Matroid Secretary via Labeling Schemes

Kristóf Bérczi[a] Vasilis Livanos[b] José A. Soto[c] Victor Verdugo[d]

Abstract

The Matroid Secretary Problem (MSP) is one of the most prominent settings for online resource allocation and optimal stopping. A decision-maker is presented with a ground set of elements E revealed sequentially and in random order. Upon arrival, an irrevocable decision is made in a take-it-or-leave-it fashion, subject to a feasibility constraint on the set of selected elements captured by a matroid defined over E. The decision-maker only has ordinal access to compare the elements, and the goal is to design an algorithm that selects every element of the optimal basis with probability at least α (i.e., α-probability-competitive). While the existence of a constant probability-competitive algorithm for MSP remains a major open question, simple greedy policies are at the core of state-of-the-art algorithms for several matroid classes.

We introduce a flexible and general algorithmic framework to analyze greedy-like algorithms for MSP based on constructing a language associated with the matroid. Using this language, we establish a lower bound on the probability-competitiveness of the algorithm by studying a corresponding Poisson point process that governs the words' distribution in the language. Using our framework, we break the state-of-the-art guarantee for laminar matroids by settling the probability-competitiveness of the greedy-improving algorithm to be exactly $1 - \ln(2) \approx 0.3068$. For graphic matroids, we show a probability-competitiveness of 0.2693 when the underlying graph has no parallel edges and a guarantee of 0.2504 for general graphs, also breaking the state-of-the-art factor of 0.25.

[a]MTA-ELTE Matroid Optimization Research Group and HUN-REN–ELTE Egerváry Research Group, Department of Operations Research, Eötvös Loránd University, Budapest, Hungary (kristof.berczi@ttk.elte.hu)

[b]EPFL, Lausanne, Switzerland & Archimedes AI, Athens, Greece (vasileios.livanos@epfl.ch)

[c]Department of Mathematical Engineering and Center for Mathematical Modeling, Universidad de Chile, Santiago, Chile (jsoto@dim.uchile.cl)

[d]Institute for Mathematical and Computational Engineering, and Department of Industrial and Systems Engineering, PUC Chile, Santiago, Chile (victor.verdugo@uc.cl)

1 Introduction

One of the most celebrated problems in online decision-making is the *secretary problem*, in which one aims to hire the best out of n candidates that arrive in a uniformly random order. The decision-maker can only rank the candidates observed up to that point and, upon observing a candidate, must make an immediate and irrevocable decision to accept or reject them. The famous optimal strategy of rejecting the first $1/e$ fraction of candidates and then selecting the first one who is better than all previously seen candidates guarantees at least a $1/e$ probability of hiring the best candidate (see, e.g., Lindley [24], Dynkin [13]).

Arguably, one of the most interesting generalizations is the *matroid secretary problem* (MSP) introduced by Babaioff, Immorlica, Kempe and Kleinberg [4]. In the MSP, the decision-maker is given a matroid and can hire any set of candidates forming an independent set in the matroid. An algorithm that, for a fixed optimal basis B, selects a set in which every candidate from B appears with probability at least α is called *α-probability-competitive*. For the remainder of this paper, we assume, without loss of generality, that there exists a unique optimal basis. When the decision-maker observes a weight associated with each candidate and the selected set has weight at least an α fraction of the optimal weight basis, the algorithm is *α-utility-competitive*. Over the last two decades, there has been a series of works studying the competitiveness for general matroids, resulting in $\Omega(1/\log\log r)$ utility-competitive and $\Omega(1/\log r)$ probability-competitive algorithms [4, 8, 14, 23, 29] for rank r matroids. Babaioff, Immorlica, Kempe and Kleinberg [4] conjectured the existence of a (utility) constant-competitive algorithm for every matroid. The *matroid secretary conjecture*, as it has been known since, has been resolved for several sub-classes of matroids [10, 17, 19, 22, 28, 29], but it remains open in general and even for fundamental sub-classes such as gammoids or binary matroids.

Equally interesting is the question of the best-possible competitive ratio for matroids, assuming that the matroid secretary conjecture holds. The *strong matroid secretary conjecture* asserts that there exists a $1/e$-probability-competitive algorithm for all matroids; in other words, that the worst-case for the MSP is the rank-1 setting. Surprisingly, the strong matroid secretary conjecture is still wide open for most matroid sub-classes and has been shown to hold only for uniform [21], partition [4] and transversal matroids [20]. Perhaps the simplest matroid class for which the strong matroid secretary conjecture is still open is the class of rank-2 matroids, which fall under the class of laminar matroids and for which the best-known algorithm prior to our work is a 0.2587-utility-competitive algorithm [16]. Motivated by the strong matroid secretary conjecture, our work aims to shorten the gap between the known competitiveness guarantees and $1/e$ for several matroid sub-classes and add to the toolbox of techniques that can be used to yield improved guarantees in the future.

1.1 Our Contributions and Techniques

In this work, we develop new tools for analyzing algorithms for the MSP that are more effective at managing the complex correlations involved in determining whether an element arriving at a specific time can be selected. Conceptually, our main contribution is introducing a new framework to analyze greedy-improving algorithms for the MSP using labeling schemes. Using this, we improve the state-of-the-art probability-competitiveness for the fundamental laminar and graphic matroid classes and match the best-known probability-competitiveness guarantees for several other matroids as we extend the capabilities of the forbidden sets framework by Soto, Turkieltaub, and Verdugo [29] (see Appendix A).

The Labeling Scheme Framework. The algorithms we analyze initially skip the first p fraction of elements. Afterwards, they mark every element that improves upon the current best set, i.e., an *improving element*. Some marked elements will be added to the output ALG, when they arrive, according to an internal property P specific to the matroid. Property P filters the improving elements and guarantees the feasibility of the resulting selection, i.e., if P holds when an element e arrives, then it must be that $\text{ALG} \cup \{e\}$ is independent. To analyze the algorithm, we consider a separate procedure, called a *labeling scheme* that visits the marked elements in reverse order and assigns a label to each. At the end of the procedure, we obtain a word z by concatenating all labels assigned by our scheme. The core idea is that, for every matroid $\mathcal{M}$, we can assign a language $\mathcal{L}(\mathcal{M})$ such that, for any fixed element e^* in the optimal set, $\Pr[e^* \in \text{ALG}] \geq \Pr[z \in \mathcal{L}(\mathcal{M})]$, where ALG is the final set of our algorithm. Then, our competitiveness guarantees follow by computing $\Pr[z \in \mathcal{L}(\mathcal{M})]$, and we show that the uniformly random arrival order of the elements implies that the characters (i.e., the labels) of z are chosen uniformly at random from the label set and the length of z follows a Poisson distribution. Our new algorithmic framework using labeling schemes is described in detail in Section 3.

Laminar Matroids. One of the first algorithms proposed for the MSP was the *greedy-improving* algorithm, which greedily selects every improving element, subject to feasibility. Despite its simplicity, it has been remarkably successful for many matroid classes. For laminar matroids, the best-known guarantee before our work was the very recent ≈ 0.2105-probability-competitiveness, achieved by Huang, Parsaeian, and Zhu [16] using the greedy-improving algorithm. Using our labeling scheme framework, we settle the probability-competitiveness of the greedy-improving algorithm for laminar matroids and show a tight guarantee of $1 - \ln(2) \approx 0.3068$.

Theorem 1. *The greedy-improving algorithm is $(1 - \ln(2)) \approx 0.3068$-probability-competitive for the MSP on laminar matroids. In addition, for every $\varepsilon > 0$, there exists a laminar matroid $\mathcal{M}(\varepsilon)$ such that the probability-competitive ratio of the greedy-improving algorithm on $\mathcal{M}(\varepsilon)$ is at most $1 - \ln(2) + \varepsilon$.*

The analysis of laminar MSP via our labeling schemes framework is in Section 4. For every natural number r, we characterize the optimal competitiveness attained by the greedy-improving algorithm on laminar matroids of rank r. The competitiveness decreases as $r \to \infty$, with a limit of $1 - \ln(2)$. We also obtain a full characterization of the greedy-improving algorithm on uniform matroids for any fixed rank r and show that it grows from $1/e$ when $r = 1$ to $1 - 1/e$ as $r \to \infty$ (Proposition 4). Working towards an answer for the strong matroid secretary conjecture, we study the simplest matroids for which the existence of a $1/e$-probability-competitive algorithm is still open – rank-2 matroids. Since they are also laminar matroids, we can specialize our results for laminar matroids to this class, obtaining a probability-competitive ratio of 0.3341. We then show that one can go beyond this, as we propose an alternative algorithm that mixes between two different algorithms: the greedy-improving algorithm that behaves especially well when the matroid is close to uniform and an oblivious partition algorithm that is good when there are many elements with high value parallel to the optimal elements of the matroid.

Theorem 2. *There exists a 0.3462-probability-competitive algorithm for the MSP on rank-2 matroids.*

Graphic Matroids. We next turn our attention to the graphic matroid case. Here, we adjust our labeling scheme to account for an algorithm that is not greedy-improving, but avoids selecting edges that seem like they will induce cycles in the final set. The previous best algorithm, by Soto, Turkieltaub, and Verdugo [29], is 1/4-probability-competitive as it satisfies the 2-forbidden property.

We improve over this guarantee via our labeling framework, showing that one can go beyond the forbidden sets technique.

Theorem 3. *There exists a* 0.2504*-probability-competitive algorithm for the MSP on graphic matroids. In addition, for simple graphs (i.e., without parallel edges), there exists a* 0.2693*-probability-competitive algorithm.*

The analysis of graphic MSP via our labeling schemes framework can be found in Section 5. Our idea is to start with a basic algorithm that orients the improving edges and greedily constructs an auxiliary digraph AUX with a maximum in-degree of one. Consequently, AUX has at most one cycle per connected component. The algorithm in [29] selects a subset of AUX that includes only edges connecting nodes with in-degree 0 when they appear. Hence, the resulting set does not contain the last edge of any cycle, and is always feasible. However, this procedure may drop too many arcs from AUX. Our labeling schemes allow a more careful study of which edges we must delete to ensure that the graph is acyclic. For that, we assign a *non-negative generation* to each arc in AUX according to the order in which arcs were added. The basic algorithm outputs only edges of generation 0. We show that, in fact, by keeping edges with generation different from 1, we also obtain an acyclic graph. This new *generation algorithm* is 0.2693-probability-competitive for simple graphs (i.e., without parallel edges) – note that even for simple graphs, the previous algorithm by [29] is still 1/4-probability-competitive. By randomizing between the generation algorithm and an oblivious algorithm that is good for those optimal edges e^* that are likely to belong to some 2-cycles in AUX, we obtain our final algorithm for graphic matroids described in Theorem 3.

1.2 Related Work

The MSP was introduced by Babaioff, Immorlica, Kempe and Kleinberg [4], in which the authors gave a *threshold-price* $\Omega(1/\log r)$-utility-competitive algorithm for general matroids. Later, Chakraborty and Lachish [8] gave an improved $\Omega(1/\sqrt{\log r})$-utility-competitive algorithm, before the current best bound of $\Omega(1/\log\log r)$-utility-competitive ratio was obtained, first by Lachish [23] and then by Feldman, Svensson, and Zenklusen [14] via a simplified algorithm. To the best of our knowledge, the only probability-competitive algorithms for general matroids are the $\Omega(1/\log^2 r)$-competitive algorithm by Bateni, Hajiaghayi, and Zadimoghaddam [6] and a $\Omega(1/\log r)$-competitive algorithm by Soto, Turkieltaub, and Verdugo [29] that is based on the aided sample-based MSP studied by Feldman, Svensson, and Zenklusen [14].

The matroid secretary conjecture has been shown to hold for the *random assignment and random order* model by Soto [28], a setting with a slightly weaker adversary in which the weights of the elements are still chosen adversarially, but the assignment of weights to elements is done uniformly at random. Oveis-Gharan and Vondrák [15] extend this result to the setting where the arrival order of the elements is adversarial instead of uniformly random. The results in [28] require knowing the matroid upfront. Santiago, Sergeev, and Zenklusen [27] circumvent this problem by obtaining an algorithm that only requires access to an independence oracle in the random-assignment random-order model. Recently, Cristi, Dütting, Kleinberg and Paes Leme [9] gave reductions from the unknown-matroid to the known-matroid variants of MSP for several sub-classes of matroids, and showed an impossibility result for such a reduction in the case of a general matroid. Very recently, a series of works by Dughmi [11, 12] uses duality to establish an equivalence between the matroid secretary conjecture and the existence of specific correlated rounding schemes for matroids, called *random-order contention resolution schemes* (ROCRSs).

Regarding sub-classes of matroids for which we can guarantee a constant competitive ratio, even from the original paper on the MSP [4], a constant probability-competitive ratio was known for uniform, partition, graphic, and constant-degree transversal matroids. Later, Babaioff, Dinitz, Gupta, Immorlica, and Talwar [3] unified these results via the α-partition property technique. Soto, Turkieltaub, and Verdugo [29] introduced the forbidden-sets technique and obtained many of the current best competitive ratios for several sub-classes of matroids. We present a brief overview of this technique – and how our labeling scheme framework generalizes it – in Appendix A.

Laminar Matroids. Even before the initial paper on MSP [4], Kleinberg [21] showed an asymptotically optimal $\left(1 - \mathrm{O}\left(r^{-1/2}\right)\right)$-utility-competitive algorithm for uniform matroids of rank r. However, due to the algorithm's relative complexity, it is worth studying simpler algorithms like greedy-improving and analyzing their competitive ratio. Recently, Albers and Ladewig [2] presented a tighter analysis for other algorithms, including single-threshold algorithms and the so-called *optimistic* algorithm.

For laminar matroids, the 3/16000 guarantee of Im and Wang [17] was beaten by Jaillet, Soto, and Zenklusen, who gave a simpler algorithm using the α-partition property that is $1/(3e\sqrt{3}) \approx 0.0707$-competitive [18]. Later, Ma, Tang, and Wang [25] showed that the greedy-improving algorithm is 0.1041-competitive, a guarantee that was later improved by Soto, Turkieltaub, and Verdugo [29] using the forbidden sets technique to show that laminar matroids are 3-forbidden, yielding a 0.1924-probability-competitive algorithm. This bound was very recently beaten by Huang, Parsaeian, and Zhu [16] using the greedy-improving algorithm along with an improved analysis of its competitive ratio. In their paper, they showed that the greedy-improving algorithm is 0.2105-competitive for all laminar matroids.

Obtaining algorithms for rank-2 matroids that satisfy the 2-forbidden property described in [29] is straightforward, demonstrating that this class admits 1/4-competitive algorithms. Since these matroids are laminar, the results from [16] are also applicable. Their analysis establishes a lower bound of 0.2587 for rank-2 matroids, which, until this paper, represented the best competitive ratio for this class.

Graphic Matroids. The initial paper on the MSP by Babaioff, Immorlica, Kempe and Kleinberg [4] showed a 1/16-competitive algorithm for graphic matroids, and this bound was improved to $1/(3e)$ by Babaioff, Dinitz, Gupta, Immorlica, and Talwar [3]. Both of these results rely on the α-partition property. This approach was followed by Korula and Pal [22], who showed that graphic matroids have the 2-partition property, improving the bound to $1/(2e)$ using a simple algorithm. Finally, Soto, Turkieltaub, and Verdugo [29] showed that graphic matroids are 2-forbidden, yielding a 1/4-probability-competitive algorithm that, up until our results, had resisted improvement.

Impossibility Results. Babaioff, Immorlica, Kempe and Kleinberg [4] showed that the greedy-improving algorithm alone cannot be constant-competitive for graphic matroids. Then, Bahrani, Beyhaghi, Singla and Weinberg [5] generalized this result, showing that this is also the case for a bigger class of greedy-like algorithms and for algorithms that partition the ground set without looking at the weights, and then work on each part separately. Recently, Abdolazimi, Karlin, Klein, and Oveis Gharan [1] showed that algorithms based on the α-partition property cannot be constant-competitive for the full binary matroid, even if the partition obtained can depend on the initial set of sampled elements.

2 Preliminaries

Given a ground set E together with a subset $F \subseteq E$ and $e \in E$, the sets $F \setminus \{e\}$ and $F \cup \{e\}$ are abbreviated as $F - e$ and $F + e$, respectively. We denote the reals, non-negative reals, and non-negative integers by $\mathbb{R}$, $\mathbb{R}_+$, and $\mathbb{N}$, respectively. For $n \in \mathbb{N}$, we use $[n]$ to denote the set $\{1, \ldots, n\}$. For a set $S \subseteq \mathbb{N}$, we denote by S^* the set of finite sequences whose entries are elements of S and refer to these sequences as *words*. The set of words of length k is denoted by S^k. We use ϵ to denote the empty word and xy to denote the *concatenation* of two words x and y. We also make use of standard shorthand notation for regular languages such as $w = 1^a 2^b$ to denote the word $11\ldots122\ldots2$ with a many 1s and b many 2s, or $w \in 1^*2$ to denote that $w \in \{1^k 2 \colon k \in \mathbb{N}\}$.

Let $D = (V, A)$ be a *directed graph* with vertex set V and arc set A. We denote an arc a oriented from u to v by (u, v) (or sometimes uv for brevity), where u and v are called the *tail* and the *head* of a, respectively. An arc uv *enters* a subset Z of vertices if $v \in Z$, $u \notin Z$ and *exits* Z if $u \in Z$, $v \notin Z$. For a subset $F \subseteq A$ of arcs, the *in-degree* of Z is the number of arcs entering it and is denoted by $\deg^-_F(Z)$. The subscript F is dismissed when F consists of the whole arc set.

Matroids. We provide a few basic definitions for matroids and refer the reader to [26] for an extensive treatment. A *matroid* $\mathcal{M} = (E, \mathcal{I})$ is defined by its *ground set* E and its *family of independent sets* $\mathcal{I} \subseteq 2^E$ that satisfies the so-called *independence axioms*: (I1) $\emptyset \in \mathcal{I}$, (I2) $X \subseteq Y,\ Y \in \mathcal{I} \Rightarrow X \in \mathcal{I}$, and (I3) $X, Y \in \mathcal{I},\ |X| < |Y| \Rightarrow \exists e \in Y - X$ such that $X + e \in \mathcal{I}$. For $X \subseteq E$, the maximum size of an independent set in X is called the *rank* of X. A non-independent set is called *dependent*, and a *circuit* is an inclusion-wise minimal dependent set. Two elements $e, f \in E$ are *parallel* if they form a circuit of size two. The independence axioms imply that any two maximal independent sets in E have the same size, called the *rank of the matroid*, usually denoted by $r_{\mathcal{M}}$. The *rank function* of the matroid is a function $r_{\mathcal{M}} : 2^E \to \mathbb{N}$, where $r_{\mathcal{M}}(S)$ denotes the size of a maximal independent set contained in S. In both cases, the subscript is dropped if the matroid is clear from context. The *bases* of the matroid are its maximal independent sets. An element $e \in E$ is a *loop* if $\{e\}$ is a circuit and a *coloop* if e is contained in every base. Throughout the paper, we use n and r to denote the size of the ground set and the total rank of the matroid, respectively, i.e., $n = |E|$, and $r = r_{\mathcal{M}}(S)$.

A *laminar matroid* is a matroid $\mathcal{M} = (E, \mathcal{I})$ where $\mathcal{I} = \{F \subseteq E \colon |F \cap E_i| \leq g_i$ for each $i \in [q]\}$ for some laminar family $E_1, \ldots, E_q$ of subsets of E and upper bounds $g_1, \ldots, g_q \in \mathbb{N}$. For a graph $G = (V, E)$, the *graphic matroid* $\mathcal{M} = (E, \mathcal{I})$ of G is defined on the edge set by considering a subset $F \subseteq E$ to be independent if it is a forest, that is, $\mathcal{I} = \{F \subseteq E \colon F$ is acyclic$\}$.

Matroid Secretary Problem (MSP). In the Matroid Secretary Problem (MSP), one is given a matroid $\mathcal{M} = (E, \mathcal{I})$ and is presented with the elements of E in an online manner in a uniformly random order. Upon observing an element $e \in E$, one must decide immediately and irrevocably whether to select e or reject it and continue to the next element, subject to the constraint that the selected set S must be independent in $\mathcal{M}$, i.e., $S \in \mathcal{I}$. It is assumed throughout that the matroid description is given upfront to the algorithm and that the matroid is loopless, as loops can be skipped during the online process.

In the *utility MSP*, an injective weight function $w : E \to \mathbb{R}_+$ is given, which is not known a priori but is revealed upon the arrival of each element. The objective is to select a set S that maximizes the total weight $w(S) = \sum_{e \in S} w(e)$. An algorithm ALG selecting a (random) set S is said to be *α-utility-competitive*, with $\alpha \in [0, 1]$, if $\mathbb{E}[w(S)] \geq \alpha \cdot \mathrm{OPT}(E)$, where $\mathrm{OPT}(E)$ denotes the maximum weight of an independent set in $\mathcal{M}$ with respect to w.

In this paper, we consider a more general setting called the *ordinal MSP*. Here, instead of a weight function, a total order, referred to as the *value order*, is given on the elements of the matroid and denoted by $\succ$. Note that, by the greedy algorithm, any subset $S \subseteq E$ admits a unique independent set that is lexicographically maximal with respect to $\succ$, which we denote by $\mathrm{OPT}(S)$. Upon the arrival of a new element e, its relative rank is revealed by comparing it with the elements that have already arrived. An algorithm ALG selecting a (random) set S is said to be *α-probability-competitive* if every element of $\mathrm{OPT}(E)$, the lexicographically maximal independent set of E with respect to $\succ$, is included in S with probability at least α. The elements in $\mathrm{OPT}(E)$ are referred to as *optimal elements*.

3 The Labeling Scheme Framework

3.1 Improving Elements and Times

In this section, we describe our new approach to the matroid secretary problem using labeling schemes. First, we remark that an equivalent way to model a uniformly random order is to assume that every element $e \in E$ picks an arrival time t_e from the uniform distribution over $[0, 1]$, and the algorithm observes elements one by one in increasing arrival time. Although the arrival times are not part of the input, our algorithms for matroids with n elements can simulate this process by first generating n independent, uniformly distributed arrival times in $[0, 1]$. These *arrival times* are sorted as $s_1 < \cdots < s_n$ and assigned to elements as they are being presented; that is, if e is the i-th arriving element, then $t_e = s_i$. For every positive $t \in (0, 1]$, let E_t denote the set of edges arriving in the interval $[0, t)$ and E_t^+ the set of elements arriving in the interval $[0, t]$.

Definition 1 (Improving Elements and Improving Times)**.** An element $e \in E$ is improving if $e \in \mathrm{OPT}(E_{t_e}^+) = \mathrm{OPT}(E_{t_e} + e)$. The times at which improving elements arrive are called *improving times*.

Contrary to arrival times, we usually assume they are sorted in decreasing order, i.e., for a set of improving times $\{t_1, t_2, \ldots, t_k\}$, we have $t_k < t_{k-1} < \cdots < t_1$. For $0 \leq a < b \leq 1$, we denote the *last improving time* in increasing order in $[0, b)$ by $S(b)$, and the *number of improving times* in $[a, b)$ or $[a, b]$ by $N[a, b)$ and $N[a, b]$, respectively. The following is a folklore lemma for matroids (see [29, Lemma 2] for a proof), and allows us to restrict our analysis to algorithms for the matroid secretary problem that always skip non-improving elements.

Lemma 1. *If $X \subseteq Y$ and $e \in \mathrm{OPT}(Y + e)$, then $e \in \mathrm{OPT}(X + e)$. In particular, every optimal element $e \in \mathrm{OPT}(E)$ is improving.*

Our algorithms skip all elements with arrival time before a certain parameter $p \in (0, 1)$, which has the same effect as choosing a value $n_0 \sim \mathrm{Bin}(n, p)$ from the binomial distribution of parameters n and p and then skipping the first n_0 elements. Among the elements that arrive after time p, our algorithms also skip all non-improving elements. In particular, the output of all our algorithms will be a subset of the improving elements arriving in the interval $[p, 1]$. The following assumption is not necessary for our algorithms, but it simplifies their analysis.

Assumption 1. The input matroid is augmented in such a way that, at every time t, the set E_t has full rank. Every new element added in this way is added at the end of the value order.

In the context of laminar matroids, one can achieve this in the following way: for every element $f \in B_0$ in turn, we create dummy copies $f_1, f_2, \ldots$ of f one by one. To each copy f_i, we assign an arrival time s_{f_i} chosen uniformly from $[0,1]$. As soon as one of these copies has arrival time in $[0,p)$, we stop creating copies of f, we extend the value order to the copies such that $f_1 \succ\succ (f_2) \succ \dots$, even though all of these elements are at the end of the value order of the original matroid, and continue with the next element in B_0. Note that for each f, we only create $\lceil 1/p \rceil$ copies in expectation. Let F' be the set of all copies created via this process. The sets of improving elements and improving times coincide for both the instance in which F is added and the instance where only F' is added. Hence, the output of our algorithms will be the same for both instances. For graphic matroids, more care is required and simply making copies of each element is not enough to implement the assumption; one way to do it is described in Assumption 2. We remark that this assumption is made without loss of generality and it is only used for the analysis of our algorithms.

One of the main tools we use is the following lemma, which characterizes the distribution of the arrival times of improving elements as a Poisson distribution.

Lemma 2. *Under Assumption 1, the following hold.*

(i) *For every $b \in \mathbb{R}_+$, $S(b)$ has cumulative distribution function* $\Pr[S(b) \le x] = 0$ *if* $x < 0$, $\Pr[S(b) \le x] = \left(\frac{x}{b}\right)^r$ *if* $x \in [0,b)$, *and* $\Pr[S(b) \le x] = 0$ *if* $x \ge b$.

(ii) *For all $k \in \mathbb{N}$, let $x_k = -\ln(y_k)$ where $y_0 = 1$ and $y_k = S(y_{k-1})$ otherwise.*

(a) *The arrival times $x_k - x_{k-1}$ follow an exponential distribution with mean $1/r$. In particular, $\{x_k\}_{k\in\mathbb{N}}$ is a Poisson point process restricted to $\mathbb{R}_+$ with rate r.*

(b) *$N[a,b)$ distributes as a Poisson random variable with rate $\lambda = r\ln(b/a)$.*

(c) *For any finite family of disjoint intervals $\{[a_i,b_i)\}_i$, the random variables $\{N[a_i,b_i)\}_i$ are mutually independent.*

Proof. (i). We follow the approach of [16]. Recall that E_b is the set of elements arriving in $[0,b)$. Conditioned on the set $E_b = X$, the last improving element in $[0,b)$ is a uniformly chosen random element of $\mathrm{OPT}(E_b)$. Therefore, its arrival time is $S(b) = \max\{t_e : e \in \mathrm{OPT}(X)\}$. First, notice that, by definition, $S(b) \in [0,b)$. Also, since the arrival times are independent,

$$\Pr[S(b) \le x] = \prod_{e\in \mathrm{OPT}(X)} \Pr(t_e \le x) = \left(\frac{x}{b}\right)^r,$$

for every $x \in [0,b)$.

(ii). For condition (a), notice that for $x \in [0,+\infty)$ we have

$$\begin{aligned}
\Pr[x_k - x_{k-1} \le x] &= \Pr[-\ln y_k + \ln y_{k-1} \le x] = \Pr\left[\ln\left(\frac{y_{k-1}}{y_k}\right) \le x\right] \\
&= \Pr\left[\ln\left(\frac{y_{k-1}}{S(y_{k-1})}\right) \le x\right] = \Pr\left[S(y_{k-1}) \ge y_{k-1}e^{-x}\right] \\
&= 1 - \Pr\left[S(y_{k-1}) < y_{k-1}e^{-x}\right] = 1 - \left(\frac{y_{k-1}e^{-x}}{y_{k-1}}\right)^r \\
&= 1 - e^{-xr}.
\end{aligned}$$

In other words, for every positive $k \in \mathbb{N}$, $x_k - x_{k-1}$ is an exponential random variable of rate r. We conclude that $\{x_k\}_{k\in\mathbb{N}}$ is a Poisson point process with rate r on the positive half-line.

Condition (b) holds since $N[a,b)$ counts the number of points in $\{y_k\}_{k\in\mathbb{N}}$ that lie in $[a,b)$, or equivalently, the number of points in the Poisson process $\{x_k\}_{k\in\mathbb{N}}$ that lie in $(-\ln(b), -\ln(a)]$, and this behaves like a Poisson variable of rate r multiplied by the length of the interval $(-\ln(b), -\ln(a)]$, i.e., like a Poisson with rate $r\ln\left(\frac{b}{a}\right)$. Finally, condition (c) follows from the complete independence of the Poisson process. □

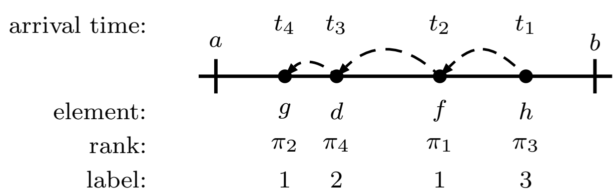

Figure 1: A sequence of improving element arrivals in $[a, b]$. Assume that $E_{t_4} = \{g\}$, $E_{t_3} = \{g, d\}$, $E_{t_2} = \{g, d, f\}$, and $E_{t_1} = \{g, d, h\}$. If the elements are ranked as $f >_\pi g >_\pi > h >_\pi d$, the improving word induced by π in $[a, b]$ is $z = 3121$.

3.2 Labeling Schemes and Improving Words

Our main contribution is a new way of analyzing algorithms based on assigning labels from $[r]$ to the elements of OPT at improving times, and then analyzing the *word* formed by the labels of the improving elements.

Definition 2 (Labeling Scheme)**.** A *labeling scheme* Λ *for an interval* $[a, b]$ assigns, for each improving time $t \in [a, b]$, a unique label from $[r]$ to each element of the set $X = \mathrm{OPT}(E_t^+)$.

It is worth emphasizing that the labels are assigned independently of the arrival order of E_t. Also note that an element's label may change from improving time to improving time and can depend on the set of improving elements and times in $(t, 1]$. We often use the following labeling scheme.

Definition 3 (Labeling Scheme Induced by a Total Order)**.** Let π be a total order on E. For every $t \in [0, 1]$, the *labeling scheme* Λ_π *induced by* π assigns a label to each $e \in \mathrm{OPT}(E_t^+)$ equal to the relative rank of e in $\mathrm{OPT}(E_t^+)$ with respect to π.

In other words, for $X = \mathrm{OPT}(E_t^+)$, the labeling scheme assigns label 1 to the element of X that appears first in π, label 2 to the element of X that appears next, and so on. Note that even for this simple type of labeling scheme, the same element may have different labels at different times. Given a labeling scheme Λ, we say that an *improving time t has label i*, denoted as $z(t) = i$, if the improving element arriving at time t receives label i. We define the following.

Definition 4 (Improving Word in Time Interval $[a, b]$)**.** Let $0 \le a < b \le 1$ and $\{e_k, \dots, e_1\}$ be the set of improving elements that arrive in time interval $[a, b]$ with improving times $t_j = t_{e_j}$, where the elements are indexed in a decreasing order with respect to their arrival times, i.e., $t_k < \cdots < t_1$. The *improving word in* $[a, b]$ is $z = z_1 \dots z_k \in [r]^k$, where $z_i = z(t_i)$ is the *label of e_i in* $\mathrm{OPT}(E_{t_i}^+)$.

For an illustration of the previous definition, see Fig. 1. Whenever we apply a labeling scheme Λ and the resulting improving word in an interval $[a, b]$ is z, we say that Λ *produces z in* $[a, b]$. The uniformly random arrival order of the elements implies the labels of z are chosen uniformly at random from $[r]$ and the length of z follows a Poisson distribution. Intuitively, the main idea of our analysis technique is to design a language $\mathcal{L}$, depending on the matroid class and rank, such that, for every optimal element $e^* \in \mathrm{OPT}(E)$ we can find a labeling scheme Λ which ensures that e^* is selected by the algorithm whenever the improving word produced by Λ in $[p, 1]$ belongs to $\mathcal{L}$. We call such a language an *improving language*. Next, we state and prove some technical results regarding labeling schemes and improving words used in analyzing our algorithms.

For a set S of labels, we denote by $N_S[a, b)$ and $N_S[a, b]$ the number of improving times in $[a, b)$, and $[a, b]$ respectively, for which the label belongs in S.

Lemma 3. *Consider a labeling scheme Λ. Let $S \subseteq [r]$ be a subset of labels and $t \in (0, 1]$ and $\Lambda(e_t)$ denote the label assigned by Λ to an element arriving at time t.*

(i) *If $x < t$ is an improving time, then* $\Pr[\Lambda(e_x) \in S] = \frac{|S|}{r}$. *Furthermore, this event is independent of the label of any improving time in (x, t).*

(ii) *The set $\{-\ln(s)\colon s \in T_S(t)\}$ is a Poisson point process in $(-\ln(t), \infty)$. In particular, for every $0 < a < b \leq t$, $N_S[a, b)$ is distributed as a Poisson random variable with rate $\lambda = |S| \ln\left(\frac{b}{a}\right)$.*

(iii) *For any finite family of disjoint intervals $\{[a_i, b_i)\}_i$ and arbitrary – not necessarily disjoint – subsets $\{S_i\}_i$ of $[r]$, the random variables $\{N_{S_i}[a_i, b_i)\}_i$ are mutually independent.*

(iv) *For any family of disjoint subsets $\{S_i\}_i$ of $[r]$ and arbitrary – not necessarily disjoint – intervals $\{[a_i, b_i)\}_i$, the random variables $\{N_{S_i}[a_i, b_i)\}_i$ are mutually independent.*

Proof. (i). Observe that for any improving time x, conditioned on the set $X = E^+_x$ of elements that have arrived in the interval $[0, x]$, the element that arrives at time x is chosen uniformly at random from $\mathrm{OPT}(E^+_x)$. Since Λ is independent of the internal arrival ordering of E^+_x, the label of x is uniformly chosen from $[r]$. Therefore, the probability that the label is in S is exactly $|S|/r$.

(ii) – (iv). Each statement follows directly by the coloring property of Poisson point processes; see [7, Pages 163–164] for further details. □

The following lemma is a direct consequence of Lemma 3 when applied for improving words, hence its proof is omitted.

Lemma 4. *Let z be the random variable representing the improving word produced in the interval $[a, b]$, $|z|$ be its length, and $\lambda = r \ln\left(\frac{b}{a}\right)$. Furthermore, let $S, T \subseteq [r]$ be two disjoint subsets of the labels, z_S (resp. z_T) be the subsequence of z obtained by deleting all symbols not in S (resp. not in T), and let $\lambda_S = |S| \ln\left(\frac{b}{a}\right)$.*

(i) *For every $k \in \mathbb{N}$,* $\Pr[|z| = k] = \Pr[N[a, b) = k] = \frac{\lambda^k e^{-\lambda}}{k!}$.

(ii) *Conditioned on $|z| = k$, z is a uniform random word in $[r]^k$.*

(iii) *For every $k \in \mathbb{N}$,* $\Pr[|z_S| = k] = \Pr[N_S[a, b) = k] = \frac{(\lambda_S)^k e^{-\lambda_S}}{k!}$.

(iv) *Conditioned on $|z_S| = k$, z_S is a uniform random word in S^k.*

(v) *z_S and z_T are independent.*

The following lemma is useful in characterizing the improving words that guarantee the selection of a specific element of OPT by our algorithms.

Lemma 5. *Let $e^* \in \mathrm{OPT}(E)$ and π be any total order of E for which $\pi_1 = e^*$. Let $p \leq t_k < \cdots < t_1 \leq 1$ be the improving times in $[p, 1]$ and $z = z_1 z_2 \ldots z_k$ be the associated improving word in $[p, 1]$ with respect to the labeling Λ_π.*

(i) *z contains a symbol 1 if and only if the arrival time of e^* is in $[p, 1]$.*

(ii) *If z contains a symbol 1 and $j = \arg\min\{\ell \in [k]\colon z_\ell = 1\}$, then $t_{e^*} = t_j$.*

Proof. (i). By Lemma 1 we know that e^* is improving, and thus $e^* \in \mathrm{OPT}(E^+_{t_{e^*}})$. Also, $\pi_1 = e^*$, and thus if $t_{e^*} \in [p, 1]$, then Λ_π assigns a label of 1 to e^* upon arrival and thus z contains a 1. Furthermore, if $t_{e^*} \in [0, p)$, then for any other improving element e arriving at $t_e \in [p, 1]$, the relative rank of e in $\mathrm{OPT}(E^+_{t_e})$ according to π cannot be 1, since $e^* \in \mathrm{OPT}(E^+_{t_e})$ and $\pi_1 = e^*$. Thus, z does not contain a 1.
(ii). Assume that z contains a 1. By the first part, this implies that the arrival time of e^* is in $[p, 1]$. Since the label of e^* in $\mathrm{OPT}(E^+_{t_{e^*}})$ is 1, t_{e^*} is among $\{t_\ell\colon \ell \in [k], z_\ell = 1\}$. Suppose that $t_{e^*} = t_i$ for some $i > j = \arg\min\{\ell \in [k]\colon z_\ell = 1\}$, and let e denote the improving element arriving at t_j. Then, since $t_j > t_i$, both e^* and e are contained in $\mathrm{OPT}(E^+_{t_j})$. However, this means that the rank of e is at least 2 in $\mathrm{OPT}(E_{t_j})$ yielding a contradiction. □

3.3 Algorithms via Improving Languages

In our framework, after skipping the elements arriving before p, we select an improving element if the sequence of different optimum sets we have seen (which correspond to the optimum sets at improving times) satisfies a property P, where P is tailored to the specific matroid class and rank r. If P holds when an element e arrives, P must guarantee that $\mathrm{ALG} + e \in \mathcal{I}$. To analyze such algorithm, our main goal is to find, for every $e^* \in \mathrm{OPT}(E)$, a labeling scheme $\Lambda := \Lambda(e^*)$ and a language $\mathcal{L} \subseteq [r]^*$ such that, if $z \in [r]^*$ is the improving word in $[p, 1]$ induced by the labeling scheme Λ, then $z \in \mathcal{L} \implies e^* \in \mathrm{ALG}$. The significance of the objective comes from the fact that if the improving languages $\mathcal{L}$ for several classes of matroids and that $\Pr[z \in \mathcal{L}]$ is much easier to compute due to z coming from a Poisson point process. In the following sections, we exploit this approach for laminar (Section 4) and graphic (Section 5) matroids.

4 Laminar Matroids

In this section, we apply the labeling framework to obtain an improved analysis of the well-known Greedy-Improving algorithm for laminar matroids. In other words, for laminar matroids, we let the property P simply be that $\mathrm{ALG} + e \in \mathcal{I}$.

Algorithm 1 Greedy-Improving Algorithm

1: Initialize $\mathrm{ALG}_0 \leftarrow \emptyset$.
2: Skip all elements with arrival time in $[0, p)$.
3: **for** each element e with $t_e \in [p, 1]$, in arrival order **do**
4:     **if** $e \in \mathrm{OPT}(t_e)$ and $\mathrm{ALG}_0 + e \in \mathcal{I}$ **then**
5:         $\mathrm{ALG}_0 \leftarrow \mathrm{ALG}_0 + e$
6: Return ALG_0.

As an illustration of our techniques, we first consider, in Section 4.1, the special case of uniform matroids and characterize the probability-competitiveness of the algorithm (Proposition 4). Then, we generalize our analysis for the broader class of laminar matroids in Section 4.2, showing that Greedy-Improving is $(1 - \ln(2))$-probability-competitive for every laminar matroid (Theorem 1); we prove this bound is tight, thereby settling the probability-competitiveness of Greedy-Improving completely. In Section 4.3, we design a new algorithm for rank-2 matroids (a subclass of the laminar family) that outperforms Greedy-Improving and provides a novel approach (Theorem 2). Recall that, in the context of laminar matroids, Assumption 1 corresponds to augmenting the matroid with a large set F of parallel copies of a single basis B_0 (i.e., each element of B_0 has a countably infinite number of parallel copies).

4.1 Uniform Matroids

As a warm-up, we present a new analysis of Greedy-Improving for the special class of uniform matroids using labeling schemes. For a rank-r uniform matroid on a ground set of size n, the independent sets are all the subsets of the ground set that have size at most r. It is worth noting that the algorithm is not optimal for uniform matroids; we describe it to aid the reader and illustrate our analysis. When $\mathcal{M} = (E, \mathcal{I})$ is a rank-r uniform matroid on n elements, the condition $\mathrm{ALG}_0 + e \in \mathcal{I}$ in Step 4 is equivalent to $|\mathrm{ALG}_0| \leq r - 1$.

Definition 5 (Labeling Scheme for Uniform Matroids)**.** Let $e^* \in \mathrm{OPT}(E)$. Consider a total order π of E in which e^* appears first. We set $\Lambda_{\mathrm{uniform}}(e^*)$ to be the labeling scheme induced by π as in Definition 3.

Lemma 6. *Let z be the improving word in $[p, 1]$ produced by the labeling scheme $\Lambda_{uniform}(e^*)$, ALG_0 be the output of* Greedy-Improving*, and $\mathcal{L}_{uniform} = \{x1y \in [r]^* \colon x \in ([r] \setminus \{1\})^* \text{ and } |y| \leq r-1\}$. Then $e^* \in \mathrm{ALG}_0$ if and only if $z \in \mathcal{L}_{uniform}$.*

Proof. By Lemma 5, $e^* \in \mathrm{ALG}_0$ if and only if z contains a symbol 1 and the first appearance of 1 in z is on one of the last r positions of z. This is equivalent to $z \in \mathcal{L}_{\mathrm{uniform}}$. □

Let

$$c(r,p) = \begin{cases} -p\ln(p) & \text{if } r = 1 \\ \dfrac{1-\left(1-\frac{1}{r}\right)^r}{\left(1-\frac{1}{r}\right)^r} p + p^r \displaystyle\sum_{k=0}^{r-1} \frac{\left(r\ln\left(\frac{1}{p}\right)\right)^k}{k!} - \frac{p^r}{\left(1-\frac{1}{r}\right)^r} \sum_{k=0}^{r-1} \frac{\left((r-1)\ln\left(\frac{1}{p}\right)\right)^k}{k!} & \text{if } r > 1 \end{cases}$$

Next, we show that $\Pr[z \in z \in \mathcal{L}_{\mathrm{uniform}}]$ is exactly $c(r,p)$. Using that we obtain the following proposition, which gives an exact formula that is easy to evaluate for small r – see Table 1 in Appendix B.

Proposition 4. *Algorithm 1 is exactly $c(r,p)$-probability-competitive for rank-r uniform matroids. Furthermore, as $r \to \infty$, the best sample size in Algorithm 1 for uniform matroids is $1/e$, achieving a probability-competitiveness of $1 - 1/e$.*

Proof. By Lemma 6,

$$\Pr[e^* \in \mathrm{ALG}_0] = \Pr[z = x1y \text{ with } x \in ([r] \setminus \{1\})^*, |y| \leq r-1].$$

Let $\lambda = r\ln(1/p) = -r\ln(p)$ and let us apply Lemma 4. For $r = 1$, we get $\Pr[e^* \in \mathrm{ALG}_0] = \Pr[z = 1] = \frac{\lambda^1 e^{-\lambda}}{1!} = -p\ln p$. For $r \geq 2$, we have

$$\begin{aligned} &\Pr[e^* \in \mathrm{ALG}_0] \\ &= \sum_{k=1}^{\infty} \sum_{m=0}^{\min(k,r)-1} \Pr[z = x1y, |z| = k, |y| = m, \text{ with } x \in ([r] \setminus \{1\})^{k-m-1}] \\ &= \sum_{k=1}^{\infty} \frac{\lambda^k e^{-\lambda}}{k!} \sum_{m=0}^{\min(k,r)-1} \left(1 - \frac{1}{r}\right)^{k-m-1} \cdot \frac{1}{r} \\ &= \sum_{k=1}^{\infty} \frac{\lambda^k e^{-\lambda}}{k!} \left(\left(1-\frac{1}{r}\right)^{k-\min(k,r)} - \left(1-\frac{1}{r}\right)^k\right) \\ &= \sum_{k=0}^{r-1} \frac{\lambda^k e^{-\lambda}}{k!} \left(1 - \left(1-\frac{1}{r}\right)^k\right) + \sum_{k=r}^{\infty} \frac{\lambda^k e^{-\lambda}}{k!} \left(\left(1-\frac{1}{r}\right)^{k-r} - \left(1-\frac{1}{r}\right)^k\right). \end{aligned}$$

The second equality holds since the probability that $z_{k-m} = 1$ is $1/r$ and the probability that $x \in ([r] \setminus \{1\})^{k-m-1}$ is $\left(1-\frac{1}{r}\right)^{k-m-1}$.

Let $\lambda' = \lambda\,(1 - {}^1\!/\!_r) = (r-1)\ln({}^1\!/\!_p)$ and note that $\lambda - \lambda' = {}^\lambda\!/\!_r = \ln({}^1\!/\!_p)$. Denote by $P(\mu)$ a random variable distributed as a Poisson with parameter μ for any μ. Then,

$$\begin{aligned}
&\Pr[e^* \in \mathrm{ALG}_0]\\
&= \Pr[P(\lambda) \le r-1] - \sum_{k=0}^{r-1} \frac{(\lambda')^k e^{-\lambda}}{k!} + \frac{1-\left(1-\frac{1}{r}\right)^r}{\left(1-\frac{1}{r}\right)^r} \sum_{k=r}^{\infty} \frac{(\lambda')^k e^{-\lambda}}{k!}\\
&= \Pr[P(\lambda) \le r-1] - e^{-\lambda+\lambda'} \Pr[P(\lambda') \le r-1]\\
&\quad + \frac{1-\left(1-\frac{1}{r}\right)^r}{\left(1-\frac{1}{r}\right)^r} e^{-\lambda+\lambda'} \Pr[P(\lambda') \ge r]\\
&= \Pr[P(\lambda) \le r-1] - p \Pr[P(\lambda') \le r-1]\\
&\quad + \frac{1-\left(1-\frac{1}{r}\right)^r}{\left(1-\frac{1}{r}\right)^r} p(1 - \Pr[P(\lambda') \le r-1])\\
&= \frac{1-\left(1-\frac{1}{r}\right)^r}{\left(1-\frac{1}{r}\right)^r} p + \Pr[P(\lambda) \le r-1] - \frac{p}{\left(1-\frac{1}{r}\right)^r} \Pr[P(\lambda') \le r-1]\\
&= \frac{1-\left(1-\frac{1}{r}\right)^r}{\left(1-\frac{1}{r}\right)^r} p + p^r \sum_{k=0}^{r-1} \frac{\left(r\ln\left(\frac{1}{p}\right)\right)^k}{k!} - \frac{p^r}{\left(1-\frac{1}{r}\right)^r} \sum_{k=0}^{r-1} \frac{\left((r-1)\ln\left(\frac{1}{p}\right)\right)^k}{k!},
\end{aligned}$$

which concludes the proof of the first part of the proposition.

Recall that for $r > 1$,

$$c(r,p) = \frac{1-\left(1-\frac{1}{r}\right)^r}{\left(1-\frac{1}{r}\right)^r} p + \Pr[P(\lambda) < r] - \frac{p}{\left(1-\frac{1}{r}\right)^r} \Pr[P(\lambda') < r],$$

and that the mean and variance of $P(\lambda)$ are both λ. For $p > {}^1\!/\!_e$, we can write ${}^1\!/\!_{\ln({}^1\!/\!_p)} = 1 + \varepsilon$ for some $\varepsilon > 0$, and so $r = \lambda(1+\varepsilon)$. Using Chebyshev's inequality and that $\lambda' < \lambda$, we get

$$\begin{aligned}
\Pr[P(\lambda') < r] &> \Pr[P(\lambda) < r]\\
&= 1 - \Pr[P(\lambda) \ge \lambda + \lambda\varepsilon] \ge 1 - \frac{1}{\lambda\varepsilon^2} = 1 - \frac{1}{r\ln\left(\frac{1}{p}\right)\varepsilon^2},
\end{aligned}$$

which goes to 1 as $r \to \infty$. We conclude that, for $p > {}^1\!/\!_e$,

$$\lim_{r\to\infty} c(r,p) = 1 + p \lim_{r\to\infty} \left(\frac{1-\left(1-\frac{1}{r}\right)^r}{\left(1-\frac{1}{r}\right)^r} - \frac{1}{\left(1-\frac{1}{r}\right)^r} \right) = 1 - p.$$

On the other hand, if $p < {}^1\!/\!_e$ then ${}^1\!/\!_{\ln({}^1\!/\!_p)} = 1 - \varepsilon$ for some $\varepsilon > 0$ and $r - 1 = \lambda'(1-\varepsilon)$. Then,

$$\begin{aligned}
\Pr[P(\lambda) < r] &< \Pr[P(\lambda') \le r-1]\\
&= \Pr[P(\lambda') \le \lambda' - \lambda'\varepsilon] \le \frac{1}{\lambda'\varepsilon^2} = \frac{1}{(r-1)\ln({}^1\!/\!_p)\,\varepsilon^2},
\end{aligned}$$

which goes to 0 as $r \to \infty$. We conclude that, for $p < {}^1\!/\!_e$,

$$\lim_{r\to\infty} c(r,p) = p \lim_{r\to\infty} \left(\frac{1-\left(1-\frac{1}{r}\right)^r}{\left(1-\frac{1}{r}\right)^r} \right) = p(e-1).$$

By the above, the limit curve for the probability-competitiveness of GREEDY-IMPROVING when $r \to \infty$ and p is a piecewise linear continuous function that increases from 0 at $p = 0$ to $1 - 1/e$ at $p = 1/e$, and then it decreases linearly to 0 again at $p = 1$. This shows that the best sample size is $1/e$ as stated. □

4.2 General Laminar Matroids

A *laminar matroid* is defined on a ground set with a laminar family of subsets. This family is organized into layers, where the sets in the i-th layer are pairwise disjoint, have rank i, and sets in lower layers are nested within sets in higher layers. In this section, we show the following theorem.

Theorem 1. *The greedy-improving algorithm is* $(1 - \ln(2)) \approx 0.3068$*-probability-competitive for the MSP on laminar matroids. In addition, for every* $\varepsilon > 0$*, there exists a laminar matroid* $\mathcal{M}(\varepsilon)$ *such that the probability-competitive ratio of the greedy-improving algorithm on* $\mathcal{M}(\varepsilon)$ *is at most* $1 - \ln(2) + \varepsilon$*.*

In order to prove this result, we extend the analysis of GREEDY-IMPROVING from uniform to laminar matroids. We consider the lower bound and upper bound of the theorem separately.

Lower Bound in Theorem 1. For uniform matroids, our analysis works for total order π of the elements on which e^* appears first. In the laminar case, however, we need to consider specific orders that depends on the laminar structure.

Definition 6 (Chain Order)**.** Let $e^* \in \mathrm{OPT}(E)$ and $C_1 \subseteq \cdots \subseteq C_q = E$ be the members of the laminar family containing e^*. We refer to these sets as the *chain induced by* e^*. We say that a total order $(\pi_1, \ldots, \pi_n)$ of E is a *chain order with respect to* e^* if $\pi_1 = e^*$ and for $1 \leq i < j \leq r$, we have $\{\ell \in [q] \colon \pi_i \in C_\ell\} \subseteq \{\ell \in [q] \colon \pi_j \in C_\ell\}$. In other words, in the enumeration π, e^* appears first, then the rest of the elements from $S \cap C_1$, then all the elements from $(S \cap C_2) \setminus C_1$, then all the elements from $(S \cap C_3) \setminus C_2$ and so on.

Definition 7 (Chain Labeling Scheme for Laminar Matroids)**.** Let $e^* \in \mathrm{OPT}(E)$ be an element of the optimal basis and π be a chain order with respect to e^*. We set $\Lambda^{\pi}_{\text{chain}}(e^*)$ as the labeling scheme induced by π. Since the choice of π is arbitrary, we write $\Lambda_{\text{chain}}(e^*)$ to denote a labeling scheme induced by an arbitrary but fixed chain order π with respect to e^*.

Definition 8 (Well-indexed Word)**.** We say that a word z is *well-indexed* if it satisfies the following:

1. There exists a unique symbol 1 in z; i.e. $z = x1y$ with $x, y \in \{[r] \setminus \{1\}\}^*$.
2. For each $c \in [|y|]$, it holds that

$$|\{i \in [|y|] \colon y_i \leq c\}| \leq c - 1. \tag{1}$$

   In particular, since $1 \leq y_i \leq r$ for all i, this condition implies $|y| \leq r - 1$.

Denote by $\mathcal{L}_{\text{laminar}} = \{z \in [r]^* \colon z \text{ is well-indexed}\}$.

Lemma 7. *Let* z *be the improving word of* $e^* \in \mathrm{OPT}(E)$ *produced by the labeling scheme* $\Lambda_{chain}(e^*)$*. If* $z \in \mathcal{L}_{laminar}$ *then* $e^* \in \mathrm{ALG}_0$*.*

Proof. Let z be the improving word of e^* and suppose that $z = x1y$ is well-indexed, where the symbol 1 refers to the entry of z corresponding to e^*. Then, by Lemma 5, the arrival time t_{e^*} of e^* is in $[p, 1]$ and there are exactly $m := |y|$ improving elements arriving in $[p, t)$. Let us denote these elements by $f_m, \dots, f_1$ with $t(f_m) < \dots < t(f_1)$, so that the label of f_i is y_i for all $i \in [m]$.

Let $\mathrm{ALG}_- \subseteq \{f_m, \dots, f_1\}$ denote the set of elements chosen by the algorithm before the arrival of e^*. Take an arbitrary member C of the chain induced by e and let $c = r(C)$ be its rank. Let $s < t_{e^*}$ and sort the elements of $\mathrm{OPT}(E_s)$ increasingly by their labels, i.e., by the chain order π with respect to e^*. Since all elements of $\mathrm{OPT}(E_s) \cap C$ appear before the elements of $\mathrm{OPT}(E_s) \setminus C$ in this order, we conclude that all elements with label $c + 1$ or higher are not in C. Since this is true also for all improving times s before t_{e^*}, we get that $C \cap \mathrm{ALG}_- \subseteq \{f_i : y_i \le c\}$. As z is well-indexed, we have

$$|C \cap \mathrm{ALG}_-| \le \{i \in [|y|] : y_i \le c\} \le c - 1.$$

We conclude that $|C \cap (\mathrm{ALG}_- + e^*)| \le 1 + |C \cap \mathrm{ALG}_-| \le r(C)$ for each C in the chain induced by e^*, therefore $\mathrm{ALG}_- + e^*$ is independent, and thus e^* is added to ALG_0. $\square$

Thus, to lower bound the competitiveness of Algorithm 1, it suffices to compute the probability that the improving word z produced in $[p, 1]$ is well-indexed.

Lemma 8. *Let $m \le r - 1$ and $y \in [r]^m$ be a word chosen uniformly at random from $[r]^m$. Then, the probability that condition (1) holds is equal to $1 - m/r$.*

Proof. Let $m \ge 0, r \ge 1$ be integers. We call a sequence $b \in [r]^m$ *good* if it satisfies (1) and *bad* otherwise. Let $B(m, r)$ denote the number of bad sequences. Then, $\Pr[b \text{ is good}] = 1 - B(m, r)/r^m$. We claim that

$$B(m, r) = \begin{cases} mr^{m-1} & \text{if } 0 \le m \le r - 1, \\ r^m & \text{otherwise.} \end{cases}$$

If $m \ge r$ then any sequence in $[r]^m$ will contain $m > r - 1$ indices that are at most r, so every sequence is bad and $B(m, r) = r^m$. Assume now that $m \le r - 1$. The rest of the proof is by induction on r. For the base case, observe that if $m = 0$, then the empty sequence is good by definition, so $B(0, 1) = 0$. Suppose now that $r \ge 2$, and let us count the number of bad sequences in $[r]^m$ for some $1 \le m \le r - 1$. Since the length m of the sequence is at most $r - 1$, any sequence $b \in [r]^m$ is bad if and only if the subsequence b' of b induced by those indices j for which $b_j \le r - 1$ is bad. Hence

$$B(m, r) = \sum_{j=0}^{m} \binom{m}{j} B(j, r - 1) = \sum_{j=1}^{m} \binom{m}{j} j(r - 1)^{j-1} = \sum_{j=1}^{m} m\binom{m - 1}{j - 1}(r - 1)^{j-1} = mr^{m-1},$$

where the first equality follows by the induction hypothesis, and the last one follows from the Binomial Theorem. The lemma then follows since if $m = 0$, the probability of selecting a good sequence is 1, and if $1 \le m \le r$, this probability is $1 - mr^{m-1}/r^m = 1 - m/r$. $\square$

For any $p \in (0, 1)$, let us define $a(r, p) = -p\ln(p)$ if $r = 1$ and

$$\begin{aligned} a(r, p) = & -2p + (2 + \ln p)\Pr[P(-r\ln p) < r - 1] \\ & + 2\Pr[P(-r\ln p) = r - 1] + \frac{p}{\left(1 - \frac{1}{r}\right)^r}\Pr[P(-(r - 1)\ln p) \ge r], \end{aligned}$$

for $r \ge 2$, where $P(\mu)$ is a Poisson random variable with rate μ.

Lemma 9. *Algorithm 1 is at least $a(r,p)$-probability-competitive for rank-r laminar matroids.*

Proof. Let $e^* \in \mathrm{OPT}(E)$ be any element of the optimal basis and z be the improving word of e^*. By Lemma 7, it suffices to show that $a(r,p) \triangleq \Pr[z \text{ is well-indexed}]$. By setting $\lambda = r\ln(1/p)$, we have

$$\begin{aligned}
a(r,p) &= \sum_{k=1}^{\infty} \frac{\lambda^k e^{-\lambda}}{k!} \Pr[z = x1y \text{ is well-indexed} \mid |z| = k] \\
&= \sum_{k=1}^{\infty} \sum_{m=0}^{\min(k,r)-1} \frac{\lambda^k e^{-\lambda}}{k!} \Pr[z = x1y \text{ is well-indexed and } |y| = m \mid |z| = k] \\
&= \sum_{k=1}^{\infty} \frac{\lambda^k e^{-\lambda}}{k!} \sum_{m=0}^{\min(k,r)-1} \frac{1}{r} \cdot \left(1 - \frac{1}{r}\right)^{k-m-1} \left(1 - \frac{m}{r}\right),
\end{aligned}$$

since $\left(1-\frac{1}{r}\right)^{k-m-1}$ is the probability that z contains no 1 symbol, $1/r$ is the probability that the symbol at position $m+1$ from the right is a 1, and $1 - m/r$ is the probability that y is good by Lemma 8. For $r = 1$, this simplifies to $a(r,p) = \frac{\lambda^1 e^{-\lambda}}{1!} = -p\ln(p)$. For $r \geq 2$, define $\lambda' = \lambda(r-1)/r = (r-1)\ln(1/p)$. We obtain that

$$\begin{aligned}
a(r,p) &= \sum_{k=1}^{\infty} \frac{\lambda^k e^{-\lambda}}{k!} \frac{1}{r} \left(\frac{r-1}{r}\right)^{k-1} \sum_{m=0}^{\min(k,r)-1} \left(\frac{r}{r-1}\right)^m \left(1 - \frac{m}{r}\right) \\
&= \sum_{k=1}^{\infty} \frac{\lambda^k e^{-\lambda}}{k!} \frac{1}{r} \left(\frac{r-1}{r}\right)^{k-1} \left(2 - 2r + \left(\frac{r}{r-1}\right)^{\min(k,r)-1} (2r - \min(k,r))\right) \\
&= \frac{(2-2r)}{r-1} e^{-\lambda+\lambda'} \sum_{k=1}^{\infty} \frac{(\lambda')^k e^{-\lambda'}}{k!} \\
&\quad + \sum_{k=1}^{r-1} \frac{\lambda^k e^{-\lambda}}{k!} \frac{1}{r}(2r-k) + \sum_{k=r}^{\infty} \frac{\lambda^k e^{-\lambda}}{k!} \left(\frac{r-1}{r}\right)^{k-r} \\
&= -2e^{-\lambda+\lambda'}(1 - e^{-\lambda'}) + 2\sum_{k=1}^{r-1} \frac{\lambda^k e^{-\lambda}}{k!} - \sum_{k=1}^{r-1} \frac{\lambda^k e^{-\lambda}}{k!} \frac{k}{r} \\
&\quad + \left(\frac{r-1}{r}\right)^{-r} e^{-\lambda+\lambda'} \sum_{k=r}^{\infty} \frac{(\lambda')^k e^{-\lambda'}}{k!} \\
&= -2e^{-\lambda+\lambda'} + 2e^{-\lambda} + 2\sum_{k=1}^{r-1} \frac{\lambda^k e^{-\lambda}}{k!} - \frac{\lambda}{r} \sum_{k=1}^{r-1} \frac{\lambda^{k-1} e^{-\lambda}}{(k-1)!} \\
&\quad + \left(\frac{r-1}{r}\right)^{-r} e^{-\lambda+\lambda'} \sum_{k=r}^{\infty} \frac{(\lambda')^k e^{-\lambda'}}{k!} \\
&= -2e^{-\lambda/r} + 2\Pr[P(\lambda) < r] - \frac{\lambda}{r} \Pr[P(\lambda) < r-1] \\
&\quad + \left(\frac{r-1}{r}\right)^{-r} e^{-\lambda/r} \Pr[P(\lambda') \geq r].
\end{aligned}$$

Using that $\lambda = r\ln(1/p)$, we obtain

$$a(r,p) = -2p + (2 + \ln p)\Pr[P(\lambda) < r-1] + 2\Pr[P(\lambda) = r-1] + \frac{p}{\left(1-\frac{1}{r}\right)^r} \Pr[P(\lambda') \geq r],$$

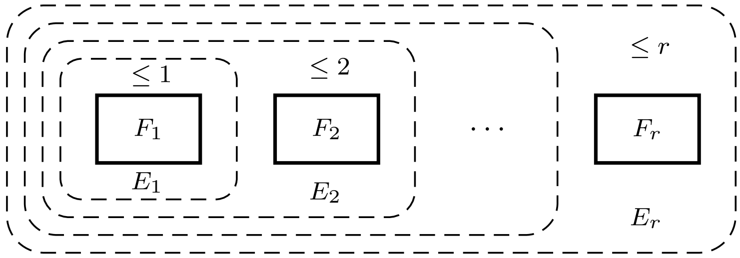

Figure 2: A tight example for Algorithm 1. Sets $F_1, \ldots, F_r$ are presented with solid lines; $E_1, \ldots, E_r$ are presented with dashed lines. The numbers denote the bounds on the E_i's.

as stated. □

Using this lemma, we obtain the lower bound in Theorem 1 by studying the function $a(r, p)$. One can evaluate $a(r, p)$ for small r – see Table 2 in Appendix B. We are now ready to prove the lower bound in Theorem 1.

Proof of the Lower Bound in Theorem 1. We use the following claim.

Claim 1. *For $p \in (1/e, 1)$, $\lim_{r\to\infty} a(r, p) = 2-2p+\ln(p)$. For $p \in (0, 1/e)$, $\lim_{r\to\infty} a(r, p) = (e-2)p$.*

Proof. In the proof of Proposition 4, we showed that as $r \to \infty$, $\Pr[P(\lambda) < r]$ and $\Pr[P(\lambda') < r]$ tend to 1 if $p > 1/e$, and to 0 if $p < 1/e$. Since $\max_{k\in\mathbb{N}} \Pr[P(\lambda) = k]$ tends to 0 as $\lambda \to \infty$, we conclude that for $p > 1/e$, $a(r, p)$ tends to $-2p + (2 + \ln(p)) \cdot 1 + 2 \cdot 0 + (ep/e^{-1}) \cdot 0 = 2 - 2p + \ln(p)$ as $r \to \infty$; and for $p < 1/r$, $a(r, p)$ tends to $-2p + (2 + \ln(p)) \cdot 0 + 2 \cdot 0 + (p/e^{-1}) \cdot 1 = p(e - 2)$. This ends the proof of the claim. □

Since any algorithm for rank-$(r + 1)$ laminar matroids can be directly applied to rank-r laminar matroids by simply adding a dummy co-loop to the matroid, we deduce that for every rank r, and every value $p \in (0, 1]$, the competitive ratio of Algorithm 1 is at least $\sup_{r'\geq r} a(r', p) \geq \lim_{r'\to\infty} a(r', p)$.

Noting that for all $p < 1/e$, $p(e - 2) < 1 - 2/e$, and this value is exactly $2 - 2x + \ln x$ for $x = 1/e$, we deduce that the best single value p we can use to maximize the competitive ratio on all laminar matroids is the one that maximizes $2 - 2p + \ln p$. This expression is maximized on $p = 1/2$, achieving a value of $1 - \ln 2 \approx 0.3068$. □

Upper Bound in Theorem 1. We show that the analysis of Algorithm 1 is tight for laminar matroids. For $q, r \in \mathbb{N}$, we define a laminar matroid $\mathcal{M}_{q,r} = (E_{q,r}, \mathcal{I}_{q,r})$ together with a weight function; the value order is then defined by setting $e \succ f$ if and only if $w_e > w_f$ for $e, f \in E_{q,r}$. The ground set of the matroid is $E_{q,r} = F_1 \sqcup \cdots \sqcup F_r$, where each F_i has cardinality q. Let $E_i = \bigcup_{j=1}^{i} F_j$. The independent sets are $\mathcal{I}_{q,r} = \{I \subseteq E_{q,r} : |I \cap E_i| \leq i \text{ for all } i \in [r]\}$. Finally, for each $i \in [r]$, we set the weight of each element $e \in F_i$ to be $w_e \sim U[r - i, r - i + 1]$ independently from all other elements (see Fig. 2). Choosing q and r carefully results in an instance that provides an upper bound on the probability-competitiveness of Greedy-Improving for laminar matroids.

Next, we show the upper bound in Theorem 1.

Proof of the Upper Bound in Theorem 1. We will a slightly stronger statement. Let $p \in (0, 1)$ and define $a(p) = 2 - 2p + \ln(p)$ if $p > 1/e$, and $a(p) = (2 - e)p$ if $p \leq 1/e$. Fix an $\varepsilon > 0$. By Claim 1, there exists r_ε large enough such that

$$a(r_\varepsilon, p) < a(p) + \varepsilon/2.$$

Fix a positive integer q and consider the instance of laminar matroid secretary corresponding to r_ε as described above. Let $\mathcal{M} = (E, \mathcal{I})$ denote the instance obtained by adding an infinite number of parallel copies at the end of the value order, so that $\mathcal{M}$ satisfies Assumption 1. By a slight abuse of notation, we still denote by F_i the i-th set of elements described in the above construction together with their parallel copies, and the same applies to the E_i's.

Let e^* denote the unique element in $\mathrm{OPT}(E) \cap E_1$, and consider the labeling scheme $\Lambda_{\text{chain}}(e^*)$ corresponding to a chain order π with respect to e^*. Clearly, e^* is accepted by Algorithm 1 if and only if $t_{e^*} > p$ and the algorithm has accepted at most $c-1$ element from E_c in the time interval $(1/2, t_{e^*})$ for each $c \in [r_\varepsilon]$. By definition, for every pair of (original) elements $e \in F_i, e' \in F_j$ with $i < j$, we have $w_e > w_{e'}$.

Let $T = \{e \in E \colon t_e < p\}$ denote the set of elements that arrived during the sampling period and $\mathcal{E}$ be the event that $|\mathrm{OPT}(T) \cap F_i| = 1$ for all $i \in [r_\varepsilon]$. Since r_ε is constant, we can choose q large enough so that $\Pr[\mathcal{E}] \geq 1 - \varepsilon/2$. Note that if $\mathcal{E}$ holds then any improving element $e \in F_i$ arriving at time $t_e \in [p, t_{e^*})$ will be assigned a label i upon arrival by the labeling scheme $\Lambda_{\text{chain}}(e^*)$

Let us now consider the improving word z produced by the labeling scheme in $[p, 1]$. Whenever e^* arrives in $[p, 1]$, it has label 1 since it is the element of rank 1 of $\mathrm{OPT}(t_{e^*}^+)$, and so z has the form $x1y$ for some $x, y \in \{[r_\varepsilon] \setminus \{1\}\}^*$, where the symbol 1 refers to the entry of z corresponding to e^*. Furthermore, the constraints described above imply that e^* is accepted if and only if y contains at most $c-1$ copies of label c for each $c \in [r_\varepsilon]$. It follows that under event $\mathcal{E}$, e^* is accepted if and only if z is well-indexed. Since the probability that z is well-indexed is $a(r_\varepsilon, p)$ we get $\Pr[e^* \in \mathrm{ALG}_0] \leq \Pr[z \text{ is well-indexed} \vee \neg\mathcal{E}] \leq a(r_\varepsilon, p) + \Pr[\neg\mathcal{E}] < a(p) + \varepsilon/2 + \varepsilon/2$. We finish the proof by setting $p = 1/2$ on the previous expression. □

4.3 Rank-2 Matroids

An interesting class of matroids for which the best achievable competitiveness is still unknown is the class of rank-2 matroids. A loopless rank-2 matroid $\mathcal{M} = (E, \mathcal{I})$ has very simple structure since its circuits have only 2 or 3 elements. Recall that two elements e and f are called *parallel* if $\{e, f\}$ is a circuit, and being parallel is an equivalence relation on E. Consider the equivalence classes $C_1, \ldots, C_k$ of this relation. A set $F \in \mathcal{I}$ is then independent if and only if $|F \cap C_i| \leq 1$ and $|F| \leq 2$. So, in particular, rank-2 matroids are laminar matroids and, by Lemma 9, Greedy-Improving is at least $a(2, p) = p(2 - 2p + p\ln(p))$-competitive. By setting $p = 0.4241$, this leads to a 0.3341-competitive algorithm for this class.

On the other hand, there are laminar matroids for which Greedy-Improving performs better. For example, on a uniform matroid of rank 2, the algorithm is $c(p, 2) = p(3 - 3p + 2p\ln(p))) = a(2, p) + p(1-p)$-competitive. Below we analyze more carefully the competitiveness of Greedy-Improving for this class. Then we propose and analyze a second algorithm, called Oblivious-Partition that has a higher competitiveness on those instances in which Greedy-Improving behaves badly. Their combination then leads to a new algorithm with better competitiveness for rank-2 matroids. In this section, we show the following result.

Theorem 2. *There exists a* 0.3462*-probability-competitive algorithm for the MSP on rank-2 matroids.*

Suppose that $\mathrm{OPT}(E) = \{e^*, f^*\}$. Let z' be the improving word of e^* generated by the labeling scheme $\Lambda_{\text{chain}}(e^*)$. As before, denote by $p \leq t_k < \cdots < t_1 \leq 1$ and $e_k, \ldots, e_1$ the improving times and elements in $[p, 1]$, respectively, and suppose that $e^* = e_i$. In particular, we know that $z'_1 = z'_2 = \cdots = z'_{i-1} = 2$ and $z'_i = 1$. Note that e^* is selected as long as e^* is not spanned by $A = \{e_{i+1}, \ldots, e_k\}$. If the matroid is uniform, it is enough that $|A| \leq 1$ to achieve this. For general rank 2 matroids though, e^* is selected if $|A| = 0$, if $|A| = 1$ and the unique element e_k of A is not parallel to e^*, or if $|A| \geq 2$ and all elements of A are parallel to each other but not parallel to e^*. Let ALG_0 be the output of Greedy-Improving. The previous discussion implies that

$$\Pr[e^* \in \text{ALG}_0] \geq \Pr[z' \in 2^*1] + \Pr[z' \in 2^*12] + \Pr[z' \in 2^*11 \wedge e_{|z'|} \text{ is not parallel to } e^*].$$

Recall that $a(2,p) = \Pr[z' \text{ is well-indexed}] = \Pr[z' \in 2^*1] + \Pr[z' \in 2^*12]$, so we want to understand in which situations the extra term above is strictly positive.

In what follows, we will use an alternative analysis in which we condition on the arrival time t of element e^*. Let $\{g,h\} = \text{OPT}(E_t)$ be the optimal set immediately before the arrival of e^*, and $\text{OPT}(E_t^+) = \text{OPT}(E_t \cup \{e^*\}) = \{g, e^*\}$. In particular, h is the element of $\text{OPT}(E_t)$ that e^* replaces on arrival and g is not parallel to e^*. We define a new labeling scheme Λ_τ only for the interval $[p,t)$. The order τ is chosen to be any total order of E_t having g as its first element. In particular, if $p < s < t$ and there is no improving time in $[s,t)$, then $\text{OPT}(E_s) = \text{OPT}(E_t) = \{g,h\}$ and the labels of g and h at time s are 1 and 2, respectively.

Let z denote the improving word on $[p,t)$ with respect to Λ_τ. Note that if $z = \epsilon$ is the empty word, then e^* is selected by Greedy-Improving. If $z = 1$, then g is the only improving element that arrived before e^*, so $\text{ALG}_0 = \{e^*, g\}$. If $z = 2$, then e^* is selected by the algorithm if and only if h is not parallel to e^*. The algorithm might select e^* for longer improving words as well, but we only focus on these three scenarios.

For each $s \in [0,1]$, let $\text{Par}(s)$ denote the event that $\text{OPT}(E_s^+ - \{e^*\})$ contains an element parallel to e^*, and let $\alpha(s) = \Pr[\text{Par}(s)]$. Based on the facts that the arrival time of e^* is uniform in $[0,1]$, that the length of z behaves like a Poisson random variable with rate $\lambda = 2\ln(t/p)$, and that each symbol of z is chosen uniform from the set $\{1,2\}$, the above discussion implies that

$$\begin{aligned}
&\Pr[e^* \in \text{ALG}_0] \\
&= \int_p^1 \Pr[e^* \in \text{ALG}_0 \,|t_{e^*} = t] \, \mathrm{d}t \\
&\geq \int_p^1 (\Pr[z = \epsilon] + \Pr[z = 1] + \Pr[z = 2 | \neg \text{Par}(t)] \Pr[\neg \text{Par}(t)]) \, \mathrm{d}t \\
&= \int_p^1 \left(e^{-\lambda} + \frac{1}{2}\frac{e^{-\lambda}\lambda}{1!} + \frac{1}{2}\frac{e^{-\lambda}\lambda^1}{1!}(1 - \alpha(t)) \right) \mathrm{d}t \\
&= \int_p^1 \left(\left(\frac{p}{t}\right)^2 + \left(\frac{p}{t}\right)^2 \ln\left(\frac{t}{p}\right) + \left(\frac{p}{t}\right)^2 \ln\left(\frac{t}{p}\right)(1 - \alpha(t)) \right) \mathrm{d}t \qquad (2) \\
&= p(2 - 2p + p\ln(p)) + \int_p^1 \left(\frac{p}{t}\right)^2 \ln\left(\frac{t}{p}\right)(1 - \alpha(t)) \, \mathrm{d}t,
\end{aligned}$$

where the third line follows since for fixed t, the event $z = 2$ has the same probability independent from $\neg \text{Par}(t)$, so $\Pr[z = 2 \mid \neg \text{Par}(t)] = \Pr[z = 2]$.

For certain rank-2 matroids and value orders, $(1 - \alpha(t))$ is equal to 1 as a function of t on the interval $[p,1]$ (e.g., for rank-2 uniform matroids), while for others the function is close to 0. Motivated by this observation, we propose a different algorithm that has higher probability of accepting e^* whenever $(1 - \alpha(t))$ is small, i.e., when it is likely that $\text{OPT}(E_t)$ contains an element parallel to e^*. At the same time, our goal is not to loose too much on the case when $\text{OPT}(E_t)$ does not contain an element parallel to e^*. The algorithm, called Oblivious-Partition, is presented as Algorithm 2.

Algorithm 2 Oblivious-Partition Algorithm for Rank-2 Matroids

Initialize $\mathrm{ALG}_1 \leftarrow \emptyset$.
Skip all elements with arrival time in $[0, p)$.
Let $\mathrm{OPT}(E_p) = \{g_1, g_2\}$, with $g_1 \succ g_2$. Let C_1 be the parallel class of g_1.
Include into ALG_1 the first improving element f_1 arriving in $[p, 1]$ such that f_1 is parallel to g_1, and the first improving element f_2 arriving in $[p, 1]$ such that f_2 is not parallel to g_1.
Return ALG_1.

Let ALG_1 be the output of the algorithm. For the analysis, we choose $e^* \in \mathrm{OPT}(E)$ with arriving time $t = t_{e^*} \in [p, 1]$, denote by $\mathrm{OPT}(E_t^+) = \{g, e^*\}$, and consider as before a labeling scheme Λ_τ where τ is any total order of E_t having g as its first element. Similarly as before, let z denote the improving word on $[p, t]$ with respect to Λ_τ. We distinguish two cases.

Case 1: $\mathrm{Par}(t)$ **holds.** In this case, $\mathrm{OPT}(E_t) = \{g, h\}$ is such that h is parallel to e^*. If h arrives before time p then it is also part of $\mathrm{OPT}(E_p)$, and the word z does not contain a symbol 1. In fact, h is contained in $\mathrm{OPT}(E_s)$ for all $s \in [p, t]$, therefore no other improving element parallel to h will appear in $[p, t)$. Here we have two sub-cases depending on whether h is g_1 or g_2.

If $h = g_1$, then e^* is selected by the algorithm being the first improving element parallel to g_1 that arrives. If $h = g_2$, then $g_1 \succ h$. This means that $\mathrm{OPT}(E_t)$ consists of the elements h and g, where $g \succ h$ in the value order. In fact, since the matroid has rank 2, every element f in E_t that satisfies $f \succ h$ must be parallel to g. In particular, g_1 is parallel to g. Observe also that if f' is an improving element arriving at time $s \in [p, t)$, then $\mathrm{OPT}(E_s^+) = \{f', h\}$. Since $\{g_1, h\}$ is an independent subset of E_s^+, we conclude that $f' \succ g_1 \succ h$ which implies that f' is parallel to g, and so it is parallel to g_1 as well. This means that e^* is the first improving element arriving that is not parallel to g_1, thus it is selected by the algorithm.

Summarizing, if $\mathrm{Par}(t)$ holds and h arrives before time p, then e^* is selected. Therefore, the events $\mathrm{Par}(t)$ and $z \in 2^*$ together imply that e^* is selected.

Case 2: $\mathrm{Par}(t)$ **does not hold.** In this case, $\mathrm{OPT}(E_t) = \{g, h\}$, $\mathrm{OPT}(E_t^+) = \{g, e^*\}$ and g, h and e^* are all from different classes. Note that if no improving elements arrive in $[p, t)$, then e^* is selected by the algorithm as f_2. This means that the events $\neg\,\mathrm{Par}(t)$ and $z = \epsilon$ together imply that e^* is selected.

To conclude, note that $z \in 2^*$ if and only if its restriction z_1 to alphabet 1 is empty. Recall that $|z_1|$ and $|z|$ both follow Poisson distributions with parameters $\lambda_1 = \ln({}^t\!/\!_p)$ and $\lambda = 2\ln({}^t\!/\!_p)$, respectively. We conclude that

$$\begin{aligned}
&\Pr[e^* \in \mathrm{ALG}_1] \\
&\quad = \int_p^1 \Pr[e^* \in \mathrm{ALG}_1 \mid t_{e^*} = t]\,\mathrm{d}t \\
&\quad \geq \int_p^1 \left(\Pr[z_1 = \epsilon \mid \mathrm{Par}(t)]\Pr[\mathrm{Par}(t)] + \Pr[z = \epsilon \mid \neg\,\mathrm{Par}(t)]\Pr[\neg\,\mathrm{Par}(t)]\right)\mathrm{d}t \\
&\quad = \int_p^1 \left(e^{-\lambda_1}\alpha(t) + e^{-\lambda}(1-\alpha(t))\right)\mathrm{d}t \\
&\quad = \int_p^1 \left(\left(\frac{p}{t}\right)\alpha(t) + \left(\frac{p}{t}\right)^2(1-\alpha(t))\right)\mathrm{d}t,
\end{aligned} \tag{3}$$

where the third line follows since the words z_1 and z are independent from $\mathrm{Par}(t)$.

Mixture Algorithm. The above analysis shows that GREEDY-IMPROVING performs well when $\Pr[\neg \operatorname{Par}(t)]$ is large, while OBLIVIOUS-PARTITION performs well when $\Pr[\operatorname{Par}(t)]$ is large. The idea is to combine these two algorithms to get a better probability-competitive guarantee. Therefore, our final algorithm for rank-2 matroids is a mixture of the two. The algorithm, called MIXTURE, is presented as Algorithm 3.

Algorithm 3 MIXTURE ALGORITHM FOR RANK-2 MATROIDS

1: Fix the sample size $p \in (0,1)$.
2: With probability ε, run OBLIVIOUS-PARTITION and return ALG_1.
3: Otherwise, run GREEDY-IMPROVING and return ALG_0.

Theorem 2. *There exists a 0.3462-probability-competitive algorithm for the MSP on rank-2 matroids.*

Proof. We choose $e^* \in \mathrm{OPT}(E)$ with arriving time $t = t_{e^*} \in [p,1]$. Let ALG denote the output of Algorithm 3. Then, by (2) and (3), we conclude that

$$\begin{aligned}\Pr[e^* \in \mathrm{ALG}] &\geq (1-\varepsilon)p(2-2p+p\ln(p)) + \int_p^1 \alpha(t)\varepsilon\frac{p}{t}\,\mathrm{d}t \\ &\quad + \int_p^1 (1-\alpha(t))\left((1-\varepsilon)\left(\frac{p}{t}\right)^2\ln\left(\frac{t}{p}\right) + \varepsilon\left(\frac{p}{t}\right)^2\right)\mathrm{d}t \\ &\geq (1-\varepsilon)p(2-2p+p\ln(p)) \\ &\quad + \int_p^1 \min\left(\varepsilon\frac{p}{t}, (1-\varepsilon)\left(\left(\frac{p}{t}\right)^2\ln\left(\frac{t}{p}\right) + \varepsilon\left(\frac{p}{t}\right)^2\right)\right)\mathrm{d}t.\end{aligned}$$

Fix $p \in [0,1]$ and consider the expressions inside the minimum, that is,

$$\begin{aligned}G_1(\varepsilon,t) &= \varepsilon\frac{p}{t}, \\ G_2(\varepsilon,t) &= (1-\varepsilon)\left(\frac{p}{t}\right)^2\ln\left(\frac{t}{p}\right) + \varepsilon\left(\frac{p}{t}\right)^2 \\ &= \left(\frac{p}{t}\right)^2\ln\left(\frac{t}{p}\right) + \varepsilon\left(1-\ln\left(\frac{t}{p}\right)\right)\left(\frac{p}{t}\right)^2.\end{aligned}$$

Note that for every given value $q > p$, both $G_1(\varepsilon,q)$ and $G_2(\varepsilon,q)$ are linear functions in ε. Furthermore, we have $G_1(0,q) = 0 < G_2(0,q) = (p/q)^2\ln(q/p)$. Therefore, $G_1(\varepsilon,q) \leq G_2(\varepsilon,q)$ if and only if

$$\varepsilon \leq \frac{p\ln\left(\frac{q}{p}\right)}{q-p+p\ln\left(\frac{q}{p}\right)}.$$

Let $q_\varepsilon > p$ be the unique solution of $\varepsilon\left(q_\varepsilon - p + p\ln(q_\varepsilon/p)\right) = p\ln(q_\varepsilon/p)$. Then,

$$\min\left(G_1(\varepsilon,t), G_2(\varepsilon,t)\right) = \begin{cases} G_1(\varepsilon,t) & \text{if } p < t < q_\varepsilon, \\ G_2(\varepsilon,t) & \text{if } t \geq q_\varepsilon. \end{cases}$$

In particular, the lower bound for $\Pr[e^* \in \mathrm{ALG}]$ can be rewritten as

$$(1-\varepsilon)p(2-2p+p\ln(p)) + \int_p^{\min(q_\varepsilon,1)} \varepsilon\frac{p}{t}\,\mathrm{d}t + \int_{\min(q_\varepsilon,1)}^1 \left((1-\varepsilon)\left(\frac{p}{t}\right)^2\ln\left(\frac{t}{p}\right) + \varepsilon\left(\frac{p}{t}\right)^2\right)\mathrm{d}t.$$

The value of this expression is

$$\begin{cases} (1-\varepsilon)p(2-2p+p\ln p)+\varepsilon(p\ln(p)) & \text{if } q_\varepsilon \geq 1 \\ (1-\varepsilon)\left(p(2-2p+p\ln(p))+\dfrac{p^2}{q_\varepsilon}(1-q_\varepsilon)(1-\ln(p))+\dfrac{p^2}{q_\varepsilon}\ln(q_\varepsilon)\right) & \\ \quad +\varepsilon\left(\dfrac{p^2}{q_\varepsilon}(1-q_\varepsilon)+p\ln\left(\dfrac{q_\varepsilon}{p}\right)\right) & \text{if } q_\varepsilon \in (p,1) \end{cases}$$

Note that whenever $q_\varepsilon > 1$, then we recover a ratio that is at most $p(2-2p+p\ln p)$ which is a lower bound for the ratio of Greedy-Improving, so the best ratio is achieved for $q_\varepsilon \leq 1$. To find the maximum, we set $\varepsilon = {p\ln(q/p)}/{-p+q+\ln(q/p)}$ and obtain an explicit maximization problem in two variables $p \leq q \leq 1$. The maximum can be computed numerically to get $p \approx 0.4067$, $q \approx 0.9194$, $\varepsilon \approx 0.3928$, showing a probability-competitive ratio of 0.3462 for Mixture. □

5 Graphic Matroids

Given a graph with edges as the ground set, the independent sets in the *graphic matroid* are edge sets that do not contain any cycles. In this section, we use our labeling schemes framework to improve the state-of-the-art competitive ratio for graphic matroid secretary. Our main result is the following.

Theorem 3. *There exists a* 0.2504*-probability-competitive algorithm for the MSP on graphic matroids. In addition, for simple graphs (i.e., without parallel edges), there exists a* 0.2693*-probability-competitive algorithm.*

To show the theorem above, in Section 5.1 we first describe a basic algorithm that uses a simple labeling scheme and can be considered a variant of the 2-forbidden algorithm of [29]. In Section 5.2, we introduce one of the key notions of the paper, the *generation* of an edge, which suggests a new algorithm that we analyze through our labeling scheme framework to show the improved 0.2693-probability-competitiveness for simple graphs, i.e., with no parallel edges. To tackle the case of general graphs, we combine this algorithm with another that performs well on different types of instances, breaking the 1/4 barrier.

5.1 Basic Algorithm and Auxiliary Digraph

Let $G' = (V', E')$ denote the input graph of the graphic MSP. For graphic matroids, we can implement Assumption 1 with the following construction.

Assumption 2. The input graph of our algorithm is the augmented graph $G = (V + w, E)$ with $E = E' \cup F$, where w is a new node considered a *root* and F is a collection of dummy edges consisting of an infinite number of parallel copies of the edges $\{wv \colon v \in V\}$. These dummy edges are put at the end of the value order.

Recall that every edge e in $E = E' \cup F$ picks its arrival time t_e from the uniform distribution over $[0, 1]$. Under Assumption 2, for every $t > 0$, the set E_t of all edges in $E = E' \cup F$ arriving in the interval $[0, t)$ has full rank $r = (|V + w| - 1) = |V|$, and thus $\mathrm{OPT}(E_t)$ is a tree with vertex set $V + w$.

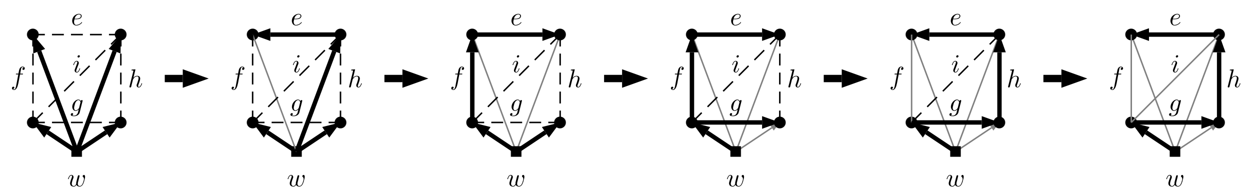

Figure 3: An example for the canonical orientation of improving edges. Vertex w is chosen as the root, the original edges of the canonical orientation are drawn with thick lines, and a set $\{e, f, g, h, i\}$ arrives in the order e, f, g, h, i, with ranking $e > g > h > i > f$. Note that e, f, g and h are improving edges whereas i is not, the orientation of e changes several times, and f leaves the optimal tree upon the arrival of h.

Definition 9 (Canonical Orientation of an Improving Edge)**.** The arborescence $A(t)$ (resp. $A(t^+)$) obtained by orienting the tree $\mathrm{OPT}(E_t)$ (resp. $\mathrm{OPT}(E_t^+)$) away from the root w is called the *canonical orientation* of the optimum at t (resp. t^+). For an edge $e \in E$, we say that e *is oriented from u to v at time t* if $e \in \mathrm{OPT}(E_t^+)$ and e is oriented as (u, v) in $A(t^+)$.

Note that the orientation of an edge changes from improving time to improving time, and that only elements that are improving are oriented. Indeed, if $e = \{u, v\}$ is an edge arriving at time t_e then, at that time, it receives an orientation (u, v) according to the arborescence $A(t_e^+)$. When the next improving edge arrives, say f, the optimum tree $\mathrm{OPT}(E_{t_f}^+)$ is different from $\mathrm{OPT}(E_{t_e}^+)$ since f enters and some other edge leaves the tree. If e is still part of $\mathrm{OPT}(E_{t_f}^+)$, then its orientation may have changed; see Fig. 3 for an illustration. The following algorithm is a variant of the 2-forbidden algorithm of [29] and is the foundation of our algorithms for graphic matroids. Algorithm 4 is $p(1-p)$-probability-competitive; see Theorem 5. In particular, it is $1/4$-probability-competitive for $p = 1/2$.

Algorithm 4 Basic Algorithm

Initialize $\mathrm{ALG}_0 \leftarrow \emptyset$.
Initialize $\mathrm{AUX} \leftarrow$ digraph $(V + w, \emptyset)$.
Skip all edges with arrival time in $[0, p)$.
for each edge e with $t_e \in [p, 1]$, in arrival order **do**
    if $e \in \mathrm{OPT}(E_{t_e}^+)$ is oriented from u to v at time t_e and $\deg^-_{\mathrm{AUX}}(v) = 0$ **then**
        $\mathrm{AUX} \leftarrow \mathrm{AUX} + (u, v)$
        if $\deg^-_{\mathrm{AUX}}(u) = 0$ **then**
            $\mathrm{ALG}_0 \leftarrow \mathrm{ALG}_0 + e$
Return ALG_0.

Theorem 5. *Algorithm 4 is $p(1-p)$-probability-competitive. In particular, it is $1/4$-probability-competitive when $p = 1/2$.*

To show the theorem above, we will need the following lemmas.

Lemma 10. *At every step, the digraph* AUX *has maximum in-degree at most one. Therefore, every cycle of the underlying undirected graph has to be a directed cycle in* AUX*, and* AUX *has at most one directed cycle per connected component.*

Proof. An edge e oriented from u to v is added to AUX only if the in-degree of v is 0, showing the first half of the statement. The second half is easy to see and is a folklore result in graph theory. □

Lemma 11. ALG_0 *is an independent set in the graphic matroid.*

Proof. Since ALG_0 is a subset of the underlying graph of AUX, the only possibility for a cycle C to be present in ALG_0 is that C corresponds to some directed cycle in AUX. However, the last arriving edge e, with orientation (u, v) in AUX, cannot enter ALG_0 since its tail u has, at that moment, in-degree one in AUX. Therefore, there are no cycles in ALG_0. $\square$

Let $e^* \in \mathrm{OPT}(E)$ be an optimal edge. In particular, $e^* \in E'$ is not a dummy edge. Let $t^* = t_{e^*}$ be its arrival time and $\vec{e} = (u^*, v^*)$ be the orientation of e^* at time t^*. We are ready to define the labeling scheme that we use to analyze the basic algorithm.

Definition 10 (Labeling Scheme $\Lambda_0(e^*)$)**.** On the interval $[t^*, 1]$, we use the usual labeling scheme Λ_π induced by any total order of E such that $\pi_1 = e^*$. For times $s < t_{e^*}$, we use a different labeling that depends on u^* and v^*. Recall that for every vertex $v \in V$, there is exactly one edge oriented toward v at time s, namely the unique edge in $\mathrm{OPT}(E_s)$ whose associated arc in $A(s)$ is oriented toward v. We assign label 1 to the edge oriented toward the head v^* of e^*, and label 2 to the edge oriented toward the tail u^*. The remaining $r - 2$ labels are assigned arbitrarily to the other elements of $\mathrm{OPT}(E_s)$.

Denote by $\mathcal{L}_{\text{basic}} = \{z \in [r]^* \colon z = x1y$, where x is a word not containing the symbol 1 and y is a word not containing the symbols 1 and $2\}$. The next lemma connects the basic algorithm with this language.

Lemma 12. *Let z be the improving word of $e^* \in \mathrm{OPT}(E)$ produced by the labeling scheme $\Lambda_0(e^*)$. If $z \in \mathcal{L}_{basic}$ then $e^* \in \mathrm{ALG}_0$.*

Proof. By Lemma 5, the arrival time t^* of e^* is in $[p, 1]$ if and only if there exists a symbol 1 in z. Furthermore, all the symbols after the first 1 are labels of improving edges arriving in $[p, t^*)$. Observe that $\vec{e} \in \mathrm{AUX}$ if and only if $t^* \in [p, 1]$ and no improving edge arriving in $[p, t^*)$ is oriented toward v^* upon arrival, which corresponds to the condition that there is a unique symbol 1 in z by the definition of the dynamic labeling we are using.

Observe that $e^* \in \mathrm{ALG}_0$ if and only $\vec{e} \in \mathrm{AUX}$ and no improving edge arriving in $[p, t^*)$ is oriented toward u^* upon arrival. By the previous paragraph and the labeling we use, this is equivalent to $z = x1y$, where x is a word without symbols in $\{1\}$ and y is a word without symbols in $\{1, 2\}$. $\square$

We are now ready to show Theorem 5.

Proof of Theorem 5. Let $e^* \in \mathrm{OPT}(E)$ and z be the improving word produced by the associated labeling scheme $\Lambda_0(e^*)$ in the interval $[p, 1]$. Instead of looking at the produced word $z \in [r]^*$ directly, we look at the subsequences $z_{\{1,2\}}$, $z_{\{1\}}$ and $z_{\{2\}}$ obtained by restricting z to alphabets $\{1, 2\}$, $\{1\}$ and $\{2\}$, respectively. By Lemma 4, $z_{\{1\}} \in 1^*$ and $z_{\{2\}} \in 2^*$ are random words whose length is distributed as a Poisson with parameter $\lambda_1 = \log(1/p)$. By Lemma 12, we get

$$\begin{aligned}
\Pr[e^* \in \mathrm{ALG}_0] &\geq \Pr[z \in ([r] \setminus \{1\})^* 1 ([r] \setminus \{1, 2\})] \\
&= \Pr[z_{\{1,2\}} \in 2^* 1] \\
&= \Pr[z_{\{1,2\}} \in 1^* 1] \\
&= \Pr[z_2 = \epsilon] \cdot \Pr[z_1 \neq \epsilon] \\
&= e^{-\lambda_1}(1 - e^{-\lambda_1}) \\
&= p(1 - p),
\end{aligned}$$

where the third line holds since generating the word 2^k1 has the same probability than generating the word 1^k1 for all $k \in \mathbb{N}$, while the fourth line holds by the independence of words z_1 and z_2 given by Lemma 4. ☐

5.2 Cycles and Generations of the Auxiliary Digraph

Let $G' = (V, E')$ be an undirected, not necessarily simple graph, and $G = (V + w, E)$ with $E = E' \cup F$ be defined as before. In the following, we study in more detail the cycles of the auxiliary digraph AUX in Algorithm 4 when applied to the augmented graph G. For that, enumerate all arcs in AUX in the order in which they entered as $(u_1, v_1), \dots (u_k, v_k)$. Note that $\{v_1, \dots, v_k\} \subseteq V$ and they are all different, while $\{u_1, \dots, u_k\} \subseteq V + w$ and they might repeat. For every time $t \in [0, 1]$, let $\mathrm{AUX}(t)$ be the sub-digraph of AUX consisting of all arcs that entered ALG_0 in the interval $[0, t]$.

Definition 11 (Generation of an Arc in AUX)**.** Let e be an edge such that its orientation $\vec{e} = (u, v)$ entered AUX at time t. The *generation* $\mathrm{Gen}(e) \in \mathbb{N}$ is defined as follows. If $e \in F$ is a dummy edge, we set $\mathrm{Gen}(e) = 1$. If $e \in E'$ and $\deg^-_{\mathrm{AUX}(t)}(u) = 0$, we set $\mathrm{Gen}(e) = 0$. Otherwise, there is a unique arc $\vec{f}$ with head u that entered AUX at a time $s < t$, and we set $\mathrm{Gen}(e) = \mathrm{Gen}(f) + 1$.

Let $C \subseteq E' \cup F$ be a cycle in AUX. No edge in C can be in F, since the edges of F are oriented away from w by construction, and no arc in AUX is directed toward w. Fig. 4 shows an example of a cycle and the generations of its edges. Also, the edges e that enter ALG_0 are exactly those with $\mathrm{Gen}(e) = 0$.

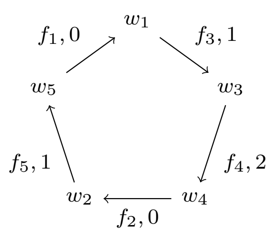

Figure 4: A directed 5-cycle in AUX arriving in the order f_1, f_2, f_3, f_4, f_5; each arc is marked as $f_i, \mathrm{Gen}(f_i)$.

Lemma 13. *Let* AUX' *be the sub-digraph of* AUX *obtained by deleting all edges with generation equal exactly to one. Then* AUX' *is acyclic.*

Proof. Suppose to the contrary that AUX' contains a directed cycle C and let $f = (u_1, u_2)$ be the first arriving arc in C. Let $h = (u_k, u_1)$ be the previous arc of C in cyclic order. Since h is added to AUX, no arc entered its head u_1 at its arrival. Consequently, no arc entered the tail of f upon its arrival. Hence, the generation of f is 0. This implies that the generation of the next arc $g = (u_2, u_3)$ in cyclic order is 1, contradicting the assumption that AUX' has no edges with generation equal to 1. Note that this proof works even if $k = 2$, that is when $h = g$. ☐

The previous discussion and the fact that the generation of each arc can be computed easily at its inclusion to AUX, suggest the following algorithm.

Lemma 14. *Algorithm 5 outputs a set* ALG_1 *that is independent.*

Proof. Note that dummy edges $e \in F$ do not enter ALG_1. For a non-dummy edge e, the condition $\deg^-_{\mathrm{AUX}}(u) = 0$ is satisfied if and only if $\mathrm{Gen}(e) = 0$. The condition $u' = w$ is satisfied if and only if f is a dummy edge, $\mathrm{Gen}(f) = 1$ and $\mathrm{Gen}(e) = 2$. Finally, the condition $u' \neq w$ and $\deg^-_{\mathrm{AUX}(t_f)}(u) = 1$ holds if and only if f is not a dummy edge, $\mathrm{Gen}(f) \geq 1$ and $\mathrm{Gen}(e) = \mathrm{Gen}(f) + 1 \geq 2$. It follows that ALG_1 contains exactly the edges with generation different from 1. By Lemma 13, ALG_1 is acyclic. ☐

Algorithm 5 GENERATION ALGORITHM

$\mathrm{ALG}_1 \leftarrow \emptyset$
Initialize AUX $\leftarrow$ digraph $(V+w, \emptyset)$.
Skip all edges with arrival time in $[0,p)$.
for each edge e with $t_e \in [p,1]$, in arrival order **do**
  if $e \in \mathrm{OPT}(E^+_{t_e})$ is oriented from u to v at time t_e and $\deg^-_{\mathrm{AUX}}(v) = 0$ **then**
    AUX $\leftarrow$ AUX $+(u,v)$
    if $u \neq w$ **then**
      if $\deg^-_{\mathrm{AUX}}(u) = 0$ **then**
        $\mathrm{ALG}_1 \leftarrow \mathrm{ALG}_1 + e$
      else let $f = (u', u)$ be the unique arc in AUX pointing toward u
        if $u' = w$ or $\deg^-_{\mathrm{AUX}(t_f)}(u') = 1$ **then**
          $\mathrm{ALG}_1 \leftarrow \mathrm{ALG}_1 + e$
Return ALG_1.

We show that if G does not contain parallel edges, the competitive ratio of GENERATION is strictly better that $1/4$. Let $G' = (V, E')$ be a simple graph. Since the case $|V| = 2$ is straightforward, we assume that $|V| \geq 3$. Let AUX and ALG_1 be obtained by applying Algorithm 5 to $G = (V+w, E' \cup F)$, and let $e^* \in \mathrm{OPT}(E)$ be an optimal edge arriving at a time $t^* = t_{e^*}$.

Definition 12 (Labeling Scheme $\Lambda_1(e^*)$)**.** On the time interval $[t^*, 1]$, we use the usual labeling scheme Λ_π induced by any total order of E such that $\pi_1 = e^*$. Suppose that e^* is oriented from a vertex u^* to a vertex v^* at time t^*, and let w_0^* be an arbitrary vertex in $V \setminus \{u^*, v^*\}$. For $s < t^*$, the labeling scheme is as follows: let $\vec{f} = (u', u^*)$ be the earliest improving arc in $(s, 1]$ having head u^*. If $\vec{f}$ exists and $u' \neq w$, then we set $w_s^* := u'$; since the original graph $G' = (V, E')$ is simple, f is not parallel to e^*, showing that $u' \neq v^*$ holds as well. Otherwise, we set $w_s^* = w_0^*$. In any case, the vertices v^*, u^* and w_s^* are different. Recall that for any vertex $v \in V$, there is a unique edge in $\mathrm{OPT}(E_s)$ that is oriented toward v. Then, at time s, we assign label 1 to the edge that is oriented toward v^*, label 2 to the edge that is oriented toward u^*, and label 3 to the edge that is oriented toward w_s^*. The remaining $(r-3)$ labels are assigned arbitrarily to the other elements of $\mathrm{OPT}(E_s)$.

Denote by $\mathcal{L}_{\text{generation}} = \{z \in [r]^* \colon z_{\{1,2,3\}} = x1y$, where x and y are words not containing the symbol 1 and y is a word that does not end with symbol 2$\}$. Here, $z_{\{1,2,3\}}$ denotes the subsequence of z restricted to the alphabet $\{1,2,3\}$.

Lemma 15. *Let z be the improving word of $e^* \in \mathrm{OPT}(E)$ produced by $\Lambda_1(e^*)$. If $z \in \mathcal{L}_{generation}$ then $\vec{e} \in$ AUX and $\mathrm{Gen}(e) \neq 1$. In particular, $e^* \in \mathrm{ALG}_1$.*

Proof. Let $z' = z_{\{1,2,3\}}$ and suppose that $z \in \mathcal{L}_{\text{generation}}$. That is $z' = x1y$; $x, y \in \{2,3\}^*$ and y does not end with symbol 2. Let $\vec{e} = (u^*, v^*)$ with $u^* \neq w$ be the orientation of e^* upon arrival and $t^* \in (0,1)$ be its arrival time. Since z' has a unique 1, the same holds for z, implying that $t^* \in [p,1]$ and that e^* is the unique improving edge that is oriented toward v^* in $[p, t^*]$. Therefore, $\vec{e} \in$ AUX. We distinguish two cases based on whether y contains a symbol 2 or not.

If y contains no symbol 2 at all, then no improving edge arriving in $[p, t^*)$ is oriented toward u^* or v^* in $[p, t^*)$. Since e^* is not a dummy edge, we get $\mathrm{Gen}(e^*) = 0$ and thus $\mathrm{Gen}(e^*) \neq 1$.

If y contains a symbol 2, then let f be the first improving edge with arrival time $p \leq s < t$ that is oriented toward u^*, that is, $f = (u', u^*)$ for some $u' \in (V+w) - u^*$. Note that f is the edge whose label is recorded as the last 2 of y. Furthermore, $\mathrm{Gen}(e^*) = \mathrm{Gen}(f) + 1$. If $u' = w$ then, by definition, f is a dummy edge in F, $\text{Gen}(f) = 1$ and $\text{Gen}(e^*) = 2$. So assume that $u' \neq w$, which means that $f \in E'$. Since the original graph $G' = (V, E')$ is simple, f is not parallel to e^*, implying $u' \neq v^*$. It follows that, for every s' with $p \leq s' < s$, we have $w^*_{s'} = u'$. Let g be the first improving edge in $[p, s)$ that is oriented toward u'. Then its orientation $\vec{g} = (u'', u')$ is in AUX and, by the previous discussion, g is the edge whose label is recorded as the last 3 of the word y. This means that $\text{Gen}(e^*) = \text{Gen}(f) + 1 = \text{Gen}(g) + 2 \geq 2$, finishing the proof. □

Using this lemma, we show that Algorithm 5 is $(\frac{1}{4}p(1-p^2) - \frac{1}{2}p\ln(p))$-probability-competitive for simple graphs. In particular, it is 0.2693-probability-competitive when $p \approx 0.4485$. Algorithm 5 may not achieve a good competitive ratio on general graphs with parallel edges, and the reason is that the probability that a given edge $e \in \text{OPT}(E)$ has generation 1 may be much larger in this case. We remedy this issue by considering an algorithm good for those elements $e \in \text{OPT}(E)$ for which $\text{OPT}(E_q)$ is likely to contain a parallel copy of e when q is a constant. We combine this algorithm with Generation, leading to an algorithm that we call Mixture, which is 0.2504-probability-competitive.

5.3 Competitiveness via the Labeling Schemes Framework

Simple graphs. We start by proving our competitiveness result for simple graphs.

Theorem 6. *Algorithm 5 is* $(\frac{1}{4}p(1-p^2) - \frac{1}{2}p\ln(p))$*-probability-competitive for simple graphs. In particular, it is* 0.2693*-probability-competitive when* $p \approx 0.4485$.

Proof. Let $e^* \in \text{OPT}(E)$ and z be the improving word produced by the associated labeling scheme $\Lambda_1(e^*)$ in the interval $[p, 1]$. For any subset of labels S, let z_S be the restriction of z to the labels in S and set $\lambda_i = i\ln(1/p)$ for all $i \in \mathbb{N}$. By Lemma 15,

$$\begin{aligned}
&\Pr[e^* \in \text{ALG}_1] \\
&\quad \geq \Pr[z_{\{1,2,3\}} = x1y \text{ such that } x, y \text{ contains no } 1,\ y \text{ does not end with } 2] \\
&\quad = \Pr[z_{\{1,2,3\}} = x1 \text{ such that } x \in \{2,3\}^*] \\
&\quad\quad + \Pr[z_{\{1,2,3\}} = x1y'3 \text{ such that } x, y' \in \{2,3\}^*].
\end{aligned}$$

We compute both term separately. First, note that

$$\begin{aligned}
&\Pr[z_{\{1,2,3\}} = x1 \text{ such that } x \in \{2,3\}^*] \\
&\quad = \Pr[z_{\{1,2,3\}} = x2 \text{ such that } x \in \{2,3\}^*] \\
&\quad = \frac{1}{2}\Pr[z_{\{1,2,3\}} = xc \text{ such that } x \in \{2,3\}^*,\ c \in \{2,3\}] \\
&\quad = \frac{1}{2}\Pr[|z_{\{1\}}| = 0] \cdot \Pr[|z_{\{2,3\}}| \geq 1] \\
&\quad = \frac{1}{2}e^{-\lambda_1}(1 - e^{-\lambda_2}) \\
&\quad = \frac{1}{2}p(1-p^2),
\end{aligned}$$

where the first and second equalities follow since the symbols of z are uniform in $\{1,2,3\}$, the third equality follows since $z_{\{1\}}$ and $z_{\{2,3\}}$ are independent random variables, and the last equality follows since the length of z_S distributes as a Poisson with parameter $|S|\ln(1/p)$.

By a similar reasoning, the second term is

$$
\begin{aligned}
&\Pr[z_{\{1,2,3\}} = x1y'3 \text{ such that } x, y' \in \{2,3\}^*] \\
&= \frac{1}{2}\Pr[z_{\{1,2,3\}} = x1y'c \text{ such that } x, y' \in \{2,3\}^*, c \in \{2,3\}] \\
&= \frac{1}{2}\big(\Pr[z_{\{1,2,3\}} \text{ has exactly one } 1] - \Pr[z_{\{1,2,3\}} = u1 \text{ such that } u \in \{2,3\}^*]\big) \\
&= \frac{1}{2}\left(\Pr[|z_{\{1\}}| = 1] - \frac{1}{2}p(1-p^2)\right) \\
&= \frac{1}{2}\left(\frac{e^{-\lambda_1}\lambda_1}{1!} - \frac{1}{2}p(1-p^2)\right) \\
&= \frac{1}{2}(p\ln(1/p)) - \frac{1}{4}p(1-p^2).
\end{aligned}
$$

Adding up the two terms, we conclude that $\Pr[e^* \in \mathrm{ALG}_1] \geq \frac{1}{4}p(1-p^2) - \frac{1}{2}p\ln(p)$. This expression attains its maximum at $p \approx 0.4485$, achieving a value of 0.2693. □

General Graphs. In what follows, we prove the 0.2504-probability-competitiveness for general graphs. Consider the following algorithm.

Algorithm 6 Oblivious Algorithm

Initialize $\mathrm{ALG}_2 \leftarrow \emptyset$.
Skip all edges with arrival time in $[0, p)$.
Let $X^p \subseteq E$ be the set of all parallel copies of the elements in $X = \mathrm{OPT}(E_p^+)$.
for each edge e with $t_e \in [p, 1]$ in arrival order **do**
    if $e \in \mathrm{OPT}(E_{t_e}^+) \cap X^p$ and $\mathrm{ALG}_2 + e$ is independent **then**
        $\mathrm{ALG}_2 \leftarrow \mathrm{ALG}_2 + e$
Return ALG_2.

We are ready to present our algorithm for matroid secretary on general graphs. As we said before, the idea that shows the theorem is similar to that of Algorithm 3: we combine Generation with Oblivious, leading to the following algorithm that we call Mixture.

Algorithm 7 Mixture Algorithm for Graphic Matroids

Fix the sample size $p \in (0, 1)$.
With probability ε, run Oblivious and return ALG_2.
Otherwise, run Generation and return ALG_1.

As before, we work under Assumption 2, and we assume that $r \geq 3$. For the analysis, we run both the oblivious and the generation algorithm in parallel, simultaneously constructing AUX, ALG_1 (for the generation algorithm), and ALG_2 (for the oblivious algorithm). Let ALG be the output of Algorithm 7.

Consider an element $e^* \in \mathrm{OPT}(E)$ with arriving time $t = t_{e^*} \in [p, 1]$. For each $s \in [0, 1]$, define $\mathrm{Par}(s)$ as the event that $\mathrm{OPT}(E_s^+ - \{e^*\})$ contains an edge f that is parallel to e^* in the graphic matroid, and let $\alpha(s) = \Pr(\mathrm{Par}(s))$. Furthermore, for $0 \leq a < b \leq 1$, define $\mathrm{Par}(a, b)$ as the event that there exists $s \in [a, b)$ for which $\mathrm{Par}(s)$ holds, and set $\alpha(a, b) = \Pr[\mathrm{Par}(a, b) | \neg\,\mathrm{Par}(b)]$. We split the analysis into two cases, depending on whether $\mathrm{Par}(t)$ holds or not.

Case 1: $\mathrm{Par}(t)$ **holds.** In this case, there exists an element $f \in \mathrm{OPT}(E_t^+ - \{e^*\}) = \mathrm{OPT}(E_t)$ that is parallel to e^*. The arrival time t_f of f is uniformly distributed on $[0,t]$. If t_f is in the interval $[0,p)$, then we know that $f \in \mathrm{OPT}(E_s)$ for all $s \in [p,t]$. In particular, $f \in \mathrm{OPT}(E_p^+)$ and e^* is the first arriving element that is parallel to f and improving, so it is selected by Oblivious.

Now focus on Generation. Let t' be the largest improving time in $[0,t)$ such that the associated improving edge $e_{t'}$ is oriented toward some vertex in $\{u^*, v^*\}$. In terms of the labeling schemes $\Lambda_0(e^*)$ and $\Lambda_1(e^*)$ we have used so far for graphic matroids, this means that the label of t' is in $\{1,2\}$. Note that if $t' < p$, then there are no arcs in AUX with head in $\{u^*, v^*\}$ when e^* arrives. This means that $\mathrm{Gen}(e^*) = 0$, therefore, e^* is selected by the algorithm.

Recall that $N_S[a,b)$ denotes the number of improving times in $[a,b)$ with label in S. Summarizing the above, by Lemma 3, we deduce that

$$\begin{aligned}\Pr[e^* \in \mathrm{ALG} \mid \mathrm{Par}(t)] &\geq \varepsilon \Pr[t_f \in [0,p) \mid f \in \mathrm{OPT}(E_t)] + (1-\varepsilon)\Pr[t' < p | t^* = t] \\ &= \varepsilon \Pr[N_{\{1\}}[p,t) = 0] + (1-\varepsilon)\Pr[N_{\{1,2\}}[p,t) = 0] \\ &= \varepsilon(p/t) + (1-\varepsilon)(p/t)^2. \end{aligned} \tag{4}$$

Case 2: $\mathrm{Par}(t)$ **does not hold.** In this case, define t' as before. Suppose first that $t' \geq p$. If $\mathrm{Par}(t',t)$ holds, then there exists a moment $s \in [t',t)$ in which $\mathrm{Par}(s)$ holds, i.e., $\mathrm{OPT}(E_s^+)$ contains an edge f parallel to e^*. At that moment, f is oriented toward u^* or v^*, implying $t_f \leq t'$. We also conclude that no improving edge parallel to f arrives in (t_f, t). Furthermore, since at time t' exactly two edges of $\mathrm{OPT}(E_{t'}^+)$ are directed toward u^* and v^*, we deduce that f is the element arriving at time t' with probability $1/2$, and $t_f < t'$ with probability $1/2$. Then, conditioned on $t_f \neq t'$, t_f is a uniform random variable in $[0,t']$. Moreover, if $t_f \in [0,p)$ then, by the same argument as in Case 1, e^* will be selected by Oblivious. By the above, if $p \leq s < t$, then $\Pr[e^* \in \mathrm{ALG}_2 \mid \neg\,\mathrm{Par}(t), t' = s, \mathrm{Par}(t',t)] = 1/2(p/s)$.

If $\mathrm{Par}(t',t)$ does not hold then, since $\mathrm{Par}(t)$ does not hold by assumption, no improving element parallel to e^* arrives in $[t',t]$. Let g be the improving edge arriving at time t' and $\vec{g}$ be its orientation. With probability $1/2$, the head of $\vec{g}$ is u^* and with probability $1/2$ it is v^*. Suppose we are in the case $\vec{g} = (u', u^*)$. If u' is the root w, then as long as no arc pointing toward u^* or v^* arrives in $[p,t')$, $\vec{g}$ and $\vec{e}$ will both enter AUX, $\mathrm{Gen}(g) = 1$ and $\mathrm{Gen}(e^*) = 2$, so Generation selects e^*. If u' is not the root w, then $u' \in V \setminus \{u^*, v^*\}$ as g is not parallel to e^*. As long as no improving edge oriented toward u^* or v^* and at least one improving edge h oriented toward u' arrive in $[p,t')$, the arcs $\vec{h}$, $\vec{g}$ and $\vec{e}$ enter AUX and $\mathrm{Gen}(e) = \mathrm{Gen}(g) + 1 = \mathrm{Gen}(h) + 2 \geq 2$, so Generation selects e^*. By the above, $\Pr[e^* \in \mathrm{ALG}_1 \mid \neg\,\mathrm{Par}(t), t' = s, \neg\,\mathrm{Par}(t',t)] = \frac{1}{2}\Pr[N_{\{1,2\}}[p,s) = 0]\Pr[N_{\{3\}}[p,s) \geq 1] = \frac{1}{2}(p/s)^2(1 - p/s)$.

Let us now consider the case when $t' < p$. Note that in this case, we immediately have that $\mathrm{Gen}(e^*) = 0$ and so e^* is selected by Generation. For Oblivious, since no arc parallel to e^* can possibly arrive in (t',t), e^* is selected by the algorithm as long as $\mathrm{OPT}(E_p^+)$ contains an element parallel to e^*, which is exactly event $\mathrm{Par}(p,t)$.

Putting everything together, using the fact that output of ALG is ALG_1 with probability $(1-\varepsilon)$ and ALG_2 with probability ε, as well as that the probability density function[1] of the variable t' is

[1] This comes from the fact that t' has the same distribution as $S(t)$ for a rank-2 matroid. See Lemma 2.

$f_{t'}(s) = 2s/t^2$, we get that

$$\begin{aligned}
&\Pr[e^* \in \mathrm{ALG} \,|\, \neg\, \mathrm{Par}(t)] \\
&\quad = \Pr[e^* \in \mathrm{ALG} \,|\, \neg\, \mathrm{Par}(t), t' < p](p/t)^2 \\
&\qquad + \int_p^1 \Pr[e^* \in \mathrm{ALG} \,|\, \neg\, \mathrm{Par}(t), t' = s] f_{t'}(s)\,\mathrm{d}s \\
&\quad = \varepsilon\alpha(p,t)(p/t)^2 + (1-\varepsilon)(p/t)^2 \\
&\qquad + \int_p^t \left(\varepsilon\alpha(s,t)\frac{1}{2}(p/s) + (1-\varepsilon)(1-\alpha(s,t))\frac{1}{2}(p/s)^2(1-p/s)\right) f_{t'}(s)\,\mathrm{d}s \\
&\quad = \varepsilon\alpha(p,t)(p/t)^2 + (1-\varepsilon)(p/t)^2 \\
&\qquad + (p/t^2)\int_p^t \varepsilon\alpha(s,t) + (1-\varepsilon)(1-\alpha(s,t))(p/s)(1-p/s)\,\mathrm{d}s. \qquad (5)
\end{aligned}$$

The bounds on the right-hand side of (4) and (5) are algebraic functions of p, ε and t. They also allude to probabilities $\alpha(\cdot)$ and $\alpha(\cdot,\cdot)$ of random events, where the only stochastic part comes from the random arrival times of the elements – they also depend on the adversarial choice of the graph and the value order on the edges. We get a better understanding of (4) and (5) if we introduce an auxiliary random variable $M \in [0,1]$ that is defined as the maximum of all times $s < t$ such that $\mathrm{Par}(s)$ holds. Then, we can write $\alpha(t) = \Pr[M = t]$, $\alpha(s,t)\alpha(t) = \Pr[s \le M < t]$ and $\alpha(s,t)(1-\alpha(t)) = \Pr[M < s]$. In particular, we can simplify (4) and (5) using random variables as follows. For any fixed value $m^* \in [0,1]$, we have

$$\begin{aligned}
&\Pr[e^* \in \mathrm{ALG} \,|\, t_{e^*} = t, M = m^*] \\
&\quad = \Pr[e^* \in \mathrm{ALG} \,|\, t_{e^*} = t, M = m^*, \mathrm{Par}(t)]\alpha(t) \\
&\qquad + \Pr[e^* \in \mathrm{ALG} \,|\, t = t_e, M = m^*, \neg\, \mathrm{Par}(t)](1-\alpha(t)) \\
&\quad \ge \mathbf{1}[m^* = t] \cdot (\varepsilon(p/t) + (1-\varepsilon)(p/t)^2) \\
&\qquad + \mathbf{1}[p \le m^* < t] \cdot \varepsilon(p/t)^2 + \mathbf{1}[m^* < t] \cdot (1-\varepsilon)(p/t)^2 \\
&\qquad + (p/t^2)\int_p^t (\mathbf{1}[s \le m^* < t] \cdot \varepsilon + \mathbf{1}[m^* < s] \cdot (1-\varepsilon)(p/s)(1-p/s))\,\mathrm{d}s.
\end{aligned}$$

Now, we define G_1, $G_2(m^*)$ and G_3 by computing the the expression above for specific values of m^*. If $m^* = t$, then

$$\Pr[e^* \in \mathrm{ALG} \mid t_{e^*} = t, M = t] \ge \varepsilon(p/t) + (1-\varepsilon)(p/t)^2 \triangleq G_1.$$

If $m^* \in [p,t)$, then

$$\begin{aligned}
&\Pr[e^* \in \mathrm{ALG} \mid t_{e^*} = t, M = m^*] \\
&\quad \ge \varepsilon(p/t)^2 + (1-\varepsilon)(p/t)^2 + (p/t^2)\int_p^{m^*} \varepsilon\,\mathrm{d}s \\
&\qquad + (p/t^2)\int_{m^*}^t (1-\varepsilon)(p/s)(1-p/s)\,\mathrm{d}s \\
&\quad = (p/t)^2 + \varepsilon\frac{p(m^*-p)}{t^2} + (1-\varepsilon)(p/t^2)\left(p^2\left(1/t - 1/m^*\right) + p\ln(t/m^*)\right) \\
&\quad \triangleq G_2(m^*).
\end{aligned}$$

If $m^* < p$, then

$$\begin{aligned}
&\Pr[e^* \in \text{ALG} \mid t_{e^*} = t, M = m^*] \\
&\geq (1-\varepsilon)(p/t)^2 + (p/t^2) \int_p^t (1-\varepsilon)(p/s)(1-p/s)\,\mathrm{d}s \\
&= (1-\varepsilon)(p/t)^2 + (1-\varepsilon)(p/t^2)\left(p^2(1/t - 1/p) + p\ln(t/p)\right) \\
&= (1-\varepsilon)\frac{p^2(p + t\ln(t/p))}{t^3} \\
&\triangleq G_3.
\end{aligned}$$

Observe that $G_3 = G_2(p) - \varepsilon(p/t)^2$, and $G_1 = G_2(t)$. Therefore,

$$\min_{m^* \leq t} \Pr[e^* \in \text{ALG} \mid t_{e^*} = t, M = m^*] = \min\left\{G_3, \min_{m^* \in [p,t]} G_2(m^*)\right\}.$$

It is not difficult to check that the second derivative of $G_2(m^*)$ with respect to m^* is $(1-\varepsilon)(m^* - 2p)p^2/((m^*)^3 t^2)$. Therefore, for $m^* \leq 2p$, the function $G_2(m^*)$ is concave in m^* and thus its minimum over the interval $[p, t]$ is attained in $m^* = p$ or in $m^* = t$. To ensure that $m^* \leq 2p$ holds, we assume from now on that $p \geq 1/2$. Since $G_2(t) = G_1$ and $G_2(p) \geq G_3$, we obtain

$$\begin{aligned}
\Pr[e^* \in \text{ALG} \mid t_{e^*} = t] &\geq \min_{m^* \leq t} \Pr[e^* \in \text{ALG} \mid t_{e^*} = t, M = m^*] \\
&= \min\{G_3, G_1\}. \qquad (6)
\end{aligned}$$

Based on the discussion above, we are ready to bound the probability-competitive ratio of Mixture.

Proof of Theorem 3. We choose $e^* \in \text{OPT}(E)$ with arriving time $t = t_{e^*} \in [p, 1]$. Let ALG denote the output of Algorithm 3. From (6), we get that

$$\begin{aligned}
&\Pr[e^* \in \text{ALG}] \\
&\geq \int_p^1 \min\left(\varepsilon(p/t) + (1-\varepsilon)(p/t)^2, (1-\varepsilon)(p/t)^3(1 + (t/p)\ln(t/p))\right)\mathrm{d}t \\
&= p\int_p^1 \min\left(\varepsilon/q + 1 - \varepsilon, (1-\varepsilon)(q - \ln(q))\right)\mathrm{d}q,
\end{aligned}$$

where the second equality holds by performing a change of variables $q = p/t$. Fix a value of $p \in [1/2, 1]$ and consider the two functions involved in the minimum inside he integral, that is,

$$\begin{aligned}
K_1(\varepsilon, q) &= \varepsilon/q + 1 - \varepsilon, \\
K_2(\varepsilon, q) &= (1-\varepsilon)(q - \ln(q)).
\end{aligned}$$

Note that for every given value $q > p$, both $K_1(\varepsilon, q)$ and $K_2(\varepsilon, q)$ are linear functions in ε. Furthermore, we have $K_1(1, q) = 1/q > K_2(1, q) = 0$. Thus, $K_1(\varepsilon, q) \leq K_2(\varepsilon, q)$ if and only if

$$\varepsilon \leq \frac{q - \ln(q) - 1}{q - \ln(q) - 1 + 1/q} = R(q). \qquad (7)$$

The function $R(q)$ is decreasing in the interval $[1/2, 1]$, with $R(1) = 0$. If $\varepsilon > R(1/2)$, inequality (7) is not satisfied for any $q \in [1/2, 1]$, and therefore $K_1(\varepsilon, q) > K_2(\varepsilon, q)$ for every $q \in [p, 1]$. If $\varepsilon \in [0, R(1/2)]$, there exists a unique value $q_\varepsilon \in [1/2, 1]$ such that $\varepsilon = R(q_\varepsilon)$. Then,

$$\min(K_1(\varepsilon, q), K_2(\varepsilon, q)) = \begin{cases} K_2(\varepsilon, q) & \text{if } \varepsilon > R(1/2), \\ K_2(\varepsilon, q) & \text{if } \varepsilon \leq R(1/2) \text{ and } q > q_\varepsilon, \\ K_1(\varepsilon, q) & \text{if } \varepsilon \leq R(1/2) \text{ and } q \leq q_\varepsilon. \end{cases}$$

In particular, the lower bound for $\Pr[e^* \in \text{ALG}]$ can be rewritten as

$$p \int_p^1 (1 - \varepsilon)(q - \ln(q))\, \mathrm{d}q = p(1 - \varepsilon)(-p^2/2 - p + p\ln(p) + 3/2)$$

if $\varepsilon > R(1/2)$,

$$p \int_p^1 (1 - \varepsilon)(q - \ln(q))\, \mathrm{d}q = p(1 - \varepsilon)(-p^2/2 - p + p\ln(p) + 3/2)$$

if $\varepsilon \leq R(1/2)$ and $p \geq q_\varepsilon$, and

$$\begin{aligned} &p \int_p^{q_\varepsilon} (\varepsilon/q + 1 - \varepsilon) + p \int_{q_\varepsilon}^1 (1 - \varepsilon)(q - \ln(q))\, \mathrm{d}q \\ &= p\varepsilon \ln(q_\varepsilon/p) + (1 - \varepsilon)(q_\varepsilon - p)p + p(1 - \varepsilon)(-q_\varepsilon^2/2 - q_\varepsilon + q_\varepsilon \ln(q_\varepsilon) + 3/2) \end{aligned}$$

if $\varepsilon \leq R(1/2)$ and $p \leq q_\varepsilon$.

To conclude, we solve an optimization problem on each three of the regions defined by these cases to find the optimal combination for ε and p. For the case $\varepsilon > R(1/2) \approx 0.0881$, the maximum of this bound is attained when $p = 1/2$, yielding a maximum value of ≈ 0.2409. When $\varepsilon \leq R(1/2)$, to find the maximum, we set $\varepsilon = R(q)$ and obtain an explicit maximization problem in two variables with constraints $1/2 \leq q \leq p \leq 1$ for the second region, and $1/2 \leq p \leq q \leq 1$ for the third region, and the maximum can be computed numerically in both cases. In the former, we get an optimal value of 0.2409. However, for the latter, the optimal solution is given by $p^\star = 0.5$ and $q^\star \approx 0.8251$, with $\varepsilon = R(q^\star) \approx 0.0141$, showing a probability-competitive ratio of at least 0.2504. □

Acknowledgement. The research was supported by the Lendület Programme of the Hungarian Academy of Sciences – grant number LP2021-1/2021, by Dynasnet European Research Council Synergy project – grant number ERC-2018-SYG 810115 and by ANID-Chile – grants FONDECYT 1231669, FONDECYT 1241846, BASAL Center for Mathematical Modeling FB210005, and ANILLO Information and Computation in Market Design ACT210005.

References

[1] D. Abdolazimi, A. R. Karlin, N. Klein, and S. Oveis Gharan. Matroid Partition Property and the Secretary Problem. In *ITCS*, volume 251, pages 2:1–2:9, 2023.

[2] S. Albers and L. Ladewig. New results for the k-secretary problem. *Theoretical Computer Science*, 863:102–119, 2021.

[3] M. Babaioff, M. Dinitz, A. Gupta, N. Immorlica, and K. Talwar. Secretary problems: Weights and discounts. In *SODA*, page 1245–1254, 2009.

[4] M. Babaioff, N. Immorlica, D. Kempe, and R. Kleinberg. Matroid secretary problems. *Journal of the ACM*, 65(6), 11 2018.

[5] M. Bahrani, H. Beyhaghi, S. Singla, and S. M. Weinberg. Formal barriers to simple algorithms for the matroid secretary problem. In *WINE*, pages 280–298, 2021.

[6] M. Bateni, M. Hajiaghayi, and M. Zadimoghaddam. Submodular secretary problem and extensions. *ACM Transactions on Algorithms*, 9(4):1–23, 2013.

[7] K. Borovkov. *Elements of stochastic modelling.* World Scientific, 2003.

[8] S. Chakraborty and O. Lachish. Improved competitive ratio for the matroid secretary problem. In *SODA*, page 1702–1712, 2012.

[9] A. Cristi, P. Dütting, R. Kleinberg, and R. P. Leme. Online matroid embeddings, 2024.

[10] M. Dinitz and G. Kortsarz. Matroid secretary for regular and decomposable matroids. *SIAM Journal on Computing*, 43(5):1807–1830, 2014.

[11] S. Dughmi. The outer limits of contention resolution on matroids and connections to the secretary problem. In *ICALP*, pages 42:1–42:18, 2020.

[12] S. Dughmi. Matroid secretary is equivalent to contention resolution. In *ITCS*, pages 58:1–58:23, 2022.

[13] E. B. Dynkin. The optimum choice of the instant for stopping a Markov process. *Soviet Math. Dokl*, 4, 1963.

[14] M. Feldman, O. Svensson, and R. Zenklusen. A simple O(log log(rank))-competitive algorithm for the matroid secretary problem. *Mathematics of Operations Research*, 43(2):638–650, 2018.

[15] S. O. Gharan and J. Vondrák. On variants of the matroid secretary problem. *Algorithmica*, 67(4):472–497, 2013.

[16] Z. Huang, Z. Parsaeian, and Z. Zhu. Laminar matroid secretary: Greedy strikes back. *To appear in ESA*, 2024.

[17] S. Im and Y. Wang. Secretary problems: Laminar matroid and interval scheduling. In *SODA*, pages 1265–1274, 2011.

[18] P. Jaillet, J. A. Soto, and R. Zenklusen. Advances on matroid secretary problems: Free order model and laminar case. In *IPCO*, pages 254–265, 2013.

[19] T. Kesselheim, K. Radke, A. Tönnis, and B. Vöcking. An optimal online algorithm for weighted bipartite matching and extensions to combinatorial auctions. In *ESA*, pages 589–600, 2013.

[20] T. Kesselheim, K. Radke, A. Tönnis, and B. Vöcking. An optimal online algorithm for weighted bipartite matching and extensions to combinatorial auctions. In H. L. Bodlaender and G. F. Italiano, editors, *Algorithms – ESA 2013*, pages 589–600, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[21] R. Kleinberg. A multiple-choice secretary algorithm with applications to online auctions. In *SODA*, page 630–631, 2005.

[22] N. Korula and M. Pál. Algorithms for secretary problems on graphs and hypergraphs. In *ICALP*, 2009.

[23] O. Lachish. O(log log rank) competitive ratio for the matroid secretary problem. In *FOCS*, pages 326–335, 2014.

[24] D. V. Lindley. Dynamic programming and decision theory. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 10(1):39–51, 1961.

[25] T. Ma, B. Tang, and Y. Wang. The Simulated Greedy Algorithm for Several Submodular Matroid Secretary Problems. In *STACS*, pages 478–489, 2013.

[26] J. Oxley. *Matroid Theory*, volume 21 of *Oxford Graduate Texts in Mathematics*. Oxford University Press, Oxford, second edition, 2011.

[27] R. Santiago, I. Sergeev, and R. Zenklusen. Constant-competitiveness for random assignment matroid secretary without knowing the matroid. In *IPCO*, pages 423–437, 2023.

[28] J. A. Soto. Matroid secretary problem in the random-assignment model. *SIAM Journal on Computing*, 42(1):178–211, 2013.

[29] J. A. Soto, A. Turkieltaub, and V. Verdugo. Strong algorithms for the ordinal matroid secretary problem. *Mathematics of Operations Research*, 46(2):642–673, 2021.

A Forbidden Sets via Labeling Schemes

Here, we briefly describe the q-forbidden technique of Soto, Turkieltaub and Verdugo [29] and show how to implement every q-forbidden algorithm via a labeling scheme. The q-forbidden algorithmic scheme of [29] also starts by first sampling the elements arriving up to a given time p and then accepting a subset of the improving elements. Specifically, for every $e^* \in \mathrm{OPT}$ arriving at a given time t, the algorithm identifies, for every time $s < t$, a set $\mathcal{F}(\mathrm{OPT}(E_s), \mathrm{OPT}(E_t), e^*)$ of at most q *forbidden* elements such that if no improving element arriving at $s \in (p, t)$ is in $\mathcal{F}(\mathrm{OPT}(E_s), \mathrm{OPT}(E_t), e^*)$ then e^* is added to the output set ALG.

Proposition 7. *For every matroid class $\mathcal{C}$ that admits a q-forbidden algorithm* ALG*, there exists a labeling scheme Λ such that, for every $p \in (0,1)$ and $e^* \in \mathrm{OPT}$,*

$$\Pr[e^* \in \mathrm{ALG}] = \begin{cases} -p\ln(p) & \text{for } q = 1, \\ \frac{p-p^q}{q-1} & \text{for } q \geq 2. \end{cases}.$$

Choosing the optimal sample size $p = p(q)$ yields a $c(q)$-probability-competitive ratio, where $(p(1), c(1)) = (1/e, 1/e)$ and $(p(q), c(q)) = \left(q^{-1/(q+1)}, q^{-q/(q+1)}\right)$.

Proof. Let $e^* \in \mathrm{OPT}(E)$. Our labeling scheme is dynamic. We start by considering a labeling scheme Λ_π induced by a total order π of all elements of E for which $\pi_1 = e^*$, and use it for the interval $[t_{e^*}, 1]$. For times $s < t_{e^*}$, we use a different labeling. By the definition of a q-forbidden algorithm, e^* can be selected if no element in $\mathcal{F}(E_s, E_{t_{e^*}}, e^*) \subseteq \mathrm{OPT}(E_s)$ arrives in $[p, s)$, where $\left|\mathcal{F}(E_s, E_{t_{e^*}}, e^*)\right| \leq q$. Our labeling scheme gives arbitrary labels from $[q]$ to the elements in $\mathcal{F}(E_s, E_{t_{e^*}}, e^*)$, and arbitrary labels from $[r] \setminus [q]$ to the remaining elements of $\mathrm{OPT}(E_s)$. In particular, any element with label greater than q is not part of $\mathcal{F}(E_s, E_{t_{e^*}}, e^*)$. Let us denote the labeling scheme thus obtained by Λ.

Consider the improving word z in $[p, 1]$ given by Λ. By Lemma 5, the first symbol 1 in z represents the arrival of e^*. From that point on, going backwards in time, labels in $\{1, 2, \ldots, q\}$ correspond to forbidden elements for e^*. Thus,

$$\Pr[e^* \in \text{ALG}] \geq \Pr[z = a1b \text{ such that } a \in ([r] \setminus 1)^*, b \in ([r] \setminus [q])^*].$$

Instead of using the word z, we use the simpler restriction z' of z to the alphabet $[q]$. By Lemma 3, z' is a uniform random word in $[q]^*$ whose length distributes as a Poisson distribution with parameter $\lambda_q = q \ln\left(\frac{1}{p}\right)$. We separate the analysis for $q = 1$ due to its simplicity, giving

$$\Pr[e^* \in \text{ALG}] \geq \Pr[z' = 1] = \frac{\lambda_1^1 e^{-\lambda_1}}{1!} = -p\ln(p). \tag{8}$$

For $q \geq 2$, let $\lambda' = \lambda_q \cdot (1 - 1/q) = (q-1)\ln(1/p)$. Then,

$$\begin{aligned}
\Pr[e^* \in \text{ALG}] &\geq \Pr\left[z' = a'1 \,\middle|\, a \in ([q] \setminus \{1\})^*\right] \\
&= \sum_{k=1}^{\infty} \frac{\lambda_q^k e^{-\lambda_q}}{k!} \left(1 - \frac{1}{q}\right)^{k-1} \cdot \frac{1}{q} = \sum_{k=1}^{\infty} \frac{(\lambda')^k e^{-\lambda'}}{k!} \frac{1}{q-1} e^{\lambda' - \lambda_q} \\
&= \frac{1 - e^{-\lambda'}}{q-1} e^{\lambda' - \lambda_q} = \frac{1 - p^{q-1}}{q-1} e^{-\ln(1/p)} = \frac{p - p^q}{q-1}.
\end{aligned} \tag{9}$$

Optimizing (8) and (9), we get that the optimal p's are $p(1) = 1/e$ and $p(q) = q^{-1/(q+1)}$ for $q \geq 2$, yielding a probability-competitive ratio of $c(1) = 1/e$ and $c(q) = q^{-q/(q+1)}$ for $q \geq 2$. □

We remark that the previous result gives an alternative proof of, for example, the $1/e$-competitive algorithm for transversal matroids [20] (which are 1-forbidden).

B Competitive Ratios for Small Ranks

Proposition 4 gives an exact formula that can be seen as a polynomial in p and $\ln(p)$ of total degree at most $2r$. Hence, it is easy to evaluate for small values of r. Table 1 presents the values of $c(r, p)$, $p(r) = \arg\max\{c(r, p) \colon p \in (0, 1)\}$ and $c(r, p(r))$ for small values of r.

r	$c(r,p)$	$p(r)$	$c(r,p(r))$
1	$-p\ln(p)$	$1/e \approx 0.3678$	$1/e \approx 0.3678$
2	$p(3 - 3p + 2p\ln(p))$	0.3824	0.4273
3	$(p/8)(19 - 19p^2 + 30p^2\ln(p) - 18p^2\ln(p)^2)$	0.3867	0.4575
4	$(p/81)(175 - 175p^3 + 444p^3\ln(p) - 504p^3\ln(p)^2 + 288p^3\ln(p)^3)$	0.3883	0.4769

Table 1: Greedy-Improving is $c(r, p(r))$-probability-competitive for uniform matroids of rank r.

Next, we consider general laminar matroids. Using that $\Pr[P(\lambda') \geq r] = 1 - \Pr[P(\lambda') < r]$ and the exact expression for a Poisson distribution taking a value of k for $k \in \{0, \ldots, r\}$, we get a formula for $a(r, p)$ that can be seen as a polynomial in p and $\ln(p)$ of total degree at most $2r$, with coefficients that depend on r. Hence, it is easy to evaluate for small values of r. Table 2 presents the values of $a(r, p)$, $p(r) = \arg\max\{a(r, p) \colon p \in (0, 1)\}$ and $a(r, p(r))$ for small values of r.

r	$a(r,p)$	$p(r)$	$a(r,p(r))$
1	$-p\ln(p)$	$1/e \approx 0.3678$	$1/e \approx 0.3678$
2	$p(2-2p+p\ln(p))$	0.4241	0.3341
3	$(p/8)(11-11p^2+14p^2\ln(p)-6p^2\ln(p)^2)$	0.4490	0.3225
4	$(p/81)(94-94p^3+201p^3\ln(p)-180p^3\ln(p)^2+72p^3\ln(p)^3)$	0.4629	0.3169

Table 2: Greedy-Improving is $a(r,p(r))$-probability-competitive for laminar matroids of rank r.