Submodular Secretary Problems: Cardinality, Matching, and Linear Constraints*

Thomas Kesselheim[1] and Andreas Tönnis[2]

1 Department of Computer Science, TU Dortmund, Dortmund, Germany†
thomas.kesselheim@cs.tu-dortmund.de

2 Department of Computer Science, University of Bonn, Bonn, Germany‡
atoennis@uni-bonn.de

Abstract

We study various generalizations of the secretary problem with submodular objective functions. Generally, a set of requests is revealed step-by-step to an algorithm in random order. For each request, one option has to be selected so as to maximize a monotone submodular function while ensuring feasibility. For our results, we assume that we are given an offline algorithm computing an α-approximation for the respective problem. This way, we separate computational limitations from the ones due to the online nature. When only focusing on the online aspect, we can assume $\alpha = 1$.

In the *submodular secretary problem*, feasibility constraints are cardinality constraints, or equivalently, sets are feasible if and only if they are independent sets of a k-uniform matroid. That is, out of a randomly ordered stream of entities, one has to select a subset of size k. For this problem, we present a 0.31α-competitive algorithm for all k, which asymptotically reaches competitive ratio α/e for large k. In *submodular secretary matching*, one side of a bipartite graph is revealed online. Upon arrival, each node has to be matched permanently to an offline node or discarded irrevocably. We give a 0.207α-competitive algorithm. This also covers the problem, in which sets of entities are feasible if and only if they are independent with respect to a transversal matroid. In both cases, we improve over previously best known competitive ratios, using a generalization of the algorithm for the classic secretary problem.

Furthermore, we give an $O(\alpha d^{-\frac{2}{B-1}})$-competitive algorithm for submodular function maximization subject to linear packing constraints. Here, d is the column sparsity, that is the maximal number of none-zero entries in a column of the constraint matrix, and B is the minimal capacity of the constraints. Notably, this bound is independent of the total number of constraints. We improve the algorithm to be $O(\alpha d^{-\frac{1}{B-1}})$-competitive if both d and B are known to the algorithm beforehand.

1998 ACM Subject Classification F.1.2 [Models of Computation] Online Computation

Keywords and phrases Secretary Problem, Online Algorithms, Submodular Maximization

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2017.16

* The full version of this article can be found at http://arxiv.org/abs/1607.08805.
† Work was done while this author was at Max Planck Institute for Informatics and Saarland University, supported in part by the DFG through Cluster of Excellence MMCI.
‡ Work was done while this author was at RWTH Aachen University, supported by the DFG GRK/1298 "AlgoSyn".

Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2017).
Editors: Klaus Jansen, José D. P. Rolim, David Williamson, and Santosh S. Vempala; Article No. 16; pp. 16:1–16:22
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

In the classic secretary problem, one is presented a sequence of items with different scores online in random order. Upon arrival of an item, one has to decide immediately and irrevocably whether to accept or to reject the current item. The objective is to accept the best of these items. Recently, combinatorial generalizations of this problem have attracted attention. In these settings, feasibility of solutions are stated in terms of matroid or linear constraints. In most cases, these combinatorial generalizations consider linear objective functions. This way, the profit gained by the decision in one step is independent of the other steps.

In this paper, we consider general monotone submodular functions[1]. For example, the *submodular secretary problem*, independently introduced by Bateni et al. [4] and Gupta et al. [15], is an online variant of monotone submodular maximization subject to cardinality constraints. In this problem, we are allowed to select up to k items from a set of n items. The value of a set is represented by a monotone, submodular function. Now, stated as an online problem, items arrive one after the other and every item can only be selected right at the moment when it arrives. The values of the submodular function are only known on subsets of the items that have already arrived. The objective function is designed by an adversary, but the order of the items is uniformly at random.

We call an algorithm (asymptotically) c-competitive if for any objective function v chosen by the adversary, the set of selected items ALG satisfies $\mathbf{E}\left[v(\mathrm{ALG})\right] \geq (c - o(1)) \cdot v(\mathrm{OPT})$, where OPT is a size-$k$ subset of items that maximizes v and the $o(1)$-term is asymptotical with respect to the length of the sequence n. Note that any algorithm can pretend n to be larger by adding dummy elements at random positions. Therefore, it is safe to assume that n is large compared to k.

Previous algorithms for submodular secretary problems were designed by modifying offline approximation algorithms for submodular objectives so that they could be used in the online environment [4, 11, 26]. In this paper, we take a different approach. Our algorithms are inspired by algorithms for linear objective functions [17, 18]. We repeatedly solve the respective offline optimization problem and use this outcome as a guide to make decisions in the current round. Generally, it is enough to only compute approximate solutions. Our results nicely separate the loss due to the online nature and due to limited computational power. Using polynomial-time computations and existing offline algorithms, we significantly outperform existing online algorithms. Certain submodular functions or kinds of constraints allow better approximations, which immediately transfer to even better competitive ratios. This is, for example, true for submodular maximization subject to a cardinality constraint if the number of allowed items is constant. Also, if computational complexity is no concern like in classical competitive analysis, our competitive ratios become even better.

1.1 Our Contribution

Given an α-approximate algorithm for monotone submodular maximization subject to a cardinality constraint, we present an $\frac{\alpha}{e}\left(1 - \frac{\sqrt{k-1}}{(k+1)\sqrt{2\pi}}\right)$-competitive algorithm for the submodular secretary problem. That is, we achieve a competitive ratio of at least 0.31α for any $k \geq 2$. Asymptotically for large k, we reach $\frac{\alpha}{e}$.

[1] A function $f\colon 2^U \to \mathbb{R}$ for given ground set U is called *submodular* if for all $S \subseteq T \subseteq U$ and every $x \in U \backslash T$ holds $f(S \cup \{x\}) - f(S) \geq f(T \cup \{x\}) - f(T)$. Additionally for all sets $S, T \subseteq U$, we call $f(S|T) = f(S \cup T) - f(T)$ the marginal gain of S to T.

Our algorithm follows the following natural paradigm. We reject the first $\frac{n}{e}$ items. Afterwards, for each arriving item, we solve the offline optimization problem of the instance that we have seen so far. If the current item is included in this solution and we have not yet accepted too many items, we accept it. Otherwise, we reject it. For the analysis, we bound the expected value obtained by the algorithm recursively. It then remains to solve the recursion and to bound the resulting term. Generally, the recursive approach can be used for any secretary problems with cardinality constraints. It could be of independent interest, especially because it allows to obtain very good bounds also for rather small values of k.

One option for the black-box offline algorithm is the standard greedy algorithm by Nemhauser and Wolsey [29]. It always picks the item of maximum marginal increase until it has picked k items. Generally, this algorithm is $1-\frac{1}{e}$-approximate. However, it is known that if one compares to the best solution with only $k' \leq k$ items the approximation factor improves to $1 - \exp\left(-\frac{k}{k'}\right)$. We exploit this fact to give a better analysis of our online algorithm when using the greedy algorithm in each step. We show that the algorithm is 0.238-competitive for any k and asymptotically for large k it is 0.275-competitive.

Additionally, we consider the *submodular secretary matching problem*. In this problem, one side of a bipartite graph arrives online in random order. Upon arrival, vertices are either matched to a free vertex on the offline side or rejected. The objective is a submodular function on the set of matched pairs or edges. It is easy to see that the submodular secretary problem is a special case of this more general problem. Fortunately, similar algorithmic ideas work here as well. Again, we combine a sampling phase with a black box for the offline problem and get an 0.207α-competitive algorithm. Notably, the analysis turns out to be much simpler compared to the submodular secretary algorithm.

Finally, we show how our new analysis technique can be used to generalize previous results on linear packing programs towards submodular maximization with packing constraints. Here, we use a typical continuous extension towards the expectation on the submodular objective. We parameterize our results in d, the column sparsity of the constraint matrix, and B, the minimal capacity of the constraints. We achieve a competitive ratio of $\Omega(\alpha d^{-\frac{2}{B-1}})$ if both parameters are not known to the algorithm. If d and B are known beforehand we give different algorithm that is $\Omega(\alpha d^{-\frac{1}{B-1}})$-competitive.

1.2 Related Work

Although the secretary problem itself dates back to the 1960s, combinatorial generalizations only gained considerable interest within the last 10 years. One of the earliest combinatorial generalizations and probably the most famous one is the *matroid secretary problem*, introduced by Babaioff et al. [3]. Here, one has to pick a set of items from a randomly ordered sequence that is an independent set of a matroid. The objective is to maximize the sum of weights of all items picked. It is still believed that there is an $\Omega(1)$-competitive algorithm for this problem; the currently best known algorithms achieve a competitive ratio of $\Omega(1/\log\log(\rho))$ for matroids of rank ρ [13, 24]. Additionally, there are constant competitive algorithms known for many special cases, e.g., for transversal matroids there is an $1/e$-competitive algorithm [17] and for k-uniform matroids there is an $1 - O(1/\sqrt{k})$-competitive algorithm [19]. Both are known to be optimal. Other examples include graphical matroids, for which there is a $1/2e$-competitive algorithm [21], and laminar matroids, for which a $1/9.6$-competitive algorithm is known [26]. Further well-studied generalizations feature *linear constraints*. This includes online packing LPs [8, 27, 2, 18] and online edge-weighted matching [17, 21], for which optimal algorithms are known. Also the online variant of the generalized assignment problem [18] has been studied.

All these secretary problems have in common that the objective function is linear. Compared to other objective functions this has the clear advantage that the gain due to a choice in one round is independent of choices in other rounds. Interdependencies between the rounds only arise due to the constraints. Bateni et al. [4] and Gupta et al. [15] independently started work on submodular objective functions in the secretary setting. To this point, the best known results are a $\frac{e-1}{e^2+e} \approx 0.170$-competitive algorithm for k-uniform matroids [11] and a $\frac{1}{95}$-competitive algorithm for submodular secretary matching [26]. In case there are m linear packing constraints, the best known algorithm is $O(\frac{1}{m})$-competitive [4]. For matroid constraints, Feldman and Zenklusen [14] give a reduction, turning a c-competitive algorithm for linear objective functions to an $\Omega(c^2)$-competitive one for submodular objective functions. Furthermore, they give the first $\Omega(1/\log\log\rho)$-competitive algorithm for the submodular matroid secretary problem. Feldman and Izsak [10] consider more general objective functions, which are not necessarily submodular. They give competitive algorithms for cardinality constraint secretary problems that are parameterized in the *supermodular degree* of the objective function.

Agrawal and Devanur [1] study concave constraints and concave objective functions. These results, however, do not generalize submodular objectives because they require the dimension of the vector space to be low. Representing an arbitrary submodular function would require the dimension to be as large as n. Another related problem is submodular welfare maximization. In this case, even the greedy algorithm is known to be $1/2$-competitive in adversarial order, which is optimal [16], but at least 0.505-competitive in random order [20].

In the offline setting, submodular function maximization is computationally hard if the function is given through a value oracle. There are efficient algorithms that approximate a monotone, submodular function over a matroid or under a knapsack-constraint with a factor of $(1 - 1/e)$ [7, 30]. As a special case, the generalized assignment problem can also be efficiently approximated up to a factor of $(1 - 1/e)$ [7]. For a constant number of linear constraints, there is also a $(1-\epsilon)(1-1/e)$-approximation algorithm [23]. In the non-monotone domain, a number of recent results achieve approximation guarantees close to but strictly better than $1/e$ [6, 9, 5].

2 Submodular Secretary Problem

Let us first turn to the submodular secretary problem. Here, a set of items from a universe U, $|U| = n$, is presented to the algorithm in random order. For each arriving $j \in U$, the algorithm has to decide whether to accept or to reject it, being allowed to accept up to k items in total. The objective is to maximize a monotone submodular function $v\colon 2^U \to \mathbb{R}_{\geq 0}$. This function is defined by an adversary and known to the algorithm only restricted to the subsets of items that have already arrived. This problem extends the secretary problem for k-uniform matroids with linear objective functions, which was solved by Kleinberg [19]. The previously best known competitive factor is $\frac{e-1}{e^2+e} \approx 0.170$ [11].

Depending on the kind of the submodular function and its representation, the corresponding offline optimization problem (monotone submodular maximization with cardinality constraint) can be computationally hard. In order to focus on the online nature of the problem, we assume that we are given an offline algorithm $\mathcal{A}$ that for any $L \subseteq U$ returns an α-approximation of the best solution within L. Formally, $v(\mathcal{A}(L)) \geq \alpha \max_{T \subseteq L, |T| \leq k} v(T)$. Note that $\mathcal{A}$ is allowed to exploit any additional structure of the function v. For different L and L', $\mathcal{A}(L)$ and $\mathcal{A}(L')$ do not have to be consistent, but the output $\mathcal{A}(L)$ must be identical, irrespective of the arrival order on L. It may also be randomized. In this case, let $v(\mathcal{A}(L))$ refer to the *expected* value achieved on set L.

Algorithm 1: Submodular k-secretary

Drop the first $\lceil pn \rceil - 1$ items;
for *item j arriving in round* $\ell \geq \lceil pn \rceil$ **do** `// online steps` $\ell = \lceil pn \rceil$ `to` n
  Set $U^{\leq \ell} := U^{\leq \ell - 1} \cup \{j\}$;
  Let $S^{(\ell)} = \mathcal{A}(U^{\leq \ell})$; `// black box` α`-approximation`
  if $j \in S^{(\ell)}$ **then** `// tentative allocation`
    if $|\text{Accepted}| < k$ **then** `// feasibility test`
      Add j to Accepted; `// online allocation`

Our online algorithm, Algorithm 1, uses algorithm $\mathcal{A}$ as a subroutine as follows. It starts by rejecting the first pn items. For every following item j, it runs $\mathcal{A}(L)$, where L is the set of items that have arrived up to this point. If $j \in \mathcal{A}(L)$ we call j tentatively selected. Furthermore if the set of accepted items S contains less than k items and j is tentatively selected, then the algorithm adds j to S. Otherwise, it rejects j.

▶ **Theorem 1.** *Algorithm 1 for the submodular secretary problem is* $\frac{\alpha}{e}\left(1 - \frac{\sqrt{k-1}}{(k+1)\sqrt{2\pi}}\right)$*-competitive with sample size* $pn = \frac{n}{e}$.

2.1 Analysis Technique

Before proving Theorem 1, let us shed some light on the way we lower-bound the value of the submodular objective function. To this end, we consider the expected value of the set of all tentatively selected items T. In other words, we pretend all selections our algorithm tries to make are actually feasible. It seems natural to bound the expected value of T by adding up the marginal gains round-by-round given the tentative selections in earlier rounds. Unfortunately, this introduces complicated dependencies on the order of arrival of previous items. Therefore, we take a different approach and bound the respective marginal gains with respect of tentative selections in *future* rounds. The important insight is that this keeps the dependencies manageable.

▶ **Proposition 2.** *The set of all items T that are tentatively selected by Algorithm 1 has an expected value of* $\mathbf{E}\left[v(T)\right] \geq \left(\frac{\alpha}{e} - \frac{\alpha}{n}\right) \cdot v(\mathrm{OPT})$ *if the algorithm is run with sample size* $pn = \frac{n}{e}$.

Proof. Let $T^{\geq \ell}$ denote the set of tentatively selected items that arrive in or after round ℓ. Formally, we have $T^{\geq \ell} = \{j\} \cup T^{\geq \ell+1}$ if $j \in \mathcal{A}(U^{\leq \ell})$ and $T^{\geq \ell} = T^{\geq \ell+1}$ otherwise.

We consider a different random process to define the $T^{\geq \ell}$ random variables, which results in the same distribution. First, we draw one item from U uniformly to come last. This determines the value of $T^{\geq n}$. Then we continue by drawing on item out of the remaining ones to come second to last, determining $T^{\geq n-1}$. Generally, this means that conditioning on $U^{\leq \ell}$ and the values of $T^{\geq \ell'}$, for $\ell' > \ell$, the item j is drawn uniformly at random from $U^{\leq \ell}$ and the respective outcome determines $T^{\geq \ell}$.

We bound the expected tentative value collected in rounds ℓ to n conditioned on the items that arrived before round ℓ and conditioned on all items that are tentatively selected. Through this condition, the value of the sets $T^{\geq \ell+1}$ to $T^{\geq n}$ is already fixed. The expectation is only over the marginal gain of j with respect to the future tentatively selected items $T^{\geq \ell+1}$

$$\mathbf{E}\left[v(T^{\geq \ell}) \;\middle|\; U^{\leq \ell}, T^{\geq \ell'} \text{for all } \ell' > \ell\right] = \frac{1}{\ell}\Big(\sum_{j \in \mathcal{A}(U^{\leq \ell})} v(\{j\}|T^{\geq \ell+1})\Big) + v(T^{\geq \ell+1}) \ .$$

Due to submodularity, the gain of the set $\mathcal{A}(U^{\leq \ell})$ is at most the sum of the individual marginal gains of the items in $\mathcal{A}(U^{\leq \ell})$. This gives us

$$\sum_{j \in \mathcal{A}(U^{\leq \ell})} v(\{j\}|T^{\geq \ell+1}) \geq v\left(\mathcal{A}(U^{\leq \ell}) \;\middle|\; T^{\geq \ell+1}\right) \geq v\left(\mathcal{A}(U^{\leq \ell})\right) - v(T^{\geq \ell+1}) \ .$$

In the last inequality, we use monotony of the objective function. This yields

$$\mathbf{E}\left[v(T^{\geq \ell}) \;\middle|\; U^{\leq \ell}, T^{\geq \ell'} \text{for all } \ell' > \ell\right] \geq \frac{1}{\ell} v\left(\mathcal{A}(U^{\leq \ell})\right) + \left(1 - \frac{1}{\ell}\right) v(T^{\geq \ell+1}) \ .$$

We take the expectation over the remaining randomization and get the following recursion

$$\mathbf{E}\left[v(T^{\geq \ell})\right] \geq \frac{1}{\ell}\mathbf{E}\left[v\left(\mathcal{A}(U^{\leq \ell})\right)\right] + \left(1 - \frac{1}{\ell}\right)\mathbf{E}\left[v(T^{\geq \ell+1})\right] \ .$$

Observe that $\mathrm{OPT} \cap U^{\leq \ell}$ is fully contained in $U^{\leq \ell}$ and has size at most k. Therefore, the approximation guarantee of $\mathcal{A}$ yields that $v(\mathcal{A}(U^{\leq \ell})) \geq \alpha v(\mathrm{OPT} \cap U^{\leq \ell})$. Furthermore, submodularity gives us $\mathbf{E}\left[v(\mathrm{OPT} \cap U^{\leq \ell})\right] \geq \frac{\ell}{n} v(\mathrm{OPT})$ because each item is included in $U^{\leq \ell}$ with probability $\frac{\ell}{n}$. In combination, this gives us

$$\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right] \geq \alpha \mathbf{E}\left[v(\mathrm{OPT} \cap U^{\leq \ell})\right] \geq \alpha \frac{\ell}{n} v(\mathrm{OPT}) \ . \tag{1}$$

Now we solve the recursion

$$\mathbf{E}\left[v(T^{\geq \ell})\right] \geq \frac{\alpha}{n} v(\mathrm{OPT}) + \left(1 - \frac{1}{\ell}\right)\mathbf{E}\left[v(T^{\geq \ell+1})\right] = \sum_{j=\ell}^{n} \prod_{i=\ell}^{j-1} \left(1 - \frac{1}{i}\right) \frac{\alpha}{n} v(\mathrm{OPT}) \ .$$

We have $\prod_{i=\ell}^{j-1}\left(1 - \frac{1}{i}\right) = \frac{\ell-1}{j-1}$ and $\sum_{j=\ell}^{n} \frac{1}{j-1} \geq \ln(\frac{n}{\ell})$ for all $\ell \geq 2$. This yields

$$\mathbf{E}\left[v(T^{\geq \ell})\right] \geq \sum_{j=\ell}^{n} \prod_{i=\ell}^{j-1} \left(1 - \frac{1}{i}\right) \frac{\alpha}{n} v(\mathrm{OPT}) = \frac{\alpha}{n} v(\mathrm{OPT}) \sum_{j=\ell}^{n} \frac{\ell - 1}{j - 1} \geq \frac{\ell - 1}{n} \ln\left(\frac{n}{\ell}\right) \alpha v(\mathrm{OPT}) \ .$$

With $\ell = pn$ and sample size $pn = \frac{n}{e}$, we get

$$\mathbf{E}\left[v(T^{\geq pn})\right] \geq \frac{pn - 1}{n} \ln\left(\frac{1}{p}\right) \alpha v(\mathrm{OPT}) = \left(\frac{1}{e} - \frac{1}{n}\right) \alpha v(\mathrm{OPT}) \ . \qquad ◀$$

The probability of a tentative selection in round ℓ is $\frac{k}{\ell}$. This means, in expectation, we make $\sum_{\ell=\frac{n}{e}}^{n} \frac{k}{\ell} \approx k$ tentative selections. Therefore, for large values of k, it is likely that most tentative selections are feasible. This way, we could already derive guarantees for large k. However, for small k, the derived bound would be far to pessimistic. This is due to the fact that we bound the marginal gain of an item based on all *tentative* future ones. If some of them are indeed not feasible, we underestimate the contribution of earlier items. Therefore, Theorem 1 requires a more involved recursion that is based on the idea from this section, but also incorporates the probability that an item is feasible directly.

2.2 Proof of Theorem 1

To prove the theorem, we will derive a lower bound on the value collected by the algorithm starting from an arbitrary round $\ell \in [n]$ with an arbitrary remaining capacity $r \in \{0, 1, \dots, k\}$. The random variables $\mathrm{ALG}_r^{\geq \ell} \subseteq U$ represent the set of first r items that a hypothetical run of the algorithm would collect if it started the *for* loop of Algorithm 1 in round ℓ. Formally, we define them recursively as follows. We set $\mathrm{ALG}_0^{\geq \ell} = \emptyset$ for all ℓ and $\mathrm{ALG}_r^{\geq n+1} = \emptyset$ for all r. For $\ell \in [n]$, $r > 0$, let j be the item arriving in round ℓ, and $U^{\leq \ell}$ be the set of items arriving until and including round ℓ. We define $\mathrm{ALG}_r^{\geq \ell} = \{j\} \cup \mathrm{ALG}_{r-1}^{\geq \ell+1}$ if $j \in \mathcal{A}(U^{\leq \ell})$ and $\mathrm{ALG}_r^{\geq \ell} = \mathrm{ALG}_r^{\geq \ell+1}$ otherwise. Note that by this definition $\mathrm{ALG} = \mathrm{ALG}_k^{\geq pn}$. Furthermore, for every possible arrival order, $\mathrm{ALG}_r^{\geq \ell}$ is pointwise a superset of $\mathrm{ALG}_{r-1}^{\geq \ell}$ for $r > 0$.

In Lemma 3, we show a recursive lower bound on the value of these sets. In this part, the precise definition of $\mathrm{ALG}_r^{\geq \ell}$ will be crucial to avoid complex dependencies. Afterwards, in Lemma 4, we solve this recursion. Given this solution, we can finally prove Theorem 1.

▶ **Lemma 3.** *For all $\ell \in [n]$ and $r \in \{0, 1, \dots, k\}$, we have*

$$\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell})\right] \geq \frac{1}{\ell}\left(\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right] + (k-1)\mathbf{E}\left[v(\mathrm{ALG}_{r-1}^{\geq \ell+1})\right] + (\ell - k)\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell+1})\right]\right).$$

Proof. As explained in Section 2.1, we first draw one item from U uniformly at random to be the item that arrives in round n. This defines the values of $\mathrm{ALG}_r^{\geq n}$ for all r. Then we draw another item to be the second to last one and so on. In this way, we can condition on $U^{\leq \ell}$ and the values of $\mathrm{ALG}_r^{\geq \ell'}$, for $\ell' > \ell$ and all r. In round ℓ, the item j is drawn uniformly at random from $U^{\leq \ell}$ and the respective outcome determines $\mathrm{ALG}_r^{\geq \ell}$ for all r. This allows us to write for $r > 0$

$$\begin{aligned}
&\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell}) \;\middle|\; U^{\leq \ell}, \mathrm{ALG}_{r'}^{\geq \ell'} \text{ for all } \ell' > \ell \text{ and all } r'\right] \\
&v = \frac{1}{\ell}\left(\sum_{j \in \mathcal{A}(U^{\leq \ell})} v(\{j\} \cup \mathrm{ALG}_{r-1}^{\geq \ell+1}) + |U^{\leq \ell} \setminus \mathcal{A}(U^{\leq \ell})| v(\mathrm{ALG}_r^{\geq \ell+1})\right) \quad .
\end{aligned}$$

By submodularity, we have

$$\sum_{j \in \mathcal{A}(U^{\leq \ell})} \left(v(\{j\} \cup \mathrm{ALG}_{r-1}^{\geq \ell+1}) - v(\mathrm{ALG}_{r-1}^{\geq \ell+1})\right) \geq v(\mathcal{A}(U^{\leq \ell}) \cup \mathrm{ALG}_{r-1}^{\geq \ell+1}) - v(\mathrm{ALG}_{r-1}^{\geq \ell+1}) \quad ,$$

and hence

$$\sum_{j \in \mathcal{A}(U^{\leq \ell})} v(\{j\} \cup \mathrm{ALG}_{r-1}^{\geq \ell+1}) \geq v(\mathcal{A}(U^{\leq \ell}) \cup \mathrm{ALG}_{r-1}^{\geq \ell+1}) + (|\mathcal{A}(U^{\leq \ell})| - 1) v(\mathrm{ALG}_{r-1}^{\geq \ell+1}) \quad .$$

This gives us

$$\begin{aligned}
&\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell}) \;\middle|\; U^{\leq \ell}, \mathrm{ALG}_{r'}^{\geq \ell'} \text{ for all } \ell' > \ell \text{ and all } r'\right] \\
&\quad \geq \frac{1}{\ell} v(\mathcal{A}(U^{\leq \ell}) \cup \mathrm{ALG}_{r-1}^{\geq \ell+1}) + \frac{|\mathcal{A}(U^{\leq \ell})| - 1}{\ell} v(\mathrm{ALG}_{r-1}^{\geq \ell+1}) \\
&\qquad + \frac{|U^{\leq \ell} \setminus \mathcal{A}(U^{\leq \ell})|}{\ell} v(\mathrm{ALG}_r^{\geq \ell+1}) \quad .
\end{aligned}$$

Furthermore, by applying the monotonicity of v and the facts that $|\mathcal{A}(U^{\leq \ell})| \leq k$ and $\mathrm{ALG}_{r-1}^{\geq \ell+1} \subseteq \mathrm{ALG}_r^{\geq \ell+1}$, we get

$$\begin{aligned}
&\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell}) \;\middle|\; U^{\leq \ell}, \mathrm{ALG}_{r'}^{\geq \ell'} \text{ for all } \ell' > \ell \text{ and all } r'\right] \\
&\quad \geq \frac{1}{\ell}\left(v(\mathcal{A}(U^{\leq \ell})) + (k-1) v(\mathrm{ALG}_{r-1}^{\geq \ell+1}) + (\ell - k) v(\mathrm{ALG}_r^{\geq \ell+1})\right) \quad .
\end{aligned}$$

Taking the expectation over all remaining randomization yields

$$\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell})\right] \geq \frac{1}{\ell}\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right] + \frac{k-1}{\ell}\mathbf{E}\left[v(\mathrm{ALG}_{r-1}^{\geq \ell+1})\right] + \frac{\ell-k}{\ell}\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell+1})\right] . \blacktriangleleft$$

The next step is to solve the recursion.

▶ **Lemma 4.** *For all $\ell \in [n]$, $\ell \geq k^2+k$, and $r \in \{0,1,\ldots,k\}$, we have*

$$\frac{\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell})\right]}{v(\mathrm{OPT})} \geq \left(\frac{r\ell}{(k-1)n} - \frac{1}{k-1}\left(\frac{\ell}{n}\right)^k \sum_{r'=0}^{r-1}\sum_{i=0}^{r'} \frac{(k-1)^i}{i!}\ln^i\left(\frac{n}{\ell}\right) - \frac{3k^2 r}{(k-1)n}\right)\alpha \ . \quad (2)$$

Proof (Outline). As a first step, we eliminate the recursive reference from $\mathrm{ALG}_r^{\geq \ell}$ to $\mathrm{ALG}_r^{\geq \ell+1}$. To this end, we count the rounds until the next item is accepted. Repeatedly inserting the bound for $\mathrm{ALG}_r^{\geq \ell+1}$ into the one for $\mathrm{ALG}_r^{\geq \ell}$ gives us

$$\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell})\right] \geq \sum_{j=\ell}^{n}\left(\prod_{i=\ell}^{j-1}\left(1-\frac{k}{i}\right)\left(\frac{k-1}{j}\mathbf{E}\left[v(\mathrm{ALG}_{r-1}^{\geq j+1})\right] + \frac{1}{j}\mathbf{E}\left[v(\mathcal{A}(U^{\leq j}))\right]\right)\right) .$$

With Equation (1) in Section 2.1 we have $\mathbf{E}\left[v(\mathcal{A}(U^{\leq j}))\right] \geq \frac{j}{n}\alpha v(\mathrm{OPT})$.

We use $\prod_{i=\ell}^{j-1}\left(1-\frac{k}{i}\right) = \frac{(\ell-1)!}{(\ell-k-1)!}\frac{(j-k-1)!}{(j-1)!} \geq \left(\frac{\ell-k}{j-k}\right)^k$ and get

$$\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell})\right] \geq \sum_{j=\ell}^{n}\left(\left(\frac{\ell-k}{j-k}\right)^k\left(\frac{k-1}{j+1}\mathbf{E}\left[v(\mathrm{ALG}_{r-1}^{\geq j+1})\right] + \frac{\alpha}{n}v(\mathrm{OPT})\right)\right) . \quad (3)$$

It can be shown that (2) provides a lower bound on the functions defined by this recursion. For details, see Appendix A.1. ◀

Proof of Theorem 1. To complete the proof of the theorem, we apply Lemma 4 for $\ell = pn$ and $r = k$. This gives us $\mathbf{E}[v(\mathrm{ALG})] = \mathbf{E}\left[v(\mathrm{ALG}_k^{\geq pn})\right]$ and thus

$$\mathbf{E}[v(\mathrm{ALG})] \geq \left(\frac{pk}{k-1} - \frac{1}{k-1}p^k\sum_{r'=0}^{k-1}\sum_{i=0}^{r'}\frac{(k-1)^i}{i!}\ln^i\left(\frac{1}{p}\right) - \frac{6k^2}{n}\right)\cdot \alpha v(\mathrm{OPT}) .$$

For p such that $pn = \lceil \frac{n}{e} \rceil$, we have $p \leq \frac{1}{e} + \frac{1}{n}$ and $\ln\left(\frac{1}{p}\right) = 1 + \ln\left(\frac{n}{n+e}\right) \leq 1$. For sake of readability, we omit the error term in the remainder of the proof. The more detailed calculation is included in Appendix A.2. With $p = \frac{1}{e}$, we have $\ln\left(\frac{1}{p}\right) = 1$, this allows us to reorder the double sum as follows

$$\sum_{r'=0}^{k-1}\sum_{i=0}^{r'}\frac{(k-1)^i}{i!} = \sum_{i=0}^{k-1}(k-i)\frac{(k-1)^i}{i!} = \sum_{i=0}^{k-1}\frac{(k-1)^i}{i!} + \frac{(k-1)^k}{(k-1)!} .$$

By definition of the exponential function $e^x = \sum_{i=0}^{\infty}\frac{x^i}{i!}$. For $x > 0$, all terms of the infinite sum are positive. This yields $e^x \geq \sum_{i=0}^{k-1}\frac{x^i}{i!} + \frac{x^k}{k!} + \frac{x^{k+1}}{(k+1)!}$ and thus by setting $x = k-1$ we get

$$\sum_{r'=0}^{k-1}\sum_{i=0}^{r'}\frac{(k-1)^i}{i!} \leq e^{k-1} - \frac{(k-1)^k}{k!} - \frac{(k-1)^{k+1}}{(k+1)!} + \frac{(k-1)^k}{(k-1)!} .$$

This implies

$$\frac{\mathbf{E}\left[v(\mathrm{ALG})\right]}{\alpha v(\mathrm{OPT})} \geq \frac{k}{e(k-1)} - \frac{1}{e^k(k-1)}\left(e^{k-1} - \frac{(k-1)^k}{k!} - \frac{(k-1)^{k+1}}{(k+1)!} + \frac{(k-1)^k}{(k-1)!}\right) - \frac{6k^2}{n}$$
$$= \frac{1}{e} - \frac{1}{e^k}\frac{k-1}{k+1}\frac{(k-1)^{k-1}}{(k-1)!} - \frac{6k^2}{n} \quad .$$

It only remains to apply the Stirling approximation $(k-1)! \geq \sqrt{2\pi(k-1)}\left(\frac{k-1}{e}\right)^{k-1}$ to get

$$\frac{\mathbf{E}\left[v(\mathrm{ALG})\right]}{\alpha v(\mathrm{OPT})} \geq \frac{1}{e}\left(1 - \frac{\sqrt{k-1}}{(k+1)\sqrt{2\pi}}\right) - \frac{6k^2}{n} \quad . \qquad \blacktriangleleft$$

2.3 Improved Analysis for the Greedy Algorithm

One possible choice for the algorithm $\mathcal{A}$ is the greedy algorithm by Nemhauser and Wolsey [29]. It repeatedly picks the item with the highest marginal increase compared to the items chosen so far until k items have been picked. As pointed out in [22], the approximation guarantee would improve further when picking more items according to the greedy rule. In other words, if we let our algorithm pick k elements but compare the outcome to the optimal solution of only k' items, the approximation factor improves to $1 - \exp\left(-\frac{k}{k'}\right)$.

We can exploit this fact in the analysis of the online algorithm that uses the greedy algorithm as $\mathcal{A}$ in Algorithm 1. The reason is that in early rounds only some items of the optimal solution have arrived. Our algorithm, however, always chooses a set of size k for $S^{(\ell)} = \mathcal{A}(U^{\leq \ell})$. In the generic analysis, we show that $\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right] \geq \alpha \frac{\ell}{n} v(\mathrm{OPT})$. In case of $\mathcal{A}$ being the greedy algorithm, we can improve this bound as follows.

▶ **Lemma 5.** $\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right] \geq \alpha_\ell \frac{\ell}{n} v(\mathrm{OPT})$ *for* $\alpha_\ell = 1 - \frac{\ell}{en} - \frac{1}{ek}$.

Proof. Consider the offline optimum OPT and $\mathrm{OPT} \cap U^{\leq \ell}$, its restriction to the items that arrive by round ℓ. Let $Z = |\mathrm{OPT} \cap U^{\leq \ell}|$ be the number of OPT items that arrive by round ℓ.

Condition on any value of Z. Observe that by symmetry the probably of every OPT item to have arrived by round ℓ is $\frac{Z}{k}$. Therefore, submodularity implies $\mathbf{E}\left[v(\mathrm{OPT} \cap U^{\leq \ell}) \mid Z\right] \geq \frac{Z}{k} v(\mathrm{OPT})$. If the greedy algorithm picks k elements, it achieves value at least $\left(1 - \exp\left(-\frac{k}{Z}\right)\right) \cdot v(\mathrm{OPT} \cap U^{\leq \ell})$. In combination, this gives us $\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell})) \mid Z\right] \geq \left(1 - \exp\left(-\frac{k}{Z}\right)\right) \frac{Z}{k} v(\mathrm{OPT})$. We now use the fact that $\exp\left(\frac{k}{Z}\right) \geq e\frac{k}{Z}$ because $Z \leq k$. Therefore $\exp\left(-\frac{k}{Z}\right) \leq \frac{Z}{ek}$ and $\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell})) \mid Z\right] \geq \left(1 - \frac{Z}{ek}\right) \frac{Z}{k} v(\mathrm{OPT})$.

It remains to take the expectation over Z. We have $\mathbf{E}\left[Z\right] = \frac{\ell}{n} k$. Letting $Z_j = 1$ if $j \in U^{\leq \ell}$ and 0 otherwise, we have and $\mathbf{E}\left[Z^2\right] = \mathbf{E}\left[\sum_{j \in \mathrm{OPT}} Z_j + \sum_{j \in \mathrm{OPT}} \sum_{j' \in \mathrm{OPT}, j' \neq j} Z_j Z_{j'}\right] = \frac{\ell}{n} k + k(k-1)\frac{\ell}{n}\frac{\ell-1}{n-1} \leq \frac{\ell}{n} k + \left(\frac{\ell}{n} k\right)^2$. This implies

$$\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right] \geq \left(\frac{\mathbf{E}\left[Z\right]}{k} - \frac{\mathbf{E}\left[Z^2\right]}{ek^2}\right) v(\mathrm{OPT}) \geq \left(\frac{\ell}{n} - \frac{\ell^2}{en^2} - \frac{\ell}{ekn}\right) v(\mathrm{OPT}) \quad . \qquad \blacktriangleleft$$

Given this lemma, we can follow similar steps as in the proof of Theorem 1 to show an improved guarantee of this particular algorithm. In more detail, we get competitive ratios of at least 0.177 for any $k \geq 2$. Asymptotically for large k we reach 0.275.

▶ **Theorem 6.** *If the greedy algorithm is used as blackbox approximation algorithm $\mathcal{A}$, then Algorithm 1 is* $\frac{1 + \frac{1}{2e^3} - \frac{3}{2e} - \frac{e-1}{e^2 k}}{e-1}\left(1 - \frac{\sqrt{k-1}}{(k+1)\sqrt{2\pi}}\right)$*-competitive with sample size* $pn = \frac{n}{e}$.

To prove Theorem 6, we combine Lemmas 3 and 5, which give us a recursive formula for $\mathrm{ALG}_r^{\geq \ell}$. We first solve the recursion (Claim 7) and then show that the occurring coefficients are non-increasing (Claim 8). This then allows to apply Chebyshev's sum inequality.

▶ **Claim 7.** *Lemma 3 implies*

$$\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell})\right] \geq \sum_{j=\ell}^{n} \frac{a_{\ell,j-1}}{j} \mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right] \sum_{r'=0}^{r-1} \sum_{\substack{M \subseteq \{\ell,\dots,j-1\} \\ |M|=r'}} \left(\prod_{i \in M} \frac{k-1}{i} \right)$$

with $a_{\ell,j-1} = \prod_{i=\ell}^{j-1} \left(1 - \frac{k}{i}\right)$.

The proof of this claim is by induction and it is included in Appendix A.3.

▶ **Claim 8.** *Let*

$$t_{\ell,j} = a_{\ell,j-1} \sum_{r'=0}^{r-1} \sum_{\substack{M \subseteq \{\ell,\dots,j-1\} \\ |M|=r'}} \left(\prod_{i \in M} \frac{k-1}{i} \right)$$

with $a_{\ell,j-1} = \prod_{i=\ell}^{j-1} \left(1 - \frac{k}{i}\right)$. *For fixed* ℓ, *the sequence* $t_{\ell,j}$ *is non-increasing in* j.

The proof of this claim is included in Appendix A.4.

Proof of Theorem 6. Now we can proceed to the proof of Theorem 6. So far, we have shown that

$$\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell})\right] \geq \sum_{j=\ell}^{n} \frac{t_{\ell,j}}{j} \mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right] \quad \text{for} \quad t_{\ell,j} = a_{\ell,j-1} \sum_{r'=0}^{r-1} \sum_{\substack{M \subseteq \{\ell,\dots,j-1\} \\ |M|=r'}} \left(\prod_{i \in M} \frac{k-1}{i} \right)$$

with $a_{\ell,j-1} = \prod_{i=\ell}^{j-1} \left(1 - \frac{k}{i}\right)$. Furthermore, Lemma 5 shows that $\frac{\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right]}{j} \geq \frac{\alpha_j v(\mathrm{OPT})}{n}$ for $\alpha_\ell = 1 - \frac{\ell}{en} - \frac{1}{ek}$.

As both $t_{\ell,j}$ and α_j are non-increasing in j, we can use Chebyshev's sum inequality to get

$$\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell})\right] \geq \sum_{j=\ell}^{n} t_{\ell,j} \frac{\alpha_j v(\mathrm{OPT})}{n} \geq \left(\sum_{j=\ell}^{n} t_{\ell,j} \frac{v(\mathrm{OPT})}{n} \right) \left(\frac{1}{n-\ell} \sum_{j=\ell}^{n} \alpha_j \right) \quad .$$

It now remains to bound these two terms.

First, we show that the sum $\sum_{j=\ell}^{n} t_{\ell,j} \frac{c}{n}$ with $c = v(\mathrm{OPT})$ is lower-bounded by a recursion of the form of Equation (3). Similar calculations to Lemma 4 will then give us the respective bound. Similar to the previous proof, we use $a_{\ell,j-1} = \prod_{i=\ell}^{j-1} \left(1 - \frac{k}{i}\right) \geq \left(\frac{\ell-k}{j-k}\right)^k$ and get

$$\begin{aligned} \sum_{j=\ell}^{n} t_{\ell,j} \frac{v(\mathrm{OPT})}{n} &= \sum_{j=\ell}^{n} a_{\ell,j-1} \sum_{r'=0}^{r-1} \sum_{\substack{M \subseteq \{\ell,\dots,j\} \\ |M|=r'}} \left(\prod_{i \in M} \frac{k-1}{i} \right) \frac{c}{n} \\ &\geq \sum_{j=\ell}^{n} \left(\frac{\ell-k}{j-k} \right)^k \sum_{r'=0}^{r-1} \sum_{\substack{M \subseteq \{\ell,\dots,j\} \\ |M|=r'}} \left(\prod_{i \in M} \frac{k-1}{i+1} \right) \frac{c}{n} \quad . \end{aligned}$$

Let now

$$b_{\ell,r'} = \sum_{j=\ell}^{n} \left(\frac{\ell-k}{j-k}\right)^k \sum_{r'=0}^{r-1} \sum_{\substack{M\subseteq\{\ell,\dots,j\}\\ |M|=r'}} \left(\prod_{i\in M} \frac{k-1}{i+1}\right) \frac{c}{n} \ .$$

We combine the two inner sums and then pull out the earliest element $m \in M \subseteq \{\ell, \dots, j\}$ recursively. We move the corresponding factor out of the product and get

$$\begin{aligned} b_{\ell,r'} &= \sum_{j=\ell}^{n} \left(\frac{\ell-k}{j-k}\right)^k \sum_{\substack{M\subseteq\{\ell,\dots,j\}\\ |M|\leq r'}} \left(\prod_{i\in M} \frac{k-1}{i+1}\right) \frac{c}{n} \\ &= \sum_{j=\ell}^{n} \left(\frac{\ell-k}{j-k}\right)^k \left(\frac{c}{n} + \sum_{m=\ell}^{j-1} \frac{k-1}{m+1} \sum_{\substack{M\subseteq\{m+1,\dots,j\}\\ |M|\leq r'-1}} \left(\prod_{i\in M} \frac{k-1}{i+1}\right) \frac{c}{n} \right) \ . \end{aligned}$$

At this point, we change the order of summation such that we sum over m first. We can keep the constant part in place, since both sums $\sum_{j=\ell}^{n} \left(\frac{\ell-k}{j-k}\right)^k = \sum_{m=\ell}^{n} \left(\frac{\ell-k}{m-k}\right)^k$ amount the same. Now the inner part matches the recursion given above

$$\begin{aligned} b_{\ell,r'} &= \sum_{m=\ell}^{n} \left(\frac{\ell-k}{m-k}\right)^k \left(\frac{c}{n} + \frac{k-1}{m} \sum_{j=m+1}^{n} \left(\frac{m-k}{j-k}\right)^k \sum_{\substack{M\subseteq\{m+1,\dots,j\}\\ |M|\leq r'-1}} \left(\prod_{i\in M} \frac{k-1}{i}\right) \frac{c}{n} \right) \\ &= \sum_{m=\ell}^{n} \left(\frac{\ell-k}{m-k}\right)^k \left(\frac{c}{n} + \frac{k-1}{m} b_{m+1,r'-1} \right) \ . \end{aligned}$$

From this point on, we follow the proof of Lemma 4 in Appendix A.1 and get the following lemma.

▶ **Lemma 9.** *Given a recursion of the form*

$$b_{\ell,r} = \sum_{j=\ell}^{n} \left(\left(\frac{\ell-k}{j-k}\right)^k \left(\frac{k-1}{j+1} b_{j+1,r-1} + \frac{c}{n}\right) \right)$$

with $b_{n+1,r} = 0$ and $b_{\ell,0} = 0$. Then

$$b_{\ell,r} \geq \left(\frac{r(\ell-k)}{(k-1)n} - \frac{1}{k-1} \left(\frac{\ell-k}{n-k}\right)^k \sum_{r'=0}^{r-1} \sum_{i=0}^{r'} \frac{(k-1)^i}{i!} \ln^i\left(\frac{n}{\ell}\right) - \frac{3k^2 r}{(k-1)n} \right) c \ .$$

Consequently, following the calculations in the proof of Theorem 1

$$\mathbf{E}\left[v(\mathrm{ALG})\right] = \mathbf{E}\left[v(\mathrm{ALG}_k^{\geq n/e})\right] \geq \frac{1}{e}\left(1 - \frac{\sqrt{k-1}}{(k+1)\sqrt{2\pi}} - \frac{6ek^2}{n}\right) \left(\frac{1}{n - n/e} \sum_{j=n/e}^{n} \alpha_j \right) v(\mathrm{OPT}).$$

For $\alpha_j = 1 - \frac{j}{en} - \frac{1}{ek}$, we can bound the last term through the integral and get

$$\frac{1}{n - n/e} \sum_{j=n/e}^{n} \left(1 - \frac{j}{en} - \frac{1}{ek}\right) \geq \frac{1}{1 - 1/e} \left(1 + \frac{1}{2e^3} - \frac{3}{2e} - \frac{e-1}{e^2 k}\right) \ .$$

For large k, we have an asymptotic competitive ratio of $\frac{1}{e}\left(1 + \frac{1}{2e^3} - \frac{3}{2e}\right) \approx 0.275$. ◀

Algorithm 2: Submodular Bipartite Online Matching

Drop the first $\lceil pn \rceil - 1$ vertices;
for *vertex* $u \in L$ *in round* $\ell \geq \lceil pn \rceil$ **do** `// online steps` $\ell = \lceil pn \rceil$ `to` n
  Set $L^{\leq \ell} := L^{\leq \ell - 1} \cup \{u\}$;
  Let $M^{(\ell)} = \mathcal{A}(L^{\leq \ell} \cup R)$; `// black box` α`-approximation`
  Let $e^{(\ell)} := (u, r)$ be the edge assigned to u in $M^{(\ell)}$; `// tentative edge`
  if Accepted $\cup\, e^{(\ell)}$ is a matching **then** `// feasibility test`
    Add $e^{(\ell)}$ to Accepted; `// online allocation`

3 Submodular Matching

Next, we consider the online submodular bipartite matching problem. In the offline version, we are given a bipartite graph $G = (L \cup R, E)$ and a monotone, submodular, non-decreasing objective function $v\colon 2^E \to \mathbb{R}_{\geq 0}$. The objective is to find a matching $M \subseteq E$ that maximizes $v(M)$. In the online version, the set L arrives online. Once a vertex in L arrives, we get to know its incident edges. At any point in time, we know the values of the objective function only restricted to subsets of the edges incident to the vertices that have already arrived. This problem also generalizes the submodular matroid secretary problem with transversal matroids.

We present a 0.207α-competitive algorithm, where α could be $\frac{1}{3}$ for a simple greedy algorithm [28]. The best known approximation algorithms are local search algorithms that give a $\frac{1}{2+\epsilon}$-approximation on bipartite matchings [25, 12]. The previously best known online algorithm is the simulated greedy algorithm with a competitive ratio of $1/95$ [26].

Algorithm 2 first samples a pn-fraction of the input sequence for some constant p. Then, whenever a new candidate arrives, it α-approximates the optimal matching on the known part of the graph with respect to the submodular objective function. If the current online vertex is matched in this matching and if its matching partner is still available, then we add the pair to the output allocation. This design paradigm has been successfully applied to linear objective functions before [17]. However, in the submodular case, the individual contribution on an edge to the eventual objective function value depends on what other edges are selected. Using an approach similar to the one in the previous section, we keep dependencies manageable.

▶ **Theorem 10.** *Algorithm 2 for the submodular secretary matching problem is $\alpha(1-p^{1/p})(p^2 - O(\frac{1}{n}))$-competitive with sample size pn. For $p = 0.614$, the algorithm is 0.207α-competitive.*

We denote the set of matching edges allocated by the algorithm in rounds ℓ to n with $\mathrm{ALG}^{\geq \ell}$ and the set of tentative edges over the same period with $T^{\geq \ell}$. For $S, S' \subseteq E$, we denote the contribution of the subset S to S' by $v(S \mid S') = v(S \cup S') - v(S')$.

We show the following two lemmas.

▶ **Lemma 11.** *In every round ℓ fix the tentative edges that will be selected in the future rounds $\ell+1, \ldots, n$. Then the marginal contribution of the tentative edge $e^{(\ell)}$ selected by the online algorithm in round ℓ is*

$$\mathbf{E}\left[v\left(\{e^{(\ell)}\} \,\middle|\, \mathrm{ALG}^{\geq \ell+1}\right) \,\middle|\, L^{\leq \ell}, T^{\geq \ell+1}\right] \geq \frac{1}{\ell}\left(v(\mathcal{A}(L^{\leq \ell})) - v(T^{\geq \ell+1})\right) .$$

Proof. We will use that $v\left(\{e^{(\ell)}\} \,\middle|\, \mathrm{ALG}^{\geq \ell+1}\right) \geq v\left(\{e^{(\ell)}\} \mid T^{\geq \ell+1}\right)$ because of submodularity of v and since $\mathrm{ALG}^{\geq \ell+1} \subseteq T^{\geq \ell+1}$. This allows us to avoid complex dependencies.

With $L^{\leq \ell}$ fixed, the algorithm's output $\mathcal{A}(L^{\leq \ell})$ is determined as well. The online vertex in round ℓ is as drawn uniformly at random from all vertices in $L^{\leq \ell}$. This gives us

$$\mathbf{E}\left[v\left(\{e^{(\ell)}\} \,\middle|\, T^{\geq \ell+1}\right) \,\middle|\, L^{\leq \ell}, T^{\geq \ell+1}\right] \geq \frac{1}{\ell} v\left(\mathcal{A}(L^{\leq \ell}) \mid T^{\geq \ell+1}\right)$$
$$\geq \frac{1}{\ell}\left(v(\mathcal{A}(L^{\leq \ell})) - v(T^{\geq \ell+1})\right) \quad . \blacktriangleleft$$

This lemma is shown in a way similar to Proposition 2.

▶ **Lemma 12.** *The probability that a tentative edge $e^{(\ell)}$ is feasible given all vertices that arrived earlier $L^{\leq \ell}$ and all future tentative edges $T^{\geq \ell+1}$ is*

$$\mathbf{Pr}\left[\text{Accepted} \cup e^{(\ell)} \textit{ is a matching} \,\middle|\, L^{\leq \ell}, T^{\geq \ell+1}\right] \geq \frac{pn-1}{\ell-1} \quad .$$

This lemma was already shown in [17].

Proof of Theorem 10. Let $\hat{e}^{(\ell)} = \{e^{(\ell)}\}$ if Accepted$\cup e^{(\ell)}$ is a matching and empty otherwise. We combine Lemmas 11 and 12, and we get that in every round ℓ for a fixed set $L^{\leq \ell}$ and $T^{\geq \ell+1}$, we have

$$\mathbf{E}\left[v\left(\hat{e}^{(\ell)} \,\middle|\, \text{ALG}^{\geq \ell+1}\right) \,\middle|\, L^{\leq \ell}, T^{\geq \ell+1}\right] \geq \frac{1}{\ell}\frac{pn-1}{\ell-1}\left(v(\mathcal{A}(L^{\leq \ell} \cup R)) - v(T^{\geq \ell+1})\right)$$

and therefore

$$\mathbf{E}\left[v\left(\hat{e}^{(\ell)} \,\middle|\, \text{ALG}^{\geq \ell+1}\right)\right] \geq \frac{1}{\ell}\frac{pn-1}{\ell-1}\left(\mathbf{E}\left[v(\mathcal{A}(L^{\leq \ell} \cup R))\right] - \mathbf{E}\left[v(T^{\geq \ell+1})\right]\right) \quad .$$

We use Lemma 12 for each future tentative edge $e^{(\ell')} \in T^{\geq \ell+1}$ and upperbound $\ell' \leq n$. This gives us $\mathbf{E}\left[v(\text{ALG}^{\geq \ell+1})\right] \geq p\mathbf{E}\left[v(T^{\geq \ell+1})\right]$. Furthermore, to bound $\mathbf{E}\left[v(\mathcal{A}(L^{\leq \ell} \cup R))\right]$, we use that the optimal solution on the subgraph induced by $L^{\leq \ell} \cup R$ is at least as good as the optimal solution restricted to the edges in this subgraph. As every edge appears with probability $\frac{\ell}{n}$ submodularity gives us $\mathbf{E}\left[v(\mathcal{A}(L^{\leq \ell} \cup R))\right] \geq \alpha\frac{\ell}{n}v(\text{OPT})$. In combination with $\ell \geq pn$, this yields

$$\mathbf{E}\left[v\left(\hat{e}^{(\ell))} \,\middle|\, \text{ALG}^{\geq \ell+1}\right)\right] \geq \frac{\alpha}{n}\frac{pn-1}{\ell-1}v(\text{OPT}) - \frac{1}{\ell}\frac{1}{p}\mathbf{E}\left[v(\text{ALG}^{\geq \ell+1})\right] \quad .$$

As $\text{ALG}^{\geq \ell} = \hat{e}^{(\ell)} \cup \text{ALG}^{\geq \ell+1}$, we get the following tail recursion

$$\mathbf{E}\left[v((\text{ALG}^{\geq \ell})\right] \geq \frac{\alpha}{n}\frac{pn-1}{\ell-1}v(\text{OPT}) + \left(1 - \frac{1/p}{\ell}\right)\mathbf{E}\left[v(\text{ALG}^{\geq \ell+1})\right]$$
$$\geq \sum_{j=\ell}^{n}\prod_{i=\ell}^{j-1}\left(1 - \frac{1/p}{i}\right)\frac{1}{j-1}\left(p - \frac{1}{n}\right)\alpha v(\text{OPT}) \quad .$$

We use $\prod_{i=\ell}^{j-1}\left(1 - \frac{1/p}{i}\right) \geq \left(\frac{\ell - 1/p}{j-1/p}\right)^{1/p}$, see Lemma 14 in Appendix B.1 for a proof. Additionally we use $\frac{1}{j-1} = \frac{1}{j-1/p}\frac{j-1/p}{j-1} = \frac{1}{j-1/p}\left(1 - \frac{1/p-1}{j-1}\right) \geq \frac{1}{j-1/p}\left(1 - \frac{1/p-1}{pn-1}\right)$ and get

$$\mathbf{E}\left[v(\text{ALG}^{\geq \ell})\right] \geq \sum_{j=\ell}^{n}\frac{(\ell - 1/p)^{1/p}}{(j-1/p)^{1/p+1}}\left(1 - \frac{1/p-1}{pn-1}\right)\left(p - \frac{1}{n}\right)\alpha\text{OPT} \quad .$$

We approximate the sum with the integral and get $\sum_{j=\ell}^{n} \frac{1}{(j-1/p)^{1/p+1}} \geq \int_{\ell}^{n} \frac{1}{(j-1/p)^{1/p+1}} dj - \frac{1}{(\ell-1-1/p)^{1/p+1}} = p\left(\frac{1}{(\ell-1/p)^{1/p}} - \frac{1}{(n-1/p)^{1/p}} - \frac{1}{(\ell-1-1/p)^{1/p+1}}\right)$. Together with $\frac{1}{n} = \frac{p-1/n}{pn-1}$ this gives us

$$\frac{\mathbf{E}\left[v(\mathrm{ALG}^{\geq \ell})\right]}{\mathrm{OPT}} \geq \alpha\left(1 - \left(\frac{\ell - 1/p}{n - 1/p}\right)^{1/p} - \frac{(\ell - 1/p)^{1/p}}{(\ell - 1 - 1/p)^{1/p+1}}\right)\left(p^2 - \frac{1 + p^2 - p - p/n}{pn - 1}\right) .$$

Now the expected value of the online algorithm is $\mathbf{E}\left[v(\mathrm{ALG}^{\geq pn})\right]$. We have $\frac{pn-1/p}{n-1/p} = p\frac{n-1/p^2}{n-1/p} \leq p$ and $\frac{(\ell-1/p)^{1/p}}{(\ell-1-1/p)^{1/p+1}} = \left(1 + \frac{1}{\ell-1-1/p}\right)^{1/p} \frac{1}{\ell-1-1/p} \in O\left(\frac{1}{n}\right)$. This gives us

$$\mathbf{E}\left[v(\mathrm{ALG}^{\geq pn})\right] \geq \left(1 - p^{1/p}\right)\left(p^2 - O\left(\frac{1}{n}\right)\right)\alpha v(\mathrm{OPT}) . \qquad \blacktriangleleft$$

This bound on the expected competitive ratio has a local maximum of 0.207α when the parameter for the sample size is $p = 0.614$.

4 Submodular Function subject to Linear Packing Constraints

We now generalize the setting to feature arbitrary linear packing constraints. That is, each item j is associated a variable y_j and there are m constraints of the form $\sum_{j \in U} a_{i,j} y_j \leq b_i$ with $a_{i,j} \geq 0$. The coefficients $a_{i,j}$ are chosen by an adversary and are revealed to the online algorithm once the respective item arrives. Immediately and irrevocably, we have to either accept or reject the item, which corresponds to setting y_j to 0 or 1. The best previous result is a constant competitive algorithm for a single constraint and $\Omega(1/m)$-competitive for multiple constraints, where m is the number of constraints [4].

Our algorithms extend the ones presented in [18] from linear to submodular objective. Again, they rely on a suitable algorithm solving the offline optimization problem. In this case we need a fractional allocation $x \in [0,1]^U$, which we evaluate in terms of the multilinear extension $F(x) = \sum_{R \subseteq U}\left(\prod_{i \in R} f(R) x_i \prod_{i \notin R}(1 - x_i)\right)$. In more detail, we assume that for any packing polytope $P \subseteq [0,1]^U$, $F(\mathcal{A}_F(P)) \geq \alpha \sup_{x \in P} F(x)$. For example, the continuous greedy process by Calinescu et al. [7] provides a $(1 - 1/e)$-approximation in polynomial time. As the set P, we use $\mathcal{P}(\frac{\ell}{n}, S)$, which is defined to be the set of vectors $x \geq 0$, for which $Ax \leq \frac{\ell}{n} b$ and $x_i = 0$ if $i \notin S$. This is the polytope of the solution space with scaled down constraints and restricted on the variables that arrived so far.

Our bounds are parameterized in the capacity ratio B and the column sparsity d. The capacity ratio B is defined by $B = \min_{i \in [m]} \frac{b_i}{\max_{j \in [n]} a_{i,j}}$. The column sparsity d is the maximal number of non-zero entries in a column of the constraint matrix A. We consider two variants of this problem, where either the B and d are known to the algorithm or not.

▶ **Theorem 13.** *There is an $\Omega\left(\alpha d^{-\frac{2}{B-1}}\right)$-competitive online algorithm for submodular maximization subject to linear constraints with unknown capacity ratio $B \geq 2$ and unknown column sparsity d. This improves to $\Omega\left(\alpha d^{-\frac{1}{B-1}}\right)$ if B and d are known.*

Note that, although the algorithm $\mathcal{A}$ returns fractional solutions, the output of our online algorithms is integral. The competitive ratio is between the integral solution of the online algorithm and the optimal fractional allocation with respect to the multilinear extension.

The proof for Theorem 13 combines ideas from Section 2 and 3 with [18]. Due to space limitations, the details are only included in the full version.

Algorithm 3: Submodular Function Maximization subject to Linear Constraints

```
Let x := 0 and S := ∅ be the index set of known requests;
for each arriving request j do                                   // steps ℓ = 1 to n
    Set S := S ∪ {j} and ℓ := |S|;
    Let x̃^(ℓ) := A_F(P(ℓ/n, S));        // fractional α-approximation on scaled polytope
    Set x̂_j^(ℓ) = 1 with probability x̃_j^(ℓ);   // tentative allocation after rand. rounding
    if A(x + x̂^(ℓ)) ≤ b then                                    // feasibility test
        Set x^(ℓ) := x̂^(ℓ), x := x + x̂^(ℓ);                      // online allocation
```

References

1 Shipra Agrawal and Nikhil R. Devanur. Fast algorithms for online stochastic convex programming. In *Proc. 26th Symp. Discr. Algorithms (SODA)*, pages 1405–1424, 2015. doi:10.1137/1.9781611973730.93.

2 Shipra Agrawal, Zizhuo Wang, and Yinyu Ye. A dynamic near-optimal algorithm for online linear programming. *Operations Research*, 62(4):876–890, 2014. doi:10.1287/opre.2014.1289.

3 Moshe Babaioff, Nicole Immorlica, and Robert Kleinberg. Matroids, secretary problems, and online mechanisms. In *Proc. 18th Symp. Discr. Algorithms (SODA)*, pages 434–443, 2007. URL: http://dl.acm.org/citation.cfm?id=1283383.1283429.

4 MohammadHossein Bateni, Mohammad Taghi Hajiaghayi, and Morteza Zadimoghaddam. Submodular secretary problem and extensions. *ACM Trans. Algorithms*, 9(4):32, 2013. doi:10.1145/2500121.

5 Niv Buchbinder and Moran Feldman. Constrained submodular maximization via a non-symmetric technique. *CoRR*, abs/1611.03253, 2016. URL: http://arxiv.org/abs/1611.03253.

6 Niv Buchbinder, Moran Feldman, Joseph Naor, and Roy Schwartz. Submodular maximization with cardinality constraints. In *Proc. 25th Symp. Discr. Algorithms (SODA)*, pages 1433–1452, 2014. doi:10.1137/1.9781611973402.106.

7 Gruia Călinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM J. Comput.*, 40(6):1740–1766, 2011. doi:10.1137/080733991.

8 Nikhil R. Devenur and Thomas P. Hayes. The adwords problem: online keyword matching with budgeted bidders under random permutations. In *Proc. 10th Conf. Econom. Comput. (EC)*, pages 71–78, 2009. doi:10.1145/1566374.1566384.

9 Alina Ene and Huy L. Nguyen. Constrained submodular maximization: Beyond 1/e. In *Proc. 57th Symp. Foundations of Computer Science (FOCS)*, pages 248–257, 2016. doi:10.1109/FOCS.2016.34.

10 Moran Feldman and Rani Izsak. Building a good team: Secretary problems and the supermodular degree. In *Proc. 28th Symp. Discr. Algorithms (SODA)*, pages 1651–1670, 2017. doi:10.1137/1.9781611974782.109.

11 Moran Feldman, Joseph Naor, and Roy Schwartz. Improved competitive ratios for submodular secretary problems (extended abstract). In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques – 14th International Workshop, APPROX 2011, and 15th International Workshop, RANDOM 2011, Princeton, NJ, USA,*

August 17-19, 2011. Proceedings, pages 218–229, 2011. doi:10.1007/978-3-642-22935-0_19.

12 Moran Feldman, Joseph Naor, Roy Schwartz, and Justin Ward. Improved approximations for k-exchange systems – (extended abstract). In *Proc. 19th European Symp. Algorithms (ESA)*, pages 784–798, 2011. doi:10.1007/978-3-642-23719-5_66.

13 Moran Feldman, Ola Svensson, and Rico Zenklusen. A simple $O(\log\log(\text{rank}))$-competitive algorithm for the matroid secretary problem. In *Proc. 26th Symp. Discr. Algorithms (SODA)*, pages 1189–1201, 2015. doi:10.1137/1.9781611973730.79.

14 Moran Feldman and Rico Zenklusen. The submodular secretary problem goes linear. In *Proc. 56th Symp. Foundations of Computer Science (FOCS)*, pages 486–505, 2015. doi:10.1109/FOCS.2015.37.

15 Anupam Gupta, Aaron Roth, Grant Schoenebeck, and Kunal Talwar. Constrained non-monotone submodular maximization: Offline and secretary algorithms. In *Proc. 6th Int'l Conf. Web and Internet Economics (WINE)*, pages 246–257, 2010. doi:10.1007/978-3-642-17572-5_20.

16 Michael Kapralov, Ian Post, and Jan Vondrák. Online submodular welfare maximization: Greedy is optimal. In *Proc. 24th Symp. Discr. Algorithms (SODA)*, pages 1216–1225, 2013. doi:10.1137/1.9781611973105.88.

17 Thomas Kesselheim, Klaus Radke, Andreas Tönnis, and Berthold Vöcking. An optimal online algorithm for weighted bipartite matching and extensions to combinatorial auctions. In *Proc. 21st European Symp. Algorithms (ESA)*, pages 589–600, 2013. doi:10.1007/978-3-642-40450-4_50.

18 Thomas Kesselheim, Klaus Radke, Andreas Tönnis, and Berthold Vöcking. Primal beats dual on online packing lps in the random-order model. In *Proc. 46th Symp. Theory of Computing (STOC)*, pages 303–312, 2014. doi:10.1145/2591796.2591810.

19 Robert D. Kleinberg. A multiple-choice secretary algorithm with applications to online auctions. In *Proc. 16th Symp. Discr. Algorithms (SODA)*, pages 630–631, 2005. URL: http://dl.acm.org/citation.cfm?id=1070432.1070519.

20 Nitish Korula, Vahab S. Mirrokni, and Morteza Zadimoghaddam. Online submodular welfare maximization: Greedy beats 1/2 in random order. In *Proc. 47th Symp. Theory of Computing (STOC)*, pages 889–898, 2015. doi:10.1145/2746539.2746626.

21 Nitish Korula and Martin Pál. Algorithms for secretary problems on graphs and hypergraphs. In *Proc. 36th Int'l Coll. Autom. Lang. Program. (ICALP)*, pages 508–520, 2009. doi:10.1007/978-3-642-02930-1_42.

22 Andreas Krause and Daniel Gloving. Submodular function maximization. In *Tractability: Practical Approaches to Hard Problems*, chapter 3. Cambridge University Press, 2014.

23 Ariel Kulik, Hadas Shachnai, and Tami Tamir. Approximations for monotone and nonmonotone submodular maximization with knapsack constraints. *Math. Oper. Res.*, 38(4):729–739, 2013. doi:10.1287/moor.2013.0592.

24 Oded Lachish. O(log log rank) competitive ratio for the matroid secretary problem. In *Proc. 55th Symp. Foundations of Computer Science (FOCS)*, pages 326–335, 2014. doi:10.1109/FOCS.2014.42.

25 Jon Lee, Maxim Sviridenko, and Jan Vondrák. Submodular maximization over multiple matroids via generalized exchange properties. *Math. Oper. Res.*, 35(4):795–806, 2010. doi:10.1287/moor.1100.0463.

26 Tengyu Ma, Bo Tang, and Yajun Wang. The simulated greedy algorithm for several submodular matroid secretary problems. *Theoret. Comput. Sci.*, 58(4):681–706, 2016. doi:10.1007/s00224-015-9642-4.

27 Marco Molinaro and R. Ravi. The geometry of online packing linear programs. *Math. Oper. Res.*, 39(1):46–59, 2014. doi:10.1287/moor.2013.0612.

28 George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. An analysis of approximations for maximizing submodular set functions – II. *Math. Prog.*, 14(1):265–294, 1978. doi:10.1007/BF01588971.
29 G. L. Nemhauser and L. A. Wolsey. Best algorithms for approximating the maximum of a submodular set function. *Math. Oper. Res.*, 3(3):177–188, 1978. doi:10.1287/moor.3.3.177.
30 Maxim Sviridenko. A note on maximizing a submodular set function subject to a knapsack constraint. *Oper. Res. Lett.*, 32(1):41–43, 2004. doi:10.1016/S0167-6377(03)00062-2.

A Missing Details in Section 2

A.1 Continued Proof of Lemma 4

To show the lemma, we perform an induction on r. Note that Equation (2) trivially holds for $r = 0$. In order to prove it holds for a given $r > 0$, we assume that it is fulfilled for $r - 1$ for all $\ell \in [n]$. From this, we will conclude that Equation (2) also holds for r for all $\ell \in [n]$. To show that (3) is solved by (2), we use the induction hypothesis and plug in the bound for $\mathbf{E}\left[v(\mathrm{ALG}_{r-1}^{\geq j+1})\right]$. This gives us

$$\begin{aligned}\frac{\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell})\right]}{\alpha v(\mathrm{OPT})} &\geq \sum_{j=\ell}^{n} \left(\frac{\ell-k}{j-k}\right)^k \frac{k-1}{j+1}\Bigg(\frac{(r-1)(j+1)}{(k-1)n} - \frac{3k^2(r-1)}{(k-1)n} + \frac{1}{n} \\ &\qquad - \frac{1}{k-1}\left(\frac{j+1}{n}\right)^k \sum_{r'=0}^{r-2}\sum_{i=0}^{r'} \frac{(k-1)^i}{i!} \ln^i\left(\frac{n}{j+1}\right)\Bigg) \\ &= \sum_{j=\ell}^{n} \left(\frac{\ell-k}{j-k}\right)^k \frac{r}{n} - \sum_{j=\ell}^{n} \left(\frac{\ell-k}{j-k}\right)^k \frac{3k^2(r-1)}{(j+1)n} \\ &\qquad - \sum_{j=\ell}^{n} \left(\frac{\ell-k}{j-k}\right)^k \frac{1}{j+1}\left(\frac{j+1}{n}\right)^k \sum_{r'=0}^{r-2}\sum_{i=0}^{r'} \frac{(k-1)^i}{i!} \ln^i\left(\frac{n}{j+1}\right) \enspace .\end{aligned}$$

In the negative terms, we bound $\frac{\ell-k}{j-k} \leq \frac{\ell}{j}$ and use $\left(\frac{j+1}{j}\right)^k \leq e^{\frac{k}{j}} \leq e^{\frac{k}{\ell}} \leq 1 + 2\frac{k}{\ell}$. Finally in the last sum, we bound $\frac{1}{j+1} \leq \frac{1}{\ell}$ once

$$\begin{aligned}\frac{\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell})\right]}{\alpha v(\mathrm{OPT})} &\geq \sum_{j=\ell}^{n} \left(\frac{\ell-k}{j-k}\right)^k \frac{r}{n} - \sum_{j=\ell}^{n} \left(\frac{\ell}{j}\right)^k \frac{3k^2(r-1)}{\ell n} \\ &\qquad - \left(\frac{\ell}{n}\right)^k \sum_{j=\ell}^{n} \frac{\left(1+2\frac{k}{\ell}\right)}{j+1} \sum_{r'=0}^{r-2}\sum_{i=0}^{r'} \frac{(k-1)^i}{i!} \ln^i\left(\frac{n}{j+1}\right) \enspace .\end{aligned}$$

We approximate both sums over j through integrals by using

$$\sum_{j=\ell}^{n} \frac{1}{(j-k)^k} \geq \int_{\ell}^{n} \frac{1}{(j-k)^k} dj = \frac{1}{k-1}\left(\frac{1}{(\ell-k)^{k-1}} - \frac{1}{(n-k)^{k-1}}\right)$$

and

$$\sum_{j=\ell}^{n} \frac{\ln^i(n/(j+1))}{j+1} \leq \int_{\ell-1}^{n-1} \frac{\ln^i(n/(j+1))}{j+1} dj = \left[-\frac{\ln^{i+1}(n/(j+1))}{i+1}\right]_{\ell-1}^{n-1} = \frac{\ln^{i+1}(n/\ell)}{i+1} \enspace .$$

This yields

$$\frac{\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell})\right]}{\alpha v(\mathrm{OPT})} \geq \frac{r(\ell-k)}{(k-1)n}\left(1-\left(\frac{\ell-k}{n-k}\right)^{k-1}\right) - \frac{3k^2(r-1)}{(k-1)n}\left(1-\left(\frac{\ell}{n}\right)^{k-1}\right)$$
$$-\left(\frac{\ell}{n}\right)^k\left(1+2\frac{k}{\ell}\right)\sum_{r'=0}^{r-2}\sum_{i=0}^{r'}\frac{(k-1)^i}{i!}\frac{\ln^{i+1}\left(\frac{n}{\ell}\right)}{i+1} .$$

We perform an index shift in the inner sum and propagate the shift to the outer sum

$$\begin{aligned}\sum_{r'=0}^{r-2}\sum_{i=0}^{r'}\frac{(k-1)^i}{i!}\frac{\ln(n/\ell)^{i+1}}{i+1} &= \frac{1}{k-1}\sum_{r'=0}^{r-2}\sum_{i=1}^{r'+1}\frac{(k-1)^i}{i!}\ln^i\left(\frac{n}{\ell}\right)\\ &= \frac{1}{k-1}\sum_{r'=1}^{r-1}\sum_{i=1}^{r'}\frac{(k-1)^i}{i!}\ln^i\left(\frac{n}{\ell}\right)\\ &= \frac{1}{k-1}\sum_{r'=0}^{r-1}\sum_{i=0}^{r'}\frac{(k-1)^i}{i!}\ln^i\left(\frac{n}{\ell}\right) - \frac{r}{k-1} .\end{aligned}$$

Now we solve the brackets and use the term split off in the index shift to simplify the expression. We get

$$\begin{aligned}\frac{\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell})\right]}{\alpha v(\mathrm{OPT})} &\geq \frac{r(\ell-k)}{(k-1)n} - \frac{r(\ell-k)}{(k-1)n}\left(\frac{\ell-k}{n-k}\right)^{k-1} + \left(\frac{\ell}{n}\right)^k\frac{\left(1+2\frac{k}{\ell}\right)}{k-1}r\\ &\quad - \left(\frac{\ell}{n}\right)^k\frac{\left(1+2\frac{k}{\ell}\right)}{k-1}\sum_{r'=0}^{r-1}\sum_{i=0}^{r'}\frac{(k-1)^i}{i!}\ln^i\left(\frac{n}{\ell}\right) - \frac{3k^2(r-1)}{(k-1)n}\\ &\geq \frac{r\ell}{(k-1)n} - \frac{rk}{(k-1)n} - \left(\frac{\ell}{n}\right)^k\frac{\left(1+2\frac{k}{\ell}\right)}{k-1}\sum_{r'=0}^{r-1}\sum_{i=0}^{r'}\frac{(k-1)^i}{i!}\ln^i\left(\frac{n}{\ell}\right)\\ &\quad - \frac{3k^2(r-1)}{(k-1)n} .\end{aligned}$$

At this point, we only have to show that the following inequality holds

$$\frac{rk}{(k-1)n} + \left(\frac{\ell}{n}\right)^k\frac{2\frac{k}{\ell}}{k-1}\sum_{r'=0}^{r-1}\sum_{i=0}^{r'}\frac{(k-1)^i}{i!}\ln^i\left(\frac{n}{\ell}\right) + \frac{3k^2(r-1)}{(k-1)n} \leq \frac{3k^2r}{(k-1)n} .$$

We bound the inner sum with the corresponding exponential function

$$\sum_{i=0}^{r'}\frac{(k-1)^i}{i!}\ln^i\left(\frac{n}{\ell}\right) \leq \sum_{i=0}^{\infty}\frac{(k-1)^i}{i!}\ln^i\left(\frac{n}{\ell}\right) = \exp\left((k-1)\ln\left(\frac{n}{\ell}\right)\right) = \left(\frac{n}{\ell}\right)^{k-1} .$$

This term is independent of r'. We eliminate the sum over r' and get

$$\frac{rk}{(k-1)n} + \frac{\ell}{n}\frac{r2\frac{k}{\ell}}{k-1} = \frac{3kr}{(k-1)n} \leq \frac{3k^2}{(k-1)n} .$$

A.2 Detailed Proof of Theorem 1

To complete the proof of the theorem, we apply Lemma 4 for $\ell = pn$ and $r = k$. This gives us $\mathbf{E}[v(\mathrm{ALG})] = \mathbf{E}\left[v(\mathrm{ALG}_k^{\geq pn})\right]$ and thus

$$\mathbf{E}[v(\mathrm{ALG})] \geq \left(\frac{pk}{k-1} - \frac{1}{k-1}p^k \sum_{r'=0}^{k-1}\sum_{i=0}^{r'} \frac{(k-1)^i}{i!} \ln^i\left(\frac{1}{p}\right) - \frac{6k^2}{n}\right) \cdot \alpha v(\mathrm{OPT}) \ .$$

For p such that $pn = \lceil \frac{n}{e} \rceil$, we have $\frac{1}{e} \leq p \leq \frac{1}{e} + \frac{1}{n}$ and $\ln\left(\frac{1}{p}\right) = 1 + \ln\left(\frac{n}{n+e}\right) \leq 1$. This allows us to reorder the occurring double sum as follows

$$\begin{aligned}\sum_{r'=0}^{k-1}\sum_{i=0}^{r'} \frac{(k-1)^i}{i!} &= \sum_{i=0}^{k-1}(k-i)\frac{(k-1)^i}{i!} = k\sum_{i=0}^{k-1}\frac{(k-1)^i}{i!} - (k-1)\sum_{i=1}^{k-1}\frac{(k-1)^{i-1}}{(i-1)!} \\ &= \sum_{i=0}^{k-1}\frac{(k-1)^i}{i!} + \frac{(k-1)^k}{(k-1)!} \ .\end{aligned}$$

By definition of the exponential function $e^x = \sum_{i=0}^{\infty} \frac{x^i}{i!}$. For $x > 0$, all terms of the infinite sum are positive. This yields $e^x \geq \sum_{i=0}^{k-1} \frac{x^i}{i!} + \frac{x^k}{k!} + \frac{x^{k+1}}{(k+1)!}$ and thus by setting $x = k-1$ we get

$$\sum_{r'=0}^{k-1}\sum_{i=0}^{r'} \frac{(k-1)^i}{i!} \leq e^{k-1} - \frac{(k-1)^k}{k!} - \frac{(k-1)^{k+1}}{(k+1)!} + \frac{(k-1)^k}{(k-1)!} \ .$$

We have $p^k e^{k-1} \leq \left(\frac{1}{e} + \frac{1}{n}\right)^k e^{k-1} = \left(1 + \frac{e}{n}\right)^{k-1}\left(\frac{1}{e} + \frac{1}{n}\right) \leq e^{\frac{ek}{n}}\left(\frac{1}{e} + \frac{1}{n}\right)$, this implies

$$\begin{aligned}\frac{\mathbf{E}[v(\mathrm{ALG})]}{\alpha v(\mathrm{OPT})} &\geq \frac{k}{e(k-1)} - \frac{\left(\frac{1}{e}+\frac{1}{n}\right)^k}{(k-1)}\left(e^{k-1} - \frac{(k-1)^k}{k!} - \frac{(k-1)^{k+1}}{(k+1)!} + \frac{(k-1)^k}{(k-1)!}\right) - \frac{6k^2}{n} \\ &= \frac{k}{e(k-1)} - \frac{e^{\frac{ek}{n}}}{e(k-1)} - \frac{e^{\frac{ek}{n}}}{n(k-1)} \\ &\quad + \left(\frac{1}{e} + \frac{1}{n}\right)^k \left(\frac{(k-1)^{k-1}}{k!} + \frac{(k-1)^k}{(k+1)!} - \frac{(k-1)^{k-1}}{(k-1)!}\right) - \frac{6k^2}{n} \\ &= \frac{k - e^{\frac{ek}{n}}}{e(k-1)} - \left(\frac{1}{e} + \frac{1}{n}\right)^k \frac{k-1}{k+1}\frac{(k-1)^{k-1}}{(k-1)!} - \frac{6k^2}{n} \ .\end{aligned}$$

At this point, we apply the Stirling approximation $(k-1)! \geq \sqrt{2\pi(k-1)}\left(\frac{k-1}{e}\right)^{k-1}$ and get

$$\begin{aligned}\frac{\mathbf{E}[v(\mathrm{ALG})]}{\alpha v(\mathrm{OPT})} &\geq \frac{1}{e} - \frac{e^{\frac{ek}{n}} - 1}{e(k-1)} - \left(\frac{1}{e} + \frac{1}{n}\right)^k e^{k-1} \frac{\sqrt{k-1}}{(k+1)\sqrt{2\pi}} - \frac{6k^2}{n} \\ &= \frac{1}{e} - \frac{e^{\frac{ek}{n}} - 1}{e(k-1)} - e^{\frac{ek}{n}}\left(\frac{1}{e} + \frac{1}{n}\right) \frac{\sqrt{k-1}}{(k+1)\sqrt{2\pi}} - \frac{6k^2}{n} \ .\end{aligned}$$

For every fixed k, we can assume that n is arbitrarily larger. This can be guaranteed, for example, through dummy elements with marginal gain zero for all sets. In the limit, this yields

$$\frac{\mathbf{E}[v(\mathrm{ALG})]}{\alpha v(\mathrm{OPT})} \geq \frac{1}{e}\left(1 - \frac{\sqrt{k-1}}{(k+1)\sqrt{2\pi}}\right) \ .$$

A.3 Proof of Claim 7

Proof. We perform an induction on ℓ. Assume that the claim has been shown for all r for $\ell+1$. In Lemma 3, we have shown

$$\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell})\right] \geq \frac{1}{\ell}\left(\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right] + (k-1)\mathbf{E}\left[v(\mathrm{ALG}_{r-1}^{\geq \ell+1})\right] + (\ell-k)\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell+1})\right]\right).$$

Now we use the induction hypothesis

$$\begin{aligned}\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell})\right] \geq{} & \frac{1}{\ell}\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right] \\ & + \frac{k-1}{\ell}\sum_{j=\ell+1}^{n}\frac{a_{\ell+1,j-1}}{j}\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right]\sum_{r'=0}^{r-2}\sum_{\substack{M\subseteq\{\ell+1,\ldots,j-1\}\\|M|=r'}}\left(\prod_{i\in M}\frac{k-1}{i}\right) \\ & + \frac{\ell-k}{\ell}\sum_{j=\ell+1}^{n}\frac{a_{\ell+1,j-1}}{j}\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right]\sum_{r'=0}^{r-1}\sum_{\substack{M\subseteq\{\ell+1,\ldots,j-1\}\\|M|=r'}}\left(\prod_{i\in M}\frac{k-1}{i}\right) .\end{aligned}$$

We perform an index shift, use $\frac{\ell-k}{\ell}a_{\ell+1,j-1} = a_{\ell,j-1}$ and get

$$\begin{aligned}\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell})\right] ={} & \frac{a_{\ell,\ell-1}}{\ell}\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right] \\ & + \sum_{j=\ell+1}^{n}\frac{a_{\ell+1,j-1}}{j}\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right]\sum_{r'=1}^{r-1}\frac{k-1}{\ell}\sum_{\substack{M\subseteq\{\ell+1,\ldots,j-1\}\\|M|=r'-1}}\left(\prod_{i\in M}\frac{k-1}{i}\right) \\ & + \sum_{j=\ell+1}^{n}\frac{a_{\ell,j-1}}{j}\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right]\sum_{r'=0}^{r-1}\sum_{\substack{M\subseteq\{\ell+1,\ldots,j-1\}\\|M|=r'}}\left(\prod_{i\in M}\frac{k-1}{i}\right) .\end{aligned}$$

We have $\frac{k-1}{\ell} \geq \frac{k-1}{i}$ for all $i \geq \ell$ and therefore we can merge the factor for the current round into the product. In a sense the $\frac{k-1}{\ell}$ factor stands for choosing an item in the current round, and it gets worse if we chose one in a future round instead. Additionally we use $a_{\ell+1,j-1} \geq a_{\ell,j-1}$ and omit the second large sum entirely.

For the final equality we use the fact that $\sum_{r'=0}^{r-1}\sum_{\substack{M\subseteq\emptyset\\|M|=r'}}\left(\prod_{i\in M}\frac{k-1}{i}\right) = 1$ because the inner sum is empty for all $r' > 0$

$$\begin{aligned}\mathbf{E}\left[v(\mathrm{ALG}_r^{\geq \ell})\right] \geq{} & \frac{a_{\ell,\ell-1}}{\ell}\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right] \\ & + \sum_{j=\ell+1}^{n}\frac{a_{\ell,j-1}}{j}\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right]\sum_{r'=0}^{r-1}\sum_{\substack{M\subseteq\{\ell,\ldots,j-1\}\\|M|=r'}}\left(\prod_{i\in M}\frac{k-1}{i}\right) \\ ={} & \sum_{j=\ell}^{n}\frac{a_{\ell,j-1}}{j}\mathbf{E}\left[v(\mathcal{A}(U^{\leq \ell}))\right]\sum_{r'=0}^{r-1}\sum_{\substack{M\subseteq\{\ell,\ldots,j-1\}\\|M|=r'}}\left(\prod_{i\in M}\frac{k-1}{i}\right) .\end{aligned}$$ ◀

A.4 Proof of Claim 8

Proof. Towards a proof, we show that $t_{\ell,j+1} \leq \beta_j t_{\ell,j}$ for some $\beta_j \leq 1$. We consider the definition of $t_{\ell,j+1}$ and split of a double sum that contains all terms where $j \in M$. In those terms, we know that j is selected and therefore the factor $\frac{k-1}{j}$ should always exist in the product. We get

$$t_{\ell,j+1} = a_{\ell,j} \sum_{r'=0}^{r-1} \sum_{\substack{M \subseteq \{\ell,\dots,j\} \\ |M|=r'}} \left(\prod_{i \in M} \frac{k-1}{i} \right)$$

$$= a_{\ell,j} \left(\sum_{r'=0}^{r-1} \sum_{\substack{M \subseteq \{\ell,\dots,j-1\} \\ |M|=r'}} \left(\prod_{i \in M} \frac{k-1}{i} \right) + \frac{k-1}{j} \sum_{r'=0}^{r-1} \sum_{\substack{M \subseteq \{\ell,\dots,j-1\} \\ |M|=r'-1}} \left(\prod_{i \in M} \frac{k-1}{i} \right) \right) .$$

Both double sums are nearly identical. We fill up the missing terms in the smaller one and bound by the following expression. Finally, we replace the remaining double sum with the definition of $t_{\ell,j}$

$$t_{\ell,j+1} \leq a_{\ell,j} \left(1 + \frac{k-1}{j} \right) \sum_{r'=0}^{r-1} \sum_{\substack{M \subseteq \{\ell,\dots,j-1\} \\ |M|=r'}} \left(\prod_{i \in M} \frac{k-1}{i} \right) = \frac{a_{\ell,j}}{a_{\ell,j-1}} \left(1 + \frac{k-1}{j} \right) t_{\ell,j} .$$

As we have $\frac{a_{\ell,j}}{a_{\ell,j-1}} \left(1 + \frac{k-1}{j}\right) = \left(1 + \frac{k-1}{j}\right)\left(1 - \frac{k}{j}\right) = 1 - \frac{k}{j} + \frac{k-1}{j} - \frac{k(k-1)}{j^2} \leq 1$ the claim follows. ◀

B Missing Details in Section 3: Submodular Matching

B.1 Missing Details in the Proof of Theorem 10: Competitive Ratio for Submodular Matching

In the proof of Theorem 10, we also required the following technical lemma that is not problem-specific.

▶ **Lemma 14.** *For $i > c \geq 1$, we have*

$$\prod_{i=j}^{k} \left(1 - \frac{c}{i}\right) \geq \left(\frac{j-c}{k-c+1} \right)^c .$$

Proof. As first step, we show that

$$1 - \frac{c}{i} = \frac{i-c}{i} \geq \left(\frac{i-c}{i-c+1} \right)^c = \left(1 - \frac{1}{i-c+1} \right)^c .$$

This is equivalent to

$$\frac{i-c}{(i-c)^c} \geq \frac{i}{(i-c+1)^c} .$$

Now we show that this inequality holds for all $i > c \geq 1$. We define the function $f : [0,1] \to \mathbb{R}$ such that

$$f(x) = \frac{i-cx}{(i-c+1)^c} .$$

This function has the properties that $f(0) = \frac{i}{(i-c+1)^c}$ and $f(1) = \frac{i-c}{(i-c)^c}$. We show that f is non-decreasing increasing and therefore the inequality holds as well. The derivative f' of f is

$$f'(x) = \frac{-c(i-c+1-x)^c - (i-cx)c(i-c+1-x)^{(c-1)}(-1)}{(i-c+1-x)^{2c}} .$$

It suffice to show that f' is non-negative for all $x \in [0,1]$. This holds true if the numerator is positive for all $x \in [0,1]$ because the denominator is guaranteed to be positive with $i > c$ and $x \in [0,1]$. We have

$$\begin{aligned} -c(i-c+1-x) - (i-cx)c(-1) &\geq 0 \\ c(i-cx) &\geq c(i-c+1-x) \\ c-1 &\geq (c-1)x \ . \end{aligned}$$

This directly gives us the proof for the lemma

$$\prod_{i=j}^{k} \left(1 - \frac{c}{i}\right) \geq \prod_{i=j}^{k} \left(1 - \frac{1}{i-c+1}\right)^{c} = \left(\prod_{i=j}^{k} \frac{i-c}{i-c+1}\right)^{c} = \left(\frac{j-c}{k-c+1}\right)^{c} \ . \qquad ◀$$