim *et al.*, 2015a]. Our results details are in Sec. 2, with overview in Table 1. For the full version of all sections, see [Hajjaghayi *et al.*, 2024].

Problems and preliminaries. In the *free order multiple-choice secretary* problem, we are given integers k, n , $1 \leq k \leq n$, and n boxes labeled $[n] = \{1, 2, \dots, n\}$ by an adversary, each $i \in [n]$ containing a single number $v(i)$, chosen by the adversary from an unknown distribution. The goal is to choose an order in which we will be sequentially opening these boxes. Each time we open the next box i in the chosen order, we learn the number $v(i)$ and decide to accept $v(i)$ or not. This decision is irrevocable, and we cannot revisit any box. We have to accept k of these numbers without necessarily opening all boxes, where the objective is to accept k largest among them, i.e., to accept k elements with maximum possible sum. Once we have accepted k of the numbers, we stop opening the remaining boxes, if any. If we have accepted j numbers so far and there are only $k - j$ remaining boxes, we have to open all these boxes and accept their numbers. This problem is also called the *free order k -secretary problem*, and when $k = 1$, it is *free order secretary problem*.

Notation. Let Π_n be the set of all $n!$ permutations of the sequence $(1, 2, \dots, n)$. A probability distribution p over Π_n is a function $p : \Pi_n \rightarrow [0, 1]$ s.t. $\sum_{\pi \in \Pi_n} p(\pi) = 1$. Shannon entropy, or simply, entropy, of the distribution p is

$\mathcal{H}(p) = -\sum_{\pi \in \Pi_n} p(\pi) \cdot \log(p(\pi))$, assuming $0 \cdot \log(0) = 0$, and all log's have base 2. Given a distribution \mathcal{D} on Π_n , $\pi \sim \mathcal{D}$ means that π is sampled from \mathcal{D} . A special case of a distribution is given a (multi-)set $\mathcal{L} \subseteq \Pi_n$, and we write $\pi \sim \mathcal{L}$ when a random order is selected uniformly at random (u.a.r.) from \mathcal{L} . We abbreviate “random variable” to r.v.

Competitive ratio. As is common, [Jaillet *et al.*, 2013], we quantify the performance of an algorithm A by the *competitive ratio*: A is α -competitive or has *competitive ratio* $\alpha \in (0, 1)$ if it accepts k numbers whose sum is at least α times the sum of the k largest numbers; the competitive ratio is usually in expectation using the random order randomness.

Wait-and-pick and threshold algorithms. An algorithm for the k -secretary problem is called *wait-and-pick* if it only observes the first m values (position $m \in \{1, 2, \dots, n - 1\}$ is a fixed *checkpoint*), selects one of the observed values x (x is a *value threshold*), and then selects every value of at least x received after checkpoint position m ; however, it cannot select more than k values in this way, and it may also select the last i values (even if they are smaller than x) provided it selected only $k - i$ values before that. An extension of wait-and-pick algorithms, see [Gupta and Singla, 2020], allows partition of the order into consecutive phases, and selecting potentially different threshold value for each phase (other than 1) based on the values observed in previous phases. Thresholds are computed at checkpoints – last positions of phases. These thresholds could also be set based on statistics. We call such algorithms *threshold algorithms*, or *multiple-threshold algorithms* if there are more than two phases.

Related work. Our problem is directly related to the *prophet inequality problem with order selection* [Hill, 1983]. Namely, the free order 1-secretary problem is the Hill’s problem where the numbers in the boxes are given by unknown single-point distributions, whereas the prophet inequality problem with order selection assumes arbitrary (un)known distributions. The free order k -secretary problem is also the free order matroid secretary problem in [Jaillet *et al.*, 2013] with a uniform matroid. Recently, the *free order* (a.k.a., *best order*) variants have been studied extensively for the secretary and prophet problems, cf., [Abolhassani *et al.*, 2017; Arsenis *et al.*, 2021; Beyhaghi *et al.*, 2021; Chawla *et al.*, 2010; Correa *et al.*, 2021; Jaillet *et al.*, 2013; Liu *et al.*, 2021a; Liu *et al.*, 2021b; Peng and Tang, 2022]. The fundamental question this problem asks is: *What is the best order to choose to maximize the chance of accepting the largest number?*

2 Our Results and Techniques

Main contribution: algorithmic results. Our main result is a polynomial-time construction of a low-entropy distribution, guaranteeing an almost-tight competitive factor for the free order k -secretary problem.

Theorem 1. *For any $k < \log n / \log \log n$, there exists a multi-set of n -element permutations \mathcal{L}_n s.t. a deterministic multiple-threshold algorithm for the free order k -secretary achieves an expected $(1 - 4\sqrt{\frac{\log k}{k}})$ competitive ratio, using*

	problem	competitive ratio	entropy	range of k	reference
upper bounds	k -secr.	$1 - 4\sqrt{(\log k)/k}$	$O(\log \log n)$	$\leq \frac{\log n}{\log \log n}$	Sec. 4.4
	k -secr.	$1 - O(1/k^{1/3})$	$O(\log \log n)$	$\leq (\log^{(3)} n)^\delta$	[Kesselheim <i>et al.</i> , 2015a]
	1-secr.	$\frac{1}{e} - \frac{6 \log \log n}{e \log^{1/2} n}$	$O(\log \log n)$	1	Sec. 2
	1-secr.	$\frac{1}{e} - \omega\left(\left(\frac{1}{\log^{(3)} n}\right)^c\right)$	$O(\log \log n)$	1	[Kesselheim <i>et al.</i> , 2015a]
lower bounds	k -secr.	$\geq 1 - \varepsilon$	$\geq \frac{1-\varepsilon}{9} \log \log n$	$\leq \log^a n$	Sec. 2
	k -secr.	$\geq 1 - \varepsilon$	$\min\{\log 1/\varepsilon, \log \frac{n}{2k}\}^*$	$< n/2$	Sec. 2
	1-secr.	$\geq \frac{\mathcal{H}(n)}{\log \log n}$	$\geq \mathcal{H}(n)^{**}$	1	[Kesselheim <i>et al.</i> , 2015a]
notes	$a, \delta \in (0, 1)$ – some constants		$c, \varepsilon \in (0, 1)$ – any constants	$\log^{(3)} n = \log \log \log n$	

Table 1: Our results vs previous results. The upper bounds are w.r.t. the competitive ratio (success probability, resp.) for the k -secretary (1-secretary, resp.) problems and appropriate range of k , for which the algorithm runs in polynomial time. Our lower bounds are the first results for general k -secretary, for $k > 1$. * wait-and-pick algorithms. ** $\mathcal{H}(n) = o(\log \log n) \Rightarrow$ success probability $\leq \frac{\mathcal{H}(n)}{\log \log n} = o(1)$.

the order chosen uniformly at random from \mathcal{L}_n . \mathcal{L}_n is computable in time $O(\text{poly } n)$ and the uniform distribution on it has $O(\log \log n)$ entropy.

By using our framework we obtain the following fine-grained results for the classical secretary.

Theorem 2. *There exists a multi-set of n -element permutations \mathcal{L}_n s.t. the wait-and-pick algorithm with checkpoint $\lfloor n/e \rfloor$ achieves $\frac{1}{e} - \frac{6 \log \log n}{e \log^{1/2} n}$ success probability for free order 1-secretary problem, when the adversarial elements are presented in order chosen uniformly from \mathcal{L}_n . \mathcal{L}_n is computable in time $O(\text{poly } n)$ and the uniform distribution on it has $O(\log \log n)$ entropy.*

Lower bounds. We complement our results providing the first known lower bounds on entropy of multiple-choice secretary algorithms achieving any competitive ratio $1 - \varepsilon$.

Theorem 3. *Let $k \leq \log^a n$ for constant $a \in (0, 1)$, and let $\varepsilon \in (0, 1)$. Then, any algorithm (even fully randomized) solving k -secretary problem while drawing permutations from some distribution on Π_n with an entropy $H \leq \frac{1-\varepsilon}{9} \log \log n$, cannot achieve the expected competitive ratio of at least $1 - \varepsilon$ for sufficiently large n .*

The second lower bound on entropy is stronger as it holds for $k < n/2$, but it applies only to the class of the wait-and-pick algorithms.

Theorem 4. *Any wait-and-pick algorithm solving k -secretary problem, for $k < n/2$, with expected competitive ratio of at least $(1 - \varepsilon)$ requires entropy $\Omega(\min\{\log 1/\varepsilon, \log \frac{n}{2k}\})$.*

For a detailed discussion of our theorems 1-4, see Sec. 5.

Our techniques. We explain our framework on the k -secretary problem. Our goal is to compute a small (multi)set $\mathcal{L} \subseteq \Pi_n$, such that there is an algorithm \mathcal{A} for the problem that chooses $\pi \sim \mathcal{L}$ as its random order, achieving good expected competitive ratio. Let us start from a fully random permutation $\pi \sim \Pi_n$ as the random order. Algorithm \mathcal{A} faces adversarial values $v(1) \geq \dots \geq v(n)$ with indices $\text{ind}(k') \in [n]$ for value $v(k')$, $k' \in [n]$, where $(\text{ind}(1), \dots, \text{ind}(n))$ is the adversarial permutation. Algorithm \mathcal{A} considers these values sequentially in order $(\pi(\text{ind}(1)), \dots, \pi(\text{ind}(n)))$.

Our building blocks are *atomic events*, in the uniform probability space $(\pi \sim \Pi_n)$. Suppose that we are given a partition of the positions in π into t consecutive blocks (buckets) of positions and a mapping f of the k adversarial indices $\{\text{ind}(1), \dots, \text{ind}(k)\}$ to the t buckets. Let σ be any ordering of the indices $\{\text{ind}(1), \dots, \text{ind}(k)\}$. Given f and σ , we define an atomic event as the event that contains all $\pi \in \Pi_n$ that obey the mapping f and preserve the ordering σ . Atomic events are appealing due to their symmetry, leading to simple algorithms and analysis. But, it might be impossible to preserve atomic events directly while keeping small entropy. Our new approach is to *group* atomic events into *positive events*, see Def. 2, that model success probability of threshold algorithms. We will now specify the technical details behind each step of our framework:¹

1. Probabilistic analysis and defining positive events. (Section 3.1 and 4.1) The first step is a probabilistic analysis of an algorithm \mathcal{A} for the k -secretary problem, assuming that \mathcal{A} uses a random order $\pi \sim \Pi_n$. We need an algorithm whose success probability (expected competitive ratio) can be analyzed by probabilistic events modelled by atomic events. We chose as \mathcal{A} a modified multiple-threshold algorithm from the survey by Gupta and Singla [Gupta and Singla, 2020], see Algorithm 2. We use the probabilistic analysis from [Gupta and Singla, 2020], where they apply Chernoff bound to a collection of indicator random variables, which indicate if indices fall in an interval in a random permutation. These random variables are not independent, but they are *negatively associated*. We show an additional technical fact to justify the application of the Chernoff bound to these negatively-associated r.v.'s. This analysis lets us define positive events for Algorithm 2, meaning that the algorithm picks the item with i th largest adversarial value, for $i \in [k]$.

2. Decomposing positive event into atomic events. (Section 3.1 and 4.2) We show how to define any positive event as union of atomic events. Once we prove that such a decomposition exists, it is easy to find it by complete enumeration

¹Sections 3.1-3.3 present our techniques on the highest level of generality, but below we explain our framework on the k -secretary problem. Problem-specific application to the k -secretary problem is in Section 4.

over the space of atomic events, since the size of this space, $t^k \cdot k!$, basically only depends on k . Section 3.1 defines positive and atomic events abstractly. The decomposition of a positive event for Algorithm 2 for the k -secretary problem into atomic events is in Section 4.2.

3. Abstract derandomization of positive events via concentration bounds. (Section 3.2 and 4.3) Let each positive event P_γ , defined by atomic events $A \in \text{Atomic}(P_\gamma)$, hold with probability p_γ . We prove by Chernoff bound (Theorem 5) that there exists a small (multi)set $\mathcal{L} \subseteq \Pi_n$ nearly preserving probabilities p_γ of events P_γ , when Algorithm 2 uses $\pi \sim \mathcal{L}$ as random order. We show how to derandomize this theorem by the method of conditional expectations with a special pessimistic estimator of failure probability, see Sec. 3.2. This estimator is derived from the proof of Chernoff bound, inspired by Young [Young, 1995]. Our derandomization Algorithm 1 uses algorithm from Lemma 2 to compute a decomposition of P_γ into atomic events. Algorithm 1 calls $\text{Prob}(A)$ (defined in Appendix) to compute conditional probabilities. This crucially uses a symmetric nature of atomic events. Algorithm $\text{Prob}(A)$ and its analysis are much simpler compared to directly computing conditional probabilities for positive events, and this also gives a promise for further applications.

4. Dimension reduction and lifting positive events. (Section 3.3 and 4.4) The above abstract derandomization has time complexity $\Theta(n^k)$, being polynomial only for non-constant k . To make it polynomial, we design *dimension reductions* from n to $\text{poly} \log(n)$. We build on the idea of using Reed-Solomon (RS) codes from [Kesselheim *et al.*, 2015a], with two changes. First, we only use one RS code; they use a product of 2 or 3 such codes. Second, we replace their second step based on complete enumeration by our derandomization step 3. To define the dimension reduction, we propose in Lemma 1, a new technical ingredient: an algebraic construction of a family of functions that have bounded number of collisions and their preimages are of almost same sizes up to additive 1. We prove it by carefully using algebraic properties of polynomials. Our construction significantly improves and simplifies the constructions in [Kesselheim *et al.*, 2015a] by adding the constraint on sizes of preimages and using only one RS code. The last step of is to lift the lower-dimensional permutations back to the original dimension, i.e., to prove that we lose only slightly on the probability of positive events when going from the low-dimensional permutations back to dimension n , see Sections 3.3 and 4.4.

3 Generic Construction for Threshold Algorithms

3.1 Probabilistic Atomic and Positive Events for Threshold Algorithms

Let $\Omega = (\Pi_n, \mu)$ denote the probabilistic space of all $n!$ permutations of n elements with uniform probabilities: $\Pr[\pi \in \Pi_n] = \mu(\pi) = 1/n!$. We will define atomic events in space Ω . Given any integer $t \in \{2, 3, \dots, n\}$, let $\mathcal{B} := B_1, B_2, \dots, B_t$, be a *bucketing* of the sequence $(1, \dots, n)$, i.e., partition of sequence $(1, \dots, n)$ into t disjoint subsets (buckets) of consecutive numbers whose union

is the whole sequence. Formally, there are indices $\tau_1 < \tau_2 < \dots < \tau_{t-1} < \tau_t = n$, $\forall j \in [t-1] : \tau_j \in [n]$, such that $B_1 = \{1, \dots, \tau_1\}$, and $B_j = \{\tau_{j-1} + 1, \dots, \tau_j\}$, for $j \in \{2, 3, \dots, t\}$.

Definition 1 (Atomic events). Let $\sigma = (\sigma_1, \dots, \sigma_k) = (\sigma(1), \dots, \sigma(k))$ be any k -tuple of set $[n]$, i.e., injection $\sigma : [k] \rightarrow [n]$. Let $f : [k] \rightarrow [t]$ be a non-decreasing mapping of elements from the sequence into t buckets of the bucketing \mathcal{B} . Then an atomic event in probability space Ω for the chosen σ and f is defined as:

$$A_{\sigma, f} = \{\pi \in \Omega : \forall i \in [k] \pi^{-1}(\sigma_i) \in B_{f(i)} \text{ and} \\ \pi^{-1}(\sigma_1) < \pi^{-1}(\sigma_2) < \dots < \pi^{-1}(\sigma_k)\}.$$

The family $\mathcal{A}_{k, \mathcal{B}}$ of all atomic events, parameterized by k and \mathcal{B} , is defined as $\mathcal{A}_{k, \mathcal{B}} = \bigcup_{\sigma, f} A_{\sigma, f}$.

Definition 2 (Positive events). A positive event P is any subset of the atomic family $\mathcal{A}_{k, \mathcal{B}}$, denoted $\text{Atomic}(P) \subseteq \mathcal{A}_{k, \mathcal{B}}$, such that every two atomic events from P are disjoint, $P = \bigcup_{A \in \text{Atomic}(P)} A$. Any set of positive events is called a *positive family of events*.

We will propose in Sec. 4 a positive events family capturing behavior of an optimal algorithm for the k -secretary problem, showing how to express the positive events by atomic events.

3.2 Derandomization of Positive Events via Concentration Bounds: Theorem 5

Let $\Omega = (\Pi_n, \mu)$ be the probabilistic space of n -element permutations with uniform probabilities. Given integer $t \in \{2, 3, \dots, n\}$, let $\mathcal{B} = B_1, B_2, \dots, B_t$, be a *bucketing* of the sequence $(1, \dots, n)$. We will prove the following probabilistic result and its derandomization in Theorem 5.

Theorem 5. Let $k \in [n], k > 2$ and $\mathcal{A}_{k, \mathcal{B}}$ be the family of atomic events in the space Ω , where bucketing \mathcal{B} has t buckets. Let $\mathcal{P} = \{P_1, \dots, P_q\}$ be a family of positive events based on $\mathcal{A}_{k, \mathcal{B}}$, for some integer $q > 1$, such that for any $P_\gamma \in \mathcal{P}$, $\gamma \in [q]$, we have $\Pr_{\pi \sim \Pi_n}[P_\gamma] \geq p_\gamma > 0$ for some $p_\gamma \in (0, 1)$. Let $p_0 = \min\{p_1, p_2, \dots, p_q\}$, and for any $P_\gamma \in \mathcal{P}$, $\text{Atomic}(P_\gamma)$ be the set of atomic events that define P_γ . Then, for any $\delta \in (0, 1)$, there exists a multi-set \mathcal{L} of permutations with $|\mathcal{L}| \leq \ell = \frac{2 \log q}{\delta^2 p_0}$, such that $\Pr_{\pi \sim \mathcal{L}}[P_\gamma] \geq (1 - \delta) \cdot p_\gamma$, for each $P_\gamma \in \mathcal{P}$, and \mathcal{L} can be computed by Algorithm 1 in deterministic time $O(\ell n^3 q \cdot t^{2k} \cdot (k!)^2 \cdot (n + k k! + k \log^2 n))$.

In the reminder of Section 3.2 we present the proof of Theorem 5; missing details appear in [Hajiaghayi *et al.*, 2024].

Preliminaries. To derandomize the Chernoff argument in Theorem 5, we will derive a special conditional expectations method with a pessimistic estimator. We will model an experiment to choose u.a.r. a permutation $\pi_j \in \Pi_n$ by independent ‘‘index’’ r.v.’s $X_j^i : \Pr[X_j^i \in \{1, 2, \dots, n - i + 1\}] = 1/(n - i + 1)$, for $i \in [n]$, to define $\pi = \pi_j \in \Pi_n$ ‘‘sequentially’’: $\pi(1) = X_j^1$, $\pi(2)$ is the X_j^2 -th element in $I_1 = \{1, 2, \dots, n\} \setminus \{\pi(1)\}$, $\pi(3)$ is the X_j^3 -th element in $I_2 = \{1, 2, \dots, n\} \setminus \{\pi(1), \pi(2)\}$, etc, where elements are increasingly ordered.

Algorithm 1: Find permutations distribution (for positive events).

Input: Positive integers $n, k \leq n, \ell \geq 2$, such that $\log k \geq 8$.
Output: Multi-set $\mathcal{L} \subseteq \Pi_n$ of size ℓ . /* Function $\text{Prob}(A)$ is defined in Appendix. */

```

1  $\pi_1 := (1, 2, \dots, n); \mathcal{L} := \{\pi_1\};$  Let  $\mathcal{P} = \{P_1, \dots, P_q\}$  be the set of all positive events.
2 for  $\gamma \in \{1, \dots, q\}$  do  $w(P_\gamma) := 1 - \delta \cdot \phi_1(P_\gamma)$ 
3 for  $s = 1 \dots \ell - 1$  do
4   for  $r = 1 \dots n$  do
5     for  $\gamma \in \{1, \dots, q\}$  do
6       for  $\tau \in [n] \setminus \{\pi_{s+1}(1), \pi_{s+1}(2), \dots, \pi_{s+1}(r-1)\}$  do
7         for  $A \in \text{Atomic}(P_\gamma)$  do
8            $\mathbb{P}\text{r}[A | \pi_{s+1}(1), \dots, \pi_{s+1}(r-1), \pi_{s+1}(r) = \tau] := \text{Prob}(A)$ 
9            $\mathbb{E}[\phi_{s+1}(P_\gamma) | \pi_{s+1}(r) = \tau] := \sum_{A \in \text{Atomic}(P_\gamma)} \mathbb{P}\text{r}[A | \pi_{s+1}(1), \dots, \pi_{s+1}(r-1), \pi_{s+1}(r) = \tau]$ 
10          Choose  $\pi_{s+1}(r) = \tau$  for  $\tau \in [n] \setminus \{\pi_{s+1}(1), \pi_{s+1}(2), \dots, \pi_{s+1}(r-1)\}$  to minimize
11           $\sum_{\gamma=1}^q \omega(\ell, s, \gamma) \cdot w(P_\gamma) \cdot (1 - \delta \cdot \mathbb{E}[\phi_{s+1}(P_\gamma) | \pi_{s+1}(r) = \tau])$ .
12  $\mathcal{L} := \mathcal{L} \cup \{\pi_{s+1}\};$  for  $\gamma \in \{1, \dots, q\}$  do  $w(P_\gamma) := w(P_\gamma) \cdot (1 - \delta \cdot \phi_{s+1}(P_\gamma))$ 

```

Suppose random permutations $\mathcal{L} = \{\pi_1, \dots, \pi_\ell\}$ are generated using $X_j^1, X_j^2, \dots, X_j^\ell$ for $j \in [\ell]$. Given a positive event $P_\gamma \in \mathcal{P}$, $\gamma \in [q]$, let $X_j(P_\gamma) = 1$ if event P_γ holds for random permutation π_j , $j \in [\ell]$, and $X_j(P_\gamma) = 0$ otherwise. For $X(P_\gamma) = X_1(P_\gamma) + \dots + X_\ell(P_\gamma)$ and $\delta \in (0, 1)$, we have that $\mathbb{E}[X(P_\gamma)] \geq p_\gamma \ell$ and by Chernoff bound we have $\mathbb{P}\text{r}[X(P_\gamma) < (1 - \delta) \cdot p_\gamma \ell] < \exp(-\delta^2 p_\gamma \ell / 2)$, and so $(\mathbb{P}\text{r}[\exists P_\gamma \in \mathcal{P} : X(P_\gamma) < (1 - \delta) p_\gamma \ell] < 1$ if $\ell \geq \frac{2 \log q}{\delta^2 p_0}$). We call the positive event $P_\gamma \in \mathcal{P}$ *not well-covered* if $X(P_\gamma) < (1 - \delta) \cdot p_\gamma \ell$ (then a new r.v. $Y(P_\gamma) = 1$), and *well-covered* otherwise (then $Y(P_\gamma) = 0$). Let $Y = \sum_{P \in \mathcal{P}} Y(P)$. By the above argument $\mathbb{E}[Y] = \sum_{P \in \mathcal{P}} \mathbb{E}[Y(P)] < 1$ if $\ell \geq \frac{2 \log q}{\delta^2 p_0}$. We will keep the expectation $\mathbb{E}[Y]$ below 1 in each step of the derandomization, and these steps will sequentially define the permutations in \mathcal{L} .

Derandomization outline. We will choose permutations $\{\pi_1, \pi_2, \dots, \pi_\ell\}$ sequentially, with $\pi_1 = (1, 2, \dots, n)$. For some $s \in [\ell - 1]$ let permutations π_1, \dots, π_s have already been chosen (“fixed”). We will choose a “semi-random” permutation π_{s+1} position by position using X_{s+1}^i . Suppose $\pi_{s+1}(1), \pi_{s+1}(2), \dots, \pi_{s+1}(r)$ are already chosen for some $r \in [n - 1]$, where all $\pi_{s+1}(i)$ ($i \in [r - 1]$) are fixed and final, except $\pi_{s+1}(r)$ which is fixed but not final yet. We will vary $\pi_{s+1}(r) \in [n] \setminus \{\pi_{s+1}(1), \pi_{s+1}(2), \dots, \pi_{s+1}(r - 1)\}$ to choose the best value for $\pi_{s+1}(r)$, assuming that $\pi_{s+1}(r + 1), \pi_{s+1}(r + 2), \dots, \pi_{s+1}(n)$ are random. Permutations π_{s+2}, \dots, π_n are “fully-random”.

Conditional probabilities. Given $P_\gamma \in \mathcal{P}$, $r \in [n - 1]$, note that $X_{s+1}(P_\gamma)$ depends only on $\pi_{s+1}(1), \pi_{s+1}(2), \dots, \pi_{s+1}(r)$. We define $\mathbb{P}\text{r}[X_{s+1}(P_\gamma) = 1 | \pi_{s+1}(1), \pi_{s+1}(2), \dots, \pi_{s+1}(r)] = \mathbb{P}\text{r}[X_{s+1}(P_\gamma) = 1]$ when $r = 0$. We will show how to compute the conditional probabilities $\mathbb{P}\text{r}[X_{s+1}(P_\gamma) = 1 | \pi_{s+1}(1), \pi_{s+1}(2), \dots, \pi_{s+1}(r)]$, where randomness is over random positions $\pi_{s+1}(r + 1), \pi_{s+1}(r + 2), \dots, \pi_{s+1}(n)$, see Theorem 6 and [Hajiaghayi et al., 2024] for details.

Theorem 6. Suppose $\pi_{s+1}(1), \pi_{s+1}(2), \dots, \pi_{s+1}(r)$ have already been fixed for some $r \in \{0\} \cup [n]$. There is a deterministic algorithm to compute $\mathbb{P}\text{r}[X_{s+1}(P_\gamma) = 1 | \pi_{s+1}(1), \pi_{s+1}(2), \dots, \pi_{s+1}(r)]$, for any positive event P_γ , where the random event is a random choice of the semi-random permutation π_{s+1} conditioned on its first r elements already being fixed. This algorithm calls function $\text{Prob}(A)$ for $A \in \text{Atomic}(P_\gamma)$, and has running time $O(|\text{Atomic}(P_\gamma)| \cdot (n^2 + nk(k! + \log^2 n)))$.

Pessimistic estimator. For $P_\gamma \in \mathcal{P}$, let $\mathbb{E}[X_j(P_\gamma)] = \mathbb{P}\text{r}[X_j(P_\gamma) = 1] = \mu_{\gamma j}$ for each $j \in [\ell]$, and $\mathbb{E}[X(P_\gamma)] = \sum_{j=1}^\ell \mu_{\gamma j} = \mu_\gamma$. By assumption in Theorem 5, $\mu_{\gamma j} \geq p_\gamma$, for $j \in [\ell]$. We use Raghavan’s proof of Hoeffding bound, see [Young, 1995], for any $\delta > 0$, using that $\mu_{\gamma j} \geq p_\gamma$:

$$\begin{aligned} \mathbb{P}\text{r}[X(P_\gamma) < (1 - \delta) \cdot \ell \cdot p_\gamma] &\leq \prod_{j=1}^\ell \frac{1 - \delta \cdot \mathbb{E}[X_j(P_\gamma)]}{(1 - \delta)^{(1 - \delta) p_\gamma}} \\ &< \prod_{j=1}^\ell \frac{\exp(-\delta \mu_{\gamma j})}{(1 - \delta)^{(1 - \delta) p_\gamma}} \leq \prod_{j=1}^\ell \frac{\exp(-\delta p_\gamma)}{(1 - \delta)^{(1 - \delta) p_\gamma}} \\ &= \frac{1}{\exp(b(-\delta) \ell p_\gamma)} < \frac{1}{\exp(\delta^2 \ell p_\gamma / 2)}, \end{aligned}$$

where $b(x) = (1 + x) \ln(1 + x) - x$ and the last inequality follows by $b(-x) > x^2 / 2$. By the union bound:

$$\begin{aligned} \mathbb{P}\text{r}[\exists P_\gamma \in \mathcal{P} : X(P_\gamma) < (1 - \delta) \cdot \ell \cdot p_\gamma] \\ \leq \sum_{\gamma=1}^q \prod_{j=1}^\ell \frac{1 - \delta \cdot \mathbb{E}[X_j(P_\gamma)]}{(1 - \delta)^{(1 - \delta) p_\gamma}}. \end{aligned} \quad (1)$$

Inequality (1) helps us derive a pessimistic estimator of this failure probability. Let $\phi_j(P_\gamma) = 1$ if π_j makes event P_γ true, and $\phi_j(P_\gamma) = 0$ otherwise. The resulting pessimistic estimator is $\Phi = \sum_{\gamma=1}^q \omega(\ell, s, \gamma) \cdot \left(\prod_{j=1}^s (1 - \delta \cdot \phi_j(P_\gamma)) \right) \cdot (1 - \delta \cdot \mathbb{E}[\phi_{s+1}(P_\gamma)])$, where $\omega(\ell, s, \gamma) = \frac{(1 - \delta p_\gamma)^{\ell - s + 1}}{(1 - \delta)^{(1 - \delta) p_\gamma \ell}}$.

3.3 A Polynomial Time Construction of a Dimension-Reduction Set

A set \mathcal{G} of functions $g : [n] \rightarrow [\ell]$ is called a *dimension-reduction set* with parameters (n, ℓ, d) if the following two conditions hold:

(1) the number of functions that have the same value on any element of the domain is bounded:

$$\forall_{i,j \in [n], i \neq j} : |\{g \in \mathcal{G} : g(i) = g(j)\}| \leq d; \text{ and}$$

(2) for each function, the elements of the domain are almost uniformly partitioned into the elements of the image:

$$\forall_{i \in [\ell], g \in \mathcal{G}} : \frac{n}{\ell} \leq |g^{-1}(i)| \leq \frac{n}{\ell} + o(\ell).$$

The dimension-reduction set of functions is key in finding low-entropy probability distribution that implies high probability of positive events. It reduces the size of permutations from n to ℓ . [Kesselheim *et al.*, 2015a] were first to use such reduction for secretary problems. Our refinement is the new condition (2) above, which significantly strengthens the reduction. We show a general pattern for constructing a set of functions that reduce the dimension from n to $q < n$ by using refined Reed-Solomon codes.

Lemma 1. *There exists a set \mathcal{G} of functions $g : [n] \rightarrow [q]$, for some prime integer $q \geq 2$, such that for any two distinct indices $i, j \in [n]$, $i \neq j$, we have*

$$|\{g \in \mathcal{G} : g(i) = g(j)\}| \leq d \text{ and} \\ \forall q' \in [q] : |g^{-1}(q')| \in \{\lfloor n/q \rfloor, \lfloor n/q \rfloor + 1\},$$

where $1 \leq d < q$ is an integer such that $n \leq q^{d+1}$. Moreover, $|\mathcal{G}| = q$ and set \mathcal{G} can be constructed in deterministic polynomial time in n, q, d .

Corollary 1. *Observe that setting $q \in \Omega(\log n)$, $d \in \Theta(q)$ in Lemma 1 results in a dimension-reduction set of functions \mathcal{G} with parameters (n, q, \sqrt{q}) . Moreover, set \mathcal{G} has size q and as long as $q \in O(n)$, it can be computed in polynomial time in n .*

4 Generic Construction Applied to Free Order Multiple-Choice Secretary

For $\ell = \Theta(\text{poly} \log(n))$, we define atomic events with respect to the bucketing on ℓ -element permutations. We employ a refined version of an algorithm from [Gupta and Singla, 2020], see Algorithm 2, with permutation distribution $\mathcal{D} = \Pi_\ell$. Then we show that the algorithm's success can be described by atomic events. Even for such small ℓ , the family of atomic events is too rich to support the construction of a low-entropy distribution preserving all atomic events' probabilities. So we group atomic events into positive events, that have larger probabilities. Derandomization from Section 3.2 is used to construct such low-entropy distribution \mathcal{L}_ℓ supporting positive events' probabilities. Doing so on ℓ -element permutations, makes it computable in polynomial time in n , if ℓ is sufficiently small. This results in our derandomized algorithm Algorithm 2 that uses our low entropy distribution $\mathcal{D} = \mathcal{L}_\ell$. To lift the constructed distribution back to size n -permutations, we use dimension-reduction from Section 3.3.

4.1 Positive Events and Probabilistic Analysis

We show a lower bound on the measure of a positive event in the space Ω_ℓ . Let a random permutation $\pi \sim \Pi_\ell$. Let

$\hat{S} = \{j_1, \dots, j_k\}$, called a k -tuple, be an ordered subset $\{j_1, \dots, j_k\} \subseteq [\ell]$. \hat{S} models the positions in the adversarial permutation of the k largest adversarial values $v(1) \geq \dots \geq v(k)$, i.e., $j_i \in [\ell]$ is the position of value $v(i)$ in π . Let \mathcal{K} be the set of all such k -tuples.

Let us first define the following events: $H_j = \{\tau_j \geq \min\{v(i) : i = 1, \dots, k\} = v(k)\}$, and $L_j = \{\tau_j \leq v((1 - 2\varepsilon_j)k)\}$. Now, event C_i means that item j_i with value $v(i)$ will be chosen by the above algorithm for $i \in \{1, 2, \dots, (1 - \delta)k\}$. Assuming that $(1 - 2\varepsilon_j)k = 1$ for $j = -1$, we can define for each $j \in \{-1, 0, 1, \dots, \log 1/\delta - 1\}$ and for each $i \in \{(1 - 2\varepsilon_j)k, \dots, (1 - 2\varepsilon_{j+1})k\}$:

$$C_i = \{\text{item } j_i \text{ arrives after position } \ell_{j+1} \text{ in } \pi\}.$$

We define a *positive event* corresponding to the k -tuple \hat{S} and any $i \in \{1, 2, \dots, (1 - \delta)k\}$ as $P_{\hat{S}, i} = \left(\bigcap_{j \in [0, \log 1/\delta]} (L_j \cap H_j) \right) \cap C_i$. By refining the analysis in [Gupta and Singla, 2020] using Hoeffding-Chernoff bounds for *negatively associated* r.v.'s, we can prove that $\Pr_{\pi \sim \Pi_\ell} [P_{\hat{S}, i}] \geq (1 - \log(k)/k) \cdot (1 - 2^{j+1} \cdot \delta)$.

4.2 Decomposing Positive Event into Atomic Events

We will design an algorithm to express each positive event as a union of disjoint atomic events. We also give an algorithm to compute conditional probabilities.

Let us define the following bucketing \mathcal{B}_1 : $B_1 = \{1, 2, \dots, \ell_0\}$, $B_j = \{\ell_{j-2} + 1, \ell_{j-2} + 2, \dots, \ell_{j-1}\}$, for $j = 2, 3, \dots, \log(1/\delta) + 1$. We will now define all atomic events that define a given positive event P_i for some i . The bucketing has $t = \log(1/\delta) + 1$ buckets, so we will use the mappings $f : [k] \rightarrow [t]$, that fulfil (1), (2), (3) below; it can easily be shown that such mappings exist:

- (1) : $\forall j \in [0, \log 1/\delta] \exists J \subseteq \{j_1, \dots, j_k\} : |J| \geq (1 - \varepsilon_j)k_j$ & $f(J) \subseteq \{1, 2, \dots, j + 1\}$;
- (2) : $\forall j \in [0, \log 1/\delta] : |f^{-1}(\{1, 2, \dots, j + 1\}) \cap \{j_1, \dots, j_{(1-2\varepsilon_j)k}\}| \leq (1 - \varepsilon_j)k_j$;
- (3) : $f(j_i) > j + 2$, i.e., item j_i arrives after time ℓ_{j+1} .

Let \mathcal{F}_{C_i} , for $i \in \{1, 2, \dots, (1 - \delta)k\}$, be the family of all mappings $f : [k] \rightarrow [t]$ s.t. (1), (2), (3) hold. Given any $f \in \mathcal{F}_{C_i}$, define a set of permutations Σ_f of the sequence $\hat{S} = (j_1, \dots, j_k)$ consistent with f , i.e., $\sigma \in \Sigma_f$ if $\forall i, i' \in [k], i \neq i' : f(i) < f(i') \iff \sigma^{-1}(j_i) < \sigma^{-1}(j_{i'})$. We express the positive events $P_{\hat{S}, i}$ as follows:

$$P_{\hat{S}, i} = \bigcup_{f \in \mathcal{F}_{C_i}} \bigcup_{\sigma \in \Sigma_f} A_{\sigma, f}, \text{ for any } i \in \{1, 2, \dots, (1 - \delta)k\}.$$

Family $\mathcal{P}_{\mathcal{G}, S}$ of positive events of interest of Algorithm 2 is $\mathcal{P}_{\mathcal{G}, S} = \{P_{\hat{S}, i} : \hat{S} \subseteq [\ell], i \in [(1 - \delta)k]\}$.

Lemma 2. *For any k -tuple \hat{S} and any positive event $P_{\hat{S}, i}$, $i \in \{1, 2, \dots, (1 - \delta)k\}$, for the Algorithm 2 for the multiple-choice secretary problem, we can compute the set *Atomic*($P_{\hat{S}, i}$) of all atomic events defining $P_{\hat{S}, i}$ in time $O(t^k \cdot k! \cdot \ell)$, where $t = \log(1/\delta)$.*

Algorithm 2: Multiple-choice secretary algorithm with adaptive thresholds and permutation distribution $\mathcal{D} \in \{\Pi_\ell, \mathcal{L}_\ell\}$.

Input: Integers $\ell \geq 2$, $k \leq \ell$; sequence of ℓ items each with an adversarial value; π s.t. $\pi \sim \mathcal{D}$.

Output: Selected k items from the input sequence.

- 1 Set $\delta := \sqrt{\frac{\log k}{k}}$. Consider the ℓ items in the order given by the random permutation π .
 - 2 Denote $\ell_j := 2^j \delta \ell$ and ignore the first $\ell_0 = \delta \ell$ items.
 - 3 **for** $j \in [0, \log 1/\delta)$, *phase j runs on arrivals in window $W_j := (\ell_j, \ell_{j+1}]$* **do**
 - 4 Let $k_j := (k/\ell)\ell_j$ and let $\varepsilon_j := \sqrt{3\delta/2^j}$.
 - 5 Set threshold τ_j to be the $(1 - \varepsilon_j)k_j$ -th-largest value among the first ℓ_j items.
 - 6 Choose any item in window W_j with value above τ_j (until budget k is exhausted).
-

4.3 Derandomization of Positive Events via Concentration Bounds

The derandomization from Section 3.2 implies the existence of multi-set \mathcal{L}_ℓ of ℓ -element permutations where the uniform distribution on \mathcal{L}_ℓ preserves probabilities of positive events:

Lemma 3. *There is (multi)set $\mathcal{L}_\ell \subseteq \Pi_\ell$ with entropy $O(\log \ell)$ of the uniform distribution on \mathcal{L}_ℓ s.t.*

$$\mathbb{P}_{\pi \sim \mathcal{L}_\ell} [\pi \in P_{\hat{S}, i}] \geq \left(1 - \frac{1}{\sqrt{k}}\right) \mathbb{P}_{\pi \sim \Pi_\ell} [\pi \in P_{\hat{S}, i}],$$

for any positive event $P_{\hat{S}, i} \in \mathcal{P}_{G-S}$. The (multi)set \mathcal{L}_ℓ can be computed in $O(\ell^{10} \cdot \ell^{2k} \cdot (k!)^3)$ time.

4.4 Lifting Lower-Dimension Permutations Distribution Satisfying Positive Events

We apply a dimension-reduction from Section 3.3, to turn a set \mathcal{L}_ℓ of ℓ -element permutations to a set \mathcal{L}_n of n -element permutations such that the competitive ratio of Algorithm 2 executed on the uniform distribution over \mathcal{L}_ℓ is carried to the ratio of this algorithm on the uniform distribution over \mathcal{L}_n .

That is, we use a dimension-reduction set \mathcal{G} from Corollary 1 with parameters $(n, \ell, \sqrt{\ell})$, and \mathcal{L}_ℓ from Lemma 3. For given $g \in \mathcal{G}$, $\pi \in \mathcal{L}_\ell$, let $\pi \circ g : [n] \rightarrow [n]$ be any permutation $\sigma \in \Pi_n$ obtained as follows: function $g \in \mathcal{G}$, $g : [n] \rightarrow [\ell]$ assigns each element from $[n]$ to one of ℓ blocks; permutation $\pi \in \mathcal{L}_\ell$ determines the order of the blocks; and permutation σ is obtained by listing elements from the blocks according to π where order of elements inside blocks is irrelevant. The final set \mathcal{L}_n is defined as $\mathcal{L}_n = \{\pi \circ g : \pi \in \mathcal{L}_\ell, g \in \mathcal{G}\}$.

Lemma 4. *Let $ALG_{G-S}(\pi)$ be the output of Algorithm 2 on the permutation π . Assuming $\ell^2 < \frac{n}{\ell}$, one can compute a multi-set of n -element permutations \mathcal{L}_n s.t.*

$$\mathbb{E}_{\pi \sim \mathcal{L}_n} (ALG_{G-S}(\pi)) > \left(1 - \frac{k^2}{\sqrt{\ell}}\right) \left(1 - \frac{1}{\sqrt{k}}\right) \left(1 - \sqrt{\frac{\log k}{k}}\right) v^*$$

where $v^* = v(1) + \dots + v(k)$ is the sum of k largest adversarial elements. Given \mathcal{L}_ℓ we can construct \mathcal{L}_n in $O(n \cdot k^2 \log(\ell))$ time and the entropy of the uniform distribution over \mathcal{L}_n is $O(\log \ell + \log |\mathcal{L}_\ell|)$.

Theorem 7. *For any $k < \frac{\log n}{\log \log n}$, there exists a multi-set of n -element permutations \mathcal{L}_n s.t. Algorithm 2 has $1 - 4\sqrt{\frac{\log k}{k}}$ expected competitive ratio for the free order k -secretary problem, when elements are presented in order chosen uniformly*

from \mathcal{L}_n . \mathcal{L}_n is computable in time $O(\text{poly}(n))$ and uniform distribution over \mathcal{L}_n has entropy $O(\log k) = O(\log \log n)$.

Proof. Set $k, k < \log n / \log \log n$, $\ell := k^8$, then the claim follows by applying Lemma 3 and 4 to the parameters. The running time follows because if $k < \log n / \log \log n$, the time of the construction by Lemma 3 is $O(\ell^{10} \cdot \ell^{2k} \cdot (k!)^3) = \text{poly}(n)$, as the dominant time is $k! \leq k^k = \text{poly}(n)$. \square

5 Discussion and Conclusions

Theorem 3 uses a non-trivial extension of semitone sequences from [Kesselheim *et al.*, 2015a]. One of technical challenges we had to overcome is that the algorithm picks k values, instead of one, creating additional probabilistic dependencies.

By Theorem 3, entropy $\Omega(\log \log n)$ is necessary for any algorithm to achieve even a constant competitive ratio $1 - \epsilon$, for $k = O(\log^a n)$, $a < 1$, for k -secretary problem, implying that our upper bound in Thm. 1 is tight. Thm. 4 implies that entropy $\Omega(\log \log n)$ is necessary for any wait-and-pick algorithm to get a close-to-optimal competitive ratio $1 - \Omega(\frac{1}{k^a})$, for any $k < \frac{n}{2}$, where $a \leq \frac{1}{2}$. Moreover, the entropy $\Omega(\log k)$ is necessary, which could be $\Omega(\log n)$ for $k = \text{poly}(n)$.

Theorem 1 implies an (almost)² optimal competitive ratio $1 - O(\sqrt{\log k/k})$, with minimal entropy $O(\log \log n)$, when $k < \log n / \log \log n$, for k -secretary in non-uniform arrival model. This improves over competitive ratio $(1 - O(1/k^{1/3}) - o(1))$ of [Kesselheim *et al.*, 2015a], who used entropy $O(\log \log n)$ and constructed the distribution for $k = O((\log \log \log n)^\epsilon)$ only. We improve the range of working values k exponentially. Entropy optimality follows by our new lower bounds in Thm. 3 and 4. Such lower bounds were unknown before for the k -secretary problem, for $k > 1$. For 1-secretary, Thm. 2 improves, over doubly-exponentially, on the additive error to $\frac{1}{e}$ of $\omega(\frac{1}{(\log \log \log(n))^c})$ by [Kesselheim *et al.*, 2015b], which holds for any constant $c \in [0, 1]$, by a polynomial-time construction with additive error $\Theta(\frac{\log \log n}{\log^{1/2} n})$.

We have applied our new derandomization techniques to two different secretarial problems: 1-secretary and multiple-choice secretary problems. Further promising candidates are the online bipartite matching and matroid secretary problems, see [Gupta and Singla, 2020].

²The competitive ratio is optimal up to a factor of $\sqrt{\log k}$, see [Kleinberg, 2005; Gupta and Singla, 2020; Agrawal *et al.*, 2014] for a matching lower bound.

Acknowledgements

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

References

- [Abolhassani *et al.*, 2017] Melika Abolhassani, Soheil Ehsani, Hossein Esfandiari, MohammadTaghi Haji-Aghayi, Robert Kleinberg, and Brendan Lucier. Beating $1-1/e$ for ordered prophets. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 61–71. ACM, 2017.
- [Agrawal *et al.*, 2014] Shipra Agrawal, Zizhuo Wang, and Yinyu Ye. A dynamic near-optimal algorithm for online linear programming. *Operations Research*, 62(4):876–890, 2014.
- [Antoniadis *et al.*, 2020] Antonios Antoniadis, Themis Gouleakis, Pieter Kleer, and Pavel Kolev. Secretary and online matching problems with machine learned advice. *Advances in Neural Information Processing Systems*, 33:7933–7944, 2020.
- [Arsenis *et al.*, 2021] Makis Arsenis, Odysseas Drosis, and Robert Kleinberg. Constrained-order prophet inequalities. In *Proc. 2021 ACM-SIAM Symposium on Discrete Algorithms, SODA 2021*, pages 2034–2046, 2021.
- [Assadi *et al.*, 2019] Sepehr Assadi, Eric Balkanski, and Renato Leme. Secretary ranking with minimal inversions. *Advances in Neural Information Processing Systems*, 32, 2019.
- [Babaioff *et al.*, 2007] Moshe Babaioff, Nicole Immorlica, and Robert Kleinberg. Matroids, secretary problems, and online mechanisms. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 434–443. Society for Industrial and Applied Mathematics, 2007.
- [Babaioff *et al.*, 2008] Moshe Babaioff, Nicole Immorlica, David Kempe, and Robert Kleinberg. Online auctions and generalized secretary problems. *SIGecom Exch.*, 7(2), 2008.
- [Bateni *et al.*, 2013] Mohammadhossein Bateni, Mohammadtaghi Hajiaghayi, and Morteza Zadimoghaddam. Submodular secretary problem and extensions. *ACM Trans. Algorithms*, 9(4):Art. 32, 23, 2013.
- [Beyhaghi *et al.*, 2021] Hedyeh Beyhaghi, Negin Golrezaei, Renato Paes Leme, Martin Pál, and Balasubramanian Sivan. Improved revenue bounds for posted-price and second-price mechanisms. *Oper. Res.*, 69(6):1805–1822, 2021.
- [Bierhorst *et al.*, 2018] Peter Bierhorst, Emanuel Knill, Scott Glancy, Yanbao Zhang, Alan Mink, Stephen Jordan, Andrea Rommal, Yi-Kai Liu, Bradley Christensen, Sae Woo Nam, et al. Experimentally generated randomness certified by the impossibility of superluminal signals. *Nature*, 556(7700):223–226, 2018.
- [Buchbinder *et al.*, 2023] Niv Buchbinder, Joseph Naor, and David Wajc. Lossless online rounding for online bipartite matching (despite its impossibility). In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2030–2068. SIAM, 2023.
- [Chawla *et al.*, 2010] Shuchi Chawla, Jason D. Hartline, David L. Malec, and Balasubramanian Sivan. Multi-parameter mechanism design and sequential posted pricing. In *Proceedings of the Forty-Second ACM Symposium on Theory of Computing*, page 311–320, 2010.
- [Chow *et al.*, 1964] Y. S. Chow, S. Moriguti, H. Robbins, and S. M. Samuels. Optimal selection based on relative rank (the “secretary” problem). *Israel J. Math.*, 2:81–90, 1964.
- [Correa *et al.*, 2021] José R. Correa, Raimundo Saona, and Bruno Ziliotto. Prophet secretary through blind strategies. *Math. Program.*, 190(1):483–521, 2021.
- [Dynkin, 1963] Eugene B Dynkin. The optimum choice of the instant for stopping a markov process. In *Soviet Math. Dokl*, volume 4, 1963.
- [Esfandiari *et al.*, 2017] Hossein Esfandiari, Mohammad-Taghi Hajiaghayi, Vahid Liaghat, and Morteza Monezadeh. Prophet secretary. *SIAM J. Discret. Math.*, 31(3):1685–1701, 2017.
- [Gilbert and Mosteller, 1966] J. Gilbert and F. Mosteller. Recognizing the maximum of a sequence. *J. Amer. Statist. Assoc.*, 61:35–73, 1966.
- [Gupta and Singla, 2020] Anupam Gupta and Sahil Singla. Random-order models. In Tim Roughgarden, editor, *Beyond the Worst-Case Analysis of Algorithms*, pages 234–258. Cambridge University Press, 2020.
- [Hajiaghayi *et al.*, 2004] Mohammad Taghi Hajiaghayi, Robert D. Kleinberg, and David C. Parkes. Adaptive limited-supply online auctions. In *Proceedings 5th ACM Conference on Electronic Commerce (EC-2004), New York, NY, USA, May 17-20, 2004*, pages 71–80. ACM, 2004.
- [Hajiaghayi *et al.*, 2007] Mohammad Taghi Hajiaghayi, Robert Kleinberg, and Tuomas Sandholm. Automated online mechanism design and prophet inequalities. In *AAAI*, volume 7, pages 58–65, 2007.
- [Hajiaghayi *et al.*, 2024] Mohammad Taghi Hajiaghayi, Dariusz R. Kowalski, Piotr Krysta, and Jan Olkowski. Online sampling and decision making with low entropy. *arXiv preprint arXiv:2111.13203*, 2024.
- [Hill, 1983] T. P. Hill. Prophet inequalities and order selection in optimal stopping problems. *Proc. American Mathematical Society*, 88(1):131–137, 1983.
- [Jaillet *et al.*, 2013] Patrick Jaillet, José A. Soto, and Rico Zenklusen. Advances on matroid secretary problems: Free order model and laminar case. In *Integer Programming and Combinatorial Optimization - 16th International*

Conference, IPCO 2013, Valparaíso, Chile, March 18-20, 2013. Proceedings, 2013.

- [Jiang *et al.*, 2021] Zhihao Jiang, Pinyan Lu, Zhihao Gavin Tang, and Yuhao Zhang. Online selection problems against constrained adversary. In *International Conference on Machine Learning*, pages 5002–5012. PMLR, 2021.
- [Kesselheim *et al.*, 2015a] Thomas Kesselheim, Robert D. Kleinberg, and Rad Niazadeh. Secretary problems with non-uniform arrival order. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 879–888. ACM, 2015.
- [Kesselheim *et al.*, 2015b] Thomas Kesselheim, Robert D. Kleinberg, and Rad Niazadeh. Secretary problems with non-uniform arrival order. *CoRR*, abs/1502.02155, 2015.
- [Kleinberg, 2005] Robert Kleinberg. A multiple-choice secretary algorithm with applications to online auctions. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 630–631. Society for Industrial and Applied Mathematics, 2005.
- [Knuth, 1981] Donald Knuth. The art of computer programming, 2 (seminumerical algorithms). (*No Title*), 1981.
- [Lindley, 1961] D. V. Lindley. Dynamic programming and decision theory. *Appl. Statist.*, 10:39–51, 1961.
- [Liu *et al.*, 2021a] Allen Liu, Renato Paes Leme, Martin Pál, Jon Schneider, and Balasubramanian Sivan. Variable decomposition for prophet inequalities and optimal ordering. In *EC '21: The 22nd ACM Conference on Economics and Computation, Budapest, Hungary, July 18-23, 2021*, page 692. ACM, 2021.
- [Liu *et al.*, 2021b] Allen Liu, Renato Paes Leme, Martin Pál, Jon Schneider, and Balasubramanian Sivan. Variable decomposition for prophet inequalities and optimal ordering. In *EC '21: The 22nd ACM Conference on Economics and Computation, Budapest, Hungary, July 18-23, 2021*, page 692. ACM, 2021.
- [Mahmud *et al.*, 2020] Mohammad Sultan Mahmud, Joshua Zhexue Huang, Salman Salloum, Tamer Z Emara, and Kuanishbay Sadatdiyev. A survey of data partitioning and sampling methods to support big data analysis. *Big Data Mining and Analytics*, 3(2):85–101, 2020.
- [Peng and Tang, 2022] Bo Peng and Zhihao Gavin Tang. Order selection prophet inequality: From threshold optimization to arrival time design. In *Proceedings of the 63rd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2022.
- [Rubinstein and Singla, 2017] Aviad Rubinstein and Sahil Singla. Combinatorial prophet inequalities. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1671–1687, 2017.
- [Rubinstein, 2016] Aviad Rubinstein. Beyond matroids: Secretary problem and prophet inequality with general constraints. *arXiv preprint arXiv:1604.00357*, 2016.
- [Turan *et al.*, 2015] Meltem Sonmez Turan, John M. Kelsey, and Kerry A. McKay. How random is your RNG? In *Shmocon Proceedings, Washington, DC. National Institute of Standards and Technology (NIST)*, 2015.
- [Vitter, 1987] Jeffrey Scott Vitter. An efficient algorithm for sequential random sampling. *ACM transactions on mathematical software (TOMS)*, 13(1):58–67, 1987.
- [Young, 1995] Neal E. Young. Randomized rounding without solving the linear program. In *Proceedings of the Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, 22-24 January 1995. San Francisco, California, USA*, pages 170–178, 1995.