SAPIENZA
UNIVERSITÀ DI ROMA

Approximation algorithms for stochastic optimization

Dipartimento di Ingegneria Informatica Automatica e Gestionale "Antonio Ruberti"

Dottorato di Ricerca in Informatica – XXVII Ciclo

Candidate

Marek Adamczyk
ID number 1527406

Thesis Advisor

Prof. Stefano Leonardi

A thesis submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Computer Science

September 2015

Thesis defended on
in front of a Board of Examiners composed by:

Prof. Nome Cognome (chairman)
Prof. Nome Cognome

Approximation algorithms for stochastic optimization
Ph.D. thesis. Sapienza – University of Rome
ISBN: 000000000-0

This thesis has been typeset by LaTeX and the Sapthesis class.

Version: September 30, 2015

Author's email: adamczyk@dis.uniroma1.it

Abstract

In recent years a lot of attention was brought to approximation algorithms for stochastic optimization. In such settings the algorithms are executed on an object which is random, for example, in a matching problem the graph can be random or in the case of a knapsack problem the size of an item is random. This area, although recent, is very vibrant due to its practical impact, mathematical appeal, and new challenges. In this thesis we contribute to this line of work by presenting new algorithms for the problems in this setup.

First we consider the so-called stochastic adaptive problems on matroid environments. Here, we obtain improved bounds for the stochastic matching problem. Later we consider general stochastic probing problem on matroids, and present improved bounds here as well. We also show how to adapt this algorithm to yield good approximations in a setting where we want to maximize a submodular function.

A very important feature of problems for stochastic optimization that is apparent in stochastic adaptive problems and also two-stage stochastic optimization is the fact that we compare our solutions with the optimum solution that follows the same rules of the model as we do. Such a thing is impossible in the most popular model of optimization under uncertainty, i.e., the online model, where the solution constructed in an online manner is compared with the optimum solution that knows the whole input in advance. Inspired by this possibility we initiate the study of stochastic universal optimization under such a benchmark model.

In the last chapter we investigate the properties of the procedure that we developed for stochastic probing on matroids, although this time in a deterministic setting. We show a randomized rounding procedure for a point $(x_e)_{e \in E} \in [0,1]^E$ such that $\sum_e x_e = r \in \mathbb{Z}_+$, which outputs a solution $(\hat{x}_e)_{e \in E} \in \{0,1\}^E$ that satisfies the hard capacity constraints: $\sum_e \hat{x}_e \leq r$, preserves approximately the marginals, i.e., $\mathbb{P}[\hat{x}_e = 1] \geq x_e/2$, and satisfies negative correlation of the elements. Such a setup was already considered by Srinivasan (FOCS 2001), but without a loss in approximation factor, i.e., $\mathbb{P}[\hat{x}_e = 1] = x_e$. Therefore, our result is weaker, but unlike the result of Srinivasan, our procedure can be extended to k-column sparse 0-1 packing problems, e.g., k-dimensional b-matching, guaranteeing that $\mathbb{P}[\hat{x}_e = 1] \geq \frac{x_e}{k+1}$. As of the writing, we are able to give a proof of negative correlation only in the single matroid case, but we conjecture that it is possible to extend the proof to arbitrary k, and leave it as a future work.

Acknowledgments

Contents

Chapter 1

Introduction

Uncertainty in input data is a common feature of most practical problems, and research in finding good solutions (both experimental and theoretical) for such problems has a long history dating back to 1950 [10, 21]. In recent years a lot of attention was given to study stochastic optimization problems from the point of view of approximation algorithms. The richest in results line of work are two-stage stochastic problems which started with the seminal paper of Immorlica et al. [38], with later coverage of almost all fundamental combinatorial problems considered in this setup like set cover [48], Steiner tree [37], facility location [46], etc.. Second most covered line of work are stochastic adaptive problems, which started with the paper of Dean et al. [23] on stochastic knapsack. In the most basic setup of the problem we are given a knapsack of capacity 1, and n items with deterministic rewards $r_1, ..., r_n$, but random sizes $S_1, ..., S_n$. We get to know the size of an item only when we decide to place it in the knapsack. The goal is to develop a(n adaptive) strategy, which one-by-one would decide which item to place in the knapsack next, and which would maximize the expected total value of items that fit in the knapsack. Dean et al. were first to explicitly address the challenges of this setting, however, already Mohring et al. [44] considered a stochastic scheduling problem that belongs to this family of problems — problems for which a solution is an adaptive process. Later on, improved algorithms were presented for the stochastic knapsack [11, 12], and the problem was considered in many variants like with preemptions [35, 42], or with random rewards r_i that would be correlated with sizes S_i [35, 43]. Stochastic knapsack also inspired algorithms for multiarmed bandit problems with martingale [33] and without the martingale assumption [35, 43]. To the class of stochastic adaptive problems belongs also the setup of stochastic probing model in which the information that we get to know is the existence of an item, which is a $0-1$ variable — unlike in the stochastic knapsack where we getting to know the sizes S_i. This thesis is devoted mostly to the class of stochastic probing problems, and we elaborate on the model below. Another set of problems that can be considered in a stochastic setup are stochastic universal problems. Here we consider covering problems, and the solution is an assignment where every element chooses a set that will cover it if the element will happen to exist; we elaborate on the model below. To the best of our knowledge, so far there were three papers sharing similar features and that make them belong to such a setup [40, 31, 28], all on different problems: Steiner tree, set cover, and scheduling. So far this line of work did not emerge in a separate class of problems with a bit more general techniques that would capture more than one problem.

Below we describe the problems results for which are presented in this thesis.

1.1 Problems

1.1.1 Stochastic probing

The above problem of stochastic matching belongs to a larger class of adaptive stochastic optimization problems in the framework of Dean et al. [23]. Here the solution is in fact a process, and the optimal one might even require larger than polynomial space to describe. Since the work of Dean et al. a number of such problems were introduced [18, 30, 33, 34, 6, 35, 22]. Gupta and Nagarajan [36] present an abstract framework for a subclass of adaptive stochastic problems giving a unified view for Stochastic Matching [18] and Sequential Posted Pricing [16].

We describe the framework following [36]. We are given a universe E, where each element $e \in E$ is *active* with probability $p_e \in [0, 1]$ independently. The only way to find out if an element is active, is to *probe* it. We call a probe *successful* if an element turns out to be active. On universe E we execute an algorithm that probes the elements one-by-one. If an element is active, the algorithm must add it to the current solution. In this way, the algorithm gradually constructs a solution consisting of active elements.

Here, we consider the case in which we are given constraints on both the elements probed and the elements included in the solution. Formally, suppose that we are given two independence systems of downward-closed sets: an *outer* independence system $(E, \mathcal{I}^{out})$ restricting the set of elements probed by the algorithm, and an *inner* independence system $(E, \mathcal{I}^{in})$, restricting the set of elements taken by the algorithm. We denote by Q^t the set of elements probed in the first t steps of the algorithm, and by S^t the subset of active elements from Q^t. Then, S^t is the partial solution constructed by the first t steps of the algorithm. We require that, at each time t, $Q^t \in \mathcal{I}^{out}$ and $S^t \in \mathcal{I}^{in}$. Thus, at each time t, the element e that we probe must satisfy both $Q^{t-1} \cup \{e\} \in \mathcal{I}^{out}$ and $S^{t-1} \cup \{e\} \in \mathcal{I}^{in}$. Gupta and Nagarajan [36] considered many types of systems $\mathcal{I}^{in}$ and $\mathcal{I}^{out}$, but we focus only on matroid intersections, i.e. on the special case in which $\mathcal{I}^{in}$ is an intersection of k^{in} matroids $\mathcal{M}^{in}_1, \ldots, \mathcal{M}^{in}_{k^{in}}$, and $\mathcal{I}^{out}$ is an intersection of k^{out} matroids $\mathcal{M}^{out}_1, \ldots, \mathcal{M}^{out}_{k^{out}}$. We always assume that $k^{out} \geq 1$ and $k^{in} \geq 0$. We assume familiarity with matroid algorithmics (see [47], for example) and, above all, with principles of approximation algorithms (see [50], for example).

Considering submodular objective functions is a common practice in combinatorial optimization as it extends the range of applicability of many methods. So far, the framework of stochastic probing has been used to maximize the expected weight of the solution found by the process. We were given weights $w_e \geq 0$ for $e \in E$ and, if S denotes the solution at the end of a process, the goal was to maximize $\mathbb{E}_S\left[\sum_{e \in S} w_e\right]$. We generalize the framework as we consider a monotone as well as non-monotone submodular function $f : 2^E \mapsto \mathbb{R}_{\geq 0}$, and objective of maximizing $\mathbb{E}_S\left[f(S)\right]$.

Applications: Bayesian mechanism design [36] Consider the following mechanism design problem. There are n agents and a single seller providing a certain service. Agent's i value for receiving service is v_i, drawn independently from a distribution D_i over set $\{0, 1, \ldots, B\}$. The valuation v_i is private, but the distribution D_i is known. The seller can provide service only for a subset of agents that belongs to system $\mathcal{I} \in 2^{[n]}$, which specifies feasibility constraints. A mechanism accepts bids of agents, decides on subset of agents to serve, and sets individual prices for the service. A mechanism is called truthful if agents bid their true valuations. Myerson's theory of virtual valuations yields *truthful* mechanisms that maximize the expected revenue of a seller, although they sometimes might be impractical. On the other hand, practical mechanisms are often non-truthful. The Sequential Posted Pricing Mechanism (SPM) introduced by Chawla et al. [16] gives a nice trade-off — it is truthful, simple to implement, and gives near-optimal revenue. An SPM offers each agent a "take-it-or-leave-it" price for the service. Since after a refusal a service won't be provided, it is easy to see that an SPM is a truthful mechanism.

To see an SPM as a stochastic probing problem, we consider a universe $E = [n] \times \{0, 1, \ldots, B\}$, where element (i, c) represents an offer of price c to agent i. The probability that i accepts the offer is $\mathbb{P}[v_i \geq c]$, and seller earns c then. Obviously, we can make only one offer to an agent, so outer constraints are given by a partition matroid; making at most one probe per agent also overcomes the problem that probes of $(i, 1), ..., (i, B)$ are not independent. The inner constraints on universe $[n] \times \{0, 1, \ldots, B\}$ are simply induced by constraints $\mathcal{I}$ on $[n]$.

Gupta and Nagarajan [36] give an LP relaxation for any single-seller Bayesian mechanism design problem. Provided that we can optimize over $\mathcal{P}(\mathcal{I})$, the LP can be used to construct an efficient SPM. Moreover, the approximation guarantee of the constructed SPM is with respect to the optimal mechanism, which needs not be an SPM.

1.1.1.1 Stochastic matching

This problem is motivated by applications in kidney exchange and online dating. Here we are given an undirected graph $G = (V, E)$. Each edge $e \in E$ is labeled with an (existence) probability $p_e \in (0, 1]$ and a weight (or profit) $w_e > 0$, and each node $v \in V$ with a *timeout* (or *patience*) $t_v \in \mathbb{N}^+$. An algorithm for this problem probes edges in a possibly adaptive order. Each time an edge is probed, it turns out to be *present* with probability p_e, in which case it is (irrevocably) included in the matching under construction and provides a profit w_e. We can probe at most t_u edges among the set $\delta(u)$ of edges incident to node u (independently from whether those edges turn out to be present or absent). Furthermore, when an edge e is added to the matching, no edge $f \in \delta(e)$ (i.e., incident on e) can be probed in subsequent steps. Our goal is to maximize the expected weight of the constructed matching.

One can also consider the *Online Stochastic Matching with Timeouts* problem introduced in [8]. Here we are given in input a bipartite graph $G = (A \cup B, A \times B)$, where nodes in B are *buyer types* and nodes in A are *items* that we wish to sell. Like in the offline case, edges are labeled with probabilities and profits, and nodes are assigned timeouts. However, in this case timeouts on the item side are assumed to be unbounded. Then a second bipartite graph is constructed in an online fashion.

Initially this graph consists of A only. At each time step one random buyer $\tilde{b}$ of some type b is sampled (possibly with repetitions) from a given probability distribution. The edges between $\tilde{b}$ and A are copies of the corresponding edges in G. The online algorithm has to choose at most t_b unmatched neighbors of $\tilde{b}$, and probe those edges in some order until some edge $a\tilde{b}$ turns out to be present (in which case $a\tilde{b}$ is added to the matching and we gain the corresponding profit) or all the mentioned edges are probed. This process is repeated n times, and our goal is to maximize the final total expected profit[1].

Applications: On-line dating and kidney exchange [18] Consider an online dating service. For each pair of users, machine learning algorithms estimate the probability that they will form a happy couple. However, only after a pair meets do we know for sure if they were successfully matched (and together leave the dating service). Users have individual patience numbers that bound how many unsuccessful dates they are willing to go on until they will leave the dating service forever. The objective of the service is to maximize the number of successfully matched couples.

To model this as a stochastic probing problem, users are represented as vertices V of a graph $G = (V, E)$, where edges represent couples of users. Set E of edges is our universe on which we make probes, with p_e being the probability that a couple $e = (u_1, u_2)$ forms a happy couple after a date. The inner constraints are matching constraints — a user can be in at most one couple —, and outer constraints are b–matching — we can probe at most $t(u)$ edges adjacent to user u, where $t(u)$ denotes the patience of u. Both inner and outer constraints are intersections of two matroids for bipartite graphs.

In similar way we can model kidney exchanges. A patient awaiting a kidney transplant can receive the organ from a living friend or a family member. Unfortunately, even if someone is willing to donate a kidney to a patient, it may happen that the donor is incompatible. However, it is possible to find two such incompatible patient/donor pairs where each donor is compatible with the patient from other pair. Four operations are then performed simultaneously resulting in two kidney transplants. In year 2000 the United Network for Organ Sharing (UNOS) launched a program of such kidney exchanges.

In order to maximize the number of transplanted kidneys we need to find the maximum matching in a graph, nodes of which represent incompatible patient/donor pairs. However, this graph is not given entirely upfront. For two incompatible pairs we need to run three tests to find out if we can perform a kidney exchange between them. First two tests are "easy", and estimate the probability that the third "hard" test will be successful — three passed tests allow to perform a kidney exchange. Medical characteristics of the program, imply that the third test cannot be performed for every two patient/donor pairs. Also, the transplants have to be performed immediately after we have matched two pairs. Thus we can see that the graph of patient/donor pairs is in fact random. Moreover, we also need to model the fact that a patient has limited time for awaiting a kidney.

[1] As in [8], we assume that the probability of a buyer type b is an integer multiple of $1/n$.

1.1.1.2 Stochastic k-set packing

Bansal et al. [8] presented a more general problem of stochastic k-set packing which includes the problem of stochastic matching. We are given n elements/columns, where each item $e \in E = [n]$ has a profit $v_e \in \mathbb{R}_+$, and a random d-dimensional size $S_e \in \{0,1\}^d$. The sizes are independent for different items. Additionally, for each item e, there is a set C_e of at most k coordinates such that each size vector S_e takes positive values only in these coordinates, i.e., $S_e \subseteq C_e$ with probability 1. We are also given a capacity vector $b \in \mathbb{Z}_+^d$ into which items must be packed. We assume that v_e is a random variable that can be correlated with S_e. The coordinates of S_e also might be correlated between each other. The goal is to design a strategy that will be one-by-one packing elements e, and that will maximize the expected outcome obtained from fully packed elements.

One can view stochastic matching as an instance of stochastic 4-set packing. We set $d = 2|V|$, and associate the vth and $(|V| + v)$ th coordinate with the vertex v — the first $|V|$ coordinates capture whether the vertex is free or not, and the second $|V|$ coordinates capture how many probes have been made involving that vertex. Now each edge (u, v) is an item which has the following distribution: with probability p_{uv} the value is w_{uv} and size is $e_u + e_v + e_{|V|+u} + e_{|V|+v}$, and with remaining probability $1 - p_{uv}$ the value is 0 and size is $e_{|V|+u} + e_{|V|+v}$; here $e_i \in \{0,1\}^{2|V|}$ is a vector with zeros and single 1 on ith position. Note that for each item, its size and value are correlated. If we set the capacity vector to be $b = (1, 1, ..., 1, t_1, t_2, ..., t_{|V|})$, this precisely captures the stochastic matching problem. In this special case each size vector has at most $k = 4$ ones.

Important thing to notice is that in this setting, unlike in the previous ones, here when we probe an element, there is no success/failure outcome. The size S_e of an element e materializes, and the reward v_e is just drawn (and is possibly correlated with S_e).

As for the way we represent the probability distribution of S_e and v_e, we assume we can compute in polynomial time the following: 1) $p_e^j = \mathbb{E}[S_e(j)]$, i.e., the probability that the jth coordinate of e will appear, 2) $\mathbb{P}[S_e \neq 0]$, i.e., the probability that it will be a non-empty edge, 3) $\mathbb{E}[v_e]$.

1.1.2 Stochastic universal optimization

It is very natural scenario that often appears in practice that the solution we construct has to be mantained while the underlying instance of a problem is being modified. Such problems belong to the rich field of robust optimization. In this chapter we investigate problems in such a setup, but under new benchmark model.

In their seminal work, Jia et al. [39] define a universal variant of the set cover problem. Suppose we are given universe $E = \{1, 2, ..., n\}$ of elements, and family $\mathcal{S} = \{S_1, ..., S_m\}$ where each set $S_j \subseteq E$, and set weights $c : \mathcal{S} \mapsto \mathbb{R}_+$. We need to find a mapping $\phi : E \mapsto \mathcal{S}$ such that $e \in \phi(e)$. After fixing the mapping a set $X \subseteq E$ is realized, and we want the cover $\phi(X)$ of X to be efficient. Jia et al. were considering the problem of finding a mapping which minimizes the worst-case ratio

$$\max_{X \subseteq E} \frac{c(\phi(X))}{c(OPT(X))}$$

between the cost of the set cover given by ϕ (which is computed without knowing X), and the cost of the optimal "offline" solution $OPT(X)$ (which is based on the knowledge of X). Jia et al. gave a tight $\tilde{\Theta}(\sqrt{n})$ bound for this problem. In a follow-up work, Grandoni et al. [31] considered a version of this problem where set X was to be chosen randomly — a distribution $\pi : E \mapsto [0,1]$ was given, and k times an element was drawn proportionally to π. Then the quantity to be minimized is

$$\frac{\mathbb{E}\left[c(\phi(X))\right]}{\mathbb{E}\left[c(OPT(X))\right]}.$$

For this problem Grandoni et al. presented a $O(\lg n \cdot \lg m)$-approximation. The benchmark here still may seem a bit too harsh — we compare our solution with the OPT that knows the actual set to be covered in advance.

However, from the line of work on 2-stage stochastic optimization, and on the stochastic adaptive problems, we know that it is possible in some settings to compare ourselves with the optimum strategy that obeys the same constraints of the model as we do, and to get better and in some sense more fair approximations. Inspired by this fact, we asked the question, if we can improve the $O(\lg n \cdot \lg m)$ bound of Grandoni et al. if one shall seek to minimize

$$\frac{\mathbb{E}\left[c(\phi(X))\right]}{\mathbb{E}\left[c(\phi_{OPT}(X))\right]},$$

where this time we compare ourselves not with the $OPT(X)$, optimum solution on X, but with the solution for X implied by the best possible mapping. In this chapter we answer this question in affirmative, showing that one can obtain $O(\lg n)$-approximation in such a setup, which matches the bound from the deterministic variant of set cover. Moreover, we show this bound for any possible probability distribution of X that is given as a black-box, and not only for the case when it consists of independently drawn elements.

We also show that for metric facility location problem in this setup, we can adapt the classic primal-dual scheme to obtain a 4-approximation, again matching, up to constant factors, the best bound from its deterministic counterpart. This result also works in the arbitrary distribution model.

As a major open problem, we leave the question: is it possible to give a constant factor approximation for a Steiner tree variant in such a setting. Here each terminal would have to choose upfront a path to the root, and after that a subset of terminals would be drawn from a probability distribution. Karger and Minkoff [40] considered the so-called *maybecast* problem, which is the same problem but where each terminal independently belonged to the realized set. They obtained a constant factor approximation for the problem. However, their techniques heavily relied on the fact that terminals appear independently. Therefore their algorithm do not carry over if the distribution of the to-be-covered set is arbitrary, but still can serve as an inspiration.

1.1.3 Randomized rounding with negative correlation

While analyzing random structures and algorithm it is prevalently the case that the random variables we investigate are dependent, and this often makes the analysis complicated if not impossible. However, if one can show that the dependence between the random variables is **negative**, then we can go through with analysis and often draw conclusions that would hold as if the random variables were in fact independent — with the most important manifestation of this phenomenon being Chernoff-like concentration bounds [45]. Paper of Dubhashi and Ranjan [24] presents various notions of negative dependence on the balls and bins models. Here, the underlying source of randomization is produced by a set of independent events — each ball chooses a bin independently at random, but when we ask about the number of balls in two bins, then these random variables are negatively dependent.

Negative dependence was brought to the line of research on LP-based approximation algorithms by Srinivasan [49]. He presented a procedure that takes a point $(x_i)_{i\in[n]} \in [0,1]^n$ such that $\sum_{i=1}^n x_i = r \in \mathbb{Z}_+$, and rounds such an x to an integral solution $\hat{x} \in \{0,1\}^n$ that satisfies the following properties. First, the sum is retained, i.e., $\sum_{i=1}^n \hat{x}_i = r$. Second, for any $i \in [n]$, $\mathbb{P}[\hat{x}_i = 1] = x_i$. The third property states that $\hat{x}_1, \hat{x}_2, .., \hat{x}_n$are negatively correlated:

Property 1. [Negative correlation] 0-1 random variables $X_1, X_2, ..., X_n$ are negatively correlated if for any subset $S \subseteq [n]$ and any $b \in \{0,1\}$ it holds that

$$\mathbb{P}\left[\bigwedge_{i\in S}(X_i = b)\right] \leq \prod_{i\in S}\mathbb{P}[X_i = b]\,. \tag{1.1}$$

Technique of Srinivasan was extended later to work on bipartite graphs in Gandhi et al. [29] to round fractional b-matchings. There the sum $\sum_{e\in\delta(v)} x_e$ was preserved for each $v \in V$ in the rounding, but the negative correlation was holding only for edges incident to a given vertex.

Chekuri, Vondrák and Zenklusen [17] developed a rounding procedure with negative correlation that worked for arbitrary matroids. Their result generalized Srinivasan's as his constraint can be seen as a uniform matroid of rank r. More precisely, they have shown the following: let $\mathcal{M} = (E, \mathcal{I})$ be an arbitrary matroid, and $\mathcal{B}(\mathcal{M}) \subseteq [0,1]^E$ be its base-polytope, i.e., a convex hull of all $\mathcal{M}$'s bases. Their procedure takes a point $x \in \mathcal{B}(\mathcal{M})$ and rounds it into $(\hat{x}_e)_{e\in E} \in \{0,1\}^E$ so that:

1. $\hat{x} \in \mathcal{B}(\mathcal{M})$, i.e., $\hat{x}$ represents a base of $\mathcal{M}$,
2. $\mathbb{P}[\hat{x}_e = 1] = x_e$,
3. $(\hat{x}_e)_{e\in E}$ are negatively correlated.

Moreover, Chekuri et al. have shown that the only structures admitting this property are matroids — if there exists a polytope $\mathcal{P}$ for which one has analogs of Property 1, and Properties 2 and 3, then $\mathcal{P}$ has to be a projection of a matroid polytope. Still, it is natural to ask about a rounding procedure that guarantees negative correlation that would work for more complicated structures than matroids. Naturally, we need to waive Property 2 in this case, and settle for approximation, i.e., $\mathbb{P}[\hat{x}_e = 1] = \Omega(x_e)$. Following this avenue it seems that the ultimate goal would be a rounding procedure for intersection of k matroids that would guarantee $\mathbb{P}[\hat{x}_e = 1] \geq \frac{1}{k}x_e$ on top of negative correlation, and $\hat{x}$ being independent in all k matroids. Procedures of Srinivasan and Chekuri et al. and their analyses do not seem to carry over easily to a non-single matroid settings. Hence one may has to resort to other techniques to obtain them. In the next Section we suggest why the iterative randomized rounding, used extensively in this thesis, may be such a technique.

1.2 Previous work and our results

1.2.1 Stochastic Probing

For the stochastic probing on k^{in} inner and k^{out} outer matroids Gupta and Nagarajan [36] presented an algorithm based on contention resolution schemes [54] that yield $4(k^{in}+k^{out})$-approximation for linear objectives. In this thesis we present an algorithm based on iterative randomized rounding that improves this approximation down to $\left(k^{in}+k^{out}\right)$.

Theorem 1. *For a stochastic probing problem with k^{in}, k^{out} inner and outer matroids and linear objective functions, there exists a $\left(k^{in}+k^{out}\right)$-approximation algorithm.*

Our algorithm allows us however to capture also monotone submodular objectives.

Theorem 2. *For a stochastic probing problem with k^{in}, k^{out} inner and outer matroids and monotone submodular function as an objective, there exists a $\frac{e}{e-1}\left(k^{in}+k^{out}+1\right)$-approximation algorithm.*

The approximation ratio from the above Theorem can be optimized to $\frac{1-\Theta(\frac{1}{\sqrt{k}})}{k+\Theta(\sqrt{k})}$ for $k=k^{in}+k^{out}$. These results can be found in Chapter 4.

The proof of Theorem 2 does not carry over to the case of non-monotone submodular functions. To handle these we need to resort to contention resolution schemes.

Theorem 3. *For a stochastic probing problem with k^{in}, k^{out} inner and outer matroids and non-negative submodular function as an objective, there exists an algorithm with approximation ratio of* $\max_{b\in[0,1]} b\cdot e^{-b}\cdot\left(\frac{1-e^{-b}}{b}\right)^{k^{out}}\cdot(1-b)^{k^{in}}$.

To put this Theorem into perspective. For example when $k^{in}=k^{out}=1$, the above Theorem, after plugging appropriate number b, yield approximation of 0.13.

The above theorem can be improved when we deal with transversal matroids.

Theorem 4. *For a stochastic probing problem with k^{in}, k^{out} inner and outer transversal matroids and non-negative submodular function as an objective, there exists a $e\left(k^{in}+k^{out}+1\right)-$approximation algorithm.*

The approximation factor can be optimized down to $\left(k+\sqrt{k+\frac{1}{4}}+\frac{1}{2}\right)\frac{1}{1-\Theta\left(\frac{1}{\sqrt{k}}\right)}$.

1.2.2 Stochastic Matching

The Stochastic Matching problem was originally presented by Chen et al. [18] together with applications in kidney exchange and online dating. The authors consider the unweighted version of the problem, and prove that a greedy algorithm is a 4-approximation. In Adamczyk [1] it was later proven that the same algorithm is in fact a 2-approximation, and this result is tight. The greedy algorithm does not provide a good approximation in the weighted case, and all known algorithms for this case are LP-based. Here, Bansal et al. [8] showed a 3-approximation for the bipartite case. Via a reduction to the bipartite case, Bansal et al. also obtained a 4-approximation algorithm for general graphs.

Our main result is an approximation algorithm for the bipartite case which improves the 3-approximation of Bansal et al. (see Section 3.1.1).

Theorem 5. *There is an expected* 2.845*-approximation algorithm for Stochastic Matching in bipartite graphs.*

Our algorithm for the bipartite case is similar to the one from [8], which works as follows. After solving a proper LP and rounding the solution via a rounding technique from [29], Bansal et al. probe edges in uniform random order. Then they show that every edge e is probed with probability at least $x_e \cdot g(p_{max})$, where x_e is the fractional value of e, $p_{max} := \max_{f \in \delta(e)}\{p_f\}$ is the largest probability of any edge incident to e (e excluded), and $g(\cdot)$ is a decreasing function with $g(1) = 1/3$.

Our idea is to rather consider edges in a carefully chosen *non-uniform* random order. This way, we are able to show (with a slightly simpler analysis) that each edge e is probed with probability $x_e \cdot g(p_e) \geq \frac{1}{3}x_e$. Observe that we have the same function $g(\cdot)$ as in [8], but depending on p_e rather than p_{max}. In particular, according to our analysis, small probability edges are more likely to be probed than large probability ones (for a given value of x_e), regardless of the probabilities of edges incident to e. Though this approach alone does not directly imply an improved approximation factor, it is not hard to patch it with a simple greedy algorithm that behaves best for large probability edges, and this yields an improved approximation ratio altogether.

We also improve on the 4-approximation for general graphs in [8]. This is achieved by reducing the general case to the bipartite one as in prior work, but we also use a refined LP with blossom inequalities in order to fully exploit our large/small probability patching technique.

Theorem 6. *There is an expected* 3.709*-approximation algorithm for Stochastic Matching in general graphs.*

For the online case Bansal et al. gave a 7.92-approximation. This was later improved to a 5.16-approximation by Li [41]. Similar arguments that allowed us to obtain the two above theorems can also be successfully applied to the online case. By applying our idea of non-uniform permutation of edges we would get a 5.16-approximation (the same as in [41]). However, due to the way edges have to be probed in the online case, we are able to finely control the probability that an edge is probed via *dumping factors*. This allows us to improve the approximation from 5.16 to 4.16. Our idea is similar in spirit to the one used by Ma [43] in his neat 2-approximation algorithm for correlated non-preemptive stochastic knapsack. Further application of the large/small probability trick gives an extra improvement down to 4.07.

Theorem 7. *There is an expected* 4.07*-approximation algorithm for Online Stochastic Matching with Timeouts.*

1.2.3 Stochastic k-set packing

For this problem Bansal et al. [8] gave a $(k+1)$-approximation algorithm, but they had to assume that the column outcomes are monotone, i.e., for any two possible realizations of vector S_e one will be dominated by another. A $(k+1)$-approximation without this assumption can be shown using the technique of iterative randomized rounding from Chapter 4. In Chapter 6 we combine these two results — we analyze the algorithm of Bansal et al. using the iterative rounding framework, and as a result we show that their algorithm is in fact a $(k+1)$-approximation even without the monotonicity assumption.

Theorem 8. *There exists a $(k+1)$-approximation algorithm for general problem of k-set packing.*

If we would assume that the packing vector $b \in \mathbb{Z}_+^d$ is equal to 1^d, i.e., is composed of only 1s, then we can exploit the technique of exponential clocks from Chapter 3. This yields an approximation of $\frac{k}{1-e^{-k}}$, which very fast goes to k. We shall call this setup *stochastic k-hypergraph matching*. Such a variant was studied by Baveja et al. [9] who gave a $\left(k+\frac{1}{2}\right)$-approximation for stochastic k-hypergraph matching, but where they assumed that a hyperedge, i.e., column, either exists or not. In our case a hyperdge of k vertices can realize over these vertices in an arbitrary way.

Theorem 9. *There exists a $\frac{k}{1-e^{-k}}$-approximation for stochastic k-set packing when the packing vector b consists of $1s$ only.*

1.2.4 Stochastic universal optimization

In a follow up paper when it was assumed that the realized set is random Grandoni et al. [31] have shown a solution that guarantees $O(\lg(nm))$-approximate solution versus expected optimum for the sample, and this result is tight. In Chapter 8 we give another algorithm for this problem, but this time we compare ourselves with the optimum universal solution. This allows us to obtain a better bound which matches the approximation ratio from deterministic setup. Also, while Grandoni et al. considered probability distributions given by sampling elements independently, our algorithm works for any probability distribution that is given as a black-box.

Theorem 10. *For the stochastic universal set cover problem there exists a $O(\lg n)$-approximation algorithm when we compare ourselves with the optimum universal solution.*

Given that in this setup we are able to match the bound from deterministic case, we asked the question for which other problems we can get close to the best bound from its deterministic counterpart. The algorithm for set cover can be easily adapted to yield also $O(\lg n)$ bound for non-metric facility location. But can we do better for the metric facility location variant? We answer this question as well, showing an algorithm based on primal dual scheme:

Theorem 11. *For the stochastic universal metric facility location problem there exists a 4-approximation algorithm when we compare ourselves with the optimum universal solution.*

One can ask the question, is it in general possible to match (up to constant factors) the deterministic bounds for covering problems. We conjecture that it should be true as well for the stochastic universal steiner tree problem. Here, every terminal has to choose upfront a path to the root. Once a subset of terminals is realized we need to build a tree from exactly the paths chosen before. By using the tree-metric embeddings of Fakcharoenphol et al. [25] with $O(\lg n)$ distortion, we can easily show a $O(\lg n)$-approximation factor for this problem. As of the writing we cannot prove a constant factor approximation, and this we leave as the future work. There is a strong evidence indicating that it should be true — Karger and Minkoff [40] obtained a constant factor approximation for the so-called maybecast problem. This problem can be viewed as exactly stochastic universal Steiner tree problem, but where terminals are drawn independently rather than from a black-box.

1.2.5 Randomized rounding with negative correlation

In his seminal work Srinivasan has proven the following Theorem.

Theorem 12. *Let $(x_e)_E \in [0,1]^E$ be so that $\sum_e x_e = r$ for integer r. There exists a randomized rounding procedure that outputs $(\hat{x}_e)_{e\in E} \in \{0,1\}^E$ such that:*

1. $\sum_e \hat{x}_e = r$,
2. $\mathbb{P}[\hat{x}_e = 1] = x_e$,
3. *for any $b \in \{0,1\}$ and any subset $S \subseteq E$ it holds that*

$$\mathbb{P}\left[\bigwedge_{e\in S} (\hat{x}_e = b)\right] \leq \prod_{e\in S} \mathbb{P}[\hat{x}_e = b].$$

In Chapter 7 we show that iterative randomized rounding plus transversal mappings from Chapter 5 yield negative correlation in Srinivasan's setting, although preserving marginals only approximately. More precisely we show the following Theorem.

Theorem 13. *Let $(x_e)_E \in [0,1]^E$ be so that $\sum_e x_e = r$ for integer r. There exists a randomized rounding procedure that outputs $(\hat{x}_e)_{e\in E} \in \{0,1\}^E$ such that:*

1. $\sum_e \hat{x}_e \leq r$,
2. $\mathbb{P}[\hat{x}_e = 1] \geq \frac{1}{2}x_e$,
3. for any subset $S \subseteq E$ it holds that

$$\mathbb{P}\left[\bigwedge_{e\in S} (\hat{x}_e = 1)\right] \leq \prod_{e\in S} \mathbb{P}[\hat{x}_e = 1].$$

Our result is weaker than Srinivasan's in two ways: we do not preserve the sum, and negative correlation holds only for events where we take the elements. As for the events $(\hat{x} = 0)$, the technique of Srinivasan allowed for basically the same lines of proof to cover both cases. In our setup, a totally different set of inequalities have to be covered to show that $\forall_{S\subseteq E}\mathbb{P}\left[\bigwedge_{e\in S}(\hat{x}_e=0)\right]\leq\prod_{e\in S}\mathbb{P}\left[\hat{x}_e=0\right]$. However, modest numerical evidence suggests this should hold as well.

Still we believe our result has a merit. Unlike Srinivasan's procedure and the one of Chekuri et al. , our procedure can be extended to k-column sparse 0-1 packings problems — such constraints are for example k-dimensional b-matchings, and more generally k-hypegraph b-matchings. Here, our procedure guarantees preserving marginals with approximation of $\frac{1}{k+1}$ — this we are able to show as it follows from the contention resolution scheme for intersection of k-transversal matroids that we present in Chapter 5. However, as of the writing we are unable to show the negative correlation property, because the proof from single matroid case complicates slightly. Still, we conjecture that one can extend the proof for the single matroid case to show that negative correlation in the general k-set packing setup is true. More precisely we conjecture the following.

Conjecture 14. *Let $A\in\{0,1\}^{|E|\cdot d}$ be a matrix, where every column has at most k non-zero entries, and let $b\in\mathbb{Z}_{+}^{d}$. Let $(x_e)_E\in[0,1]^E$ be a point so that*

$$A\cdot x\leq b.$$

There exists a randomized rounding procedure that outputs $(\hat{x}_e)_{e\in E}\in\{0,1\}^E$ such that:

1. $A\cdot\hat{x}\leq b$,
2. $\mathbb{P}\left[\hat{x}_e=1\right]\geq\frac{1}{k+1}x_e$,
3. *for any subset $S\subseteq E$ it holds that*

$$\mathbb{P}\left[\bigwedge_{e\in S}(\hat{x}_e=1)\right]\leq\prod_{e\in S}\mathbb{P}\left[\hat{x}_e=1\right].$$

Chapter 2

Preliminaries

For set $S \subseteq E$ and element $e \in E$ we use $S + e$ to denote $S \cup \{e\}$, and $S - e$ to denote $S \setminus \{e\}$. For set $S \subseteq E$ we shall denote by $\mathbf{1}_S$ a characteristic vector of set S, and for a single element e we shall write $\mathbf{1}_e$ instead of $\mathbf{1}_{\{e\}}$. For random event $\mathcal{A}$ we shall denote by $\chi[\mathcal{A}]$ a 0-1 random variable that indicates whether $\mathcal{A}$ occurred. The optimal strategy will be denoted by OPT, and we shall denote the expected objective value of its outcome as $\mathbb{E}[f(OPT)]$.

2.1 Matroids and polytopes

Let $\mathcal{M} = (E, \mathcal{I})$ be a matroid, where E is the universe of elements and $\mathcal{I} \subseteq 2^E$ is a family of independent sets. For element $e \in E$, we shall denote the matroid $\mathcal{M}$ with e contracted by $\mathcal{M}/e$, i.e. $\mathcal{M}/e = (E - e, \{S \subseteq E - e \,|S + e \in \mathcal{I}\})$.

The following lemma is a slightly modified[1] basis exchange lemma, which can be found in [47].

Lemma 15. *Let $A, B \in \mathcal{I}$ and $|A| = |B|$. There exists a bijection $\phi : A \mapsto B$ such that: 1) $\phi(e) = e$ for every $e \in A \cap B$, 2) $B - \phi(e) + e \in \mathcal{I}$.*

We shall use the following corollary, where we consider independent sets of possibly different sizes.

Corollary 16. *Let $A, B \in \mathcal{I}$. We can find assignment $\phi_{A,B} : A \mapsto B \cup \{\perp\}$ such that:*

$\phi_{A,B}(e) = e$ for every $e \in A \cap B$,

for each $f \in B$ there exists at most one $e \in A$ for which $\phi_{A,B}(e) = f$,

for $e \in A \setminus B$, if $\phi_{A,B}(e) = \perp$ then $B + e \in \mathcal{I}$, otherwise $B - \phi_{A,B}(e) + e \in \mathcal{I}$.

Proof. Suppose $|A| \geq |B|$. We use the matroid augmentation property a sufficient number of times to extend B into $B' \in \mathcal{I} : |A| = |B'|$ using elements of A. From Lemma 15 we get a bijection $\phi : A \mapsto B'$. For each $e \in A$, we set $\phi_{A,B}(e) = \perp$, if $\phi(e) \in B' \setminus B$, and $\phi_{A,B}(e) = \phi(e)$ otherwise. Case $|A| \leq |B|$ is similar. $\square$

[1] The difference is that we do not assume that A, B are bases, but independent sets of the same size.

We consider optimization over *matroid polytopes* which have the general form:

$$\mathcal{P}(\mathcal{M}) = \left\{ x \in \mathbb{R}^E_{\geq 0} \middle| \forall_{A \in \mathcal{I}} \sum_{e \in A} x_e \leq r_{\mathcal{M}}(A) \right\},$$

where $r_{\mathcal{M}}$ is the rank function of $\mathcal{M}$. We know [47] that the matroid polytope $\mathcal{P}(\mathcal{M})$ is equivalent to the convex hull of $\{\mathbf{1}_A \,|A \in \mathcal{I}\}$, i.e. characteristic vectors of all independent sets of $\mathcal{M}$. Thus, we can represent any $x \in \mathcal{P}(\mathcal{M})$ as $x = \sum_{i=1}^{m} \beta_i \cdot \mathbf{1}_{B_i}$, where $B_1, \ldots, B_m \in \mathcal{I}$ and $\beta_1, \ldots, \beta_m$ are non-negative weights such that $\sum_{i=1}^{m} \beta_i = 1$. We shall call sets $B_1, \ldots, B_m$ a *support* of x in $\mathcal{P}(\mathcal{M})$.

We shall also deal with a specific type of matroids which are transversal matroids.

Definition 17. Consider bipartite graph $(E \cup V, \subseteq E \times V)$. Let $\mathcal{I}$ be a family of all subsets S of E such that there exists a matching between S and V of size exactly $|S|$. Then $M = (E, \mathcal{I})$ is a matroid, called *transversal* matroid.

We shall assume that we know the graph representation $(E \cup V, \subseteq E \times V)$ of the given transversal matroid. This assumption is quite natural and common, e.g., [7].

2.2 Submodular functions

A set function $f : 2^E \mapsto \mathbb{R}_{\geq 0}$ is *submodular*, if for any two subsets $S, T \subseteq E$ we have $f(S \cup T) + f(S \cap T) \leq f(S) + f(T)$. We call function f *monotone*, if for any two subsets $S \subseteq T \subseteq E : f(S) \leq f(T)$. For a set $S \subseteq E$, we let $f_S(A) = f(A \cup S) - f(S)$ denote the marginal increase in f when the set A is added to S. Note that if f is monotone submodular, then so is f_S for all $S \subseteq E$. Moreover, we have $f_S(\emptyset) = 0$ for all $S \subseteq E$, so f_S is normalized. Without loss of generality, we assume also that $f(\emptyset) = 0$.

Multilinear extension We consider the *multilinear extension* $F : [0,1]^E \mapsto \mathbb{R}_{\geq 0}$ of f, whose value at a point $y \in [0,1]^E$ is given by

$$F(y) = \sum_{A \subseteq E} f(A) \prod_{e \in A} y_e \prod_{e \notin A} (1 - y_e).$$

Note that $F(\mathbf{1}_A) = f(A)$ for any set $A \subseteq E$, so F is an extension of f from discrete domain 2^E into a real domain $[0,1]^E$. For $y \in [0,1]^E$ let $R(y)$ denote a random subset $A \subseteq E$ that is constructed by taking each element $e \in E$ with probability y_e. Then, we note that $F(y) = \mathbb{E}[f(R(y))]$. Following this interpretation, Calinescu et al. [13] show that $F(y)$ can be estimated to any desired accuracy in polynomial time, using a sampling procedure.

Additionally, they show that F has the following properties, which we shall make use of in our analysis:

Lemma 18. *The multilinear extension F is linear along the coordinates, i.e. for any point $x \in [0,1]^E$, any element $e \in E$, and any $\xi \in [-1,1]$ such that $x + \xi \cdot \mathbf{1}_e \in [0,1]^E$, it holds that $F(x + \xi \cdot \mathbf{1}_e) - F(x) = \xi \cdot \frac{\partial F}{\partial y_e}(x)$, where $\frac{\partial F}{\partial y_e}(x)$ is the partial derivative of F in direction y_e at point x.*

Lemma 19. *If $F : [0,1]^E \mapsto \mathbb{R}$ is a multilinear extension of monotone submodular function $f : 2^E \mapsto \mathbb{R}$, then 1) function F has second partial derivatives everywhere; 2) for each $e \in E$, $\frac{\partial F}{\partial y_e} \geq 0$ everywhere; 3) for any $e_1, e_2 \in E$ (possibly equal), $\frac{\partial^2 F}{\partial y_{e_1} \partial y_{e_2}} \leq 0$, which means that $\frac{\partial F}{\partial y_{e_2}}$ is non-increasing with respect to y_{e_1}.*

Concave closure Another extension of f studied in [13] is given by:

$$f^+(y) = \max \left\{ \sum_{A \subseteq E} \alpha_A f(A) \middle| \sum_{A \subseteq E} \alpha_A \leq 1,\ \forall A \subseteq E : \alpha_A \geq 0,\ \forall j \in E : \sum_{A : j \in A} \alpha_A \leq y_j \right\} \tag{2.1}$$

Intuitively, the solution $(\alpha_A)_{A \subseteq E}$ above represents the distribution over 2^E that maximizes the value $\mathbb{E}\left[f(A)\right]$ subject to the constraint that its marginal values satisfy $\mathbb{P}\left[i \in A\right] \leq y_i$. The value $f^+(y)$ is then the expected value of $\mathbb{E}\left[f(A)\right]$ under this distribution, while the value of $F(y)$ is the value of $\mathbb{E}\left[f(A)\right]$ under the particular distribution that places each element i in A independently. From this interpretation one can easily conclude that for avery point $y \in [0,1]^E$ we have

$$f^+(y) \geq F(y).$$

We remark that calculating the value of f^+ for an arbitrary point y is an NP-hard problem, and even APX-hard [52].

Convex closure and Lovasz extension One can consider an extension very similar to the concave extension, i.e., a convex extension:

$$f^-(y) = \min \left\{ \sum_{A \subseteq E} \alpha_A f(A) \middle| \sum_{A \subseteq E} \alpha_A \geq 1,\ \forall A \subseteq E : \alpha_A \geq 0,\ \forall j \in E : \sum_{A : j \in A} \alpha_A \geq y_j \right\}.$$

Unlike the concave closure, function f^- is efficiently computable for every point. This is due to the fact that f^- is equal to the so-called Lovasz extension f^L.

Definition 20. For a function $f : \{0,1\}^E \mapsto \mathbb{R}$, extension $f^L : [0,1]^E \mapsto \mathbb{R}$ is defined by

$$f^L(y) = \sum_{i=0}^{|E|} \lambda_i f(S_i),$$

where $\emptyset = S_0 \subset S_1 \subset ... \subset S_{|E|}$is a chain such that $\sum \lambda_i \mathbf{1}_{S_i} = y$ and $\sum \lambda_i = 1$ for $\lambda_i \geq 0$.

An equivalent way of defining Lovasz extension is $f^L(y) = \mathbb{E}\left[f(\{e : x_e \geq \lambda\})\right]$, where λ is uniformly chosen at random from $[0,1]$. Due to the submodularity one can show that f^L is actually equal to f^- [53]. Given this, one can efficiently compute f^- either by sampling λ — this gives a $(1+\epsilon)$-approximation, which often may be enough —, or one can solve the linear program defining f^-, since its solution will have non-zero variables α only for at most $|E|$ subsets that will form a chain.

2.3 Martingale theory

Here, we give a brief overview of the basic notions from martingale theory necessary for the proof of Lemma 31. See [55] for extended background on martingale theory.

Definition 21. Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space, where Ω is a sample space, $\mathcal{F}$ is a σ-algebra on Ω, and $\mathbb{P}$ is a probability measure on $(\Omega, \mathcal{F})$. Sequence $\{\mathcal{F}_t : t \geq 0\}$ is called a *filtration* if it is an increasing family of sub-σ-algebras of $\mathcal{F}$: $\mathcal{F}_0 \subseteq \mathcal{F}_1 \subseteq \ldots \subseteq \mathcal{F}$.

Intuitively speaking, when considering a stochastic process, σ-algebra $\mathcal{F}_t$ represents all information available to us right after making step t. In our case σ-algebra $\mathcal{F}_t$ contains all information about each randomly chosen element to probe, about outcome of each probe, and about each support update for every matroid, that happened before or at step t.

Definition 22. A process $(X_t)_{t\geq 0}$ is called a *martingale* if for every $t \geq 0$ all following conditions hold:
random variable X_t is $\mathcal{F}_t$-measurable,
$\mathbb{E}[|X_t|] < \infty$,
$\mathbb{E}[X_{t+1}| \mathcal{F}_t] = X_t$.
Random variable $\tau : \Omega \mapsto \{0, 1, \ldots\}$ is called a *stopping time* if $\{\tau = t\} \in \mathcal{F}_t$ for every $t \geq 0$.

Intuitively, τ represents a moment when an event happens. We have to be able to say whether it happened at step t given only the information from steps $0, 1, 2, \ldots, t$. In our case we define τ as the first moment when the fractional solution becomes zero. It is clear that this is a stopping time according to the above definition.

Theorem 23 (Doob's optional stopping Theorem)**.** *Let τ be a stopping time. Let $(X_t)_{t\geq 0}$ be a martingale. If there exists a constant N such that always $\tau < N$, then $\mathbb{E}[X_\tau] = \mathbb{E}[X_0]$.*

The above is not Doob's theorem in its full generality, but rather the simplest variant that still holds in our setting.

Sometimes we shall actually consider a *sub-martingale* $(X_t)_{t\geq 0}$ which satisfies $\mathbb{E}[X_{t+1}| \mathcal{F}_t] \geq X_t$ instead of equality in the above definition. In this case the Doob's Theorem we are guaranteed that $\mathbb{E}[X_\tau] \geq \mathbb{E}[X_0]$.

Chapter 3

Stochastic matching

In this chapter we present two algorithms for the weighted stochastic matching problem. First one is a refined version of the algorithm of Bansal et al. [8]. It turns out that the idea used there is a stochastic analog of the technique of exponential clocks which were used in the design of approximation algorithm for various problems. The algorithm improves the bound of 3-approximation of Bansal et al. down to 2.85, which is the current best. It also improves the bound of 4 for arbitrary graphs down to 3.709. This one is not the current best however, as Baveja et al. recently presented a 3.22 approximation.

Second algorithm is based on the idea of iterative randomized rounding. This algorithm matches the 3-approximation from [8]. In fact, it is the same algorithm, but we provide a different analysis. We present this algorithm for various reasons. First, it is the simplest possible setup where we use the technique of iterative randomized rounding, which is used in, basically, all other chapters. Second, the analysis of Bansal et al. , and the analysis of our improvement in the next section, give a non-linear dependency between probabilities of taking edges, while the analysis vie iterative randomized rounding gives a linear dependency. We believe that this simplicity may allow for future improvement of the approximation ratio by using, e.g., factor revealing LPs.

3.1 Stochastic matching via stochastic exponential clocks

Exponential clocks is an idea for designing algorithms which consider underlying elements in a random order that is much more structured than just a plain uniform permutation. Suppose we are given elements $\{1, 2, ..., n\}$ and corresponding parameters $\{x_1, x_2, ..., x_n\}$. Element i is assigned a random variable Y_i that has exponential distribution with parameter x_i, i.e., $\mathbb{P}[Y_i < t] = 1 - e^{x_i t}$. After drawing random variables Y_i we consider them in an increasing order. This idea was successfully used for many different problems []. In this chapter we develop its stochastic analogue: element i, which on top of x_i also has probability p_i, is assigned a random variable $Y_i : \mathbb{P}[Y_i < t] = \frac{1}{p_i}\left(1 - e^{-p_i x_i t}\right)$. This new idea leads to improved approximation bounds for the Stochastic Matching problem.

3.1.1 Improved approximation for bipartite graphs

Let us denote by OPT the optimum probing strategy, and let $\mathbb{E}[OPT]$ denote its expected outcome. Consider the following LP:

$$\max \sum_{e \in E} w_e p_e x_e \qquad \text{(LP-BIP)}$$

$$\text{s.t.} \sum_{e \in \delta(u)} p_e x_e \leq 1, \qquad \forall u \in V; \tag{3.1}$$

$$\sum_{e \in \delta(u)} x_e \leq t_u, \qquad \forall u \in V; \tag{3.2}$$

$$0 \leq x_e \leq 1, \qquad \forall e \in E. \tag{3.3}$$

The proof of the following Lemma is already quite standard [2, **?**, 23] — just note that $x_e = \mathbb{P}[OPT \text{ probes } e]$ is a feasible solution of LP-BIP.

Lemma 24. *[8]Let LP_{bip} be the optimal value of* LP-BIP*. It holds that $LP_{bip} \geq \mathbb{E}[OPT]$.*

Our approach is similar to the one of Bansal et al. [8] (see also Algorithm 1 in the figure). We solve LP-BIP: let $x = (x_e)_{e \in E}$ be the optimal fractional solution. Then we apply to x the rounding procedure by Gandhi et al. [29], which we shall call just GKPS. Let $\hat{E}$ be the set of rounded edges, and let $\hat{x}_e = 1$ if $e \in \hat{E}$ and $\hat{x}_e = 0$ otherwise. GKPS guarantees the following properties of the rounded solution:

1. (Marginal distribution) For any $e \in E$, $\mathbb{P}[\hat{x}_e = 1] = x_e$.
2. (Degree preservation) For any $v \in V$, $\sum_{e \in \delta(v)} \hat{x}_e \leq \lceil \sum_{e \in \delta(v)} x_e \rceil \leq t_v$.
3. (Negative correlation) For any $v \in V$, any subset $S \subseteq \delta(v)$ of edges incident to v, and any $b \in \{0, 1\}$, it holds that $\mathbb{P}[\wedge_{e \in S}(\hat{x}_e = b)] \leq \prod_{e \in S} \mathbb{P}[\hat{x}_e = b]$.

Our algorithm sorts the edges in $\hat{E}$ according to a random permutation and probes each edge $e \in \hat{E}$ according to that order, but provided that the endpoints of e are not matched already. It is important to notice that, by the degree preservation property, in $\hat{E}$ there are at most t_v edges incident to each node v. Hence, the timeout constraint of v is respected even if the algorithm probes all the edges in $\delta(u) \cap \hat{E}$.

Our algorithm differs from [8] and subsequent work in the way edges are randomly ordered. Prior work exploits a random uniform order on $\hat{E}$. We rather use the following, more complex strategy. For each $e \in \hat{E}$ we draw a random variable Y_e distributed on the interval $\left[0, \frac{1}{p_e} \ln \frac{1}{1-p_e}\right]$ according to the following cumulative distribution: $\mathbb{P}[Y_e \leq y] = \frac{1}{p_e}\left(1 - e^{-p_e y}\right)$. Observe that the density function of Y_e in this interval is e^{-yp_e} (and zero otherwise). Edges of $\hat{E}$ are sorted in increasing order of the Y_e's, and they are probed according to that order. We next let $Y = (Y_e)_{e \in \hat{E}}$.

Define $\hat{\delta}(v) := \delta(v) \cap \hat{E}$. We say that an edge $e \in \hat{E}$ is *safe* if, at the time we consider e for probing, no other edge $f \in \hat{\delta}(e)$ is already taken into the matching. Note that the algorithm can probe e only in that case, and if we do probe e, it is added to the matching with probability p_e.

Algorithm 1 Approximation algorithm for bipartite Stochastic Matching.

1. Let $(x_e)_{e \in E}$ be the solution to LP-BIP.
2. Round the solution $(x_e)_{e \in E}$ with GKPS; let $(\hat{x}_e)_{e \in E}$ be the rounded 0-1 solution, and $\hat{E} = \{e \in E | \hat{x}_e = 1\}$.
3. For every $e \in \hat{E}$, sample a random variable Y_e distributed as $\mathbb{P}[Y_e \le y] = \frac{1-e^{-yp_e}}{p_e}$.
4. For every $e \in \hat{E}$ in increasing order of Y_e:
   (a) If no edge $f \in \hat{\delta}(e) := \delta(e) \cap \hat{E}$ is yet taken, then probe edge e

The main ingredient of our analysis is the following lower-bound on the probability that an arbitrary edge e is safe.

Lemma 25. *For every edge e it holds that $\mathbb{P}\left[e \text{ is safe} | e \in \hat{E}\right] \ge g(p_e)$, where*

$$g(p) := \frac{1}{2+p}\left(1 - \exp\left(-(2+p)\frac{1}{p}\ln\frac{1}{1-p}\right)\right).$$

Proof. In the worst case every edge $f \in \hat{\delta}(e)$ that is before e in the ordering can be probed, and each of these probes has to fail for e to be safe. Thus

$$\mathbb{P}\left[e \text{ is safe} | e \in \hat{E}\right] \ge \mathbb{E}_{\hat{E}\setminus e, Y}\left[\prod_{f \in \hat{\delta}(e): Y_f < Y_e} (1-p_f) \middle| e \in \hat{E}\right].$$

Now we take expectation on Y only, and using the fact that the variables Y_f are independent, we can write the latter expectation as

$$\mathbb{E}_{\hat{E}\setminus e}\left[\int_0^{\frac{1}{p_e}\ln\frac{1}{1-p_e}} \left(\prod_{f \in \hat{\delta}(e)} \left(\mathbb{P}[Y_f \le y](1-p_f) + \mathbb{P}[Y_f > y]\right)\right) e^{-p_e \cdot y} \mathrm{d}y \middle| e \in \hat{E}\right]. \tag{3.4}$$

Observe that $\mathbb{P}[Y_f \le y](1-p_f) + \mathbb{P}[Y_f > y] = 1 - p_f \mathbb{P}[Y_f \le y]$. When $y > \frac{1}{p_f}\ln\frac{1}{1-p_f}$, then $\mathbb{P}[Y_f \le y] = 1$, and moreover, $\frac{1}{p_f}(1-e^{-p_f \cdot y})$ is an increasing function of y. Thus we can upper-bound $\mathbb{P}[Y_f \le y]$ by $\frac{1}{p_f}(1-e^{-p_f \cdot y})$ for any $y \in [0, \infty]$, and obtain that $1 - p_f \mathbb{P}[Y_f \le y] \ge 1 - p_f \frac{1}{p_f}(1-e^{-p_f \cdot y}) = e^{-p_f \cdot y}$. Thus (3.4) can be lower bounded by

$$\begin{aligned} &\mathbb{E}_{\hat{E}\setminus e}\left[\int_0^{\frac{1}{p_e}\ln\frac{1}{1-p_e}} e^{-\sum_{f \in \hat{\delta}(e)} p_f \cdot y - p_e \cdot y} \mathrm{d}y \middle| e \in \hat{E}\right] \\ =&\mathbb{E}_{\hat{E}\setminus e}\left[\frac{1}{\sum_{f \in \hat{\delta}(e)} p_f + p_e}\left(1 - e^{-\left(\sum_{f \in \hat{\delta}(e)} p_f + p_e\right)\frac{1}{p_e}\ln\frac{1}{1-p_e}}\right) \middle| e \in \hat{E}\right]. \end{aligned}$$

□

From the negative correlation and marginal distribution properties we know that $\mathbb{E}_{\hat{E}\setminus e}\left[\hat{x}_f \middle| e \in \hat{E}\right] \leq \mathbb{E}_{\hat{E}\setminus e}\left[\hat{x}_f\right] = x_f$ for every $f \in \delta(e)$, and therefore $\mathbb{E}_{\hat{E}\setminus e}\left[\sum_{f\in\hat{\delta}(e)} p_f \middle| e \in \hat{E}\right] \leq \sum_{f\in\delta(e)} p_f x_f \leq 2$, where the last inequality follows from the LP constraints. Consider function $f(x) := \frac{1}{x+p_e}\left(1 - e^{-(x+p_e)\frac{1}{p_e}\ln\frac{1}{1-p_e}}\right)$. This function is decreasing and convex. From Jensen's inequality we know that $\mathbb{E}\left[f(X)\right] \geq f(\mathbb{E}\left[X\right])$. Thus

$$\mathbb{E}_{\hat{E}\setminus e}\left[f\left(\sum_{f\in\hat{\delta}(e)} p_f\right)\middle| e \in \hat{E}\right] \geq f\left(\mathbb{E}_{\hat{E}\setminus e}\left[\sum_{f\in\hat{\delta}(e)} p_f \middle| e \in \hat{E}\right]\right)$$
$$\geq f(2) = \frac{1}{2+p_e}\left(1 - e^{-(2+p_e)\frac{1}{p_e}\ln\frac{1}{1-p_e}}\right) = g(p_e). \qquad \square$$

From Lemma 25 and the marginal distribution property, the expected contribution of edge e to the profit of the solution is

$$w_e p_e \cdot \mathbb{P}\left[e \in \hat{E}\right] \cdot \mathbb{P}\left[e \text{ is safe} \middle| e \in \hat{E}\right] \geq w_e p_e x_e \cdot g(p_e) \geq w_e p_e x_e \cdot g(1) = \frac{1}{3} w_e p_e x_e.$$

Therefore, our analysis implies a 3 approximation, matching the result in [8]. However, by playing with the probabilities appropriately we can do better.

Patching with Greedy. We next describe an improved approximation algorithm, based on the patching of the above algorithm with a simple greedy one. Let $\delta \in (0, 1)$ be a parameter to be fixed later. We define E_{large} as the (*large*) edges with $p_e \geq \delta$, and let E_{small} be the remaining (*small*) edges. Recall that LP_{bip} denotes the optimal value of LP-BIP. Let also LP_{large} and LP_{small} be the fraction of LP_{bip} due to large and small edges, respectively; i.e., $LP_{large} = \sum_{e\in E_{large}} w_e p_e x_e$ and $LP_{small} = LP_{bip} - LP_{large}$. Define $\gamma \in [0, 1]$ such that $\gamma LP_{bip} = LP_{large}$. By refining the above analysis, we obtain the following result.

Lemma 26. *Algorithm 1 has expected approximation ratio* $\frac{1}{3}\gamma + g(\delta)(1 - \gamma)$.

Proof. The expected profit of the algorithm is at least:

$$\sum_{e\in E} w_e p_e x_e \cdot g(p_e) \geq \sum_{e\in E_{large}} w_e p_e x_e \cdot g(1) + \sum_{e\in E_{small}} w_e p_e x_e \cdot g(\delta)$$
$$= \frac{1}{3} LP_{large} + g(\delta) LP_{small} = \left(\frac{1}{3}\gamma + g(\delta)(1-\gamma)\right) LP_{bip}. \qquad \square$$

$\square$

Consider the following greedy algorithm. Compute a maximum weight matching M_{grd} in G with respect to edge weights $w_e p_e$, and probe the edges of M_{grd} in any order. Note that the timeout constraints are satisfied since we probe at most one edge incident to each node (and timeouts are strictly positive by definition and w.l.o.g.).

Lemma 27. *The greedy algorithm has expected approximation ratio* $\delta\gamma$.

Proof. It is sufficient to show that the expected profit of the obtained solution is at least $\delta \cdot LP_{large}$. Let $x = (x_e)_{e \in E}$ be the optimal solution to LP-BIP. Consider the solution $x' = (x'_e)_{e \in E}$ that is obtained from x by setting to zero all the variables corresponding to edges in E_{small}, and by multiplying all the remaining variables by δ. Since $p_e \geq \delta$ for all $e \in E_{large}$, x' is a feasible fractional solution to the following matching LP:

$$\begin{aligned}
\max \sum_{e \in E} w_e p_e z_e & \qquad\qquad \text{(LP-MATCH)} \\
\text{s.t.} \sum_{e \in \delta(u)} z_e \leq 1, & \qquad\qquad \forall u \in V; \\
0 \leq z_e \leq 1, & \qquad\qquad \forall e \in E. \qquad (3.5)
\end{aligned}$$

The value of x' in the above LP is $\delta \cdot LP_{large}$ by construction. Let LP_{match} be the optimal profit of LP-MATCH. Then $LP_{match} \geq \delta \cdot LP_{large}$. Given that the graph is bipartite, LP-MATCH defines the matching polyhedron, and we can find an integral optimal solution to it. But such a solution is exactly a maximum weight matching according to weights $w_e p_e$, i.e. $\sum_{e \in M_{grd}} w_e p_e = LP_{match}$. The claim follows since the expected profit of the greedy algorithm is precisely the weight of M_{grd}. □ □

The overall algorithm, for a given δ, simply computes the value of γ, and runs the greedy algorithm if $\gamma\delta \geq \left(\frac{1}{3}\gamma + g(\delta)(1-\gamma)\right)$, and Algorithm 1 otherwise[1].

The approximation factor is given by $\max\{\frac{\gamma}{3} + (1-\gamma)g(\delta), \gamma\delta\}$, and the worst case is achieved when the two quantities are equal, i.e., for $\gamma = \frac{g(\delta)}{\delta + g(\delta) - \frac{1}{3}}$, yielding an approximation ratio of $\frac{\delta \cdot g(\delta)}{\delta + g(\delta) - \frac{1}{3}}$. Maximizing (numerically) the latter function in δ gives $\delta = 0.6022$, and the final 2.845-approximation ratio claimed in Theorem 5.

3.1.2 Improved approximation for general graphs

For general graphs, we consider the linear program LP-GEN which is obtained from LP-BIP by adding the following *blossom inequalities*:

$$\sum_{e \in E(W)} p_e x_e \leq \frac{|W| - 1}{2} \qquad \forall W \subseteq V, |W| \text{ odd.} \qquad (3.6)$$

Here $E(W)$ is the subset of edges with both endpoints in W. We remark that, using standard tools from matching theory, we can solve LP-GEN in polynomial time despite its exponential number of constraints; see the book of Schrijver for details [47]. Also in this case $x_e = \mathbb{P}[OPT \text{ probes } e]$ is a feasible solution of LP-GEN, hence the analogue of Lemma 24 still holds.

Our Stochastic Matching algorithm for the case of a general graph $G = (V, E)$ works via a reduction to the bipartite case. First we solve LP-GEN; let $x = (x_e)_{e \in E}$ be the optimal fractional solution. Second we randomly split the nodes V into two sets A and B, with E_{AB} being the set of edges between them. On the bipartite graph $(A \cup B, E_{AB})$ we apply the algorithm for the bipartite case, but using the

[1] Note that we cannot run both algorithms, and take the best solution.

fractional solution $(x_e)_{e\in E_{AB}}$ induced by LP-GEN rather than solving LP-BIP. Note that $(x_e)_{e\in E_{AB}}$ is a feasible solution to LP-BIP for the bipartite graph $(A \cup B, E_{AB})$.

The analysis differs only in two points w.r.t. the one for the bipartite case. First, with $\hat{E}_{AB}$ being the subset of edges of E_{AB} that were rounded to 1, we have now that $\mathbb{P}\left[e \in \hat{E}_{AB}\right] = \mathbb{P}\left[e \in E_{AB}\right] \cdot \mathbb{P}\left[e \in \hat{E}_{AB} \middle| e \in E_{AB}\right] = \frac{1}{2}x_e$. Second, but for the same reason, using again the negative correlation and marginal distribution properties, we have

$$\mathbb{E}\left[\sum_{f\in\hat{\delta}(e)} p_f \middle| e \in \hat{E}_{AB}\right] \leq \sum_{f\in\delta(e)} p_f \mathbb{P}\left[f \in \hat{E}_{AB}\right] = \frac{1}{2}\sum_{f\in\delta(e)} p_f x_f \leq \frac{1}{2}(2 - 2p_e x_e) \leq 1.$$

Repeating the steps of the proof of Lemma 25 and including the above inequality we get the following.

Lemma 28. *For every edge e it holds that $\mathbb{P}\left[e \text{ is safe} \middle| e \in \hat{E}_{AB}\right] \geq h(p_e)$, where*

$$h(p) := \frac{1}{1+p}\left(1 - \exp\left(-(1+p)\frac{1}{p}\ln\frac{1}{1-p}\right)\right).$$

Since $h(p_e) \geq h(1) = \frac{1}{2}$, we directly obtain a 4-approximation which matches the result in [8]. Similarly to the bipartite case, we can patch this result with the simple greedy algorithm (which is exactly the same in the general graph case). For a given parameter $\delta \in [0, 1]$, let us define γ analogously to the bipartite case. Similarly to the proof of Lemma 26, one obtains that the above algorithm has approximation factor $\frac{\gamma}{4} + \frac{1-\gamma}{2}h(\delta)$. Similarly to the proof of Lemma 27, the greedy algorithm has approximation ratio $\gamma\delta$ (here we exploit the blossom inequalities that guarantee the integrality of the matching polyhedron). We can conclude similarly that in the worst case $\gamma = \frac{h(\delta)}{2\delta + h(\delta) - 1/2}$, yielding an approximation ratio of $\frac{\delta \cdot h(\delta)}{2\delta + h(\delta) - 1/2}$. Maximizing (numerically) this function over δ gives, for $\delta = 0.5580$, the 3.709 approximation ratio claimed in Theorem 6.

3.1.3 Online Stochastic Matching with Timeouts

Let $G = (A \cup B, A \times B)$ be the input graph, with items A and buyer types B. We use the same notation for edge probabilities, edge profits, and timeouts as in Stochastic Matching. Following [8], we can assume w.l.o.g. that each buyer type is sampled uniformly with probability $1/n$. Consider the following linear program:

$$\begin{aligned}
\max \sum_{a\in A, b\in B} & w_{ab} p_{ab} x_{ab} && \text{(LP-ONL)} \\
\text{s.t.} \sum_{b\in B} & p_{ab} x_{ab} \leq 1, && \forall a \in A \\
\sum_{a\in A} & p_{ab} x_{ab} \leq 1, && \forall b \in B \\
\sum_{a\in A} & x_{ab} \leq t_b, && \forall b \in B \\
0 \leq & x_{ab} \leq 1, && \forall ab \in E.
\end{aligned}$$

The above LP models a bipartite Stochastic Matching instance where one side of the bipartition contains exactly one buyer per buyer type. In contrast, in the online case several buyers of the same buyer type (or none at all) can arrive, and the optimal strategy can allow many buyers of the same type to probe edges. Still, this is not a problem since the following lemma from [8] allows us just to look at the graph of buyer types and not at the actual realized buyers.

Lemma 29 ([8], Lemmas 9 and 11)**.** *Let* $\mathbb{E}[OPT]$ *be the expected profit of the optimal online algorithm for the problem. Let* LP_{onl} *be the optimal value of* LP-ONL. *It holds that* $\mathbb{E}[OPT] \le LP_{onl}$.

We will devise an algorithm whose expected outcome is at least $\frac{1}{4.07} \cdot LP_{onl}$, and then Theorem 7 follows from Lemma 29.

3.1.3.1 The algorithm.

We initially solve LP-ONL and let $(x_{ab})_{ab \in A \times B}$ be the optimal fractional solution. Then buyers arrive. When a buyer of type b is sampled, then 1) if a buyer of the same type b was already sampled before we simply discard her, do nothing, and wait for another buyer to arrive, 2) if it is the first buyer of type b, then we execute the following *subroutine for buyers.* Since we take action only when the first buyer of type b comes, we shall denote such a buyer simply by b, as it will not cause any confusion.

Subroutine for buyers. Let us consider the step of the online algorithm in which the first buyer of type b arrived, if any. Let A_b be the items that are still available when b arrives. Our subroutine will probe a subset of at most t_b edges ab, $a \in A_b$. Consider the vector $(x_{ab})_{a \in A_b}$. Observe that it satisfies the constraints $\sum_{a \in A_b} p_{ab} x_{ab} \le 1$ and $\sum_{a \in A_b} x_{ab} \le t_b$. Again using GKPS, we round this vector in order to get $(\hat{x}_{ab})_{a \in A_b}$ with $\hat{x}_{ab} \in \{0, 1\}$, and satisfying the marginal distribution, degree preservation, and negative correlation properties[2]. Let $\hat{A}_b$ be the set of items a such that $\hat{x}_{ab} = 1$. For each ab, $a \in \hat{A}_b$, we independently draw a random variable Y_{ab} with distribution: $\mathbb{P}[Y_{ab} < y] = \frac{1}{p_{ab}}(1 - \exp(-p_{ab} \cdot y))$ for $y \in \left[0, \frac{1}{p_{ab}} \ln \frac{1}{1-p_{ab}}\right]$. Let $Y = (Y_{ab})_{a \in \hat{A}_b}$.

Next we consider items of $\hat{A}_b$ in increasing order of Y_{ab}. Let $\alpha_{ab} \in [\frac{1}{2}, 1]$ be a *dumping factor* that we will define later. With probability α_{ab} we probe edge ab and as usual we stop the process (of probing edges incident to b) if ab is present. Otherwise (with probability $1 - \alpha_{ab}$) we *simulate* the probe of ab, meaning that with probability p_{ab} we stop the process anyway — like if edge ab were probed and turned out to be present. Note that we do not get any profit from the latter simulation since we do not really probe ab.

[2] Actually in this case we have a bipartite graph where one side has only one vertex, and here GKPS reduces to Srinivasan's rounding procedure for level-sets [49].

Dumping factors. It remains to define the dumping factors. For a given edge ab, let

$$\beta_{ab} := \mathbb{E}_{\hat{A}_b \setminus a, Y}\left[\prod_{a' \in A_b : Y_{a'b} < Y_{ab}} (1 - p_{a'b}) \middle| a \in \hat{A}_b\right].$$

Using the inequality $\sum_{a \in A_b} p_{ab} x_{ab} \leq 1$, by repeating the analysis from Section 3.1.1 we can show that

$$\beta_{ab} \geq h(p_{ab}) = \frac{1}{1+p_{ab}}\left(1 - \exp\left(-(1+p_{ab})\frac{1}{p_{ab}}\ln\frac{1}{1-p_{ab}}\right)\right) \geq \frac{1}{2}.$$

Let us assume for the sake of simplicity that we are able to compute β_{ab} exactly. We will show in Section 3.1.3.3 how to remove this assumption. We set $\alpha_{ab} = \frac{1}{2\beta_{ab}}$. Note that α_{ab} is well defined since $\beta_{ab} \in [1/2, 1]$.

3.1.3.2 Analysis.

Let us denote by $\mathcal{A}_b$ the event that at least one buyer of type b arrives. The probability that an edge ab is probed can be expressed as:

$$\mathbb{P}[\mathcal{A}_b] \cdot \mathbb{P}[\text{no } b' \text{ takes } a \text{ before } b | \mathcal{A}_b] \cdot \mathbb{P}[b \text{ probes } a | \mathcal{A}_b \wedge a \text{ is not yet taken}].$$

The probability that b arrives is $\mathbb{P}[\mathcal{A}_b] = 1 - \left(1 - \frac{1}{n}\right)^n \geq 1 - \frac{1}{e}$. We shall show first that

$$\mathbb{P}[b \text{ probes } a | \mathcal{A}_b \wedge a \text{ is not yet taken}]$$

is exactly $\frac{1}{2}x_{ab}$, and later we shall show that $\mathbb{P}[\text{no } b' \text{ takes } a \text{ before } b | \mathcal{A}_b]$ is at least $\frac{1}{1+\frac{1}{2}\left(1-\frac{1}{e}\right)}$. This will yield that the probability that ab is probed is at least

$$\left(1 - \frac{1}{e}\right)\frac{1}{1+\frac{1}{2}\left(1-\frac{1}{e}\right)} \cdot \frac{1}{2}x_{ab} = \frac{e-1}{3e-1}x_{ab} > \frac{1}{4.16}x_{ab}.$$

Consider the probability that some edge $a'b$ appearing before ab in the random order *blocks* edge ab, meaning that ab is not probed because of $a'b$. Observe that each such $a'b$ is indeed considered for probing in the online model, and the probability that $a'b$ blocks ab is therefore $\alpha_{a'b}p_{a'b} + (1 - \alpha_{a'b})p_{a'b} = p_{a'b}$. We can conclude that the probability that ab is not blocked is exactly β_{ab}.

Due to the dumping factor α_{ab}, the probability that we actually probe edge $ab \in \hat{A}_b$ is exactly $\alpha_{ab} \cdot \beta_{ab} = \frac{1}{2}$. Recall that $\mathbb{P}\left[a \in \hat{A}_b\right] = x_{ab}$ by the marginal distribution property. Altogether

$$\mathbb{P}[b \text{ probes } a | \mathcal{A}_b \wedge a \text{ is not yet taken}] = \frac{1}{2}x_{ab}. \tag{3.7}$$

Next let us condition on the event that buyer b arrived, and let us lower bound the probability that ab is not blocked on the a's side in such a step, i.e., that no other buyer has taken a already. The buyers, who are first occurrences of their type, arrive uniformly at random. Therefore, we can analyze the process of their arrivals as if it was constructed by the following procedure: every buyer b' is given an independent random variable $Y_{b'}$ distributed exponentially on $[0, \infty]$, i.e., $\mathbb{P}[Y_{b'} < y] = 1 - e^y$; buyers arrive in increasing order of their variables $Y_{b'}$. Once buyer b' arrives, it probes edge ab' with probability (exactly) $\alpha_{ab'}\beta_{ab'}x_{ab'} = \frac{1}{2}x_{ab'}$ — these probabilities are independent among different buyers. Thus, conditioning on the fact that b arrives, we obtain the following expression for the probability that a is safe at the moment when b arrives:

$$\begin{aligned} &\mathbb{P}\left[\text{no } b' \text{ takes } a \text{ before } b \middle| \mathcal{A}_b\right] \\ \geq\ & \mathbb{E}\left[\prod_{b' \in B \setminus b : Y_{b'} < Y_b} \left(1 - \mathbb{P}\left[\mathcal{A}_{b'} \middle| \mathcal{A}_b\right] \mathbb{P}\left[b' \text{ probes } ab' \middle| \mathcal{A}_{b'}\right] p_{ab'}\right) \middle| \mathcal{A}_b\right] \\ =\ & \int_0^\infty \prod_{b' \in B \setminus b} \left(1 - \mathbb{P}\left[\mathcal{A}_{b'} \middle| \mathcal{A}_b\right] \cdot \mathbb{P}\left[Y_{b'} < y \middle| \mathcal{A}_{b'}\right] \cdot \mathbb{P}\left[b' \text{ probes } ab' \middle| \mathcal{A}_{b'}\right] p_{ab'}\right) e^{-y} \mathrm{d}y. \end{aligned}$$

Now let us upper-bound each of the probability factors in the above product. First of all $\mathbb{P}\left[\mathcal{A}_{b'} \middle| \mathcal{A}_b\right] = 1 - \left(1 - \frac{1}{n}\right)^{n-1} \leq 1 - \frac{1}{e}$. Second, $\mathbb{P}\left[Y_{b'} < y \middle| \mathcal{A}_{b'}\right] = 1 - e^{-y}$ just by definition[3]. Third, from (3.7) we have that $\mathbb{P}\left[b' \text{ probes } ab' \middle| \mathcal{A}_{b'}\right] = \frac{x_{ab}}{2}$.

Thus the above integral can be lower bounded by

$$\begin{aligned} & \int_0^\infty \prod_{b' \in B \setminus b} \left(1 - \left(1 - \frac{1}{e}\right)(1 - e^{-y}) \cdot \frac{1}{2} x_{ab'} \cdot p_{ab'}\right) e^{-y} \mathrm{d}y \\ \geq\ & \int_0^\infty \prod_{b' \in B \setminus b} \exp\left(-\left(1 - \frac{1}{e}\right) \frac{1}{2} x_{ab'} \cdot p_{ab'} \cdot y\right) e^{-y} \mathrm{d}y \\ =\ & \frac{1}{1 + \left(1 - \frac{1}{e}\right) \frac{1}{2} \left(\sum_{b' \in B \setminus b} p_{ab'} \cdot x_{ab'}\right)} \\ \geq\ & \frac{1}{1 + \frac{1}{2}\left(1 - \frac{1}{e}\right)} = \frac{2e}{3e - 1}. \end{aligned}$$

Above in the first inequality we used the fact that $1 - c(1 - e^{-y}) \geq e^{-cy}$ for $c \in [0, 1]$ and any $y \in \mathbb{R}$: here $c = \left(1 - \frac{1}{e}\right) \frac{1}{2} x_{ab'} \cdot p_{ab'}$. In the first equality we used $\int_0^\infty e^{-ax} \mathrm{d}x = \frac{1}{a}$. In the last inequality we used the LP constraint $\sum_{b' \in B \setminus b} p_{ab'} \cdot x_{ab'} \leq 1$.

Altogether, as anticipated earlier,

$$\mathbb{P}[ab \text{ is probed}] \geq \left(1 - \frac{1}{e}\right) \frac{x_{ab}}{2} \cdot \frac{2e}{3e - 1} = x_{ab} \cdot \frac{e - 1}{3e - 1} > \frac{1}{4.16} \cdot x_{ab}.$$

In next Section we show how to compute the dumping factors so that the above probability is $\frac{e-1}{3e-1} + \varepsilon$ for an arbitrarily small constant $\varepsilon > 0$. In particular, by choosing a small enough ε the factor 4.16 is still guaranteed.

We can again use the approach with big and small probabilities, thus reducing the approximation factor to 4.07. The details are given in Section 3.1.3.4. Theorem 7 follows.

[3] The $\mathcal{A}_{b'}$ event in the condition simply indicates that $Y_{b'}$ was drawn.

3.1.3.3 Computing Dumping Factors

Recall that we assumed the knowledge of quantities β_{ab}, which are needed to define the dumping factors α_{ab}. Though we are not able to compute the first quantities exactly in polynomial time, we can efficiently estimate them and this is sufficient for our goals. Let us focus on a given edge ab. Recall that

$$\begin{aligned}\beta_{ab} := \mathbb{E}_{\hat{A}_b \setminus a, Y}\left[\prod_{a' \in A_b : Y_{a'b} < Y_{ab}} (1 - p_{a'b}) \middle| a \in \hat{A}_b\right] \\ \geq \frac{1}{1+p_{ab}}\left(1 - \exp\left(-(1+p_{ab})\frac{1}{p_{ab}}\ln\frac{1}{1-p_{ab}}\right)\right) = h(p_{ab}).\end{aligned}$$

Let us simulate the subroutine for buyers N times without the dumping factors —- in a simulation we run GKPS, we sample the Y variables, but we simulate probes of edges, and we never really probe any edge. We shall set N later. Let $S^1, S^2, ..., S^N$ be 0-1 indicator random variables of whether a was safe or not in each simulation. Note that $\mathbb{E}\left[S^i\right] = \beta_{ab}x_{ab} \in [h(p_{ab})\, x_{ab}, x_{ab}]$.

Suppose that $x_{ab} \geq \frac{\epsilon}{n}$, where n is the number of buyers. The expression $\hat{s}_{ab} = \frac{1}{N}\sum_{i=1}^{N} S^i$ should be a good estimation of $\beta_{ab} \cdot x_{ab}$, i.e., $\hat{s}_{ab} \in [\beta_{ab}x_{ab}(1-\epsilon), \beta_{ab}x_{ab}(1+\epsilon)]$ with probability $1 - \frac{1}{n^C}$. Set $N = \frac{6n}{\epsilon^3}\ln\left(2n^2 Z\right)$ for $Z = 3\frac{1}{\varepsilon} + 1$.

Applying Chernoff's bound $\mathbb{P}\left[|X - \mathbb{E}[X]| > \varepsilon\mathbb{E}[X]\right] \leq 2e^{-\frac{\varepsilon^2}{3}\mathbb{E}[X]}$ with $X = \sum_{i=1}^{N} S_i$ one obtains:

$$\begin{aligned}&\mathbb{P}\left[\sum_{i=1}^{N} S_i \notin \left[(1-\epsilon)\beta_{ab}x_{ab}\cdot N, (1+\epsilon)\beta_{ab}x_{ab}\cdot N\right]\right] \\ &\leq 2\exp\left(-\frac{\epsilon^2}{3}\beta_{ab}x_{ab}\cdot N\right) \leq 2\exp\left(-\frac{\epsilon^2}{3}\frac{x_{ab}}{2}\cdot N\right) \leq 2\exp\left(-\frac{\epsilon^3}{6n}\cdot N\right) = \frac{1}{n^2}\frac{1}{Z}.\end{aligned}$$

From the union-bound, with probability at least $1 - \frac{1}{Z}$ we have that $\hat{s}_{ab} \in [\beta_{ab}x_{ab}(1-\epsilon), \beta_{ab}x_{ab}(1+\epsilon)]$ for every edge ab such that $x_{ab} \geq \frac{\epsilon}{n}$.

Now let us assume this happened, i.e., we have good estimates. We set $\alpha_{ab} = \max\{\frac{1}{2}, \min\{\frac{1}{2}\frac{x_{ab}}{\hat{s}_{ab}}, 1\}\}$ which belongs to $\left[\frac{1}{2}\frac{1}{\beta_{ab}(1+\epsilon)}, \frac{1}{2}\frac{1}{\beta_{ab}(1-\epsilon)}\right]$, but only for edges ab such that $x_{ab} \geq \frac{\epsilon}{n}$. For edges ab such that $x_{ab} < \frac{\epsilon}{n}$ we just put $\alpha_{ab} = 1$ (so we do not dump such edges actually). Two elements of the proof were depending on the dumping factors. First, now the probability that edge is taken is $\alpha_{ab}\beta_{ab}x_{ab} \in \left[\frac{x_{ab}}{2(1+\epsilon)}, \frac{x_{ab}}{2(1-\epsilon)}\right]$. Second, recall the probability of an edge ab not to be blocked:

$$\frac{1}{1 + \left(1 - \frac{1}{e}\right)\left(\sum_{b' \in B \setminus b} p_{ab'} \cdot \alpha_{ab'}\beta_{ab'} \cdot x_{ab'}\right)}. \tag{3.8}$$

We have that

$$\begin{aligned}
&\sum_{b'\in B\backslash b} p_{ab'}\cdot\alpha_{ab'}\beta_{ab'}\cdot x_{ab'}\\
=&\sum_{b'\in B\backslash b: x_{ab'}\geq\frac{\epsilon}{n}} p_{ab'}\cdot\alpha_{ab'}\beta_{ab'}\cdot x_{ab'}+\sum_{b'\in B\backslash b: x_{ab'}<\frac{\epsilon}{n}} p_{ab'}\cdot\alpha_{ab'}\beta_{ab'}\cdot x_{ab'}\\
\leq&\sum_{b'\in B\backslash b: x_{ab'}\geq\frac{\epsilon}{n}} p_{ab'}\cdot\frac{1}{2(1-\epsilon)}x_{ab'}+\sum_{b'\in B\backslash b: x_{ab'}<\frac{\epsilon}{n}} x_{ab'}\\
\leq&\frac{1}{2(1-\epsilon)}+\epsilon=\frac{1}{2}+O(\epsilon).
\end{aligned}$$

So the probability that a is not blocked is at least $\frac{1}{1+\left(1-\frac{1}{e}\right)\left(\frac{1}{2}+O(\epsilon)\right)}$. The final probability that edge ab is probed is at least

$$\begin{aligned}
\left(1-\frac{1}{e}\right)\frac{x_{ab}}{2(1+\epsilon)}\cdot\frac{1}{1+\left(1-\frac{1}{e}\right)\left(\frac{1}{2}+O(\epsilon)\right)}&=\frac{x_{ab}}{1+\varepsilon}\cdot\frac{e-1}{2e+(e-1)(1+O(\epsilon))}\\
&=x_{ab}\cdot\frac{e-1}{3e-1+O(\epsilon)}>\frac{1}{4.16}\cdot x_{ab}.
\end{aligned}$$

In the last inequality above we assumed ε to be small enough.

With probability at most $\frac{1}{Z}$ we did not obtain good estimates of the dumping factors. Still we have that $\alpha_{ab}\in\left[\frac{1}{2},1\right]$, and therefore $\alpha_{ab}\beta_{ab}\in\left[\frac{1}{4},1\right]$. In this case quantity (3.8) can be just lower-bounded by $\frac{1}{1+\left(1-\frac{1}{e}\right)}$, and the probability that edge ab is probed in the subroutine for buyers is at least $\frac{x_{ab}}{4}$. Thus the probability that edge ab is probed during the algorithm is at least $\left(1-\frac{1}{e}\right)\frac{x_{ab}}{4}\cdot\frac{1}{1+\left(1-\frac{1}{e}\right)}=\frac{x_{ab}}{4}\cdot\frac{e-1}{2e-1}>\frac{1}{10.33}x_{ab}$. The total expected outcome of the algorithm is therefore, for sufficiently small ε, at least

$$LP_{onl}\left(\left(1-\frac{1}{Z}\right)\frac{e-1}{3e-1+O(\epsilon)}+\frac{1}{Z}\frac{1}{4}\cdot\frac{e-1}{2e-1}\right)\overset{Z=3\frac{1}{\varepsilon}+1}{\geq}\frac{1}{4.16}LP_{onl}.$$

The above approach can be combined with the small/big probability trick from the next Section. By choosing ε small enough the approximation ratio is 4.07 as claimed.

3.1.3.4 Combination with Greedy in the Online Case

Recall that $h(p)=\frac{1}{1+p}\left(1-\exp\left(-(1+p)\frac{1}{p}\ln\frac{1}{1-p}\right)\right)$. We are again applying the big/small probabilities trick, so let $\delta\in(0,1)$ be a parameter to be fixed later. Consider back again the subroutine for buyers. Previously we have used dumping factors $\alpha_{ab}=\frac{1}{2\beta_{ab}}$, where — recall — $\beta_{ab}\geq h(p_{ab})$.

This time we define $\alpha_{ab}=\frac{1}{\beta_{ab}}h(\delta)$ for ab such that $p_{ab}\leq\delta$, and $\alpha_{ab}=\frac{1}{\beta_{ab}}\frac{1}{2}$ otherwise. We again assume here that we can calculate β_{ab} (see subsection 3.1.3.3). Define $E_{large}=\{ab\in E\,|p_{ab}\geq\delta\}$ and $E_{small}=E\backslash E_{large}$, and let $LP_{large}=\gamma\cdot LP_{onl}$. Therefore, for edge ab the probability that ab is probed when b scans items is exactly

$h(\delta)$ for $ab \in E_{small}$ and $\frac{1}{2}$ for $ab \in E_{large}$. Now by repeating the steps in the proof of Section 3.1.3, we obtain that the probability that ab is not blocked on a's side is at least

$$\begin{aligned}\frac{1}{1+\left(1-\frac{1}{e}\right)\left(\sum_{b'\in B\setminus b} p_{ab'}\cdot \alpha_{ab'}\beta_{ab'}\cdot x_{ab'}\right)} &\geq \frac{1}{1+\left(1-\frac{1}{e}\right)h(\delta)\left(\sum_{b'\in B\setminus b} p_{ab'}\cdot x_{ab'}\right)}\\ &\geq \frac{1}{1+\left(1-\frac{1}{e}\right)h(\delta)},\end{aligned}$$

since $\alpha_{ab'}\cdot\beta_{ab'} = h(\delta)$ for small edges and $\alpha_{ab'}\cdot\beta_{ab'} = \frac{1}{2} \leq h(\delta)$ for large edges. Therefore, the approximation ratio of such an algorithm is at least

$$\begin{aligned}&\left(1-\frac{1}{e}\right)\left(\gamma\frac{1/2}{1+h(\delta)\left(1-\frac{1}{e}\right)}+(1-\gamma)\frac{h(\delta)}{1+h(\delta)\left(1-\frac{1}{e}\right)}\right)\\ &\qquad = \left(1-\frac{1}{e}\right)\frac{1}{1+h(\delta)\left(1-\frac{1}{e}\right)}\left(\gamma\frac{1}{2}+(1-\gamma)h(\delta)\right).\end{aligned}$$

An alternative algorithm simply computes a maximum weight matching w.r.t. weights $p_e w_e$ in the graph corresponding to LP-ONL, and upon arrival of the first copy of a buyer type b probes only the edge incident to b in the matching (if any). By the same argument as in the offline case, this matching has weight at least $\gamma\cdot\delta\cdot LP_{onl}$, and every buyer type is sampled with probability at least $1-\frac{1}{e}$. So the approximation ratio of the greedy algorithm is at least $\left(1-\frac{1}{e}\right)\gamma\delta$.

For a fixed δ, depending on the value of γ (that we can compute offline) we can run the algorithm with best approximation ratio according to the above analysis. Thus the overall approximation ratio is

$$(1-\frac{1}{e})\max\left\{\frac{1}{1+h(\delta)\left(1-\frac{1}{e}\right)}\left(\gamma\frac{1}{2}+(1-\gamma)h(\delta)\right),\gamma\cdot\delta\right\}.$$

Optimizing over δ gives $\delta = 0.525$ and a final approximation factor strictly less than 4.07.

3.2 Stochastic matching via iterative randomized rounding

In this Section we give a different analysis of the 3-approximation algorithm of Bansal et al.

Consider the following LP for stochastic matching problem:

$$\max \sum_{e} w_e p_e x_e \tag{3.9}$$
$$\sum_{e \in \delta(v)} p_e x_e \leq 1 \qquad \forall v \in V \tag{3.10}$$
$$\sum_{e \in \delta(v)} x_e \leq t_v \qquad \forall v \in V \tag{3.11}$$
$$0 \leq x_e \leq 1 \qquad \forall e \in E. \tag{3.12}$$

Suppose that $(x_e)_{e\in E}$ is the optimal solution to this LP. We round the solution with dependent rounding of Gandhi et al. [29]; we call the algorithm GKPS. Let $\left(\hat{X}_e\right)_{e\in E}$ be the rounded solution, and denote $\hat{E} = \left\{e \in E \,\middle|\, \hat{X}_e = 1\right\}$. From the definition of dependent rounding we know that:

1. (Marginal distribution) $\mathbb{P}\left[\hat{X}_e = 1\right] = x_e$;
2. (Degree preservation) For any $v \in V$ it holds that
$$\sum_{e\in\delta(v)} \hat{X}_e \leq \left\lceil \sum_{e\in\delta(v)} x_e \right\rceil \leq t_v;$$
3. (Negative correlation) For any $v \in V$ and any subset $S \subseteq \delta(v)$ of edges incident to v it holds that:
$$\forall_{b\in\{0,1\}} \mathbb{P}\left[\bigwedge_{e\in S} \left(\hat{X}_e = b\right)\right] \leq \prod_{e\in S} \mathbb{P}\left[\hat{X}_e = b\right].$$

Negative correlation property and constraint (3.10) imply that

$$\mathbb{E}\left[\sum_{f\in\delta(e)} p_f \hat{X}_f \,\middle|\, e \in \hat{E}\right] \leq \mathbb{E}\left[\sum_{f\in\delta(e)} p_f \hat{X}_f\right] = \sum_{f\in\delta(e)} p_f x_f \leq 2 - 2p_e x_e. \tag{3.13}$$

Given the solution $\left(\hat{X}_e\right)_{e\in E}$ we execute the selection algorithm presented on Figure 2. Because of the Degree preservation property we will not exceed the patience of any vertex. We say that an edge e is safe if no other edge adjacent to e was already successfully probed; otherwise edge is *blocked*. Initially all edges are safe.

The expected outcome of our algorithm is

$$\sum_e \mathbb{P}[e \text{ probed}] \cdot p_e \cdot w_e = \sum_e \mathbb{P}\left[\hat{X}_e = 1\right] \cdot \mathbb{P}\left[e \text{ probed} \middle| \hat{X}_e = 1\right] \cdot p_e \cdot w_e$$
$$= \sum_e w_e p_e x_e \cdot \mathbb{P}\left[e \text{ probed} \middle| \hat{X}_e = 1\right],$$

and we shall show that $\mathbb{P}\left[e \text{ probed} \middle| \hat{X}_e = 1\right] \geq \frac{1}{3}$ for any edge e, which will imply 1/3-approximation of the algorithm.

Algorithm 2 Algorithm for Stochastic Matching

Solve the LP; let x an optimal solution;
let $\hat{X} \in \{0,1\}^E$ be a solution rounded using GKPS; let $\hat{E} = \left\{ e \middle| \hat{X}_e = 1 \right\}$; call every $e \in \hat{E}$ *safe*
while there are still safe elements in $\hat{E}$ **do**
  pick element e uniformly at random from safe elements of $\hat{E}$
  probe e
  if probe successful **then**
    $S \leftarrow S \cup \{e\}$
    call every $f \in \hat{E} \cap \delta(e)$ *blocked*
return S

From now let us condition that we know the set of edges $\hat{E}$ and we know that $\hat{X}_e = 1$.

Consider a random variable Y_e^t which indicates if edge e is still in the graph after step t. We consider variable Y_f^t for any edge $f \in E$. Initially we have $Y_f^0 = 1$ for any $f \in \hat{E}$, and $Y_f^0 = 0$ for $f \notin \hat{E}$. Let variable P_e^t denote if edge e was probed in one of steps $0, 1, ..., t$; we have $P_e^0 = 0$.

Let Σ^t be the number of edges that are left after t steps. Variable $P_e^{t+1} - P_e^t$ indicates whether edge e was probed in step $t+1$. Given the information $\mathcal{F}^t$ about the process up to step t, probability of this event is $\mathbb{E}\left[P_e^{t+1} - P_e^t \middle| \mathcal{F}^t, \hat{E}, \hat{X}_e = 1 \right] = \frac{Y_e^t}{\Sigma^t}$, i.e., if edge e still exists in the graph after step t (i.e. $Y_e^t = 1$), then the probability is $\frac{1}{\Sigma^t}$, otherwise it is 0.

Variable $Y_e^t - Y_e^{t+1}$ indicates whether edge e was blocked from the graph in step $t+1$. Given $\mathcal{F}^t$, probability of this event is $\mathbb{E}\left[Y_e^t - Y_e^{t+1} \middle| \mathcal{F}^t, \hat{E}, \hat{X}_e = 1 \right] = \frac{Y_e^t}{\Sigma^t} \cdot \left(\sum_{f \in \delta(e)} p_f Y_f^t + 1 \right)$.

It is immediate to note that $Y_f^t \leq \hat{X}_f$ for any edge f, and that $P_e^{t+1} - P_e^t$ is always nonnegative. Hence

$$\begin{aligned} &\mathbb{E}\left[\left(\sum_{f \in \delta(e)} p_f \hat{X}_f + 1 \right) \cdot \left(P_e^{t+1} - P_e^t \right) - \left(Y_e^t - Y_e^{t+1} \right) \middle| \mathcal{F}^t, \hat{E}, \hat{X}_e = 1 \right] \\ \geq &\mathbb{E}\left[\left(\sum_{f \in \delta(e)} p_f Y_f^t + 1 \right) \cdot \left(P_e^{t+1} - P_e^t \right) - \left(Y_e^t - Y_e^{t+1} \right) \middle| \mathcal{F}^t, \hat{E}, \hat{X}_e = 1 \right] = 0, \end{aligned}$$

which means that the sequence

$$\left(\left(\sum_{f \in \delta(e)} p_f \hat{X}_f + 1 \right) \cdot P_e^t - \left(1 - Y_e^t \right) \right)_{t \geq 0}$$

is a super-martingale.

Let $\tau = \min \left\{ t \middle| Y_e^t = 0 \right\}$ be the step in which edge e was either blocked or probed. It is clear that τ is a stopping time. Thus from Doob's Stopping Theorem

— this time in the variant for super-martingales, i.e., if $\mathbb{E}\left[Z^{t+1} - Z^t \middle| \mathcal{F}^t\right] \geq 0$, then $\mathbb{E}\left[Z^\tau\right] \geq \mathbb{E}\left[Z^0\right]$ — we get that

$$\mathbb{E}_\tau \left[\left(\sum_{f \in \delta(e)} p_f \hat{X}_f + 1 \right) \cdot P_e^\tau - (1 - Y_e^\tau) \middle| \hat{E}, \hat{X}_e = 1 \right] \\ \geq \mathbb{E} \left[\left(\sum_{f \in \delta(e)} p_f \hat{X}_f + 1 \right) \cdot P_e^0 - \left(1 - Y_e^0\right) \middle| \hat{E}, \hat{X}_e = 1 \right],$$

where the expectation above is over the random variable τ only. Since $P_e^0 = 0, Y_e^0 = 1, Y_e^\tau = 0$ the above inequality implies that

$$\mathbb{E}_\tau \left[\left(\sum_{f \in \delta(e)} p_f \hat{X}_f + 1 \right) \cdot P_e^\tau \middle| \hat{E}, \hat{X}_e = 1 \right] \geq 1.$$

Since we condition all the time on $\hat{E}$ and $\hat{X}_e = 1$ we can write that

$$\left(\sum_{f \in \delta(e)} p_f \hat{X}_f + 1 \right) \cdot \mathbb{E}_\tau \left[P_e^\tau \middle| \hat{E}, \hat{X}_e = 1 \right] \geq 1.$$

Let us notice that $\mathbb{E}_\tau \left[P_e^\tau \middle| \hat{E}, \hat{X}_e = 1 \right]$ is exactly equal to $\mathbb{P}\left[e \text{ probed} \middle| \hat{E}, \hat{X}_e = 1 \right]$. Thus we can write that

$$\mathbb{P}\left[e \text{ probed} \middle| \hat{E}, \hat{X}_e = 1 \right] \geq \frac{1}{\sum_{f \in \delta(e)} p_f \hat{X}_f + 1}.$$

Now we can apply to both sides of the above inequality expectation over $\hat{E}$ but still conditioned on $\hat{X}_e = 1$:

$$\mathbb{P}\left[e \text{ probed} \middle| \hat{X}_e = 1 \right] = \mathbb{E}_{\hat{E}} \left[\mathbb{P}\left[e \text{ probed} \middle| \hat{E}, \hat{X}_e = 1 \right] \middle| \hat{X}_e = 1 \right] \\ \geq \mathbb{E}_{\hat{E}} \left[\frac{1}{\sum_{f \in \delta(e)} p_f \hat{X}_f + 1} \middle| \hat{X}_e = 1 \right],$$

and from Jensen's inequality, and the fact that $x \mapsto \frac{1}{x}$ is convex, we get that

$$\mathbb{E}_{\hat{E}} \left[\frac{1}{\sum_{f \in \delta(e)} p_f \hat{X}_f + 1} \middle| \hat{X}_e = 1 \right] \geq \frac{1}{\mathbb{E}_{\hat{E}} \left[\sum_{f \in \delta(e)} p_f \hat{X}_f + 1 \middle| \hat{X}_e = 1 \right]}.$$

From inequality (3.13) we get $\mathbb{E}_{\hat{E}} \left[\sum_{f \in \delta(e)} p_f \hat{X}_f + 1 \middle| \hat{X}_e = 1 \right] \leq \mathbb{E}_{\hat{E}} \left[\sum_{f \in \delta(e)} p_f \hat{X}_f + 1 \right] \leq 3 - 2x_e p_e \leq 3$ and we conclude that

$$\mathbb{P}\left[e \text{ probed} \middle| \hat{X}_e = 1 \right] \geq \frac{1}{3}.$$

Chapter 4

Stochastic probing via iterative randomized rounding

Our result is a new algorithm for stochastic probing problem based on iterative randomized rounding of linear programs and the *continuous greedy algorithm* introduced by Calinescu et al. [13]. We show that an iterative, randomized rounding algorithm combined with the continuous greedy algorithm gives a $\frac{1-e^{-T}}{T(k^{out}+k^{in})+1}$-approximation for the stochastic probing problem with monotone submodular objective function, where $T \in (0,1]$ is the time at which the continuous greedy algorithm is stopped. By running the continuous greedy algorithm until $T = 1$, we obtain a $\frac{1-e^{-1}}{k^{out}+k^{in}+1}$-approximation. However, we show that for $k = k^{out} + k^{in} > 1$, the optimal value of the stopping time is given by $T = -1 - \frac{1}{k} - W_{-1}\left(-e^{-1-\frac{1}{k}}\right) \approx \sqrt{\frac{2}{k}} - \frac{1}{3k}$, where W is the Lambert W function.

Additionally, we improve the bound of $\frac{1}{4(k^{in}+k^{out})}$ given by Gupta and Nagarajan [36] in the case of a linear objective. Specifically, we show that our iterative randomized rounding algorithm is a $\frac{1}{k^{in}+k^{out}}$-approximation for the stochastic probing problem with a linear objective function.

4.1 Overview of the iterative randomized rounding approach

We now give a description of the general rounding approach that we employ in both the linear and submodular case. We consider an instance of a stochastic probing problem, with objective function f, outer matroid constraints $\mathcal{M}^{out}j$, where $1 \leq j \leq k^{out}$ and inner matroid constraints $\mathcal{M}^{in}j$, where $1 \leq j \leq k^{in}$. Our rounding procedure is guided by the solution of the following mathematical programming relaxation, where f^+ is the relaxation given in Section 2.2:

$$\begin{aligned} \text{maximize } & f^+(p \cdot x) \\ \text{subject to: } & x \in \mathcal{P}(\mathcal{M}^{out}j), && 1 \leq j \leq k^{out} \\ & p \cdot x \in \mathcal{P}(\mathcal{M}^{in}j), && 1 \leq j \leq k^{in} \\ & x \in [0,1]^E \end{aligned} \tag{4.1}$$

We now show that the solution of the relaxation (4.1) is an upper bound the expected value of the optimal feasible strategy for the related stochastic probing problem. Henceforth, we let x^+ denote the optimal solution to (4.1).

Lemma 30. *Let OPT be the optimal feasible strategy for the stochastic probing problem in our general setting, then,* $\mathbb{E}\left[f(OPT)\right] \leq f^+(x^+)$.

Proof. We construct a feasible solution x of (4.1) by setting $x_e = \mathbb{P}\left[OPT \text{ probes } e\right]$. First, we show that this is indeed a feasible solution of (4.1). Since OPT is a feasible strategy, the set of elements Q probed by any execution of OPT is always an independent set of each outer matroid $\mathcal{M} = \left(E, \mathcal{I}_j^{out}\right)$, i.e. $\forall_{j\in[k^{out}]} Q \in \mathcal{I}_j^{out}$. Thus, for any $j \in \left[k^{out}\right]$, the vector $\mathbb{E}\left[\mathbf{1}_Q\right] = x$ may be represented as a convex combination of vectors from $\left\{\mathbf{1}_A \,\middle|\, A \in \mathcal{I}_j^{out}\right\}$, and so $x \in \mathcal{P}\left(\mathcal{M}^{out} j\right)$. Analogously, the set of elements S that were successfully probed by OPT satisfy $\forall_{j\in[k^{in}]} S \in \mathcal{I}_j^{in}$ for every possible execution of OPT. Hence, for any $j \in \left[k^{in}\right]$ the vector $\mathbb{E}\left[\mathbf{1}_S\right] = p{\cdot}x$ may be represented as a convex combination of vectors from $\left\{\mathbf{1}_A \,\middle|\, A \in \mathcal{I}_j^{in}\right\}$ and so $p \cdot x \in \mathcal{P}\left(\mathcal{M}^{in} j\right)$.

The value $f^+(p \cdot x)$ gives the maximum value of $\mathbb{E}_{S\sim\mathcal{D}}\left[f(S)\right]$ over all distributions $\mathcal{D}$ satisfying $\mathbb{P}_{S\sim\mathcal{D}}\left[e \in S\right] = x_e p_e$. The solution S returned by OPT satisfies $\mathbb{P}\left[e \in S\right] = \mathbb{P}\left[\text{OPT probes } e\right] p_e = x_e p_e$. Thus, OPT defines one such distribution, and so we have $\mathbb{E}\left[f(OPT)\right] \leq f^+(p \cdot x) \leq f^+(p \cdot x^+)$. ☐

In the case of a submodular objective function f, it is NP-hard to solve the relaxation (4.1) exactly. We postpone our discussion of this difficulty until the relevant section.

We now describe our general rounding procedure. We are given an instance of a stochastic probing problem over universe E, specified by a set of k^{in} inner matroids, k^{out} outer matroids, an objective function f, and a probability p_e for each $e \in E$. We first obtain a feasible solution x^0 to a relaxation (4.1), using either linear programming or the continuous greedy algorithm. Next, we iteratively rounds the solution, carrying out a single probe in each iteration. In each iteration, we randomly select a single element $\bar{e}$ in the support of x to probe, choosing $\bar{e}$ with probability proportional to $x_{\bar{e}}$. We probe and update S accordingly, then update the inner and outer constraints to obtain a new relaxation of the form (4.1) representing the remaining problem. Finally, we update x to obtain a feasible solution for this new relaxation. The algorithm terminates when there are no elements remaining in the support of x.

We now describe in more detail how the algorithm carries out the updates for a single step. Suppose that at some step of the algorithm, we select $\bar{e}$ to probe. We carry out the probe, adding $\bar{e}$ to S if we are successful. Next, we replace each outer matroid $\mathcal{M}^{out} j$ with the contracted matroid $\mathcal{M}^{out} j/\bar{e}$, to reflect the fact that $\bar{e}$ has been probed. If the probe succeeds, we must similarly update each inner matroid constraint, replacing $\mathcal{M}^{in} j$ by $\mathcal{M}^{in} j/\bar{e}$, to reflect the fact that $\bar{e}$ was taken by the algorithm. If the probe fails, we do not need to update the inner constraints. Finally, we remove $\bar{e}$ from E. This gives us a new relaxation of the form (4.1).

Next, we need to further update the solution x to obtain a feasible solution for the new relaxation. We now consider the update process for x in more detail. Let us describe how to perform a single update corresponding to an inner or outer matroid constraint.

Suppose that $\bar{e}$ was the element probed, and consider some outer matroid $\mathcal{M}^{out}j$, where $1 \leq j \leq k^{out}$. Before selecting $\bar{e}$, we have $x \in \mathcal{P}(\mathcal{M}^{out}j)$. We can thus represent x as a convex combination: $x = \sum_{i=1}^{m} \beta_i^{out} \mathbf{1}_{B_i^{out}}$, where $B_1^{out}, \ldots, B_m^{out}$ are independent sets in $\mathcal{M}^{out}j$. We modify x to obtain a solution x' such that $x' \in \mathcal{P}(\mathcal{M}^{out}j)$ with $x'_{\bar{e}} = 1$, as follows. First, we pick one set B_a^{out} with $\bar{e} \in B_a^{out}$ to guide the update process. Specifically, we choose a set $B_a^{out} \ni \bar{e}$ at random with probability $\beta_a^{out}/x_{\bar{e}}$ (note that for any element e, $\sum_{a:e \in B_a^{out}} \beta_a^{out} = x_e$). Then, for any set B_b^{out}, let $i = \phi_{a,b}$, where $\phi_{a,b}$ is the mapping from B_a^{out} into B_b^{out} given by Corollary 16. If $i = \bar{e}$, then $\bar{e} \in B_b^{out}$ and we do nothing. If $i = \perp$, then $B_b^{out} + \bar{e} \in \mathcal{M}^{out}j$, and so we replace B_b^{out} by $B_b^{out} + \bar{e}$. Otherwise, we substitute B_b^{out} with $B_b^{out} - i + \bar{e}$ in the support of x. Each such substitution decreases the value of coordinate i by β_b. After performing all such substitutions, we obtain a vector $x' \in \mathcal{P}(\mathcal{M}^{out}j)$. Moreover, x' is a convex combination of independent sets all containing $\bar{e}$. Thus, we have $x'_{\bar{e}} = 1$, while for all $i \neq \bar{e}$, we have $x'_i = x_i - \delta_i$ for some $0 \leq \delta_i \leq x_i$.

Similarly, if $\bar{e}$ is successfully probed we must perform a support update for each inner matroid $\mathcal{M}^{in}j$. Here, we proceed as in the case of the outer matroids, except now we have $p \cdot x \in \mathcal{M}^{in}j$, and we choose a set $B_a^{out} \ni \bar{e}$ to guide the support update with probability $\beta_a^{out}/(x_{\bar{e}}p_{\bar{e}})$. Similarly, we consider now the vector $y = p \cdot x$ instead of x. We obtain a vector $y' \in \mathcal{M}^{in}j$ with $y'_{\bar{e}} = 1$ and for all $i \neq \bar{e}$, $y'_i = y_i - \delta_i p_i = (x - \delta_i)p_i$ for some $0 \leq \delta_i \leq x_i$.

We now show how to combine the updates obtain a feasible solution for the updated relaxation. Consider some element $i \neq \bar{e}$. Each matroid update requires decreasing x_i by some value $0 \leq \delta_i \leq x_i$. We decrease each such x_i by the *maximum* such δ_i required by any of the $k^{out} + k^{in}$ updates, and call the resulting solution x'. Then, we have both $\{x'_i\}_{i \neq \bar{e}} \in \mathcal{P}(\mathcal{M}^{in}j/\bar{e})$ for each $1 \leq j \leq k^{out}$ and $\{x'_i p_i\}_{i \neq \bar{e}} \in \mathcal{P}(\mathcal{M}^{out}j/\bar{e})$ for each $1 \leq j \leq k^{in}$. It remains to remove $\bar{e}$ from E. Note that once our algorithm sets some coordinate x_i to 0, i will never be probed, and so x_i will remain 0 for the remainder of the algorithm. Thus, in order to simplify our discussion, we do not explicitly remove $\bar{e}$ from E in each iteration. Rather, we just set $x_{\bar{e}}$ to 0. Thus, all solutions we consider will be vectors in $[0,1]^E$. Note that the coordinates of our current solution x are always decreasing throughout the algorithm, either due to a matroid update step, or because we set $x_{\bar{e}}$ to 0 after probing $\bar{e}$.

We now turn to the general analysis of our rounding procedure. In order to analyze the approximation performance of our algorithm, we shall keep track of a *potential* value z, depending on the current solution x, which intuitively represents the expected value of the remaining fractional solution x, given the choices that have been made so far. Initially, our potential z will be at least some constant fraction of the optimal value of (4.1), and in the final step z will be equal to 0. Let x^t, S^t, and z^t be the current value of x, z, and S at the beginning of the $(t+1)$th iteration, and let x^+ be the optimal solution of (4.1). Our analysis proceeds by first showing that $z^0 \geq \beta \cdot f^+(x^+)$. for some constant $\beta \in (0,1]$. Then, we consider an arbitrary step $t+1$ and analyze the expected decrease $\mathbb{E}\left[z^t - z^{t+1}\right]$ in the potential due to this step. We bound this decrease in terms of the expected increase $\mathbb{E}\left[S^{t+1} - S^t\right]$ in the probed solution S at this step, showing that:

$$\alpha \cdot \mathbb{E}\left[z^t - z^{t+1}\right] \leq \mathbb{E}\left[f(S^{t+1}) - f(S^t)\right],$$

for some $\alpha < 1$. Then, we employ the following Lemma to conclude that the algorithm is an $\alpha\beta$-approximation in expectation. The proof is based on Doob's optional stopping theorem for martingales.

Lemma 31. *Suppose our algorithm runs for τ iterations and that the potential function z satisfies $z^0 =\geq \beta \cdot f^+(p \cdot x^+)$ and $z\tau = 0$. Let $(\mathcal{F}_t)_{t\geq 0}$ be the filtration associated with our iterative algorithm, where $\mathcal{F}_i$ represents all information available after the ith iteration. Finally, suppose that in each step in our iterative rounding procedure, $\mathbb{E}\left[f\left(S^{t+1}\right) - f\left(S^t\right) \middle| \mathcal{F}_t\right] \geq \alpha \cdot \mathbb{E}\left[z^t - z^{t+1} \middle| \mathcal{F}_t\right]$. Then, the final solution S^τ produced by the algorithm satisfies $\mathbb{E}\left[f(S^\tau)\right] \geq \alpha\beta \cdot \mathbb{E}\left[f\left(OPT\right)\right]$.*

Proof. Define the random variable $G_t = f(S^t) - f(S^{t-1})$, representing the gain in f in iteration t, and similarly let $L_t = z^{t-1} - z^t$ represent the loss in z in iteration t. Additionally, define $G_0 = L_0 = 0$. For each $0 \leq t \leq \tau$, define $D_t = G_t - \alpha \cdot L_t$. The sequence of random variables $(D_0 + D_1 + ... + D_t)_{t\geq 0}$ forms a sub-martingale, i.e.

$$\mathbb{E}\left[\sum_{i=0}^{t+1} D_i \middle| \mathcal{F}_t\right] = \sum_{i=0}^{t} D_i + \mathbb{E}\left[G_{t+1} - \alpha \cdot L_{t+1} \middle| \mathcal{F}_t\right] \geq \sum_{i=0}^{t} D_i.$$

Let τ be the step in which the algorithm terminates, i.e. $\tau = \min\left\{t \middle| x^t = 0^E\right\}$. Then, the event $\tau = t$ depends only on $\mathcal{F}_0, \ldots, \mathcal{F}_t$, so τ is a stopping time. Also, by the definition of the algorithm $x^\tau = 0^E$. It is easy to verify that all the assumptions of Doob's optional stopping theorem are satisfied, and from this theorem we get that $\mathbb{E}\left[\sum_{i=0}^{\tau} D_i\right] \geq \mathbb{E}\left[D_0\right]$. Since $D_0 = 0$, we have

$$0 \leq \mathbb{E}\left[\sum_{i=0}^{\tau} D_i\right] = \mathbb{E}\left[\sum_{i=0}^{\tau} G_i - \alpha \cdot \sum_{i=0}^{\tau} L_i\right] = \mathbb{E}\left[\sum_{i=0}^{\tau} G_i\right] - \alpha \cdot \mathbb{E}\left[\sum_{i=0}^{\tau} L_i\right].$$

Finally, we note that

$$\sum_{i=0}^{\tau} G_i = \sum_{i=1}^{\tau}\left[f(S^i) - f(S^{i-1})\right] = f(S^\tau) - f(\emptyset) = f(S^\tau)$$

and so $\mathbb{E}\left[\sum_{i=0}^{\tau} G_i\right] = \mathbb{E}\left[f(S^\tau)\right]$, and similarly,

$$\sum_{i=0}^{\tau} L_i = \sum_{i=1}^{\tau}\left[z^{i-1} - z^i\right] = z^0 - z^\tau \geq \beta \cdot f^+(x^+ \cdot p) - 0 = \beta \cdot f^+(x^+ \cdot p)$$

and so from Lemma 30, $\mathbb{E}\left[f(S^\tau)\right] \geq \alpha \cdot \mathbb{E}\left[\sum_{i=0}^{\tau} L_i\right] \geq \alpha\beta \cdot f^+(x^+ \cdot p) \geq \alpha\beta \cdot \mathbb{E}\left[f(OPT)\right]$. □

Henceforth, we will implicitly condition on all information $\mathcal{F}_t$ available to the algorithm just before it makes step $t + 1$. That is, when discussing step $t + 1$ of the algorithm, we write simply $\mathbb{E}\left[\cdot\right]$ in place of $\mathbb{E}\left[\cdot \middle| \mathcal{F}_t\right]$.

4.2 Improved bounds for linear stochastic probing

In this setting, we are given a weight w_e and a probability p_e for each element $e \in E$ and $f(S)$ is simply $\sum_{e \in S} w_e$. We note that because f is linear, we in fact have

$$f^+(x \cdot p) = \sum_{e \in E} w_e x_e p_e,$$

and so we can solve (4.1) exactly via linear programming, to obtain an initial solution x^0 satisfying $f^+ (x^0 \cdot p) = f^+ (x^+ \cdot p)$.

At each step t, our algorithm randomly selects an element $\bar{e}$ to probe. Let $\Sigma = \sum_{e \in E} x_e$. Then, our algorithm chooses $\bar{e} = e$ with probability $x^[e]t/\Sigma$. As discussed in the previous overview, it then probes $\bar{e}$ and updates the matroid constraints to reflect both the choice of $\bar{e}$ and the probe. Finally, it updates x^t to obtain a new fractional solution x^{t+1} for the new set of constraints, and removing $\bar{e}$ from the support of x.

We now turn to the analysis of the algorithm. The potential z^t at step t will be given by:

$$z^t = \sum_{e \in E} w_e x_e^t p_e.$$

In particular, we have $z^0 = f^+ (x^0 \cdot p) = f^+ (x^+ \cdot p)$. Suppose that the algorithm terminates after τ steps. Then, $z^\tau = \sum_{e \in E} w_e \cdot 0 \cdot p_e = 0$. Hence, the conditions of Lemma 31 are satisfied with $\beta = 1$.

We now bound the expected loss $\mathbb{E}\left[z^t - z^{t+1}\right]$ in step $t+1$. In order to do this, we consider the value $\delta_i = p_i \left(x_i^t - x_i^{t+1}\right)$ for each $i \in E$. The decrease δ_i may be caused either by selecting i to probe, in case which we set x_i^{t+1} to 0, or by the matroid update step, in which we decrease several other coordinates of x^t to obtain x^{t+1}. Let us first consider the losses due to each matroid update.

Lemma 32. *Consider the update step performed for a given outer matroid $\mathcal{M}^{out}j$ in step $t+1$, and let δ_i^{out} be the amount that x_i^t is decreased by this step. Then, $\mathbb{E}\left[\delta_i^{out}\right] \leq \frac{1}{\Sigma}\left(1 - x_i^t\right) p_i x_i^t$.*

Proof. The expectation $\mathbb{E}\left[\delta_i^{out}\right]$ is over the random choice of an element $\bar{e}$ to probe and the random choice of an independent set to guide the update. Let $\mathcal{E}_a^{out}$ denote the event that the set B_a^{out} is chosen to guide a support update for $\mathcal{M}^{out}j$.

In a given step the probability that the set B_a^{out} is chosen to guide the support update is given by

$$\mathbb{P}\left[\mathcal{E}_a^{out}\right] = \sum_{e \in B_a^{out}} \frac{x_e^t}{\Sigma^t} \frac{\beta_a^{out}}{x_e^t} = \sum_{e \in B_a^{out}} \frac{\beta_a^{out}}{\Sigma^t} = \left|B_a^{out}\right| \frac{\beta_a^{out}}{\Sigma^t}.$$

Moreover, conditioned on the fact B_a^{out} was chosen, the probability that an element $e \in B_a^{out}$ was probed is uniform over the elements of B_a^{out}:

$$\mathbb{P}\left[e \text{ probed} \mid \mathcal{E}_a^{out}\right] = \mathbb{P}\left[e \text{ probed} \wedge \mathcal{E}_a^{out}\right] \Big/ \mathbb{P}\left[\mathcal{E}_a^{out}\right] = \frac{x_e^t}{\Sigma^t} \frac{\beta_a^{out}}{x_e^t} \Big/ \left|B_a^{out}\right| \frac{\beta_a^{out}}{\Sigma^t} = \frac{1}{\left|B_a^{out}\right|}. \tag{4.2}$$

We can write the expected decrease as $\mathbb{E}\left[\delta_i^{out}\right] = \sum_{a=1}^{m} \mathbb{P}\left[\mathcal{E}_a^{out}\right] \cdot \mathbb{E}\left[\delta_i^{out} \middle| \mathcal{E}_a^{out}\right]$. Note that for all $i \in B_a^{out}$, we have $\phi_{a,b}(i) = i$ for *every* set B_b^{out} such that $i \in B_b^{out}$. Thus, the support update will not change the current value of x_i for any $i \in B_a^{out}$, and so $\sum_{a=1}^{m} \mathbb{P}\left[\mathcal{E}_a^{out}\right] \cdot \mathbb{E}\left[\delta_i^{out} \middle| \mathcal{E}_a^{out}\right] = \sum_{a:i\notin B_a^{out}} \mathbb{P}\left[\mathcal{E}_a^{out}\right] \cdot \mathbb{E}\left[\delta_i^{out} \middle| \mathcal{E}_a^{out}\right]$.

Now let us condition on taking B_a^{out} to guide the support update. Consider a set B_b^{out} containing i. If we remove i from B_b^{out}, and hence decrease x_i by β_b^{out}, it must be the case that we chose to probe the single element $\phi_{a,b}^{-1}(i) \in B_a^{out}$. As shown in (4.2), the probability that we probe this element is $\frac{1}{|B_a^{out}|}$. Hence

$$
\begin{aligned}
&\sum_{a:i\notin B_a^{out}} \mathbb{P}\left[\mathcal{E}_a^{out}\right] \cdot \mathbb{E}\left[\delta_i^{out} \middle| \mathcal{E}_a^{out}\right] \\
&= \sum_{a:i\notin B_a^{out}} \mathbb{P}\left[\mathcal{E}_a^{out}\right] \cdot \left(\sum_{b:i\in B_b^{out}} \beta_b^{out} \cdot \mathbb{P}\left[\phi_{a,b}^{-1}(i) \text{ is probed} \middle| \mathcal{E}_a^{out}\right] \right) \\
&\leq \sum_{a:i\notin B_a^{out}} \mathbb{P}\left[\mathcal{E}_a^{out}\right] \cdot \left(\sum_{b:i\in B_b^{out}} \beta_b^{out} \cdot \frac{1}{|B_a^{out}|} \right) \\
&= \sum_{a:i\notin B_a^{out}} \mathbb{P}\left[\mathcal{E}_a^{out}\right] \cdot \frac{x_i^t}{|B_a^{out}|} \\
&= \sum_{a:i\notin B_a^{out}} \left|B_a^{out}\right| \frac{\beta_a^{out}}{\Sigma^t} \cdot \frac{x_i^t}{|B_a^{out}|} = \frac{1}{\Sigma^t} \sum_{a:i\notin B_a^{out}} \beta_a^{out} x_i^t = \frac{1}{\Sigma^t}\left(1 - x_i^t\right) x_i^t.
\end{aligned}
$$

□

Lemma 33. *Consider the update step performed for a given inner matroid $\mathcal{M}^{in}j$ in step $t+1$, and let δ_i^{in} be the amount that x_i^t is decreased by this step. Then,* $\mathbb{E}\left[\delta_i^{in}\right] \leq \frac{1}{\Sigma^t}\left(1 - p_i x_i^t\right) x_i^t$.

Proof. Because we only perform a support update when the probe of a chosen element is successful, the expectation $\mathbb{E}\left[\delta_i^{in}\right]$ is over the random result of the probe, as well as the random choice of element $\bar{e}$ to probe and the random choice of a base to guide the update. We proceed as in the case of Lemma 32, now letting $\mathcal{E}_a^{in}$ denote the event that the probe was successful and B_a^{in} is chosen to guide the support update. We have:

$$
\mathbb{P}\left[\mathcal{E}_a^{in}\right] = \sum_{e\in B_a^{in}} p_e \frac{x_e^t}{\Sigma^t} \frac{\beta_a^{in}}{p_e x_e^t} = \sum_{e\in B_a^{in}} \frac{\beta_a^{in}}{\Sigma^t} = \left|B_a^{in}\right| \frac{\beta_a^{in}}{\Sigma^t},
$$

$$
\mathbb{P}\left[e \text{ probed} \mid \mathcal{E}_a^{in}\right] = \mathbb{P}\left[e \text{ probed} \wedge \mathcal{E}_a^{in}\right] \Big/ \mathbb{P}\left[\mathcal{E}_a^{in}\right] = p_e \frac{x_e^t}{\Sigma^t} \frac{\beta_a^{in}}{p_e x_e^t} \Bigg/ \left|B_a^{in}\right| \frac{\beta_a^{in}}{\Sigma^t} = \frac{1}{|B_a^{in}|}.
$$

By a similar argument as in Lemma 32 we then have that $\mathbb{E}\left[\delta_i^{in}\right]$ is at most:

$$\begin{aligned}
\sum_{a:i\notin B_a^{in}} \mathbb{P}\left[\mathcal{E}_a^{in}\right]\cdot\left(\sum_{b:i\in B_b^{in}}\beta_b^{in}\cdot\frac{1}{|B_a^{in}|}\right) &= \sum_{a:i\notin B_a^{in}} \mathbb{P}\left[\mathcal{E}_a^{in}\right]\cdot\frac{x_i^t}{|B_a^{in}|} \\
&= \sum_{a:i\notin B_a^{out}}\left|B_a^{in}\right|\frac{1}{\Sigma^t}\beta_a^{in}\cdot\frac{x_i^t}{|B_a^{in}|} = \frac{1}{\Sigma^t}\sum_{a:i\notin B_a^{in}}\beta_a^{in}x_i^t = \frac{1}{\Sigma^t}(1-p_ix_i^t)x_i^t. \quad \square
\end{aligned}$$

We perform the matroid updates sequentially for each of the k^{in} and k^{out} matroids to obtain a new solution x^{t+1}. Now, we consider the expected decrease of a single coordinate of x due to both the initial probing step, in which we decrease the probed element's coordinate to 0, and the matroid updates.

Lemma 34. *For each step $t+1$ in the iterative rounding procedure,*

$$\mathbb{E}\left[p_i\delta_i\right] = \mathbb{E}\left[p_i(x_i^t - x_i^{t+1})\right] \le \frac{k^{in}+k^{out}}{\Sigma^t}p_ix_i^t$$

for all $i \in E$.

Proof. We must decrease x_i^t either by x_i^t, in the case that i is probed, or by the maximum value δ_i^{out} or δ_{out}^{in} required by any matroid update. We note that the total decrease in $x^[i]t$ is at most the *sum* of all the decreases required by the probing step and each individual update step. Thus, we have

$$\begin{aligned}
\mathbb{E}\left[p_i\delta_i\right] &\le \mathbb{P}\left[i \text{ probed}\right]p_ix_i^t + k^{out}\mathbb{E}\left[\delta_i^{out}\right] + k^{in}\mathbb{E}\left[\delta_i^{in}\right] \\
&\le \frac{x_i^t}{\Sigma^t}p_ix_i^t + k^{out}\frac{1}{\Sigma^t}(1-x_i^t)p_ix_i^t + k^{in}\frac{1}{\Sigma^t}(1-p_ix_i^t)p_ix_i^t \\
&\le \frac{1}{\Sigma^t}\left(k^{out}p_ix_i^t + k^{in}(1-p_ix_i^t)p_ix_i^t\right) \\
&\le \frac{1}{\Sigma^t}(k^{out}+k^{in})p_ix_i^t
\end{aligned}$$

where the second inequality follows from Lemmas 32 and 33 and the third one uses the fact that $k^{out} \ge 1$. $\square$

Using Lemma 34 we can now prove our main result for linear stochastic probing.

Theorem 35. *For a linear objective function f, the solution S produced by our randomized rounding algorithm satisfies $\mathbb{E}\left[f(S)\right] \ge \frac{1}{k^{out}+k^{in}}\mathbb{E}\left[OPT\right]$, where OPT is the value of the solution produced by the optimal policy.*

Proof. Because z^t is a linear function of x^t, the expected total decrease of z in step $t+1$ is given by:

$$\mathbb{E}\left[z^t - z^{t+1}\right] = \sum_i w_i\mathbb{E}\left[p_i(x_i^t - x_i^{t+1})\right] \le \frac{k^{out}+k^{in}}{\Sigma^t}\sum_i w_ip_ix_i^t.$$

where the final inequality follows from Lemma 34.

On the other hand, the expected gain in $f(S)$ is

$$\sum_{e\in E} w_e p_e \mathbb{P}\left[e \text{ probed}\right] = \frac{1}{\Sigma^t}\sum_{e\in E} w_e p_e x_e^t \geq \frac{1}{k^{out}+k^{in}}\mathbb{E}\left[z^t - z^{t+1}\right]$$

Applying Lemma 31, with $\beta = 1$ and $\alpha = \frac{1}{k^{out}+k^{in}}$, the final solution S^τ produced by the algorithm satisfies $\mathbb{E}\left[f(S^\tau)\right] \geq \frac{1}{k^{out}+k^{in}}\mathbb{E}\left[f(OPT)\right]$. □

4.3 Monotone submodular stochastic probing

4.3.1 Stronger bound for continuous greedy algorithm

In [13] the authors utilized the multilinear extension in order to maximize a submodular monotone function over a matroid constraint. They showed that a *continuous greedy algorithm* finds a $(1-1/e)$-approximate maximum of the above extension F over any solvable, downward closed polytope. In the special case of the matroid polytope, they show how to employ the pipage rounding [3] technique to the fractional solution to obtain an integral solution. Later, Feldman et al. [26] developed the *measured* continuous greedy algorithm, which gives improved approximations in a variety of cases. In the case of monotone submodular functions, Feldman et al. show that stopping the measured continuous greedy algorithm at time $T \in [0,1]$ yields a solution x satisfying optimal value attained by f on integral solutions in $\mathcal{P}$. They note that these $F(x) \geq (1-e^{-T})OPT$ and $x/T \in \mathcal{P}$, where OPT is the particular guarantees hold for the standard continuous greedy algorithm, as well. Because these guarantees are sufficient for our purposes, we shall focus on the simpler, standard continuous greedy algorithm. However, the following allows us to relate the value of F on the solution of the continuous greedy algorithm to the optimal value of the relaxation f^+.

Lemma 36. *Let f be a submodular function with multilinear extension F, and let $\mathcal{P}$ be any downward closed polytope. Let x be solution produced by the continuous greedy algorithm on F and $\mathcal{P}$ until time $T \in (0,1]$. Then:*
$x/T \in \mathcal{P}$
$F(x) \geq (1-e^{-T})\max_{y\in\mathcal{P}} f^+(y)$

This follows from a simple modification of the continuous greedy analyses of [13], provided by Vondrák [51], together with the observations from [26]. The remainder of the subsection is the proof of the Lemma.

We consider the extension:

$$f^*(y) = \min_{S\subseteq E}\left[f(S) + \sum_{j\in E} y_j f_S(j)\right].$$

Calinescu et al. [13] show that for any value $y \in [0,1]^E$, $f^+(y) \leq f^*(y)$.

The analysis of the continuous greedy algorithm in Calinescu et al. [13], depends on the following two lemmas, where $OPT = \max_{I\in\mathcal{I}} f(I)$ for some independence system $\mathcal{I}$. We show that we can replace OPT by $f^+(x^+)$ in both lemmas. The remainder of the analysis from [13] then follows as before.

Lemma 37 (Lemma 3.1 in [13])**.** *Consider any $y \in [0,1]^E$ and let R denote a random set in which each element $e \in E$ occurs independently with probability y_e. Then,*

$$OPT \leq F(y) + \max_{I \in \mathcal{I}} \sum_{j \in I} \mathbb{E}\left[f_R(j)\right].$$

The fractional solution x^+ can be represented as a convex combination of independent sets $B \in \mathcal{I}$. For each such set B, we have the inequality

$$\sum_{j \in B} \mathbb{E}\left[F_R(j)\right] \leq \max_{I \in \mathcal{I}} \sum_{j \in I} \mathbb{E}\left[f_R(j)\right].$$

Taking the same convex combination of these inequalities, we obtain the single inequality:

$$\sum_{j \in E} x^+ \mathbb{E}\left[f_R(j)\right] \leq \max_{I \in \mathcal{I}} \sum_{j \in I} \mathbb{E}\left[f_R(j)\right]. \tag{4.3}$$

We note that $F(y) = \mathbb{E}\left[f(R)\right]$, and so (4.3) implies

$$\begin{aligned} F(y) + \max_{I \in \mathcal{I}} \sum_{j \in I} \mathbb{E}\left[f_R(j)\right] &\geq \mathbb{E}\left[f(R)\right] + \sum_{j \in E} x^+ \mathbb{E}\left[f_R(j)\right] \\ &= \mathbb{E}\left[f(R) + \sum_{j \in E} x^+ f_R(j)\right] \\ &\geq \min_{S \subseteq E} \left[f(S) + \sum_{j \in E} x^+ f_S(j)\right] \\ &= f^*(x^+) \geq f^+(x^+) \end{aligned}$$

Thus, we obtain the analogous inequality

$$f^+(x^+) \leq F(y) + \max_{I \in \mathcal{I}} \sum_{j \in I} \mathbb{E}\left[f_R(j)\right]. \tag{4.4}$$

Lemma 38 (Lemma 3.1 in [13])**.** *With high probability, the continuous greedy algorithm for every t finds a set $I(t)$ such that*

$$\sum_{j \in I(t)} \mathbb{E}\left[f_{R(t)}(j)\right] \geq (1 - 2d\delta) OPT - F(y(t)).$$

Replacing Lemma 3.1 with bound (4.4) in the analysis of [13] we obtain:

$$\sum_{j \in I(t)} \mathbb{E}\left[f_{R(t)}(j)\right] \geq f^+(x^+) - 2d\delta OPT - F(y(t)).$$

Then, because f^+ is a relaxation of f, we have $f^+(x^+) \geq OPT$. Hence, we in fact have the analogous inequality

$$\sum_{j \in I(t)} \mathbb{E}\left[f_{R(t)}(j)\right] \geq f^+(x^+) - 2d\delta f^+(x^+) - F(y(t)) = (1 - 2d\delta) f^+(x^+) - F(y(t)). \tag{4.5}$$

The remainder of the analysis in [13] does not depend on the definition of the quantity OPT. Thus, replacing the inequalities from Lemmas 37 and 38 with inequalities (4.4) and (4.5), respectively, we obtain the same guarantees with respect to $f^+(x^+)$ as those stated with respect to OPT in [13].

4.3.2 Analysis of iterative randomized rounding with a monotone submodular function

We now consider the case in which the objective function $f : 2^E \to \mathbb{R}_{\geq 0}$ is a monotone submodular function. Obtaining an optimal solution to the relaxation (4.1) is NP-hard in this case [52], but we can obtain a constant-factor approximation using the continuous greedy algorithm. That is, we run the continuous greedy algorithm on the multilinear relaxation F of f and the polytope $\mathcal{P} = \bigcap_{j=1}^{k^{out}} \mathcal{P}(\mathcal{M}^{out}j) \cap \bigcap_{j=1}^{k^{in}} \mathcal{P}(\mathcal{M}^{in}j)$. We consider the solution $\hat{x}$ produced by the algorithm when it is terminated at time $T \in (0,1]$. According to Lemma 36, we have $\hat{x}/T \in \mathcal{P}$ and $F(p \cdot \hat{x}) \geq (1-e^{-T})f^+(p \cdot x^+)$. We then start our iterative rounding procedure with initial solution $x^0 = \hat{x}/T$, and define the potential z^t by

$$z^t = F(\mathbf{1}_{S^t} + T \cdot (p \cdot x^t)) - F(\mathbf{1}_{S^t}),$$

for all $0 \leq t \leq \tau$. Then, we have

$$\begin{aligned} z^0 = F(\mathbf{1}_{S^0} + T \cdot (p \cdot x^0)) - F(\mathbf{1}_{S^0}) &= F(\mathbf{1}_{\emptyset} + T \cdot (p \cdot x^0)) - F(\mathbf{1}_{\emptyset}) \\ &= F(T \cdot (p \cdot x^0)) = F(p \cdot \hat{x}) \geq (1-e^{-T})f^+(p \cdot x^+). \end{aligned}$$

If the probing algorithm stops after τ steps, then we have $z^\tau = F(\mathbf{1}_{S^\tau} + T \cdot p \cdot 0^E) - F(\mathbf{1}_{S^\tau}) = F(\mathbf{1}_{S^\tau}) - F(\mathbf{1}_{S^\tau}) = 0$. Thus, the potential z satisfies the conditions of Lemma 31 with $\beta = 1 - e^{-T}$.

Given the initial value x^0, our iterative rounding algorithm proceeds exactly as in the linear case. We now use Lemma 31 with the potential z to analyze the expected performance of our probing algorithm.

Theorem 39. *For a monotone submodular objective function f, and any stopping time $T \in (0,1]$ for the continuous greedy phase, the solution S produced by our randomized rounding algorithm satisfies $\mathbb{E}[f(S)] \geq \frac{(1-e^{-T})}{T(k^{out}+k^{in})+1}\mathbb{E}[f(OPT)]$, where OPT is the solution produced by the optimal policy.*

Proof. We now analyze the expected decrease $\mathbb{E}[z^t - z^{t+1}]$ due to step $t+1$ of the algorithm. Suppose that the algorithm selects element $\bar{e}$ to probe. Then, we have $S^{t+1} = S^t + \bar{e}$ with probability $p_{\bar{e}}$, and $S^{t+1} = S^t$ otherwise. Thus, we have

$$\begin{aligned} &\mathbb{E}\left[z^t - z^{t+1}\right] \\ &= \mathbb{E}\left[F(\mathbf{1}_{S^t} + T \cdot (p \cdot x^t) - F(\mathbf{1}_{S^t})\right] - \mathbb{E}\left[F(\mathbf{1}_{S^{t+1}} + T \cdot (p \cdot x^{t+1}) - F(\mathbf{1}_{S^{t+1}})\right] \\ &= \mathbb{E}\left[F(\mathbf{1}_{S^{t+1}}) - F(\mathbf{1}_{S^t})\right] + \mathbb{E}\left[F(\mathbf{1}_{S^t} + T \cdot (p \cdot x^t) - F(\mathbf{1}_{S^{t+1}} + T \cdot (p \cdot x^{t+1})\right] \\ &\leq \mathbb{E}\left[F(\mathbf{1}_{S^{t+1}}) - F(\mathbf{1}_{S^t})\right] + \mathbb{E}\left[F(\mathbf{1}_{S^t} + T \cdot (p \cdot x^t) - F(\mathbf{1}_{S^t} + T \cdot (p \cdot x^{t+1}))\right] \quad (4.6) \end{aligned}$$

where in the last line, we have used the fact that $S^{t+1} \supseteq S^t$ and F is increasing in all directions (Lemma 19).

We shall first bound the second expectation in (4.6). For each $i \in E$, we define

$$w_i = \frac{\partial F}{\partial x_i}(\mathbf{1}_{S^t}) = F(\mathbf{1}_{S^t+i}) - F(\mathbf{1}_{S^t}) = f(S^t + i) - f(S^t).$$

As in the linear case, let $\delta = x^t - x^{t+1}$, and note that $\delta_i = 0$ for all $i \in S^t$. Let $y = \mathbf{1}_{S^t} + T \cdot (p \cdot x^t)$ and suppose that we decrease the coordinates of y sequentially in some arbitrary order to obtain the vector $\mathbf{1}_{S^t} + T \cdot (p \cdot x^{t+1}) = y - T \cdot (p \cdot \delta)$. We consider the total change in F when an arbitrary coordinate i is decreased. Let y' be the vector obtained from y after all coordinates preceding i have been decreased. Lemma 18 states that F behaves as a linear function when only one coordinate changes, and so the total change in F from decreasing coordinate i is given by:

$$F(y') - F(y' - T \cdot ((p \cdot \delta) \cdot \mathbf{1}_i)) = T \cdot p_i \delta_i \frac{\partial F}{\partial x_i}(y') \leq T \cdot p_i \delta_i \frac{\partial F}{\partial x_i}(\mathbf{1}_{S^t}) = T \cdot w_i p_i \delta_i,$$

where the inequality follows from $y' \geq \mathbf{1}_{S^t}$ and Lemma 19, which states that the partial derivatives of F are coordinate-wise non-increasing. Thus, we have:

$$\begin{aligned}\mathbb{E}\left[F(\mathbf{1}_{S^t} + T \cdot (p \cdot x^t)) - F(\mathbf{1}_{S^t} + T \cdot (p \cdot x^{t+1}))\right] &= \mathbb{E}\left[F(y) - F(y - T \cdot (p \cdot \delta))\right] \\ \leq \mathbb{E}\left[\sum_{i \in E} T \cdot w_i p_i \delta_i\right] = \sum_{i \in E} T \cdot w_i p_i \mathbb{E}\left[\delta_i\right] &\leq \frac{1}{\Sigma^t}(k^{out} + k^{in}) \sum_{i \in E} T \cdot w_i p_i x^[i]t, \quad (4.7)\end{aligned}$$

where the last inequality follows from Lemma 34, since our algorithm alters the solution x^t exactly as in the linear case.

Returning to the first expectation in (4.6), we have

$$\mathbb{E}\left[F(\mathbf{1}_{S^{t+1}}) - F(\mathbf{1}_{S^t})\right] = \sum_i \mathbb{P}\left[i \text{ probed}\right] p_i (F(S^t + i) - F(S^t)) = \frac{1}{\Sigma^t} \sum_{i \in E} x_i^t p_i w_i. \quad (4.8)$$

Combining inequalities (4.6), (4.7), and (4.8) we obtain:

$$\begin{aligned}\mathbb{E}\left[z^t - z^{t+1}\right] &\leq \frac{1}{\Sigma^t} \sum_{i \in E} w_i p_i x_i^t + \frac{1}{\Sigma^t}(k^{out} + k^{in}) \sum_{i \in E} T \cdot x_i^t p_i w_i \\ &= (T \cdot (k^{out} + k^{in}) + 1) \frac{1}{\Sigma^t} \sum_i w_i p_i x_i^t.\end{aligned}$$

On the other hand, the expected increase of $f(S^{t+1}) - f(S^t)$ in this step is:

$$\frac{1}{\Sigma^t} \sum_{i \in E} x_i^t p_i (f(S^t + i) - f(S^t)) = \frac{1}{\Sigma^t} \sum_{i \in E} x_i^t p_i w_i \geq \frac{1}{T \cdot (k^{out} + k^{in}) + 1} \mathbb{E}\left[z^t - z^{t+1}\right].$$

Applying Lemma 31, with $\beta = 1 - e^{-T}$ and $\alpha = \frac{1}{T \cdot (k^{out} + k^{in}) + 1}$, the final solution S^τ produced by the algorithm satisfies

$$\mathbb{E}\left[f(S^\tau)\right] \geq \frac{1 - e^{-T}}{T \cdot (k^{out} + k^{in}) + 1} \mathbb{E}\left[f(OPT)\right]. \qquad \square$$

Let $k = k^{out} + k^{in}$. Then, Theorem 39 shows that our probing algorithm returns a solution that has expected value at least $\rho(T) \triangleq \frac{1 - e^{-T}}{Tk + 1}$ times that of the optimal policy. Setting just $T = 1$ gives us an approximation factor of $\left(1 - \frac{1}{e}\right) / (k + 1)$, but in Section 4.3.3, we derive the optimal value T_{opt} for T, showing that for $k = 1$, $T_{opt} = 1$ and for all $k > 1$,

$$T_{opt} = -1 - \frac{1}{k} - W_{-1}(-e^{-1-\frac{1}{k}}),$$

where W_{-1} is the lower, real-valued branch of the Lambert W function. We refer the reader to [19] for a thorough introduction to the Lambert W function. Chatzigeorgiou [15] gives the following bounds on $W_{-1}(e^{-u-1})$ that hold for all $u \in (0, 1)$:

$$-1 - \sqrt{2u} - \frac{3}{4}u \leq W_{-1}(-e^{-u-1}) \leq -1 - \sqrt{2u} - \frac{2}{3}u \leq W_{-1}(e^{-u-1}).$$

Using $u = \frac{1}{k}$, in this bound, we obtain

$$-\sqrt{\frac{2}{k}} + \frac{1}{4k} \leq 1 + \frac{1}{k} + W_{-1}(-e^{-1-\frac{1}{k}}) \leq -\sqrt{\frac{2}{k}} + \frac{1}{3k},$$

and hence

$$T_{opt} = \sqrt{\frac{2}{k}} - \frac{1}{\gamma_k k},$$

for $\gamma_k \in [3, 4]$, and then the optimal approximation ratio satisfies

$$\rho\left(T_{opt}\right) = \frac{1 - e^{-\sqrt{\frac{2}{k}} - \frac{1}{\gamma_k k}}}{\left(\sqrt{\frac{2}{k}} - \frac{1}{\gamma_k k}\right) k + 1} = \frac{\sqrt{\frac{2}{k}} + O\left(\frac{1}{k}\right)}{\sqrt{2k} + 1 - \frac{1}{\gamma_k}} = \frac{1 + O\left(\frac{1}{\sqrt{k}}\right)}{k + \sqrt{\frac{k}{2}}\left(1 - \frac{1}{\gamma_k}\right)}.$$

4.3.3 Derivation of T_{opt}

Recall that our submodular stochastic probing algorithm's approximation performance is given by:

$$\rho(T) = \frac{1 - e^{-T}}{Tk + 1},$$

where T is the stopping time of the continuous greedy algorithm and $k = k^{out} + k^{in} \geq 1$.

The first derivative of ρ with respect to T is given by:

$$\frac{d}{dT}\rho(T) = \frac{(Tk + 1)e^{-T} - k\left(1 - e^{-T}\right)}{(Tk + 1)^2}$$

We have $\frac{d}{dT}\rho(T) = 0$ if and only if

$$k\left(T + \frac{1}{k} + 1\right)e^{-T} - k = 0,$$

or, equivalently (since $k \geq 1$):

$$\begin{aligned}
\left(T + 1 + \frac{1}{k}\right)e^{-T} &= 1 \\
-\left(T + 1 + \frac{1}{k}\right)e^{-T-1-\frac{1}{k}} &= -e^{-1-\frac{1}{k}} \\
-T - 1 - \frac{1}{k} &= W\left(-e^{-1-\frac{1}{k}}\right) \\
T &= -1 - \frac{1}{k} - W\left(-e^{-1-\frac{1}{k}}\right),
\end{aligned}$$

where W is the Lambert W function, defined by the equation $z = W(z)e^{W(z)}$ (for a detailed discussion of the Lambert W function, we refer the reader to [19]). The function $W(z)$ has 2 real-valued branches in the range $[-e^{-1}, 0]$. In this range, the upper branch W_0 of W takes values in $[-1, 0]$ and so $W_0(-e^{-1-\frac{1}{k}})$ yields a negative value for T, since $-1 - \frac{1}{k} < -1$. Thus, we restrict ourselves to the lower branch, W_{-1}, which takes values in $[-1, -\infty]$ over this range. The single real-valued critical point of $\rho(T)$ satisfying $T \geq 0$ is thus given by:

$$T_{opt} = -1 - \frac{1}{k} - W_{-1}\left(-e^{-1-\frac{1}{k}}\right). \tag{4.9}$$

Now, we show that T_{opt} is a maximizer of ρ. Let $B(T) = Tk + 1$ and note that $B(T) > 0$ for all $T \geq 0$. Then, we have:

$$\frac{d^2}{dT^2}\rho(T) = \frac{B(T)^2(-B(T)e^{-T}) - 2B(T)k\left(B(T)e^{-T} - k + ke^{-T}\right)}{B(T)^4} \tag{4.10}$$

$$= \frac{-B(T)^2e^{-T} - 2B(T)ke^{-T} + 2k^2 - 2k^2e^{-T}}{B(T)^3} \tag{4.11}$$

$$= \frac{e^{-T}}{B(T)^3}\left[-B(T)^2 - 2B(T)k + 2k^2e^T - 2k^2\right]. \tag{4.12}$$

We now show that $\frac{d^2}{dT^2}\rho(T_{opt}) \leq 0$. Because $B(T_{opt}) > 0$, it suffices to show that:

$$-B(T_{opt})^2 - 2B(T_{opt})k + 2k^2e^{T_{opt}} - 2k^2 \leq 0. \tag{4.13}$$

Let $C = e^{-1-\frac{1}{k}}$. Then, we have $T_{opt} = -1 - \frac{1}{k} - W_{-1}(-C)$, and so

$$\begin{aligned} T_{opt} + 1 + \frac{1}{k} &= -W_{-1}(-C) \\ &= -W_{-1}(-C)e^{W_{-1}(C)}e^{-W_{-1}(C)} \\ &= -(-C)e^{-W_{-1}(C)} \\ &= e^{-1-\frac{1}{k}}e^{-W_{-1}(C)} \\ &= e^{T_{opt}}. \end{aligned}$$

Thus, we have

$$2k^2e^{T_{opt}} = 2k^2\left(T_{opt} + 1 + \frac{1}{k}\right) = 2k\left(T_{opt}k + k + 1\right) = 2B(T_{opt})k + 2k^2. \tag{4.14}$$

Substituting (4.14) in (4.13) we obtain

$$-B(T_{opt})^2 - 2B(T_{opt})k + 2B(T_{opt})k + 2k^2 - 2k^2 \leq 0,$$

which is true for all values of k. Thus, T_{opt} is a local maximum for ρ. Because T_{opt} is the only critical point of $\rho(T)$ for $T > 0$, it follows that T_{opt} is a global maximum of ρ. We note that for all $k = 1$, we have $T_{opt} > 1$. In this case, the optimal value for $T \in [0, 1]$ is given by 1, since T must be non-decreasing on the interval $[0, T_{opt}]$ in order for T_{opt} to be a global maximum. In all other cases, we have $T_{opt} \leq 1$.

Chapter 5

Stochastic probing via contention resolution schemes

The algorithm from previous chapter fails when the submodular objective function is non-monotone. In this chapter we generalize the framework even further by showing that also maximizing a non-negative submodular function can be considered in the probing model. On the way we develop a randomized procedure for transversal matroids which can be used to improve the analysis of [8] for the k-set packing problem.

To handle non-monotone objectives we are making use of *contention resolution schemes* introduced by Chekuri et al. [54]. Contention resolution schemes in the context of stochastic probing already were used by Gupta and Nagarajan [36]. Recently, Feldman et al. [27] presented online version of contention resolution schemes which, on top of applications for online settings, yield good approximations for stochastic probing problem for a broader set of constraints than before — most notably, for inner knapsack constraints and *deadlines*.

This chapter fills the gaps between and merges results from basically four different paper [2, 54, 36, 26]. That is the reason why this paper comes with diverse contributions: we improve the bound on measured greedy algorithm of Feldman et al. [26]; adjust contention resolution schemes to stochastic probing setting in a way in which submodular optimization is be possible; use iterative randomized rounding technique to develop contention resolution schemes; moreover, we revisit the algorithms of Bansal et al. [8].

5.1 Stronger bound for measured greedy algorithm

Mathematical program Recall the mathematical programming relaxation that gives an upper bound on the expected value of the optimal feasible strategy for the related stochastic probing problem:

$$\text{maximize } \left\{ f^+(x \cdot p) \,\middle|\, x \in \mathcal{P}\left(\mathcal{I}^{in}, \mathcal{I}^{out}\right) \right\}. \tag{5.1}$$

Recall Lemma 30 that if OPT is the optimal feasible strategy for the stochastic probing problem in our general setting, then $\mathbb{E}\left[f\left(OPT\right)\right] \leq f^+\left(x^+ \cdot p\right)$, where x^+ is the optimum solution of the program.

However, the framework of Chekuri et al. [54] uses multilinear relaxation F and not f^+. The Lemma below allows us to make a connection. Note that $F(x)$ is exactly equal to $\mathbb{E}\left[F(R(x)\right]$, i.e., it corresponds to sampling each point $e \in E$ independently with probability x_e, while definition of $f^+(x)$ involves the best possible, most likely correlated, distribution of E's subsets. It follows immediately that for any point x we have $f^+(x) \geq F(x)$, and therefore the following Lemma states a stronger lower-bound for the measured greedy algorithm of Feldman et al. [26]. Details are presented in the next paragraph.

Lemma 40. *Let $b \in [0,1]$, let f be a submodular function with multilinear extension F, and let $\mathcal{P}$ be any downward closed polytope. Then, the solution $x \in [0,1]^E$ produced by the measured greedy algorithm satisfies 1) $x \in b \cdot \mathcal{P}$, 2) $F(x) \geq \Big(b \cdot e^{-b} - o(1)\Big) \cdot \max_{y \in \mathcal{P}} f^+(y)$.*

Stronger bound

We now briefly review the measured continuous greedy algorithm of Feldman et al. [26]. The algorithm runs for δ^{-1} discrete time steps, where δ is a suitably chosen, small constant. Let $y(t)$ be the algorithm's current fractional solution at time t. At time t, the algorithm selects vector $I(t) \in \mathcal{P}$ given by $\arg\max_{x \in \mathcal{P}} \sum_{e \in E} x_e \cdot \Big(F\Big(y(t) \vee \mathbf{1}_{\{e\}}\Big) - F(y(t))\Big)$ (where $\vee$ denotes element-wise maximum). Then, it sets $y_e(t+\delta) = y_e(t) + \delta I_e(t) \cdot (1 - y_e(t))$ and continues to time $t + \delta$.

The analysis of Feldman et al. shows that if, at every time step

$$F(y(t+\delta)) - F(y(t)) \geq \delta \cdot \Big[e^{-t} \cdot f(OPT) - F(y(t))\Big] - O\Big(n^3 \delta^2 f(OPT)\Big), \quad (5.2)$$

then we must have $F(y(T)) \geq \Big[Te^{-T} - o(1)\Big] \cdot f(OPT)$. We note that, in fact, this portion of their analysis works even if $f(OPT)$ is replaced by any constant value. Thus, in order to prove our claim, it suffices to derive an analogue of (5.2) in which $f(OPT)$ is replaced by $f^+(x^+)$, where $x^+ = \text{argmax}_{y \in \mathcal{P}} f^+(y)$. The remainder of the proof then follows as in [26].

Lemma 41 below contains the required analogue of (5.2). Hence it implies Lemma 40.

Lemma 41. *For every time $0 \leq t \leq T$*

$$F(y(t+\delta)) - F(y(t)) \geq \delta \cdot \Big[e^{-t} \cdot f^+(x^+) - F(y(t))\Big] - O(n^3\delta^2) \cdot f^+\Big(x^+\Big).$$

To prove the above Lemma we shall require the following additional facts from the analysis of [26].

Lemma 42. *[Lemma 3.3 in [26]] Consider two vectors $x, x' \in [0,1]^E$, such that for every $e \in E$, $|x_e - x'_e| \leq \delta$. Then, $F(x') - F(x) \geq \sum_{e \in E}(x'_e - x_e) \cdot \partial_e F(x) - O(n^3\delta^2) \cdot f(OPT)$.*

Lemma 43. *[Lemma 3.5 in [26]] Consider a vector $x \in [0,1]^E$. Assuming $x_e \leq a$ for every $e \in E$, then for every set $S \subseteq E$, $F(x \vee \mathbf{1}_S) \geq (1-a) f(S)$.*

Lemma 44. *[Lemma 3.6 in [26]] For every time* $0 \leq t \leq T$ *and element* $e \in E$, $y_e(t) \leq 1 - (1-\delta)^{t/\delta} \leq 1 - \exp(-t) + O(\delta)$.

Now we can move to the proof of Lemma **41.**

Proof. Applying Lemma 42 to the solutions $y(t+\delta)$ and $y(t)$, we have

$$\begin{aligned}
&F(y(t+\delta)) - F(y(t)) \\
&\geq \sum_{e\in E} \delta \cdot I_e(t)(1-y(t)) \cdot \partial_j F(y(t)) - O\left(n^3\delta^2\right)) \cdot f(OPT) \qquad (5.3)\\
&= \sum_{e\in E} \delta \cdot I_e(t)(1-y(t)) \cdot \frac{F(y(t)\vee \mathbf{1}_j) - F(y(t))}{1-y(t)} - O(n^3\delta^2)\cdot f(OPT)\\
&= \sum_{e\in E} \delta \cdot I_e(t) \cdot [F(y(t)\vee \mathbf{1}_j) - F(y(t))] - O(n^3\delta^2)\cdot f(OPT)\\
&\geq \sum_{e\in E} \delta \cdot x_e^+ [F(y(t)\vee \mathbf{1}_j) - F(y(t))] - O(n^3\delta^2)\cdot f(OPT) \qquad (5.4)
\end{aligned}$$

where the last inequality follows from our choice of $I(t)$.

Moreover, we have $f^+(x^+) = \sum_{A\subseteq E} \alpha_A f(A)$ for some set of values α_A satisfying $\sum_{A\subseteq E} \alpha_A = 1$ and $\sum_{A\subseteq E: e\in A} \alpha_A = x_e^+$. Thus,

$$\begin{aligned}
&\sum_{e\in E} x_e^+ [F(y(t)\vee \mathbf{1}_j) - F(y(t))] = \sum_{A\subseteq E} \alpha_A \sum_{j\in A} [F(y(t)\vee \mathbf{1}_j) - F(y(t))]\\
&\geq \sum_{A\subseteq E} \alpha_A [F(y(t)\vee \mathbf{1}_A) - F(y(t))] \geq \sum_{A\subseteq E} \alpha_A \left[(e^{-t} - O(\delta))\cdot f(A) - F(y(t))\right]\\
&\qquad = (e^{-t} - O(\delta)) \cdot f^+(x^+) - F(y(t)).
\end{aligned}$$

where the first inequality follows from the fact that F is concave in all positive directions, and the second from Lemmas 43 and 44. Combining this with the inequality (5.4), and noting that $f^+(x^+) \geq f^+(OPT) = f(OPT)$, we finally obtain $F(y(t+\delta)) - F(y(t)) \geq \delta \cdot [e^{-t}\cdot f^+(x^+) - F(y(t))] - O(n^3\delta^2)\cdot f^+(x^+)$. ☐

5.2 Stochastic contention resolution schemes

5.2.1 Contention Resolution Schemes

Consider a ground set of elements E and an down-closed family $\mathcal{I} \subseteq 2^E$ of E's subsets — we call $(E, \mathcal{I})$ an *independence system.* Let $\mathcal{P}(\mathcal{I})$ be the convex hull of characteristic vectors of sets from $\mathcal{I}$. Given $x \in \mathcal{P}(\mathcal{I})$ we define $R(x)$ to be a random set in which every element $e \in E$ is included in $R(x)$ with probability x_e; set $R(x)$ defined like that is used frequently throughout the paper.

Chekuri et al. [54] presented a framework of contention resolution schemes (CR schemes) that allows to maximize non-negative submodular functions for various constraints. The following definition and theorem come from [54].

Definition 45. Let $(E, \mathcal{I})$ be independence system. For $b, c \in [0,1]$, a (b,c)-balanced CR scheme π for $\mathcal{P}(\mathcal{I})$ is a randomized procedure that for every $x \in b\cdot\mathcal{P}(\mathcal{I})$ and $A \subseteq E$, returns a random set $\pi_x(A)$ such that:

always $\pi_x(A) \subseteq A \cap \operatorname{supp}(x)$ and $\pi_x(A) \in \mathcal{I}$,
$\mathbb{P}[e \in \pi_x(A_1)] \geq \mathbb{P}[e \in \pi_x(A_2)]$ whenever $e \in A_1 \subseteq A_2$,
for all $e \in \operatorname{supp}(x)$, $\mathbb{P}[e \in \pi_x(R(x)) | e \in R(x)] \geq c$.

Theorem 46. *Let* $(E, \mathcal{I})$ *be an independence system. Let* $f : 2^E \to \mathbb{R}$ *be a non-negative submodular function with multilinear relaxation* F*, and* x *be a point in* $b \cdot \mathcal{P}(\mathcal{I})$*. Let* π *be a* (b, c)*-balanced CR scheme for* $\mathcal{P}(\mathcal{I})$*, and let* $S = \pi_x(R(x))$*. If* f *is monotone then* $\mathbb{E}[f(S)] \geq c \cdot F(x)$*. Furthermore, there is a function* $\eta_f : 2^E \to 2^E$ *that depends on* f *and can be evaluated in linear time, such that even for* f *non-monotone* $\mathbb{E}[f(\eta_f(S))] \geq c \cdot F(x)$*.*

Function $\eta_f(S)$ represents a *pruning* operation that removes from S some elements. To prune a set S with pruning function η_f, an arbitrary ordering of the elements of E is fixed: for simplicity of notation let $E = \{1, ..., |E|\}$ which gives a natural ordering. Starting with $S^{prun} = \emptyset$ the final set $S^{prun} = \eta_f(S)$ is constructed by going through all elements of E in the given order. When considering an element e, S^{prun} is replaced by $S^{prun} + e$ if $f(S^{prun} + e) - f(S^{prun}) \geq 0$.

Note that a pruning operation like that is not possible to execute in the probing model since we commit to elements. We address this issue in Section 5.2.3, where we show how to perform on-the-fly pruning.

5.2.2 Stochastic contention resolution schemes

To obtain our results we extend the framework of CR schemes into the probing model, and we define a *stochastic contention resolution scheme (stoch-CR scheme)*. Define a polytope

$$\mathcal{P}\left(\mathcal{I}^{in}, \mathcal{I}^{out}\right) = \left\{x \,\middle|\, x \in \mathcal{P}\left(\mathcal{I}^{out}\right), p \cdot x \in \mathcal{P}\left(\mathcal{I}^{in}\right), x \in [0,1]^E\right\}.$$

By $act(S)$ we denote the subset of active elements of set S. Note that event $e \in act(R(x))$ means both that $e \in R(x)$ and that e is active; this event has probability $p_e x_e$.

Definition 47. Let $(E, p, \mathcal{I}^{in}, \mathcal{I}^{out})$ be a stochastic probing problem. For $b, c \in [0, 1]$, a (b, c)-balanced *stoch-CR scheme* $\bar{\pi}$ for a polytope $\mathcal{P}(\mathcal{I}^{in}, \mathcal{I}^{out})$ is a probing strategy that for every $x \in b \cdot \mathcal{P}(\mathcal{I}^{in}, \mathcal{I}^{out})$ and $A \subseteq E$, obeys outer constraints $\mathcal{I}^{out}$, and the returned random set $\bar{\pi}_x(A)$ satisfies the following:
$\bar{\pi}_x(A)$ consists only of active elements,
$\bar{\pi}_x(A) \subseteq A \cap \operatorname{supp}(x)$ and $\bar{\pi}_x(A) \in \mathcal{I}^{in}$,
$\mathbb{P}[e \in \bar{\pi}_x(R(x)) | e \in act(R(x))] \geq c$,
$\mathbb{P}[e \in \bar{\pi}_x(A_1)] \geq \mathbb{P}[e \in \bar{\pi}_x(A_2)]$ whenever $e \in A_1 \subseteq A_2$.

5.2.3 Optimization using stochastic resolution schemes

At the beginning of this Chapter, in Section 5.1, we present a mathematical program that models our problem. Solving the program, getting x^+, and running the stoch-CR scheme $\bar{\pi}_{x^+}$ on $R(x^+)$ constitute the algorithm from the below Theorem.

Theorem 48. *Consider a stochastic probing problem* $\left(E,p,\mathcal{I}^{in},\mathcal{I}^{out}\right)$*, where we need to maximize a non-negative submodular function* $f : 2^E \mapsto \mathbb{R}_{\geq 0}$*. If there exists a* (b,c)*-balanced stoch-CR scheme* $\bar{\pi}$ *for* $\mathcal{P}\left(\mathcal{I}^{in},\mathcal{I}^{out}\right)$*, then there exists a probing strategy whose expected outcome is at least* $c\left(b \cdot e^{-b} - o\left(1\right)\right) \cdot \mathbb{E}\left[f\left(OPT\right)\right]$*.*

There exists a $\left(b, \left(\frac{1-e^{-b}}{b}\right)^k\right)$-balanced scheme for intersection of k matroids [54], and there exists a $\left(b, (1-b)^k\right)$-balanced ordered scheme for intersection of k matroids [16, 27], so the above Theorem plus Lemma 51 together imply Theorem 3.

The following Lemma is implied by Theorem 46 — it follows just from the fact that set *act* $\left(R\left(x\right)\right)$ is distributed exactly as $R\left(x \cdot p\right)$.

Lemma 49. *Let* $\left(E,p,\mathcal{I}^{in},\mathcal{I}^{out}\right)$ *be a probing problem. Let* $f : 2^E \mapsto \mathbb{R}_{\geq 0}$ *be a non-negative submodular function with multilinear relaxation* F*, and* x *be a point in* $b \cdot \mathcal{P}\left(\mathcal{I}^{in},\mathcal{I}^{out}\right)$ *for* $b \in [0,1]$*. Let* $\bar{\pi}_x$ *be a* (b,c)*-balanced stoch-CR scheme for* $\mathcal{P}\left(\mathcal{I}^{in},\mathcal{I}^{out}\right)$*. Let* $\bar{\pi}_x\left(R\left(x\right)\right)$ *be the output of the CR scheme, and let* $\eta_f\left(\bar{\pi}_x\left(R\left(x\right)\right)\right)$ *be a pruned subset of* $\bar{\pi}_x\left(R\left(x\right)\right)$*. It holds that*

$$\mathbb{E}\left[f\left(\eta_f\left(\bar{\pi}_x\left(R\left(x\right)\right)\right)\right)\right] \geq c \cdot F\left(x \cdot p\right).$$

However, we cannot apply yet the above Lemma to reason about a probing strategy, because here the pruning η_f of S is done after the process. In the probing model we commit to an element once we successfully probe it, and therefore we cannot do the pruning operation after the execution of a stoch-CR scheme. However, since a probing strategy inherently includes elements one-by-one, we can naturally add to any stoch-CR scheme the pruning operation done on the fly. The idea is to simulate the probes of elements that would be rejected by the pruning criterion. To simulate a probe of e means **not** to probe e and to toss a coin with probability p_e — in case of success to behave afterwards as if e was indeed taken into the solution, and in case of failure to behave as if e was not taken. During the execution of a stoch-CR scheme $\bar{\pi}_x$ we construct two sets: S^{prun} consists of elements successfully probed, and S^{virt} consists of elements whose simulation was successful. If in a step we want to probe an element e such that $f\left(S^{prun} + S^{virt} + e\right) - f\left(S^{prun} + S^{virt}\right) < 0$, then we simulate the probe of e and if successful $S^{virt} \leftarrow S^{virt} + e$; otherwise we really probe e and $S^{prun} \leftarrow S^{prun} + e$ if successful. We can see that at any step, it holds that $S^{prun} = \eta_f\left(S^{prun} + S^{virt}\right)$. Also, the final random set $S^{prun} + S^{virt}$ is distributed exactly as $\bar{\pi}_x\left(R\left(x\right)\right)$. Hence, the outcome of such a probing strategy is $\mathbb{E}\left[f\left(S^{prun}\right)\right] = \mathbb{E}\left[f\left(\eta_f\left(S^{prun} + S^{virt}\right)\right)\right] = \mathbb{E}\left[f\left(\eta_f\left(\bar{\pi}_x\left(R\left(x\right)\right)\right)\right)\right] \geq c \cdot F\left(x \cdot p\right)$, where the inequality comes from Lemma 49. Thus we have proven what follows.

Lemma 50. *Let* f*,*x*,*$\bar{\pi}_x$ *be as in Lemma 49. There exists a probing strategy whose expected* outcome *is* $\mathbb{E}\left[f\left(\eta_f\left(\bar{\pi}_x\left(R\left(x\right)\right)\right)\right)\right] \geq c \cdot F\left(x \cdot p\right)$*.*

Now we can finish the proof of Theorem 48. Consider the following algorithm. First, use Lemma 40 to find a point x^* such that

$$F\left(x^* \cdot p\right) \geq \left(b \cdot e^{-b} - o\left(1\right)\right) \cdot \max_{x \in \mathcal{P}\left(\mathcal{I}^{in},\mathcal{I}^{out}\right)} f^+\left(x \cdot p\right).$$

Second, run the probing strategy based on a stoch-CR scheme $\bar{\pi}_{x^*}$ as described in Lemma 50. This yields, together with Lemma 30, that the outcome of such a probing strategy is

$$\mathbb{E}\left[f\left(\eta_f\left(\bar{\pi}_{x^*}\left(R\left(x^*\right)\right)\right)\right)\right] \geq c \cdot F\left(x^* \cdot p\right) \geq c \cdot \left(b \cdot e^{-b} - o\left(1\right)\right) \cdot \max_{x \in \mathcal{P}(\mathcal{I}^{in}, \mathcal{I}^{out})} f^+\left(x \cdot p\right)$$
$$\geq c \cdot \left(b \cdot e^{-b} - o\left(1\right)\right) \cdot \mathbb{E}\left[f\left(OPT\right)\right].$$

In [54] also an alternative approach than pruning was used. They defined a *strict* contention resolution scheme where the approximation guarantee $\mathbb{P}\left[e \in \pi_x\left(R\left(x\right)\right) | e \in R\left(x\right)\right] \geq c$ holds with equality rather than inequality. Since the pruning operation depends on an objective function, resigning from it allows for the algorithm to be used in maximizing many submodular functions at the same time. In our stochastic setting we can also skip the pruning operation if we have a strict scheme. No proof of this fact is needed, since we can directly use an appropriate analog of Lemma 49 (Theorem 4.1 from 46).

5.2.4 Stochastic probing for various constraints

Gupta and Nagarajan [36] introduced a notion of *ordered CR scheme*, for which there exists a (possibly random) permutation σ of E, so that for each A the set $\pi_x\left(A\right)$ is the maximal independent subset of $\mathcal{I}$ obtained by considering elements in the order of σ. Ordered scheme are required to implement probing strategies, because of the commitment to the elements. CR schemes exist for various types of constraints, e.g., matroids, sparse packing integer programs, constant number of knapsacks, unsplittable flow on trees (UFP), k-systems (including intersection of k matroids, and k-matchoids). Ordered schemes exist for k-systems and UFP on trees. See Theorem 4 in [36] for a listing with exact parameters.

The following Lemma is based on Lemma 1.6 from [54]. The proof basically carries over, the only thing we have to do is to again incorporate probes' simulations as in the proof of Lemma 50. Theorem 3.4 from [36] yields a similar result but would imply a $(b, c_{out} + c_{in} - 1)$-balanced stoch-CR scheme. Thus the below Lemma can be considered as a strengthening of Theorem 3.4 from [36], because an FKG inequality is used in the proof instead of a union-bound.

Lemma 51. *Consider a probing problem* $\left(E, p, \mathcal{I}^{in}, \mathcal{I}^{out}\right)$. *Suppose we have a* (b, c_{out})*-balanced CR-scheme* π^{out} *for* $\mathcal{P}\left(\mathcal{I}^{out}\right)$*, and a* (b, c_{in})*-balanced ordered CR scheme* π^{in} *for* $\mathcal{P}\left(\mathcal{I}^{in}\right)$*. Then there exists a* $(b, c_{out} \cdot c_{in})$*-balanced stoch-CR scheme for* $\mathcal{P}\left(\mathcal{I}^{in}, \mathcal{I}^{out}\right)$*.*

Proof. First let us recall a Lemma from [54].

Lemma[1.6 from [54]]. *Let* $\mathcal{I} = \bigcap_i \mathcal{I}_i$ *and* $\mathcal{P}_{\mathcal{I}} = \bigcap_i \mathcal{P}_i$. *Suppose each* $\mathcal{P}_{\mathcal{I}_i}$ *has a monotone* (b, c_i)*-balanced CR scheme. Then* $\mathcal{P}_{\mathcal{I}}$ *has a monotone* $(b, \prod_i c_i)$*-balanced CR scheme defined as* $\pi_x(A) = \bigcap_i \pi_x^i(A)$ *for* $A \subseteq N, x \in b\mathcal{P}_{\mathcal{I}}$*.*

Suppose we have a CR scheme π^{out} for $\mathcal{P}(\mathcal{I}^{out})$ and an ordered CR scheme π^{in} for $\mathcal{P}(\mathcal{I}^{in})$. We would like to define the stochastic contention resolution scheme for

$\mathcal{P}\left(\mathcal{I}^{in}, \mathcal{I}^{out}\right)$ just as in the Lemma above, i.e., as $\pi_x(A) = \pi_x^{out}(A) \cap \pi_{p\cdot x}^{in}(act\,(A))$. However, we cannot just simply run π_x^{out} on A, and then $\pi_{p\cdot x}^{in}$ on A again, and take the intersection, because that does not constitute a feasible probing strategy. Once again, we need to make use of simulated probes to get a stoch-CR scheme that will have a probability distribution of $\pi_x^{out}(A) \cap \pi_{p\cdot x}^{in}(act\,(A))$. How to implement such a strategy? We first run $\pi_x^{out}(A)$ on the set A. Later we use $\pi_{p\cdot x}^{in}$ to scan elements of A in the order given from the definition of an ordered scheme. If π_x^{in} considers element e such that $e \in A \setminus \pi_x^{out}(A)$, then we simulate the probe of e; if the $e \in \pi_x^{out}(A)$, then we just probe it. Therefore, the CR-scheme $\pi_{p\cdot x}^{in}$ works in fact on the set $act\,(\pi_x^{out}(A) + (A \setminus \pi_x^{out}(A))^{virt})$, where $(A \setminus \pi_x^{out}(A))^{virt}$ represents simulated probes of elements in $A \setminus \pi_x^{out}(A)$. Assuming $e \in \pi_x^{out}(A)$, it is easy to see that elements in $act\,(A)$ and elements in $act\,(\pi_x^{out}(A) + (A \setminus \pi_x^{out}(A))^{virt})$ have the same probability distribution. Therefore,

$$\mathbb{P}\left[e \in \pi_{p\cdot x}^{in}(act\,(A))\right] = \mathbb{P}\left[e \in \pi_{p\cdot x}^{in}(act\left(\pi_x^{out}(A) \cup (A \setminus \pi_x^{out}(A))^{virt}\right)) \middle| \pi_x^{out}(A), e \in \pi_x^{out}(A)\right]. \quad (5.5)$$

And the RHS corresponds to a second phase of a feasible probing strategy. Thus we have:

$$\begin{aligned}
&\mathbb{P}\left[e \in \pi_x(A)\right] \\
&= \mathbb{P}\left[e \in \pi_x^{out}(A) \cap \pi_{p\cdot x}^{in}(\pi_x^{out}(A) \cup (A \setminus \pi_x^{out}(A))^{virt})\right] \\
&= \mathbb{E}_{\pi_x^{out}}\left[\chi\left[e \in \pi_x^{out}(A)\right] \cdot \mathbb{E}_{\pi_{p\cdot x}^{in}}\left[\chi\left[e \in \pi_{p\cdot x}^{in}(act\left(\pi_x^{out}(A) \cup (A \setminus \pi_x^{out}(A))^{virt}\right))\right] \middle| \pi_x^{out}(A), e \in \pi_x^{out}(A)\right]\right].
\end{aligned}$$

Now just note that

$$\begin{aligned}
&\mathbb{E}_{\pi_{p\cdot x}^{in}}\left[\chi\left[e \in \pi_{p\cdot x}^{in}(\pi_x^{out}(A) \cup (A \setminus \pi_x^{out}(A))^{virt})\right] \middle| \pi_x^{out}(A), e \in \pi_x^{out}(A)\right] = \\
&\qquad \mathbb{P}\left[e \in \pi_{p\cdot x}^{in}(act\left(\pi_x^{out}(A) \cup (A \setminus \pi_x^{out}(A))^{virt}\right)) \middle| \pi_x^{out}(A), e \in \pi_x^{out}(A)\right] = \\
&\qquad\qquad \mathbb{P}\left[e \in \pi_{p\cdot x}^{in}(act\,(A))\right], \quad (5.6)
\end{aligned}$$

from line (5.5), and so we can simplify the previous expression to

$$\begin{aligned}
\mathbb{P}\left[e \in \pi_x(A)\right] &= \mathbb{E}_{\pi_x^{out}}\left[\chi\left[e \in \pi_x^{out}(A)\right] \cdot \mathbb{P}\left[e \in \pi_{p\cdot x}^{in}(act\,(A))\right]\right] \\
&= \mathbb{E}_{\pi_x^{out}}\left[\chi\left[e \in \pi_x^{out}(A)\right]\right] \cdot \mathbb{P}\left[e \in \pi_{p\cdot x}^{in}(act\,(A))\right] \\
&= \mathbb{P}\left[e \in \pi_x^{out}(A)\right] \cdot \mathbb{P}\left[e \in \pi_{p\cdot x}^{in}(act\,(A))\right].
\end{aligned}$$

Now the analysis just follows the lines of Lemma 1.6 from [54].We plug $R(x)$ for A, and apply expectation on $R(x)$ conditioned on $e \in R(x)$ to get:

$$\mathbb{E}_{R(x)}\left[\mathbb{P}\left[e \in \pi_x(R\,())\right] | e \in R(x)\right] = \mathbb{E}_{R(x)}\left[\mathbb{P}\left[e \in \pi_x^{out}(R(x))\right] \cdot \mathbb{P}\left[e \in \pi_{p\cdot x}^{in}(act\,(R(x)))\right] | e \in R(x)\right].$$

From the fact that both π^{out}, π^{in} are monotone, and $act\,(R)\,(x)$ is an increasing function of $R(x)$ we get that $\mathbb{P}\left[e \in \pi_x^{out}(R(x))\right]$ and also $\mathbb{P}\left[e \in \pi_{p\cdot x}^{in}(act\,(R(x)))\right]$ are increasing functions of $R(x)$. Now we shall apply the FKG inequality [4] which states that for two monotone functions $f, g : 2^E \mapsto \mathbb{R}$ we have $\mathbb{E}[f(R(x)) \cdot g(R(x))] \geq \mathbb{E}[f(R(x))] \cdot \mathbb{E}[g(R(x))]$ — this inequality holds because the distribution $R(x)$ is log-supermodular. Thus applying FKG inequality gives us that

$$\begin{aligned}
&\mathbb{E}_{R(x)}\left[\mathbb{P}\left[e \in \pi_x(R(x))\right] | e \in R(x)\right] \\
=&\mathbb{E}_{R(x)}\left[\mathbb{P}\left[e \in \pi_x^{out}(R(x))\right] \cdot \mathbb{P}\left[e \in \pi_{p\cdot x}^{in}(act\,(R(x)))\right] | e \in R(x)\right] \\
\geq&\mathbb{E}_{R(x)}\left[\mathbb{P}\left[e \in \pi_x^{out}(R(x))\right] | e \in R(x)\right] \cdot \mathbb{E}_{R(x)}\left[\mathbb{P}\left[e \in \pi_{p\cdot x}^{in}(act\,(R\,(x)))\right] | e \in R(x)\right] \\
\geq& b \cdot c_{out} \cdot \mathbb{E}_{R(x)}\left[\mathbb{P}\left[e \in \pi_{p\cdot x}^{in}(act\,(R(x)))\right] | e \in R(x)\right].
\end{aligned}$$

Now applying also expectation on $act\,(R(x))$, we get finally

$$\begin{aligned}
&\mathbb{E}_{R(x),act(R(x))}\left[\mathbb{P}\left[e \in \pi_x(R(x))\right] | e \in R(x)\right] \\
&\quad \geq c_{out} \cdot \mathbb{E}_{R(x),act(R(x))}\left[\mathbb{P}\left[e \in \pi_{p\cdot x}^{in}(act\,(R(x)))\right] | e \in R(x)\right] \geq p_e \cdot c_{out} \cdot c_{in}. \qquad (5.7)
\end{aligned}$$

Also, directly from equation $\mathbb{P}[e \in \pi_x(A)] = \mathbb{P}\left[e \in \pi_x^{out}(A)\right] \cdot \mathbb{P}\left[e \in \pi_{p\cdot x}^{in}(act\,(A))\right]$ we get the monotonicity of the stoch CR-scheme π_x, since both π_x^{out} and $\pi_{p\cdot x}^{in}\,(act\,(\cdot))$ are monotone. ☐

In light of the above Lemma, one can question Definition 47 of a stoch CR-scheme — why do we need to define it at all, if we can just be using two separate classic CR schemes π^{out}, π^{in}. The reason is that there may exist stoch-CR schemes that are not convolutions of two deterministic schemes, and that yield better approximations than corresponding convoluted ones. Such a stoch-CR scheme is the subject of Section 5.3.

In a recent paper, Feldman et al. [27] presented a variant of online contention resolution schemes. They enriched the set of constraints possible to use in the stochastic probing problem — previously inner knapsack constraints were not possible to incorporate, as well as deadlines (element e can be taken only first d_e steps) for weighted settings. Their results can be extended to monotone submodular settings by making use of a stronger bound for continuous greedy algorithm [13] presented in [2]. The stronger bound for measured greedy algorithm — which works for non-monotone functions — that we give in this paper can also be used in [27] enhancing their result by the possibility of handling non-monotone functions as well.

5.3 Resolution schemes for transversal matroids

Stochastic probing on intersection of transversal matroids We improve upon the approximation described in the previous Section, if we assume the constraints are intersections of transversal matroids. We do it by developing a new stoch-CR scheme. This scheme is direct in the sense that we do not construct it by applying first a scheme for outer and then for inner constraints, as in Lemma 51.

Lemma 52. *There exists a* $\left(b, \frac{1}{1+b\cdot(k^{in}+k^{out})}\right)$*-balanced stoch-CR scheme when constraints are intersections of* k^{in} *inner and* k^{out} *outer transversal matroids.*

For $k^{in} = k^{out} = 1$ the above Lemma together with Theorem 48 give approximation factor of 0.15, so a modest improvement over 0.13, but it gets significantly better for larger values of $k^{in} + k^{out}$. With $k = k^{in} + k^{out}$ we plug $b = \frac{2}{\sqrt{1+4k}+1}$ and we use Theorem 48 to conclude the following Theorem.

Theorem 53. *For maximizing non-negative submodular function in the probring model with k^{in} inner and k^{out} outer transversal matroids there exists an algorithm with approximation ratio of*

$$\frac{1}{k + \sqrt{k + \frac{1}{4}} + \frac{1}{2}} \left(1 - \Theta\left(\frac{1}{\sqrt{k}}\right)\right).$$

There are further applications of the techniques used in Lemma 52.

Regular CR scheme for transversal matroids When $k^{in} = 0$ this scheme yields $\left(b, \frac{1}{1+b\cdot k}\right)$-balanced CR scheme for deterministic setting. So far only a $\left(b, (1-b)^k\right)$-balanced *ordered* scheme and a $\left(b, \left(\left(1 - e^b\right)/b\right)^k\right)$-balanced scheme were known [54]; see section 5.2.4 for the definition of ordered scheme. Our scheme can be seen as an improvement when one looks at the $\max_b (b \cdot c)$ — first one yields $\frac{1}{e(k+1)}$, second $\frac{2}{e(k+1)}$(for k big[1]), and we get $\frac{1}{k+1}$.

5.3.1 The scheme

In this section we prove Lemma 52. We assume we have only one inner matroid $\mathcal{M}^{in}$ to keep the presentation simple. Also, we shall present a $\left(1, \frac{1}{2}\right)$-balanced CR scheme, instead of $\left(b, \frac{1}{b+1}\right)$. In the Section 5.3.2 we present a full scheme with arbitrary b, k^{in}, and k^{out}.

Our stoch-CR scheme on the input is given a point x such that $p \cdot x \in \mathcal{P}\left(\mathcal{M}^{in}\right)$ and a set $A \subseteq E$, and on the output it returns set $\bar{\pi}_x(A)$. The procedure is divided in two phases. First, the preprocessing phase, depends only on the point x. Second, the random selection phase, depends on the set $A \subseteq E$ and the outcome of the preprocessing phase.

Preprocessing

In what follows we shall write superscripts indicating the time in which we are in the process.

We start by finding the support $\mathcal{B}^0$ of vector $p{\cdot}x \in \mathcal{P}\left(\mathcal{M}^{in}\right)$, i.e., $p{\cdot}x = \sum_i \beta_i {\cdot} \mathbf{1}_{B_i^0}$. For every two sets $B, A \in \mathcal{B}^0$ we find a mapping $\phi^0[B, A] : B \to A \cup \{\perp\}$, which we call *transversal mapping.* This mapping will satisfy three properties.

Property 2. For each $a \in A$ there is at most one $b \in B$ for which $\phi^0[B, A](b) = a$.

Property 3. For $b \in B \setminus A$, if $\phi^0[B, A](b) = \perp$, then $A + b \in \mathcal{I}$, otherwise $A - \phi^0[B, A](b) + b \in \mathcal{I}$.

[1]The precise value is smaller than $\frac{1}{k+1}$ starting $k \geq 4$.

Note that unlike in standard exchange properties of matroids, we do not require that $\phi^0[B,A](b)=b$, if $b\in A\cap B$. Property 3 will be presented in a moment. Once we find the family ϕ^0 of transversal mappings, for each element $e\in E$ we choose one set among $B_i^0 : e\in B_i^0$ with probability $\beta_i/p_e x_e$; since $\sum_{B_i^0:e\in B_i^0}\beta_i = p_e x_e$ this is properly defined. Denote by $c(e)$ the index of the chosen set, and call *e-critical* the set $B_{c(e)}^t$ for any t (note that $c(e)$ is fixed throughout the process). We concisely denote indices of critical sets by $\mathcal{C}=(c(e))_{e\in E}$. For each element e we define $\Gamma^0(e)=\left\{f \,\middle|\, f\neq e\wedge \phi^0\left[B_{c(f)}^0, B_{c(e)}^0\right](f)=e\right\}$ — the *blocking set* of e. The Lemma below, basically speaking, follows from Property 1.

Lemma 54. *If $p\cdot x\in\mathcal{P}(\mathcal{M}^{in})$, then for any element e, it holds that*

$$\mathbb{E}_{\mathcal{C},R(x)}\left[\sum_{f\in\Gamma^0(e)} p_f\cdot\chi\left[f\in R(x)\right]\right]\leq 1,$$

where the expectation is over $R(x)$ and the choice of critical sets $\mathcal{C}$; here $\chi[\mathcal{E}]$ is a 0-1 indicator of random event $\mathcal{E}$.

Proof. In what follows let us skip writing 0 in the superscript of bases B_i^0, mappings ϕ^0, and set $\Gamma^0(e)$. Let us condition for now on the critical set $B_{c(e)}$ of element e. For f to belong to $\Gamma(e)$ it has to be the case that $\phi\left[B_{c(f)},B_{c(e)}\right](f)=e$. Therefore

$$\sum_{f\in\Gamma(e)} p_f\cdot\chi\left[f\in R(x)\right]=\sum_{f\in E\setminus\{e\}} p_f\cdot\chi\left[f\in R(x)\right]\cdot\left(\sum_{i:\phi\left[B_i,B_{c(e)}\right](f)=e}\chi\left[B_i \text{ is } f\text{-critical}\right]\right),$$

and by changing the order of summation it is equal to

$$\sum_i \sum_{f\in B_i\setminus e:\phi\left[B_i,B_{c(e)}\right](f)=e} p_f\cdot\chi\left[f\in R(x)\right]\cdot\chi\left[B_i \text{ is } f\text{-critical}\right].$$

Consider f such that $f\in B_i\setminus e : \phi\left[B_i,B_{c(e)}\right](f)=e$. Since $\chi\left[f\in R(x)\right]$ and $c(f)$ (the index critical set of f) are independent, and $\mathbb{E}\left[\chi\left[f\in R(x)\right]\right]=x_f$ and $\mathbb{P}\left[B_i \text{ is } f\text{-critical}\middle| B_{c(e)}\right]=\mathbb{P}\left[i=c(f)\middle| B_{c(e)}\right]=\frac{\beta_i}{p_f\cdot x_f}$, we get that

$$\mathbb{E}\left[p_f\cdot\chi\left[f\in R(x)\right]\cdot\chi\left[B_i^j \text{ is } f\text{-critical}\right]\middle| B_{c(e)}\right]=p_f x_f\cdot\frac{\beta_i}{p_f x_f}=\beta_i,$$

and hence

$$\mathbb{E}\left[\sum_{f\in\Gamma(e)} p_f\cdot\chi\left[f\in R(x)\right]\middle| B_{c(e)}\right]=\sum_i\sum_{f\in B_i\setminus e:\phi\left[B_i,B_{c(e)}\right](f)=e}\beta_i^j\leq\sum_i\beta_i^j=1,$$

where the inequality follows from the fact that for each B_i there can be at most one element $f\in B_i$ such that $\phi\left[B_i,B_{c(e)}\right](f)=e$. □

Algorithm 3 Stoch-CR scheme $\bar{\pi}_x(A)$

find support $\mathcal{B}^0$ of $p \cdot x$ in $\mathcal{M}^{in}$ and family ϕ^0; choose critical sets $\mathcal{C}$
remove from A all $e : x_e = 0$; mark all $e \in A$ as available
$S \leftarrow \emptyset$
while there are still available elements in A **do**
  pick element e uniformly at random from A
  if e is available **then**
    probe e
    if probe of e successful **then**
      $S \leftarrow S \cup \{e\}$
      for each set B_i^t of support $\mathcal{B}^t$ **do**
        $B_i^t \leftarrow B_i^t + e$
    call e unavailable
  else simulate the probe of e
  if probe or simulation was successful **then**
    for each set B_i^t of support $\mathcal{B}^t$ **do**
      $f \leftarrow \phi\left[B_{c(e)}^t, B_{c(f)}^t\right](e)$
      if $f \neq e$ **then** $B_{c(f)}^t \leftarrow B_{c(f)}^t - f$ and call f unavailable
  compute the family ϕ^{t+1}
  for each i **do** $B_i^{t+1} \leftarrow B_i^t$
  $t \leftarrow t + 1$;
return S as $\bar{\pi}_x(A)$

Random selection procedure

The whole stoch-CR scheme is presented in Algorithm 3. During the algorithm we modify sets of support $\mathcal{B}^t$ after each step, but we keep the weights β_i unchanged. We preserve an invariant that each B_i^t from $\mathcal{B}^t$ is an independent set of matroid $\mathcal{M}^{in}$. At the end of the algorithm the set $\bar{\pi}_x(A)$ belongs to every set $B_i^t \in \mathcal{B}^t$. Hence, the final set $\bar{\pi}_x(A)$ is independent in every matroid.

Now we define Property 3 of transversal mappings. Suppose that in the first step we update the support $\mathcal{B}^0$ according to the for loop in line 15, and we obtain $\mathcal{B}^1$. Different support $\mathcal{B}^1$ requires a different family of mappings ϕ^1, and so in step 2, the elements that can block e are $\Gamma^1(e)$. If it happens that $\Gamma^1(e) \neq \Gamma^0(e)$, then we cannot show the monotonicity property of stoch-CR scheme. However, we can require from the transversal mappings to keep the blocking sets $\Gamma^t(e)$ unchanged, as long as e is available. In Section 5.3.1.1 we show how to find such a family of transversal mappings ϕ^0 and how to construct ϕ^{t+1} given ϕ^t.

Property 4. Let ϕ^t be a family of transversal mappings for $\mathcal{B}^t$. Suppose we update the support $\mathcal{B}^t$ and obtain $\mathcal{B}^{t+1}$. Then we can find a family ϕ^{t+1} of transversal mappings such that $\Gamma^t(e) = \Gamma^{t+1}(e)$ for any element e that is still available after step t.

Analysis

First, an explanation. We allow to pick in line 5 elements that we have once probed and simulate their probe. This guarantees that the probability that an available element is blocked is equal for every step. Otherwise, again, we would not be able to guarantee the monotonicity.

In the analysis we deploy martingale theory. In particular Doob's Stopping Theorem which states that if a martingale $\left(Z^t\right)_{t\geq 0}$ and stopping time τ are both "well-behaving", then $\mathbb{E}\left[Z^\tau\right] = \mathbb{E}\left[Z^0\right]$.

The random process executed in the while loop depends on the critical sets chosen in the preprocessing phase. Therefore, when we analyze the random process we condition on the choice of critical sets $\mathcal{C}$.

We say e is *available,* if it is still possible to probe, i.e., it is not yet blocked, and it was not yet probed. Define $X_e = [[e \in A]]$; in Iverson notation $[[false]] = 0 = 1 - [[true]]$.

Let Y_e^t for $t = 0, 1, ...,$ be a random variable indicating if e is still available after step t. Initially $Y_e^0 = X_e$. Let P_e^t be a random variable indicating if e was probed in one of steps $0, 1, ...,$ or t; we have $P_e^0 = 0$ for all e. Variable $P_e^{t+1} - P_e^t$ indicates if e was probed at step $t+1$. Given the information $\mathcal{F}^t$ about the process up to step t, the probability of this event is

$$\mathbb{E}\left[P_e^{t+1} - P_e^t \middle| \mathcal{F}^t, \mathcal{C}\right] = \frac{Y_e^t}{|A|},$$

because if element e is still available after step t (i.e., $Y_e^t = 1$), then with probability $\frac{1}{|A|}$ we choose it in line 5, and otherwise (i.e. $Y_e^t = 0$) we cannot probe it.

Variable $Y_e^t - Y_e^{t+1}$ indicates whether element e stopped being available at step $t+1$, i.e., we either have picked it in line 5 and probed (with probability $\frac{Y_e^t}{|A|}$), or some $f \in \Gamma^t(e)$ has blocked e in line 17 (with probability $\frac{Y_e^t}{|A|} \cdot \sum_{f\in\Gamma^t(e)} p_f X_f$). So in total

$$\mathbb{E}\left[Y_e^t - Y_e^{t+1} \middle| \mathcal{F}^t, \mathcal{C}\right] = \frac{Y_e^t}{|A|}\left(1 + \sum_{f\in\Gamma^t(e)} p_f X_f\right),$$

and we can say that

$$\mathbb{E}\left[\left(P_e^{t+1} - P_e^t\right) \cdot \left(1 + \sum_{f\in\Gamma^t(e)} p_f X_f\right) - \left(Y_e^t - Y_e^{t+1}\right) \middle| \mathcal{F}^t, \mathcal{C}\right] = 0.$$

This means that the sequence $\left(\left(1 + \sum_{f\in\Gamma^t(e)} p_f X_f\right) \cdot P_e^t + Y_e^t\right)_{t\geq 0}$ is a martingale. Let $\tau = \min\left\{t \,\middle|\, Y_e^t = 0\right\}$ be the step in which edge e became unavailable. It is clear that τ is a stopping time. Thus from Doob's Stopping Theorem we get that

$$\mathbb{E}_\tau\left[\left(1 + \sum_{f\in\Gamma^\tau(e)} p_f X_f\right) \cdot P_e^\tau + Y_e^\tau \middle| \mathcal{C}\right] = \mathbb{E}\left[\left(1 + \sum_{f\in\Gamma^0(e)} p_f X_f\right) \cdot P_e^0 + Y_e^0 \middle| \mathcal{C}\right],$$

and this is equal to X_e, because $P_e^0 = 0$ and $Y_e^0 = X_e$. We have $Y_e^\tau = 0$, and expression $1 + \sum_{f\in\Gamma^\tau(e)} p_f X_f$ is in fact equal to $1 + \sum_{f\in\Gamma^0(e)} p_f X_f$ (Property 3 of the transversal mapping, as e was available before step τ), which depends solely on $\mathcal{C}$ and A. Hence

$$\left(1+\sum_{f\in\Gamma^0(e)} p_f X_f\right)\cdot \mathbb{E}_\tau\left[P_e^\tau\middle|\mathcal{C}\right] = X_e.$$

Now just note that $\mathbb{E}_\tau\left[P_e^\tau\middle|\mathcal{C}\right]$ is exactly the probability that e is probed, so we conclude that

$$\mathbb{P}\left[e\in\bar{\pi}_x(A)\middle|\mathcal{C}\right] = p_e\cdot\mathbb{E}_\tau\left[P_e^\tau\middle|\mathcal{C}\right] = p_e X_e \Big/ \left(1+\sum_{f\in\Gamma^0(A)} p_f X_f\right).$$

Monotonicity Set $\Gamma^0(e)$ does not depend on A, but only on the vector $p\cdot x$ and $\mathcal{C}$, so for $A_1\subseteq A_2$ we have $\sum_{f\in\Gamma^0(A)} p_f\cdot[[f\in A_1]]\leq\sum_{f\in\Gamma^0(A)} p_f\cdot[[f\in A_2]]$.

Approximation guarantee In the identity

$$\mathbb{P}\left[e\in\bar{\pi}_x(A)\middle|\mathcal{C}\right] = p_e X_e \Big/ \left(1+\sum_{f\in\Gamma^0(A)} p_f X_f\right)$$

we place random set $R(x)$ instead of A; now $X_f=\chi\left[f\in R(x)\right]$ is a random variable. Let us condition on $e\in R(x)$, take expected value $\mathbb{E}_{\mathcal{C},R(x)}\left[\cdot\middle| e\in R(x)\right]$, and apply Jensen's inequality to convex function $x\mapsto\frac{1}{x}$ to get:

$$\begin{aligned}\mathbb{P}\left[e\in\bar{\pi}_x(R(x))\middle| e\in R(x)\right] &= \mathbb{E}_{\mathcal{C},R(x)}\left[\mathbb{P}\left[e\in\bar{\pi}_x(R(x))\middle|\mathcal{C},R(x)\right]\middle| e\in R(x)\right]\\ &= \mathbb{E}_{\mathcal{C},R(x)}\left[p_e X_e \Big/ \left(1+\sum_{f\in\Gamma^0(e)} p_f X_f\right)\middle| e\in R(x)\right]\\ &\geq p_e \Big/ \mathbb{E}_{\mathcal{C},R(x)}\left[1+\sum_{f\in\Gamma^0(e)} p_f X_f\middle| e\in R(x)\right].\end{aligned}$$

Since $\mathbb{E}_{\mathcal{C},R(x)}\left[\sum_{f\in\Gamma^0(e)} p_f X_f\middle| e\in R(x)\right]\leq 1$ from Lemma 54, we conclude that

$$\mathbb{P}\left[e\in\bar{\pi}_x(R(x))\middle| e\in R(x)\right]\geq\frac{p_e}{2},$$

and therefore $\mathbb{P}\left[e\in\bar{\pi}_x(R(x))\middle| e\in \mathit{act}(R(x))\right]\geq\frac{1}{2}$, which is exactly Property 3 from the definition of stoch-CR scheme.

5.3.1.1 Transversal mappings

Recall the definition of a transversal matroid.

Definition 55. Consider a bipartite graph $(E\cup V,\subseteq E\times V)$. Let $\mathcal{I}$ be a family of all subsets S of E such that there exists an injection from S to V. Then $M=(E,\mathcal{I})$ is a matroid, called *transversal* matroid.

We assume we know the graph $(E \cup V, \subseteq E \times V)$ of the matroid.

Let $\mathcal{B}^0$ be the initial support. Let $A \in \mathcal{B}^0$ be an independent set. From the definition of the transversal matroid, there exists an injection $v^A : A \to V$; we shall say that $a \in A$ is *matched to* $v^A(a)$. There can be many such injections for a given set, but we initially pick one for every $A \in \mathcal{B}^0$. When a set of the support will be changed we shall explicitly define an injection. In fact, only for added elements we will define a new match, for all other elements they will be matched all the time to the same vertex, as long as they are available.

For any two $A, B \in \mathcal{B}^0$ we define the mapping $\phi^0[B,A] : B \to A \cup \{\bot\}$ as follows. Let v^A be the injection of A, and let v^B be the injection of B. If there exists element a such that $v^A(a) = v^B(b)$, then we set $\phi^0[B,A](b) = a$; if not, we set $\phi^0[B,A](b) = \bot$.

Let us verify that such a definition satisfies first two properties.

Property 5. For each $a \in A$ there is at most one $b \in B$ for which $\phi^0[B,A](b) = a$.

Proof. This one is trivially satisfied because there can be at most one $b \in B$ that according to v^B is matched to $v^A(a)$. ☐

Property 6. For $b \in B \setminus A$, if $\phi^0[B,A](b) = \bot$, then $A + b \in \mathcal{I}$, otherwise $A - \phi^0[B,A](b) + b \in \mathcal{I}$.

Proof. Suppose $\phi^0[B,A](b) = \bot$. It means that b is matched to $v^B(b)$ to which no element $a \in A$ is matched to. Therefore when we add edge $(b, v^B(b))$ to the injection $\left\{\left(a, v^A(a)\right)\right\}_{a\in A}$ it is still a proper injection, since $b \notin A$, and so $A + b \in \mathcal{I}$. Now suppose $\phi^0[B,A](b) = a' \neq \bot$. Now the set A changes to $A - a' + b$ and the underlying injection is $\left\{\left(a, v^A(a)\right)\right\}_{a\in A\setminus a'} \cup \{(b, v^B(b))\}$, which is a valid injection since $b \notin A$, and if so, then $A - a' + b$ is indeed an independent set. So Property 2 also holds. ☐

Now let us move to the most technically demanding property.

Property 7. Let ϕ^t be a family of transversal mappings for $\mathcal{B}^t$. Suppose we update the support $\mathcal{B}^t$ and obtain $\mathcal{B}^{t+1}$. Then we can find a family ϕ^{t+1} of transversal mappings such that $\Gamma^t(a) = \Gamma^{t+1}(a)$ for any element a that is still available after step t.

Proof. Suppose that in step t we have chosen element c, and we update the support $\mathcal{B}^t$ as described in the for loop of the algorithm in line 15. First of all assume that $c \neq a$, otherwise a becomes unavailable so there is nothing to prove. Let C^t be the critical set of c and let $v^{C^t}(c)$ be the vertex to which c is matched according to v^{C^t}. Consider set $B^t \in \mathcal{B}^t$ and let us describe how c affects B^{t+1} and injection $v^{B^{t+1}}$.

Case 1, $c \in B^t$: If it is c from B^t that is matched to $v^{C^t}(c)$, i.e., $v^{B^t}(c) = v^{C^t}(c)$, then we do not have to change anything, we set $B^{t+1} := B^t$ and $v^{B^{t+1}} = v^{B^t}$. If $v^{B^t}(c) \neq v^{C^t}(c)$, then let b_1 be such that $v^{B^t}(b_1) = v^{C^t}(c)$. We remove b_1 from B^t, i.e., $B^{t+1} := B^t \setminus b_1$ (we do not have to add c to B^{t+1} because it is already there). For every $b_3 \in B^{t+1}$ we set $v^{B^{t+1}}(b_3) = v^{B^t}(b_3)$. See Figure 5.1.

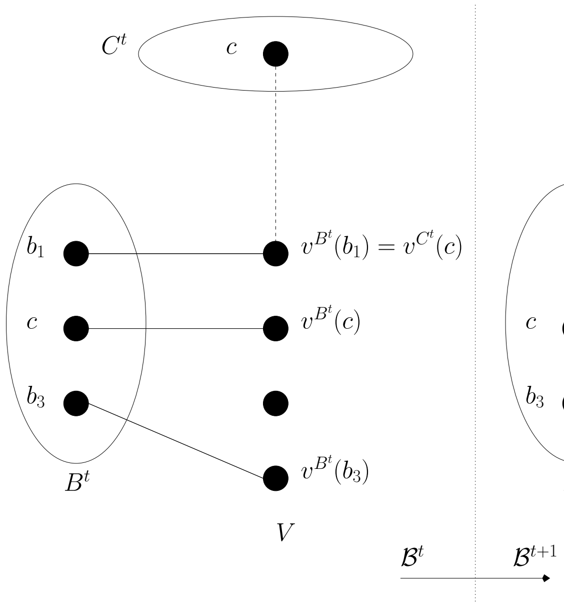

Figure 5.1. Illustration of Case 1, $c \in B^t$

Case 2, $c \notin B^t$: Let b_1 be such that $v^{B^t}(b_1) = v^{C^t}(c)$. We remove b_1 from B^t and add c instead, i.e., $B^{t+1} = B^t - b_1 + c$. The injection is defined as: $v^{B^{t+1}}(c) = v^{C^t}(c)$, and $v^{B^{t+1}}(b_3) = v^{B^t}(b_3)$ for $b_3 \in B^t \setminus b_1$. See Figure 5.2.

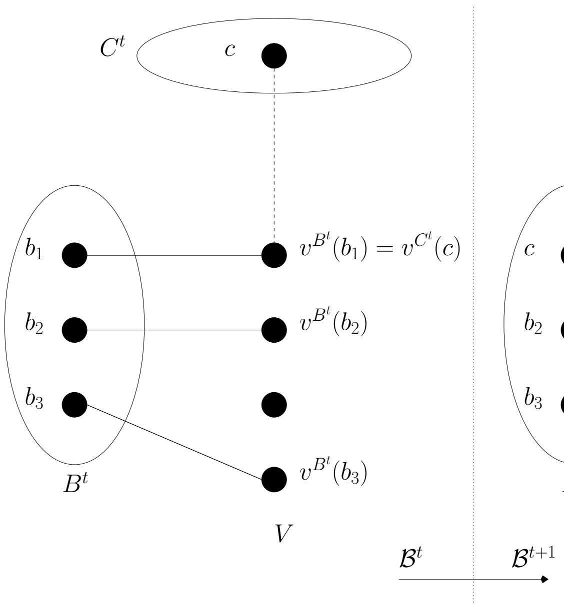

Figure 5.2. Illustration of Case 2, $c \notin B^t$

Given sets A^{t+1}, B^{t+1} with corresponding injections $v^{A^{t+1}}$,$v^{B^{t+1}}$ we define mapping ϕ^{t+1} as before, i.e., $\phi^{t+1}\left[B^{t+1}, A^{t+1}\right](b) = a$, if $v^{A^{t+1}}(a) = v^{B^{t+1}}(b)$.

Now we need to show that the set $\Gamma^{t+1}(a) = \left\{ b \,\middle|\, b \neq a \wedge \phi\left[B^{t+1}_{c(b)}, B^{t+1}_{c(a)}\right](b) = a \right\}$

is equal to $\Gamma^t(a) = \left\{b \,\middle|\, b \neq a \wedge \phi\left[B^t_{c(b)}, B^t_{c(a)}\right](b) = a\right\}$, if a is still available.

Consider again sets A^{t+1}, B^{t+1} and suppose that A^t is the critical set of a and that B^t is the critical set of b. Suppose that both a, b are matched to $v_{ab} = v^{A^t}(a) = v^{B^t}(b)$, i.e., $b \in \Gamma^t(a)$. If it happened that in step t element c removed a and b, i.e., $v^{C^t}(c) = v_{ab}$, then elements a, b are blocked and not available, so there is nothing to prove here. If $v^{C^t}(c) \neq v$, then from the reasoning in Case 1 and 2, we know that a and b are still matched to v_{ab}, i.e., $v_{ab} = v^{A^{t+1}}(a) = v^{B^{t+1}}(b)$. But if so, then $\phi^{t+1}\left[B^{t+1}, A^{t+1}\right](b) = a$, and $b \in \Gamma^{t+1}(a)$ still, because A^{t+1}, B^{t+1} remain critical sets of a, b. Conversely, if b is not matched to $v^{A^t}(a)$, i.e., $v^{B^t}(b) \neq v^{A^t}(a)$, then $b \notin \Gamma^t(a)$. But if c during the update does not block b, then b does not change its matched vertex so we still have $v^{B^{t+1}}(b) \neq v^{A^{t+1}}(a)$, and still $b \notin \Gamma^{t+1}(a)$.

Illustration is given in Figure 5.3.

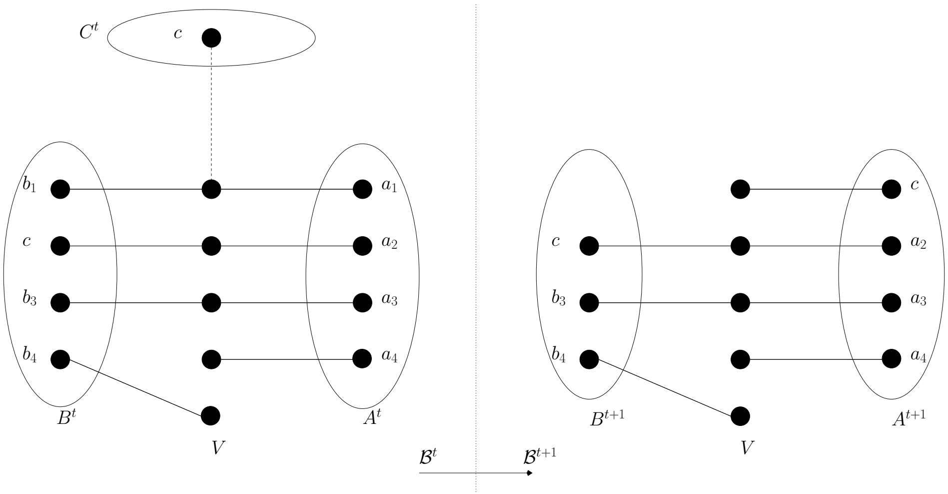

Figure 5.3. Illustration of change in Γ. We have blocked b_1, and if $b_1 \in \Gamma^t(a_1)$, then it does not matter anyway, because we have also blocked a_1. Element c was matched (w.r.t. B_t) to the same vertex as a_2, but B_t is not the critical set of c, so $c \notin \Gamma^t(a_2)$. Assume B^t is a critical set of b_3: we have $\phi^t[B^t, A^t](b_3) = a_3$ and so $b_3 \in \Gamma^t(a_3)$; after the update we still have $\phi^{t+1}[B^{t+1}, A^{t+1}](b_3) = a_3$, so $b_3 \in \Gamma^{t+1}(a_3)$. Element a_4 did not have any element $b' \in B^t$ in $\Gamma^t(a_4)$, so it does not have any $b' \in B^{t+1}$ in $\Gamma^{t+1}(a_4)$ as well.

□

5.3.2 Full proof of Lemma 52

The full scheme is presented on Figure 4. Let us concisely denote by $\mathcal{C}$ all the critical sets chosen, i.e., $\mathcal{C} = \left(\mathcal{C}^{in}_j\right)_{j \in [k^{in}]} \times \left(\mathcal{C}^{out}_j\right)_{j \in [k^{out}]}$. Transversal mappings $\phi(\mathcal{M})^0$ are found in exactly the same manner as in the single matroid version.

There are two main differences with respect to what was presented in the main body. First, since $p \cdot x \in b \cdot \mathcal{P}\left(\mathcal{M}^{in}_j\right)$, then $\frac{1}{b} p \cdot x \in \mathcal{P}\left(\mathcal{M}^{in}_j\right)$, and the support $\mathcal{B}^0_{in[j]}$ is found by decomposing $\frac{1}{b} p \cdot x$. Thus when element f chooses a critical set in matroid $\mathcal{M}^{in}_j$ it chooses with probability $\beta^{in[j]}_i / \left(\frac{1}{b} p_f x_f\right) = b \cdot \beta^{in[j]}_i / (p_f x_f)$. This results in

---

Algorithm 4 Stoch-CR scheme $\bar{\pi}_x(A)$ for $\mathcal{M}_1^{in}, ..., \mathcal{M}_{k^{in}}^{in}$ inner matroids and $\mathcal{M}_1^{out}, ..., \mathcal{M}_{k^{out}}^{out}$ outer matroids, and $x \in b \cdot \mathcal{P}(\mathcal{I}^{in}, \mathcal{I}^{out})$.

//Preprocessing:
find support $\mathcal{B}_{in[j]}^{0}$ of $\frac{1}{b}p \cdot x \in \mathcal{P}\left(\mathcal{M}_j^{in}\right)$ for each $\mathcal{M}_j^{in}$;
find support $\mathcal{B}_{out[j]}^{0}$ of $\frac{1}{b}x \in \mathcal{P}\left(\mathcal{M}_j^{out}\right)$ for each $\mathcal{M}_j^{out}$;
find family $\phi_{in[j]}^{0}$ for every $\mathcal{M}_j^{in}$; find family $\phi_{out[j]}^{0}$ for every $\mathcal{M}_j^{out}$;
choose critical sets $\mathcal{C}_j^{in}$ for each $\mathcal{M}_j^{in}$ and $\mathcal{C}_j^{out}$ for each $\mathcal{M}_j^{out}$;
//Random selection phase:
remove from A all $e : x_e = 0$; mark all $e \in A$ as available; $S \leftarrow \emptyset$
while there are still available elements in A **do**
    pick element e uniformly at random from A
    if e is available **then**
        probe e
        if probe of e successful **then**
            $S \leftarrow S \cup \{e\}$
            for each matroid $\mathcal{M}_j^{in}$ **do**
                for each set B_i^t of support $\mathcal{B}_{in[j]}^{t}$ **do**
                    $B_i^t \leftarrow B_i^t + e$
        call e unavailable
    else simulate the probe of e
    if probe or simulation was successful **then**
        for each set B_i^t of support $\mathcal{B}^t$ **do**
            $f \leftarrow \phi\left[B_{c_j^{in}(e)}^t, B_{c_j^{in}(f)}^t\right](e)$
            if $f \neq e$ **then** $B_{c_j^{in}(f)}^t \leftarrow B_{c_j^{in}(f)}^t - f$ and call f unavailable
    for each matroid $\mathcal{M}_j^{out}$ **do**
        for each set B_i^t of support $\mathcal{B}_{out[j]}^t$ **do**
            $B_i^t \leftarrow B_i^t + e$
            $f \leftarrow \phi\left[B_{c_j^{out}(e)}^t, B_{c_j^{out}(f)}^t\right](e)$
            if $f \neq e$ **then** $B_{c_j^{out}(f)}^t \leftarrow B_{c_j^{out}(f)}^t - f$ and call f unavailable
    compute families $\phi_{in[j]}^{t+1}$, $\phi_{out[j]}^{t+1}$
    $t \leftarrow t + 1$
return S as $\pi_x(A)$

---

the following modification in Lemma 54. The proof is completely analogous, so we skip it.

Lemma 56. *If $\frac{1}{b}p \cdot x \in \mathcal{P}\left(\mathcal{M}_j^{in}\right)$, then for any element e, it holds that*

$$\mathbb{E}_{\mathcal{C}_j^{in}, R(x)}\left[\sum_{f \in \Gamma^0_{in[j]}(e)} p_f \cdot \chi\left[f \in R(x)\right]\right] \leq b,$$

where the expectation is over $R(x)$ and the choice of critical sets $\mathcal{C}$.

Now we also need to deal with outer matroids. Again, the proof is completely analogous, so we skip it.

Lemma 57. *If $\frac{1}{b}x \in \mathcal{P}\left(\mathcal{M}_j^{out}\right)$, then for any element e, it holds that*

$$\mathbb{E}_{\mathcal{C}_j^{out}, R(x)}\left[\sum_{f \in \Gamma^0_{out[j]}(e)} \chi\left[f \in R(x)\right]\right] \leq b,$$

where the expectation is over $R(x)$ and the choice of critical sets $\mathcal{C}$.

Second. Let Y_e^t for $t = 0, 1, ...,$ be a random variable indicating if e is still available after step t. Initially $Y_e^0 = X_e$. Let P_e^t be a random variable indicating, if e was probed in one of steps $0, 1, ...,$ or t. In step $t+1$ element e can be blocked if, for some $j \in [k^{in}]$, we pick element $f \in \Gamma^t_{in[j]}(e)$ and successfully probe it (or successfully simulate), or if we just pick element $f \in \Gamma^t_{out[j]}(e)$, and probe f or simulate its probe, irregardless of the outcome. Let $\Gamma^t_{out}(e) = \bigcup_{j \in [k^{out}]} \Gamma^t_{out[j]}(e)$ and let $\Gamma^t_{in}(e) = \bigcup_{j \in [k^{in}]} \Gamma^t_{in[j]}(e) \setminus \Gamma^t_{out}(e)$ — this subtraction is to not count an element twice, because if element f belongs to both $\Gamma^t_{in[j]}(e)$ and $\Gamma^t_{out[j]}(e)$, then just probing f (or simulating its probe) automatically blocks e, irregardless of f's probe (simulation) outcome. Hence, we do cannot account for the excessive $p_f X_f$ influence of f on e. Therefore, the probability that e stops to be available at step $t+1$ is equal to

$$\mathbb{E}\left[Y_e^t - Y_e^{t+1} \middle| \mathcal{F}^t, \mathcal{C}\right] = \frac{Y_e^t}{|A|} + \frac{Y_e^t}{|A|} \cdot \sum_{f \in \Gamma^t_{in}(e)} p_f X_f + \frac{Y_e^t}{|A|} \cdot \sum_{f \in \Gamma^t_{out}(e)} X_f.$$

As in the single matroid case, the probability of probing e is just equal to $\mathbb{E}\left[P_e^{t+1} - P_e^t \middle| \mathcal{F}^t, \mathcal{C}\right] = \frac{Y_e^t}{|A|}$. Thus the martingale we use now in the analysis is

$$\left(\left(1 + \sum_{f \in \Gamma^t_{in}(e)} p_f X_f + \sum_{f \in \Gamma^t_{out}(e)} X_f\right) \cdot P_e^t + Y_e^t\right)_{t \geq 0}.$$

Let $\tau = \min\left\{t \middle| Y_e^t = 0\right\}$ be the step in which edge e became unavailable. It is clear that τ is a stopping time. Thus from Doob's stopping theorem we get that

$$\mathbb{E}_\tau\left[\left(1+\sum_{f\in\Gamma^\tau_{in}(e)} p_f X_f + \sum_{f\in\Gamma^\tau_{out}(e)} X_f\right)\cdot P^\tau_e + Y^\tau_e \middle| \mathcal{C}\right]$$
$$= \mathbb{E}\left[\left(1+\sum_{f\in\Gamma^0_{in}(e)} p_f X_f + \sum_{f\in\Gamma^0_{out}(e)} X_f\right)\cdot P^0_e + Y^0_e \middle| \mathcal{C}\right],$$

And as before, since the transversal mappings are controlled per matroid, we have that $\Gamma^t_{in}(e) = \Gamma^0_{in}(e)$ and $\Gamma^t_{out}(e) = \Gamma^0_{out}(e)$ for $t \leq \tau$. Thus

$$\mathbb{P}\left[e \in \bar{\pi}_x(A) | \mathcal{C}\right] = p_e \cdot \mathbb{E}_\tau\left[P^\tau_e | \mathcal{C}\right] = p_e X_e \Big/ \left(1+\sum_{f\in\Gamma^0_{in}(e)} p_f X_f + \sum_{f\in\Gamma^0_{out}(e)} X_f\right). \tag{5.8}$$

Monotonicity follows as before. The approximation guarantee similarly from Jensen.

We take random set $R(x)$ instead of A; now $X_f = \chi[f \in R(x)]$ is a random variable. Let us condition on $e \in R(x)$, take expected value $\mathbb{E}_{\mathcal{C},R(x)}[\cdot | e \in R(x)]$ on both sides of 5.8, and apply Jensen's inequality to convex function $x \mapsto \frac{1}{x}$ to get:

$$\mathbb{P}\left[e \in \bar{\pi}_x(R(x)) | e \in R(x)\right] = \mathbb{E}_{\mathcal{C},R(x)}\left[\mathbb{P}\left[e \in \bar{\pi}_x(R(x)) | \mathcal{C}, R(x)\right] | e \in R(x)\right] =$$
$$= \mathbb{E}_{\mathcal{C},R(x)}\left[p_e X_e \Big/ \left(1+\textstyle\sum_{f\in\Gamma^0_{in}(e)} p_f X_f + \sum_{f\in\Gamma^0_{out}(e)} X_f\right) \middle| e \in R(x)\right]$$
$$\geq p_e \Big/ \mathbb{E}_{\mathcal{C},R(x)}\left[1+\textstyle\sum_{f\in\Gamma^0_{in}(e)} p_f X_f + \sum_{f\in\Gamma^0_{out}(e)} X_f \middle| e \in R(x)\right].$$

Since

$$\mathbb{E}_{\mathcal{C},R(x)}\left[\sum_{f\in\Gamma^0_{in[j]}(e)} p_f X_f \middle| e \in R(x)\right] \leq b$$

from Lemma 56, and

$$\mathbb{E}_{\mathcal{C},R(x)}\left[\sum_{f\in\Gamma^0_{out[j]}(e)} X_f \middle| e \in R(x)\right] \leq b$$

from Lemma 57, we conclude that

$$
\begin{aligned}
&\mathbb{E}_{\mathcal{C},R(x)}\left[1+\sum_{f\in\Gamma^0_{in}(e)} p_f X_f + \sum_{f\in\Gamma^0_{out}(e)} X_f \,\middle|\, e\in R(x)\right] \\
&\leq \mathbb{E}_{\mathcal{C},R(x)}\left[1+\sum_{j\in[k^{in}]}\sum_{f\in\Gamma^0_{in[j]}(e)} p_f X_f + \sum_{j\in[k^{out}]}\sum_{f\in\Gamma^0_{out[j]}(e)} X_f \,\middle|\, e\in R(x)\right] \\
&\leq 1+\sum_{j\in[k^{in}]}\mathbb{E}_{\mathcal{C},R(x)}\left[\sum_{f\in\Gamma^0_{in[j]}(e)} p_f X_f \,\middle|\, e\in R(x)\right] \\
&\qquad +\sum_{j\in[k^{out}]}\mathbb{E}_{\mathcal{C},R(x)}\left[\sum_{f\in\Gamma^0_{out[j]}(e)} X_f \,\middle|\, e\in R(x)\right] \\
&\leq 1+k^{in}\cdot b+k^{out}\cdot b.
\end{aligned}
$$

And therefore

$$\mathbb{P}\left[e\in\bar{\pi}_x\left(R\left(x\right)\right)\middle|\, e\in R\left(x\right)\right]\geq\frac{p_e}{b\left(k^{in}+k^{out}\right)+1},$$

which yields

$$\mathbb{P}\left[e\in\bar{\pi}_x\left(R\left(x\right)\right)\middle|\, e\in act\left(R\left(x\right)\right)\right]\geq\frac{1}{b\left(k^{in}+k^{out}\right)+1},$$

which is exactly Property 3 from the definition of stoch-CR scheme. Lemma 52 follows.

Chapter 6

Stochastic k-set packing

In this Chapter we are showing a $(k+1)$-approximation algorithm for Stochastic k-set packing, and a $\frac{k}{1-e^{-k}}$-approximation for its special case the stochastic k-hypergraph matching. Let us recall the statement of the problem. We are given n elements/columns, where each item $e \in E = [n]$ has a profit $v_e \in \mathbb{R}_+$, and a random d-dimensional size $S_e \in \{0,1\}^d$. The sizes are independent for different items. Additionally, for each item e, there is a set C_e of at most k coordinates such that each size vector S_e takes positive values only in these coordinates, i.e., $S_e \subseteq C_e$ with probability 1. We are also given a capacity vector $b \in \mathbb{Z}_+^d$ into which items must be packed. We assume that v_e is a random variable that can be correlated with S_e. The coordinates of S_e also might be correlated between each other. The goal is to design a strategy that will be one-by-one packing elements e, and that will maximize the expected outcome obtained from fully packed elements.

Important thing to notice is that in this setting, unlike in the previous ones, here when we probe an element, there is no success/failure outcome. The size S_e of an element e materializes, and the reward v_e is just drawn.

6.1 $(k+1)$-approximation without monotonicity

Let us first formulate the LP. Let $p_e^j = \mathbb{E}[S_e(j)]$ be the expected size of the j coordinate of column e. The following is an LP that models the problem. Here $U(c)$ denotes a uniform matroid of rank c.

$$\begin{aligned} \max \quad & \sum_{e=1}^{n} \mathbb{E}[v_e] \cdot x_e & \text{(LP-}k\text{-set)} \\ \text{s.t.} \quad & p^j \cdot x \in \mathcal{P}(U(b_j)) & \forall j \in [d] \\ & x_e \in [0,1] & \forall e \in [n]. \end{aligned}$$

Where, as usual, x_e stands for $\mathbb{P}[OPT \text{ probes column } e]$. We are going to present a probing strategy in which for every element e probability that we will probe e will be at least $\frac{x_e}{k+1}$. From this the Theorem will follow.

The algorithm is presented on Figure 5.

Constraint for row j is in fact given be a uniform matroid in which we can take at most b_j elements from subset $\{e \,|\, j \in C_e\} \subseteq E$. Therefore, we can decompose $p^j \cdot x =$

Algorithm 5 Algorithm for stochastic k-set packing

```
//Preprocessing:
for each $j \in [d]$ do
    find support $\mathcal{B}_j^0$ of $p^j \cdot x$ in $\mathcal{P}(U(b_j))$
    family $\phi_j^0$
    critical sets $\mathcal{C} = (\mathcal{C}^j)_{j \in [d]}$
//Rounding:
let $A \leftarrow R(x)$; mark all $e \in A$ as available; $S \leftarrow \emptyset$
while there are still available elements in $A$ do
    pick element $e$ uniformly at random from $A$
    if $e$ is available then
        probe $e$
        $S \leftarrow S + e$
        for each $j \in C_e$ such that $S_e(j) = 1$ do
            for each set $B_i^{j,t}$ of support $\mathcal{B}^{j,t}$ do
                $B_i^{j,t} \leftarrow B_i^{j,t} + e$
        call $e$ unavailable
    else simulate the probe of $e$
    for each $j \in C_e$ such that $S_e(j) = 1$ (whether we probe or simulate) do
        for each set $B_i^{j,t}$ of support $\mathcal{B}^{j,t}$ do
            $f \leftarrow \phi\left[B_{c^j(e)}^{j,t}, B_{c^j(f)}^{j,t}\right](e)$
            if $f \neq e$ then $B_{c^j(f)}^{j,t} \leftarrow B_{c^j(f)}^{j,t} - f$ and call $f$ unavailable
        compute the family $\phi_j^{t+1}$
    for each $i$ do $B_i^{j,t+1} \leftarrow B_i^{j,t}$
    $t \leftarrow t + 1$
return $S$
```

$\sum_l \beta_l^j \cdot B_l^{j,0}$. Uniform matroid is a transversal matroid, so we use the transversal mapping ϕ_j^0 between sets $B^{j,0}$, also let $\mathcal{C} = \left(\mathcal{C}^j\right)_{j\in[d]}$ be the vector indicating the critical sets. We define in the same way as we did already in Lemma 52, the sets $\Gamma_j^0(e)$ of blocking elements, i.e.,

$$\Gamma_j^0(e) = \left\{ f \,\middle|\, f \neq e \wedge \phi_j^0[B^0_{c^j(f)}, B^0_{c^j(e)}] \right\}.$$

As before, let us from now on condition on $\mathcal{C}$. Let us analyze the impact of $f \in \Gamma_j^t(e)$ on e. Element $f \in \Gamma_j^t(e)$ blocks e when f is chosen and $S_f(j) = 1$. However, right now f can belong to $\Gamma_j^t(e)$ for many $j \in C_e$. Therefore if f is chosen in line 9 of the Algorithm 5, then the probability that f blocks e is equal to $\mathbb{P}\left[\bigvee_{j:f\in\Gamma_j^t(e)} (S_f(j) = 1) \middle| \mathcal{C}\right]$.

Let us now repeat the steps of Lemma 1. Let $X_e = \chi[e \in R(x)]$. Let Y_e^t for $t = 0, 1, \ldots$, be a random variable indicating if e is still available after step t. Initially $Y_e^0 = X_e$. Let P_e^t be a random variable indicating, if e was probed in one of steps $0, 1, \ldots,$ or t; we have $P_e^0 = 0$ for all e.

Variable $P_e^{t+1} - P_e^t$ indicates if e was probed at step $t+1$. Given the information $\mathcal{F}^t$ about the process up to step t, the probability of this event is $\mathbb{E}\left[P_e^{t+1} - P_e^t \middle| \mathcal{F}^t, \mathcal{C}\right] = \frac{Y_e^t}{|A|}$, because if element e is still available after step t (i.e., $Y_e^t = 1$), then with probability $\frac{1}{|A|}$ we choose it in line 9, and otherwise (i.e. $Y_e^t = 0$) we cannot probe it.

Variable $Y_e^t - Y_e^{t+1}$ indicates whether element e stopped being available at step $t+1$. For this to happen we need to pick $f \in \Gamma_j^t(0)$ and the probe (or simulation) of f needs to result in a vector S_f such that $S_f(j) = 1$. However, as already noted, there can be many j for which $f \in \Gamma_j^t(0)$. Therefore, probability that e stops being available in step $t+1$ is equal to

$$\mathbb{E}\left[Y_e^t - Y_e^{t+1} \middle| \mathcal{F}^t, \mathcal{C}\right] = \frac{Y_e^t}{|A|} + \frac{Y_e^t}{|A|} \cdot \sum_f X_f \cdot \mathbb{P}\left[\bigvee_{j:f\in\Gamma_j^t(e)} (S_f(j) = 1) \middle| \mathcal{C}\right].$$

Here we stress the condition on $\mathcal{C}$ because sets Γ_j^t are constructed given the choice of critical sets. Again we can reason that

$$\left(\left(1 + \sum_f X_f \cdot \mathbb{P}\left[\bigvee_{j:f\in\Gamma_j^t(e)} (S_f(j) = 1) \middle| \mathcal{C}\right]\right) \cdot P_e^t + Y_e^t\right)_{t\geq 0}$$

is a martingale. Let $\tau = \min\left\{t \,\middle|\, Y_e^t = 0\right\}$ be the step in which edge e became unavailable. It is clear that τ is a stopping time. Thus from Doob's Stopping Theorem we get that

$$\mathbb{E}_\tau\left[\left(1+\sum_f X_f\cdot\mathbb{P}\left[\bigvee_{j:f\in\Gamma_j^\tau(e)}(S_f(j)=1)\middle|\mathcal{C}\right]\right)\cdot P_e^\tau+Y_e^\tau\middle|\mathcal{C}\right]$$
$$=\mathbb{E}_\tau\left[\left(1+\sum_f X_f\cdot\mathbb{P}\left[\bigvee_{j:f\in\Gamma_j^0(e)}(S_f(j)=1)\middle|\mathcal{C}\right]\right)\cdot P_e^0+Y_e^0\middle|\mathcal{C}\right]=X_e.$$

We argue again using the properties of transversal mapping ϕ_j^t that we have $\Gamma_j^\tau(e)=\Gamma_j^0(e)$ for each $j\in C_e$, since e was available before step τ. And if so, then

$$\mathbb{E}_\tau\left[\left(1+\sum_f X_f\cdot\mathbb{P}\left[\bigvee_{j:f\in\Gamma_j^0(e)}(S_f(j)=1)\middle|\mathcal{C}\right]\right)\cdot P_e^\tau+Y_e^\tau\middle|\mathcal{C}\right]=X_e$$

and since $\left(1+\sum_f X_f\cdot\mathbb{P}\left[\bigvee_{j:f\in\Gamma_j^0(e)}(S_f(j)=1)\middle|\mathcal{C}\right]\right)$ is just a number depending on $\mathcal{C}$, we can say that

$$\left(1+\sum_f X_f\cdot\mathbb{P}\left[\bigvee_{j:f\in\Gamma_j^0(e)}(S_f(j)=1)\middle|\mathcal{C}\right]\right)\cdot\mathbb{E}_\tau\left[P_e^\tau\middle|\mathcal{C}\right]=X_e.$$

Note that $\mathbb{E}_\tau\left[P_e^\tau\middle|\mathcal{C}\right]=\mathbb{P}\left[e\text{ is probed}\middle|\mathcal{C}\right]$, and conclude that

$$\mathbb{P}\left[e\text{ is probed}\middle|\mathcal{C}\right]=X_e\Bigg/\left(1+\sum_f X_f\cdot\mathbb{P}\left[\bigvee_{j:f\in\Gamma_j^0(e)}(S_f(j)=1)\middle|\mathcal{C}\right]\right).$$

At this point we reason as follows:

$$\begin{aligned}&\sum_f X_f\cdot\mathbb{P}\left[\bigvee_{j:f\in\Gamma_j^0(e)}(S_f(j)=1)\middle|\mathcal{C}\right]\\ \le&\sum_f X_f\cdot\sum_{j:f\in\Gamma_j^0(e)}\mathbb{P}\left[S_f(j)=1\middle|\mathcal{C}\right]\\ =&\sum_f X_f\cdot\sum_{j:f\in\Gamma_j^0(e)}p_f^j\\ =&\sum_{j\in C_e}\sum_{f\in\Gamma_j^0(e)}p_f^jX_f,\end{aligned}$$

where inequality just follows simply from the union-bound. Then we have an identity since event $(S_f(j)=1)$ is independent of $\mathcal{C}$ and its probability is just equal to p_f^j. Later we just change the order of summation. Therefore we have shown that

$$\mathbb{P}\left[e\text{ is probed}\middle|\mathcal{C}\right]\ge X_e\Bigg/\left(1+\sum_{j\in C_e}\sum_{f\in\Gamma_j^0(e)}p_f^jX_f\right).$$

We apply expectation $\mathbb{E}_{\mathcal{C},R(x)}\left[\cdot\middle|\, e\in R(x)\right]$ to both sides, use Jensen's inequality to get

$$\mathbb{P}\left[e \text{ is probed}\middle|\, e\in R(x)\right] \geq 1 \Bigg/ \left(1+\mathbb{E}_{\mathcal{C},R(x)}\left[\sum_{j\in C_e}\sum_{f\in\Gamma_j^0(e)} p_f^j X_f \,\middle|\, e\in R(x)\right]\right).$$

Now we can say that

$$\mathbb{E}_{\mathcal{C},R(x)}\left[\sum_{j\in C_e}\sum_{f\in\Gamma_j^0(e)} p_f^j X_f \,\middle|\, e\in R(x)\right]$$
$$= \sum_{j\in C_e}\mathbb{E}_{\mathcal{C},R(x)}\left[\sum_{f\in\Gamma_j^0(e)} p_f^j X_f \,\middle|\, e\in R(x)\right] \leq \sum_{j\in C_e} 1 = |C_e| \leq k,$$

where the last inequality $\mathbb{E}_{\mathcal{C},R(x)}\left[\sum_{f\in\Gamma_j^0(e)} p_f^j X_f \,\middle|\, e\in R(x)\right] \leq 1$ for each j comes from Lemma 54. Hence

$$\mathbb{P}\left[e \text{ is probed}\middle|\, e\in R(x)\right] \geq \frac{1}{1+k},$$

which gives

$$\mathbb{P}\left[e \text{ is probed}\right] = \mathbb{P}\left[e\in R(x)\right]\cdot\mathbb{P}\left[e \text{ is probed}\middle|\, e\in R(x)\right] \geq \frac{x_e}{1+k}$$

as desired.

6.2 k-hypergraph matching

We use an LP very similar to the one from previous section:

$$\begin{aligned}
\max \quad & \sum_{e=1}^{n} \mathbb{E}\left[v_e\right]\cdot x_e && \text{(LP-}k\text{-HYP-MATCH)}\\
\text{s.t.} \quad & \sum_{e=1}^{n} p_e^j \cdot x_e \leq 1 && \forall j\in[d]\\
& x_e \in [0,1] && \forall e\in[n].
\end{aligned}$$

In the algorithm we first need to compute the probability that $S_e \neq 0$ for every edge e. As mentioned in the introduction, we assume that we can calculate $\mathbb{P}\left[S_e\neq 0\right]$ in polynomial time. Given this quantity we generate a random variable Y_e that is distributed according to the following cumulative distribution function $\mathbb{P}\left[Y_e\leq t\right] = \left(1-\exp\left(-t\cdot x_e\mathbb{P}\left[S_e\neq 0\right]\right)\right)/\mathbb{P}\left[S_e\neq 0\right]$. The support of Y_e is $\left[0, \frac{1}{x_e\mathbb{P}[S_e\neq 0]}\ln\frac{1}{1-\mathbb{P}[S_e\neq 0]}\right]$; let us denote this interval by I_e. We try to pack edges in the order of increasing Y_e. The density function of Y_e is $t\mapsto x_e\cdot\exp\left(-t\cdot x_e\mathbb{P}\left[S_e\neq 0\right]\right)$. The algorithm is presented in Figure 6. As in the previous Section we want to prove that every edge will be probed with probability at least $x_e\frac{1-e^{-k}}{k}$.

Algorithm 6 Approximation algorithm for stochastic k-hypergraph matching.

1. solve the linear program (LP-k-HYP-MATCH); let x be the solution;
2. for every $e \in [n]$, sample a random variable Y_e distributed according to $\mathbb{P}[Y_e \leq t] = (1 - \exp(-t \cdot x_e \mathbb{P}[S_e \neq 0]))/\mathbb{P}[S_e \neq 0]$.
3. $R \leftarrow \emptyset$
4. for every $e \in [n]$ in increasing order or Y_e's:
   (a) realize S_e
   (b) if $\sum_{f \in R} S_f + S_e \leq 1^d$, i.e., S_e can still be packed, then $R \leftarrow R + e$
5. return R

Consider two columns e and f. If f appears before e in the order, then it can block e only when $S_f \cap S_e \neq \emptyset$. Probability $\mathbb{P}[S_f \cap S_e \neq \emptyset]$ can be trivially upperbounded by $\mathbb{P}[S_f \cap C_e \neq \emptyset]$, where C_e is the set of all coordinates that can be non-zero in S_e. Therefore, the probability that f does not block e is

$$\begin{aligned}
&\mathbb{P}[Y_f > Y_e] + \mathbb{P}[Y_f < Y_e](1 - \mathbb{P}[S_f \cap S_e \neq \emptyset]) \\
\geq &\mathbb{P}[Y_f > Y_e] + \mathbb{P}[Y_f < Y_e](1 - \mathbb{P}[S_f \cap C_e \neq \emptyset]) \\
= &1 - \mathbb{P}[Y_f < Y_e]\,\mathbb{P}[S_f \cap C_e \neq \emptyset] \\
= &1 - \min\left(\frac{1 - e^{-x_f \mathbb{P}[S_f \neq \emptyset] \cdot Y_e}}{\mathbb{P}[S_f \neq \emptyset]}, 1\right) \cdot \mathbb{P}[S_f \cap C_e \neq \emptyset] \\
\geq &1 - \frac{1 - e^{-x_f \mathbb{P}[S_f \neq \emptyset] \cdot Y_e}}{\mathbb{P}[S_f \neq \emptyset]} \cdot \mathbb{P}[S_f \cap C_e \neq \emptyset] \\
\geq &e^{-x_f \mathbb{P}[S_f \cap C_e \neq \emptyset] \cdot t},
\end{aligned}$$

where the last inequality follows from $1 - (1 - e^{-t})\alpha \geq e^{-\alpha t}$ for $\alpha \in [0, 1]$. Hence, the total probability that e is not blocked is at least (we always implicitly iterate over $f \neq e$):

$$\begin{aligned}
&\int_{I_e} \left(\prod_f e^{-x_f \mathbb{P}\left[S_f \cap C_e \neq \emptyset\right] \cdot Y_e} \right) \cdot \mathrm{d}Y_e \\
=&\int_{I_E} \left(\prod_f e^{-x_f \mathbb{P}\left[S_f \cap C_e \neq \emptyset\right] \cdot t} \right) \cdot x_e \cdot e^{-x_e \mathbb{P}[S_e \neq \emptyset] \cdot t} \mathrm{d}t \\
=&x_e \int_{I_e} e^{-\sum_f x_f \mathbb{P}\left[S_f \cap C_e \neq \emptyset\right] \cdot t} \cdot e^{-x_e \mathbb{P}[S_e \neq \emptyset] \cdot t} \mathrm{d}t \\
=&x_e \int_{I_e} e^{-\sum_f x_f \mathbb{P}\left[S_f \cap C_e \neq \emptyset\right] \cdot t - x_e \mathbb{P}[S_e \neq \emptyset] \cdot t} \mathrm{d}t \\
=&x_e \cdot g(\sum_f x_f \mathbb{P}\left[S_f \cap C_e \neq \emptyset\right] + x_e \mathbb{P}\left[S_e \neq \emptyset\right]),
\end{aligned}$$

where

$$g(y) = \frac{1}{y} \left(1 - \exp\left(-y \cdot \frac{1}{x_e \mathbb{P}\left[S_e \neq \emptyset\right]} \ln \frac{1}{1 - \mathbb{P}\left[S_e \neq \emptyset\right]} \right) \right).$$

Now, function $g(y)$ is decreasing in y, so $g(\sum_f x_f \mathbb{P}\left[S_f \cap C_e \neq \emptyset\right] + x_e \mathbb{P}\left[S_e \neq \emptyset\right])$ is at least $g(k)$, because of the following Lemma.

Lemma 58. *It holds that*

$$\sum_f x_f \mathbb{P}\left[S_f \cap C_e \neq \emptyset\right] + x_e \mathbb{P}\left[S_e \neq \emptyset\right] \leq k.$$

Proof. Let us denote by C_e the support of S_e, i.e., all the coordinates that possibly can be non-zero in S_e. Let P_e^j be the $0-1$ variable stating whether S_e has 1 on jth coordinate. We have

$$\begin{aligned}
&\sum_f x_f \mathbb{P}\left[S_f \cap C_e \neq \emptyset\right] + x_e \mathbb{P}\left[S_e \neq \emptyset\right] \\
&= \sum_f x_f \mathbb{P}\left[\sum_{j \in C_f \cap C_e} P_f^j > 0 \right] + x_e \mathbb{P}\left[\sum_{j \in C_e} P_e^j > 0 \right] \\
&\leq \sum_f x_f \sum_{j \in C_f \cap C_e} \mathbb{P}\left[P_f^j > 0 \right] + x_e \sum_{j \in C_e} \mathbb{P}\left[P_e^j > 0 \right] \\
&= \sum_f x_f \sum_{j \in C_f \cap C_e} p_f^j + x_e \sum_{j \in C_e} p_e^j \\
&\leq \sum_f x_f \sum_{j \in C_e} p_f^j + x_e \sum_{j \in C_e} p_e^j \\
&= \sum_{j \in C_e} \sum_f x_f p_f^j + x_e p_e^j \\
&\leq \sum_{j \in C_e} 1 \\
&\leq k,
\end{aligned}$$

where the first inequality follows from union-bound, the second from $C_f \cap C_e \subseteq C_e$, the third from LP constraint for each j — $\sum_e p_e^j \cdot x_e \leq 1$ —, and the last one from the fact that $|C_e| \leq k$. □

Since $x_e \leq 1$, we can get rid of x_e:

$$\begin{aligned} g(k) &= \frac{1}{k}\left(1-\exp\left(-k\cdot\frac{1}{x_e\cdot\mathbb{P}\left[S_e\neq\emptyset\right]}\ln\frac{1}{1-\mathbb{P}\left[S_e\neq\emptyset\right]}\right)\right) \\ &\geq \frac{1}{k}\left(1-\exp\left(-k\frac{1}{\mathbb{P}\left[S_e\neq\emptyset\right]}\ln\frac{1}{1-\mathbb{P}\left[S_e\neq\emptyset\right]}\right)\right). \end{aligned}$$

Now just note that $\frac{1}{\mathbb{P}[S_e\neq\emptyset]}\ln\frac{1}{1-\mathbb{P}[S_e\neq\emptyset]} \geq 1$, so finally we obtain that the probability with which e is not blocked is

$$x_e\frac{1}{k}\left(1-\exp\left(-k\frac{1}{\mathbb{P}\left[S_e\neq\emptyset\right]}\ln\frac{1}{1-\mathbb{P}\left[S_e\neq\emptyset\right]}\right)\right) \geq x_e\frac{1}{k}\left(1-\exp\left(-k\right)\right)$$

as claimed.

Chapter 7

Iterative randomized rounding and negative correlation

In this Chapter we prove the following Theorem.

Theorem 59. *Let $(x_e)_{e\in E} \in [0,1]^E$ be such that $\sum_{e\in E} x_e = r$. There exists a randomized rounding procedure that outputs $(\hat{x}_e)_{e\in E} \in \{0,1\}^E$ such that:*

1. $\sum_e \hat{x}_e \leq r$,
2. $\mathbb{P}[\hat{x}_e = 1] \geq \frac{1}{2} x_e$,
3. variables $(\hat{x}_e)_{e\in E}$ are negatively correlated.

The Theorem can also be proved for any transversal matroid using the transversal mappings introduced in Chapter 5. However, for self-containment of this chapter we show it for uniform matroids, because then the construction of underlying mappings between independent sets of the convex decomposition is much easier to demonstrate.

Again, we start by using Cunningham's algorithm [20] to find a convex decomposition $x = \sum_{i=1}^{m} \beta_i \mathbf{1}_{B_i}$, where B_i are bases of uniform matroid of rank l. It holds that each B_i has size exactly l. It follows that for every e we have $x_e = \sum_{i:e\in B_i} \beta_i$. To describe better the structure of the decomposition, we introduce *atoms*, which basically represent pairs $\{(e, B_i) | e \in E, B_i : e \in B_i\}$.

7.1 Atoms and relation between them

For each pair (e, B_i) such that $e \in B_i$ define an *atom* γ_i^e. Therefore, once we say γ_i^e, we indicate both element e and base $B_i \ni e$. Sometimes we will write γ^e if the stress on the set it represents will not be important. Sometimes we will be even writing only γ if we will not be caring about the precise pair element-set it represents.

Atom γ_i^e has weight β_i, and represents the contribution of base B_i in the value of x_e. The set of atoms of element e we denote as $A(e)$, i.e., $A(e) = \{\gamma_i^e | B_i \ni e\}$. Atoms are weighted, and the weight of atom γ_i^e is just β_i; hence

$$x_e = \sum_{i\in\{i:\gamma_i^e \in A(e)\}} \beta_i.$$

However, it will be convenient to use just γ_i^e to denote the weight of atom γ_i^e; it will be always clear from the context if we mean weight of an atom or the atom itself. So now we can just write

$$x_e = \sum_{\gamma_i^e \in A(e)} \gamma_i^e$$

instead of the former equality.

In the following Lemma we present a simple construction of transversal mappings in the setting of uniform matroids.

Lemma 60. *Let $B_1, B_2, ..., B_m \in \mathcal{I}$ be independent sets of $U(r)$, i.e., sets of size at most r. There exists a family of mappings $\phi_{A,B} : A \mapsto B$ for each $A, B \in \{B_1, ..., B_m\}$ such that:*

1. *for each $b \in B$ there exists at most one $a \in A$ for which $\phi_{A,B}(a) = b$,*
2. *given $A, B, C \in \mathcal{I}$ and $a \in A, b \in B, c \in C$, if $\phi_{B,C}(b) = c$ and $\phi_{A,B}(a) = b$, then $\phi_{A,C}(a) = c$.*

Proof. In every base B_i number all elements from 1 to r — we shall call this numbers *levels*. For any two sets A, B construct the mapping as follows: $\phi_{A,B}(a) = b$ if a in A has the same number as b in B. If a has number bigger than r. Property 1 is obviously satisfied. Property 2 is also easy: if $\phi_{B,C}(b) = c$, then c has the same level as b, and since $\phi_{A,B}(a) = b$, then a has the same level as b, and therefore a is mapped to c in $\phi_{A,C}$, since a and c have the same level. ☐

We shall call the mappings from above Lemma as *level-mappings.* Suppose we obtained the family $\phi_{B,B'}$ of mappings between $B_1, B_2, ..., B_m$ using the above construction. Let γ_i^e be an atom of element e. Let $L(\gamma_i^e) = \left\{\gamma_{i'}^f \middle| \phi_{B_{i'},B_i}(f) = e\right\}$ be the set of atoms that are mapped to e in B_i, i.e., atoms $\gamma_{i'}^f$ for which f has in $B_{i'}$ the same level as e in B_i. Note that it can happen that other atoms of e can belong to $\Gamma(\gamma_i^e)$. Also, note that from the way we construct the level-mapping we have a symmetry of $N(\cdot)$, i.e., if $\gamma \in N(\gamma')$, then $\gamma' \in N(\gamma)$.

Definition 61. [Blocking set] For an atom γ^e let us define set $\Gamma(\gamma^e) = N(\gamma^e) \cup A(e) \setminus \gamma^e$, and call it a *blocking set* of atom γ^e. Recall that $A(e)$ are all atoms of element e.

Lemma 62. *The sum of all atoms from $\Gamma(\gamma_i^e)$ is at most $1 - \gamma_i^e + x_e$, i.e.,*

$$\sum_{\gamma \in \Gamma(\gamma_i^e)} \gamma \le 1 - 2\gamma_i^e + x_e$$

Proof. Since $\Gamma(\gamma^e) = N(\gamma^e) \cup A(e) \setminus \gamma^e$ we have

$$\sum_{\gamma \in \Gamma(\gamma_i^e)} \gamma \le \sum_{\gamma \in N(\gamma_i^e)} \gamma + \sum_{\gamma \in A(e) - \gamma_i^e} \gamma.$$

Recall that the weight of $\gamma_{i'}^{[.]}$ is $\beta_{i'}$The first sum $\sum_{\gamma \in N(\gamma_i^e)} \gamma$ can be bounded by $1 - \gamma_i^e$, since every base $B_{i'}$ such that $i' \neq i$ contributes at most one time the term $\beta_{i'}$ — Property 1 from Lemma 60. And we have $\sum_{i' \neq i} \beta_{i'} + \beta_i = 1$, from the definition of convex decomposition. We also have $\sum_{\gamma \in A(e) - \gamma_i^e} \gamma = x_e - \gamma_i^e$, since $\sum_{\beta_i : e \in B_i} \beta_i = x_e$. ☐

7.2 The algorithm

Algorithm 7 Rounding procedure with negative correlation

1: Find the convex decomposition $x = \sum_{i=1}^{m} \beta_i \mathbf{1}_{B_i}$
2: Define atoms γ_i^e for every B_i such that $e \in B_i$; initially mark every atom as *active*
3: Find the level-mappings between every two sets of $B_1, ..., B_m$
4: For every atom γ_i^e define its dummy atom of weight $1 + x_e - \Gamma(\gamma_i^e)$; mark it as *inactive*
5: Create a dummy element d, consisting of all dummy atoms; let $x_d = \sum_{\gamma \in D} \gamma$; $E \leftarrow E + d$
6: Let $\Sigma = \sum_{e \in E} x_e$
7: **while** there are active atoms **do**
8:    pick either an element e with probability $\frac{x_e}{\Sigma}$ or a dummy
9:    pick an atom γ_i^e with probability $\frac{\gamma_i^e}{x_e}$
10:    **if** γ_i^e is active **then**
11:       take γ_i^e
12:       and mark γ_i^e as inactive
13:    **end if**
14:    mark all atoms in $\Gamma(\gamma_i^e)$ inactive
15: **end while**
16: **return** elements whose atoms were taken

Lemma 62 states that $\sum_{\gamma \in \Gamma(\gamma_i^e)} \gamma \leq 1 - 2\gamma_i^e + x_e$. However, to simplify the analysis we shall introduce a dummy atom $\bar{\gamma}_i^e$of weight $2 - \gamma^e - \sum_{\gamma \in \Gamma(\gamma_i^e)} \gamma \geq 0$ for each γ_i^e, and insert it into $\Gamma(\gamma_i^e)$; symmetrically we set $\Gamma(\bar{\gamma}_i^e) = \{\gamma_i^e\}$. The new set of blocking atoms of $\Gamma(\gamma_i^e)$ has now total weight of exactly $2 - \gamma^e$. Note that the weight of $\bar{\gamma}_i^e$ is greater than 0, so it is properly defined. We denote the set of all dummy atoms by D. We create a dummy compound element d that consists of all dummy atoms from D. Its corresponding x value is $x_d = \sum_{\gamma \in D} \gamma$. We create this element only to simplify the algorithm, to not explicitly distinguish between dummy and real atoms.

The algorithm is presented in Figure 7.

7.2.1 Analysis of correctness and approximation

Let us first quickly observe that the solution consists of at most r elements. It is because for every number $j \in \{1, ..., l\}$ we take at most one atom whose number in his set is j, and so at most l elements in total. Thus the hard capacity constraint is satisfied.

As for the approximation guarantee. The derivation in this paragraph is basically the analysis from Chapter 4. We just rephrase it in a single matroid case using atoms for a better presentation of the ideas that are coming after.

We are going to analyze the probability that a given atom γ_i^e was chosen in line 9 and was still active. In such a case we shall say that we have *taken an atom*.

We can focus on the atoms because

$$\mathbb{P}\left[e \text{ taken}\right] = \sum_{\gamma_i^e \in A(e)} \mathbb{P}\left[\gamma_i^e \text{ taken}\right],$$

where the equality follows from the fact that we can always take at most one atom. And this is because atoms of the same element belong to each other's blocking sets — recall that $\Gamma\left(\gamma^e\right) = N(\gamma^e) \cup A(e) \setminus \gamma^e$. We shall show that $\mathbb{P}\left[\gamma_i^e \text{ taken}\right] = \frac{1}{2}\gamma_i^e$ and this will give us that

$$\mathbb{P}\left[e \text{ taken}\right] = \sum_{\gamma_i^e \in A(e)} \frac{1}{2}\gamma_i^e = \frac{1}{2}x_e.$$

We index every iteration of the while loop starting with $t = 0$ being a moment before the first iteration. Consider the following sequence $A^0, A^1, A^2 ...$, where A^t is a random variable saying $(A^t = 1)$ if atom γ_i^e is still active after step t; initially $A^0 = 1$. We also consider another sequence $P^0, P^1, ...$, where P^t says if γ_i^e was taken at step t or before; initially $P^0 = 0$. Let τ be the first moment when atom γ_i^e became inactive, i.e., when $A^\tau = 0$ for the first time. That is the moment when we stopped considering possibility of taking γ_i^e anymore. Then the expected value $\mathbb{E}\left[P^\tau\right]$ is exactly equal to $\mathbb{P}\left[\gamma_i^e \text{ taken}\right]$. To analyze $\mathbb{E}\left[P^\tau\right]$ we use a martingale-based argument together with Doob's optional stopping theorem.

Suppose the while loop already made t iterations, and assume that still $A^t = 1$, and $P^t = 0$. Let $\mathcal{F}^t$ be the filtration of the process, which includes the whole information about what happened in steps $0, 1, 2, ..., t$. Now let us ask the following question: given that $P^t = 0, A^t = 1$ what is the probability that $P^{t+1} - P^t = 1$, i.e., that we take atom γ_i^e in step $t+1$? It is easy to see that the probability of choosing an atom γ_i^e is just

$$\frac{x_e}{\Sigma} \cdot \frac{\gamma_i^e}{x_e} = \frac{\gamma_i^e}{\Sigma},$$

where the first term $\frac{x_e}{\Sigma}$ is the probability that e is chosen, and $\frac{\gamma_i^e}{x_e}$. To drop the condition on $A^t = 1$, we need to just add appropriate term in the above formula — the probability that γ_i^e is taken is in fact

$$\mathbb{E}\left[P^{t+1} - P^t \middle| \mathcal{F}^t\right] = \frac{A^t}{\Sigma} \cdot \gamma_i^e.$$

What is the probability that γ_i^e becomes inactive, i.e., taken or blocked? That is, $A^t - A^{t+1} = 1$? When $A^t = 0$, then this probability is just 0, because γ_i^e is already inactive. So suppose that $A^t = 1$. Probability that it is taken is as above. To block γ_i^e we need to pick an atom γ such that $\gamma_i^e \in \Gamma\left(\gamma\right)$. From symmetry of blocking sets, that is exactly the set $\Gamma\left(\gamma_i^e\right)$, whose total weight is $2 - \gamma^e$. Hence

$$\mathbb{E}\left[A^t - A^{t+1} \middle| \mathcal{F}^t\right] = \frac{A^t}{\Sigma} \cdot \left(\gamma^e + \Gamma\left(\gamma^e\right)\right) = \frac{A^t}{\Sigma} \cdot 2.$$

Important to note is that an atom $\gamma \in \Gamma\left(\gamma_i^e\right)$ could be deactivated long time ago, but it still can block atom γ_i^e. We need this wasteful approach to have exact values of the probabilities in order to prove negative correlation later on.

Therefore we can say that

$$\mathbb{E}\left[\left(P^{t+1}-P^{t}\right)-\frac{\gamma_i^e}{2}\cdot\left(A^{t}-A^{t+1}\right)\middle|\mathcal{F}^t\right]=\mathbb{E}\left[P^{t+1}-P^{t}\middle|\mathcal{F}^t\right]-\frac{\gamma_i^e}{2}\cdot\mathbb{E}\left[A^{t}-A^{t+1}\middle|\mathcal{F}^t\right]$$
$$=\gamma_i^e\cdot\frac{A^t}{\Sigma}-\frac{\gamma_i^e}{2}\cdot\frac{A^t}{\Sigma}\cdot 2=0.$$

This means that the sequence

$$\left(P^t+\gamma_i^e\cdot A^t\right)_{t\geq 0}$$

is a martingale. τ, i.e., the moment when γ_i^e became inactive, is a stopping moment, and therefore from Doob's stopping Theorem we have that

$$\mathbb{E}\left[P^{\tau}+\frac{\gamma_i^e}{2}\cdot A^{\tau}\right]=P^0+\frac{\gamma_i^e}{2}\cdot A^0=\frac{\gamma_i^e}{2},$$

where the last equality follows from $P^0=0$, $A^0=1$. Since $A^\tau=0$ we conclude that $\mathbb{E}\left[P^\tau\right]=\frac{\gamma_i^e}{2}$.

Fact 63. *We can actually refine the analysis to get approximation ratio of* $\mathbb{P}\left[\hat{x}_e=1\right]=\frac{x_e}{1+x_e}$*, and also the proof of negative correlation would carry over in this case. However, it does not significantly improve the approximation factor, and therefore we shall focus on the version with* $\mathbb{P}\left[\hat{x}_e=1\right]=\frac{1}{2}x_e$.

7.3 Negative correlation

7.3.1 Simple case: two elements

Let us first show pairwise negative correlation to outline all the necessary ideas behind the proof of negative correlation between any subset of elements.

As shown in previous section, for every atom γ_i^e we have

$$\mathbb{P}\left[\gamma_i^e \text{ taken}\right]=\frac{1}{2}\gamma_i^e,$$

and also $\mathbb{P}\left[\hat{x}_e=1\right]=\mathbb{P}\left[e\text{ taken}\right]=\frac{1}{2}x_e$. Negative correlation of two elements is defined by the following inequality that we need to prove:

$$\mathbb{P}\left[e\text{ and }f\text{ taken}\right]\leq\mathbb{P}\left[e\text{ taken}\right]\mathbb{P}\left[f\text{ taken}\right]=\frac{1}{4}x_e x_f.$$

As before we focus our attention on the atoms. Consider atom γ^e of element e, and atom γ^f of element f — we are not going to write $\gamma_i^e,\gamma_{i'}^f$, i.e., we do not use subscripts indicating set B_i, because the actual base will not important here. We want to calculate the probability that both γ^e and γ^f are taken. There are two cases here.

First, if it is the case that $\gamma^e\in\Gamma\left(\gamma^f\right)$ (and by symmetry that $\gamma^f\in\Gamma\left(\gamma^e\right)$), then obviously the probability of taking both these atoms is zero, because if we would take one it would immediately mark another as inactive.

So suppose that $\gamma^e \notin \Gamma\left(\gamma^f\right)$. In this case atoms γ^e and γ^f have different levels in the bases to which they belong. This means that the $L(\gamma^e)$ part of $\Gamma(\gamma^e)$ and $L(\gamma^f)$ of $\Gamma\left(\gamma^f\right)$ are disjoint — recall that $\Gamma(\gamma^e) = L(\gamma^e) \cup A(e) \setminus \gamma^e$. If it would be the case that for all atoms, always both sets $\Gamma(\gamma^e)$ $\Gamma\left(\gamma^f\right)$ would be disjoint, then we would be able to show that $\mathbb{P}\left[\gamma^e \text{ and } \gamma^f \text{ both taken}\right] = \frac{1}{4}\gamma^e\gamma^f$. From which we would just conclude that

$$\begin{aligned}\mathbb{P}[e \text{ and } f \text{ taken}] = \sum_{\gamma^e \in A(e)} \sum_{\gamma^f \in A(f)} \mathbb{P}\left[\gamma^e \text{ and } \gamma^f \text{ taken}\right] \\ \leq \sum_{\gamma^e \in A(e)} \sum_{\gamma^f \in A(f)} \frac{1}{4}\gamma^e\gamma^f = \frac{1}{4}\left(\sum_{\gamma^e \in A(e)} \gamma^e\right)\left(\sum_{\gamma^f \in A(f)} \gamma^f\right) = \frac{1}{4}x_e x_f.\end{aligned}$$

However, often it might be the case that $\Gamma(\gamma^e) \cap \Gamma\left(\gamma^f\right) \neq \emptyset$, which makes the probability of taking both γ^e and γ^f actually bigger than $\frac{1}{4}\gamma^e\gamma^f$. Hence, in the proof we need to balance this deviation from $\frac{1}{4}\gamma^e\gamma^f$ by the number of atoms for which $\mathbb{P}\left[\gamma^e \text{ and } \gamma^f \text{ taken}\right]$ is zero due to having the same levels.

So suppose that γ^e and γ^f have different levels, but $\Gamma(\gamma^e) \cap \Gamma\left(\gamma^f\right) \neq \emptyset$, in which case it has to be the that $A(e) \cap \Gamma\left(\gamma^f\right) \neq \emptyset$ or $A(f) \cap \Gamma(\gamma^e) \neq \emptyset$. Therefore, let us denote by $\Gamma_e\left(\gamma^f\right)$ the atoms of $A(e) \cap \Gamma\left(\gamma^f\right)$, and $\Gamma_f(\gamma^e) = A(f) \cap \Gamma(\gamma^e)$. For simplicity, let us denote also $\Gamma_e\left(\gamma^f\right)$ the total weight of the atoms that belong to it.

Lemma 64. *If* $\gamma^e \notin \Gamma\left(\gamma^f\right)$*, then*

$$\mathbb{P}\left[\gamma^e \textit{ taken first,} \gamma^f \textit{ taken second}\right] = \frac{1}{2}\frac{\gamma^e\gamma^f}{4 - \Gamma_f(\gamma^e) - \Gamma_e(\gamma^f)}.$$

Proof. Let us consider a sequence $A^0, A^1, ...$ where A^t says that both γ^e and γ^f are still active, i.e., $A^t = 1$ if at step t both atoms are still active, otherwise $A^t = 0$; we start with $A^0 = 1$. Let $P^0, P^1, ...$, be a sequence such that $P^t = 1$, if γ^e is taken at step t and γ^f is still active, and $P^t = 0$ otherwise; we start with $P^0 = 0$.

Consider step t, and assume that $A^t = 1$ meaning that both γ^e and γ^f are still active. Also assume that $P^t = 0$ meaning that we did not take γ^e yet. Let $\mathcal{F}^t$ represent all the history up to step t. What is the value of $\mathbb{E}\left[P^{t+1} - P^t \middle| \mathcal{F}^t\right]$, i.e., the probability that we take γ^e in step $t+1$? It is

$$\frac{A^t}{\Sigma}\gamma^e.$$

What is the probability that one of the elements becomes inactive at step $t+1$, i.e., $\mathbb{E}\left[A^t - A^{t+1} \middle| \mathcal{F}^t\right]$? When this happens we know if either we have taken γ^e with γ^fstill active, or we lost this possibility. There are two cases in which this occurs. First, if we select γ^e or γ^f, which happens with probability

$$\frac{A^t}{\Sigma}\left(\gamma^e+\gamma^f\right).$$

Or if we choose an atom from $\Gamma\left(\gamma^e\right)\cup\Gamma\left(\gamma^f\right)$, and this happens with probability

$$\frac{A^t}{\Sigma}\left(\Gamma\left(\gamma^e\right)+\Gamma\left(\gamma^f\right)-\Gamma_f\left(\gamma^e\right)-\Gamma_e\left(\gamma^f\right)\right).$$

Hence, the total probability of making one of γ^e, γ^f inactive is

$$\begin{aligned}&\mathbb{E}\left[A^t-A^{t+1}\middle|\mathcal{F}^t\right]\\=&\frac{A^t}{\Sigma}\left(\gamma^e+\gamma^f+\Gamma\left(\gamma^e\right)+\Gamma\left(\gamma^f\right)-\Gamma_f\left(\gamma^e\right)-\Gamma_e\left(\gamma^f\right)\right)\\=&\frac{A^t}{\Sigma}\left(4-\Gamma_f\left(\gamma^e\right)-\Gamma_e\left(\gamma^f\right)\right).\end{aligned}$$

So we get that

$$\mathbb{E}\left[\left(P^{t+1}-P^t\right)-\frac{\gamma^e}{4-\Gamma_f\left(\gamma^e\right)-\Gamma_e\left(\gamma^f\right)}\left(A^t-A^{t+1}\right)\middle|\mathcal{F}^t\right]=0$$

which means that the sequence

$$\left(P^t+\frac{\gamma^e}{4-\Gamma_f\left(\gamma^e\right)-\Gamma_e\left(\gamma^f\right)}A^t\right)_{t\geq 0}$$

is a martingale. Let τ_e be the moment when first of atoms γ^e or γ^f becomes inactive. By repeating the argument based on Doob's stopping theorem we get that

$$\begin{aligned}\mathbb{E}\left[P^{\tau_e}\right]=\mathbb{E}&\left[P^{\tau_e}+\frac{\gamma^e}{4-\Gamma_f\left(\gamma^e\right)-\Gamma_e\left(\gamma^f\right)}A^{\tau_e}\right]\\&=P^0+\frac{\gamma^e}{4-\Gamma_f\left(\gamma^e\right)-\Gamma_e\left(\gamma^f\right)}A^0=\frac{\gamma^e}{4-\Gamma_f\left(\gamma^e\right)-\Gamma_e\left(\gamma^f\right)},\end{aligned}$$

and hence

$$\mathbb{P}\left[\gamma^e\text{ taken, }\gamma^f\text{still active}\right]=\mathbb{E}\left[P^{\tau_e}\right]=\frac{\gamma^e}{4-\Gamma_f\left(\gamma^e\right)-\Gamma_e\left(\gamma^f\right)}.$$

Now consider the sequence $Q^{\tau_e}, Q^{\tau_e+1}, \ldots$ that starts at the moment when the γ^e is taken. We condition on the fact that γ^f is still active, and so we can analyze now the probability of taking γ^f. From the argument of the previous subsection we know that the probability of taking f later on is equal to $\frac{1}{2}\gamma^f$ — note that right now atoms of e can also block γ^f. Therefore the probability that we take first γ^e and then γ^f is equal to

$$\begin{aligned}&\mathbb{P}\left[\gamma^e\text{ taken first,}\gamma^f\text{ second}\right]\\=&\mathbb{P}\left[\gamma^e\text{ taken, }\gamma^f\text{still active}\right]\cdot\mathbb{P}\left[\gamma^f\text{ taken}|\gamma^e\text{ taken, }\gamma^f\text{still active}\right]\\=&\frac{\gamma^e}{4-\Gamma_f\left(\gamma^e\right)-\Gamma_e\left(\gamma^f\right)}\cdot\frac{\gamma^f}{2}\\=&\frac{1}{2}\cdot\frac{\gamma^e\gamma^f}{4-\Gamma_f\left(\gamma^e\right)-\Gamma_e\left(\gamma^f\right)}.\end{aligned}$$

□

Now we can prove the negative correlation. Let us use the following notation: $\gamma^e \sim \gamma^f$ to denote that atoms do not conflict each other, i.e., if they have different levels, equivalently if $\gamma^f \notin \Gamma(\gamma^e)$. Let us analyze the probability that some atom of e is taken first, and then some atom of f is taken. From the above Lemma:

$$\begin{aligned}\mathbb{P}\left[e \text{ taken first, } f \text{ taken second}\right] &= \sum_{(\gamma^e,\gamma^f):\gamma^f\sim\gamma^e} \mathbb{P}\left[\gamma^e \text{ first}, \gamma^f \text{ second}\right] \\ &= \sum_{(\gamma^e,\gamma^f):\gamma^f\sim\gamma^e} \frac{1}{2}\frac{\gamma^e\gamma^f}{4-\Gamma_f(\gamma^e)-\Gamma_e(\gamma^f)}.\end{aligned}$$

Now we use Harmonic-Arithmetic mean inequality $\frac{2}{a+b} \le \frac{1}{2}\left(\frac{1}{a}+\frac{1}{b}\right)$ to get

$$\begin{aligned}&\sum_{(\gamma^e,\gamma^f):\gamma^f\sim\gamma^e} \frac{1}{2}\frac{\gamma^e\gamma^f}{4-\Gamma_f(\gamma^e)-\Gamma_e(\gamma^f)} \\ \le &\sum_{(\gamma^e,\gamma^f):\gamma^f\sim\gamma^e} \frac{1}{2}\gamma^e\gamma^f\cdot\frac{1}{4}\left(\frac{1}{2-\Gamma_f(\gamma^e)}+\frac{1}{2-\Gamma_e(\gamma^f)}\right)\end{aligned}$$

Let us bound the first-term part after opening the brackets:

$$\begin{aligned}&\frac{1}{2}\cdot\frac{1}{4}\sum_{(\gamma^e,\gamma^f):\gamma^f\sim\gamma^e}\gamma^e\gamma^f\frac{1}{2-\Gamma_f(\gamma^e)} \\ =&\frac{1}{2}\cdot\frac{1}{4}\sum_{\gamma^e}\frac{\gamma^e}{2-\Gamma_f(\gamma^e)}\sum_{\gamma^f:\gamma^f\sim\gamma^e}\gamma^f \\ =&\frac{1}{2}\cdot\frac{1}{4}\sum_{\gamma^e}\frac{\gamma^e}{2-\Gamma_f(\gamma^e)}\left(x_f-\Gamma_f(\gamma^e)\right) \\ \le&\frac{1}{2}\cdot\frac{1}{4}\sum_{\gamma^e}\frac{\gamma^e}{2}\cdot x_f = \frac{1}{16}x_e x_f,.\end{aligned}$$

The second equality comes from the fact that $\sum_{\gamma^f:\gamma^f\sim\gamma^e}\gamma^f = x_f - \Gamma_f(\gamma^e)$, i.e., we sum over all atoms of f except those that have the same level as γ^e. The inequality follows from $\frac{x-a}{2-a} \le \frac{x}{2}$ for $a \in [0,x]$. Similarly we bound the second sum by $\frac{1}{16}x_e x_f$, and so in total we get that

$$\mathbb{P}\left[e \text{ taken first, } f \text{ taken second}\right] \le \frac{1}{8}x_e x_f,$$

and since we have two possible orderings of e and f we get that:

$$\mathbb{P}\left[e \text{ taken, } f \text{ taken}\right] \le \frac{1}{4}x_e x_f = \mathbb{P}\left[e \text{ taken}\right]\mathbb{P}\left[f \text{ taken}\right].$$

7.3.2 Arbitrary number of elements

Consider elements $e_1, ..., e_k$, and any of their atoms $\gamma^{e_1}, \gamma^{e_2}, ..., \gamma^{e_k}$. If for any two e_i, e_j we have $\gamma^{e_i} \in \Gamma(\gamma^{e_j})$, then $\mathbb{P}[\gamma^{e_1}, \gamma^{e_2}, ..., \gamma^{e_k} \text{ are taken}] = 0$. So suppose they do not conflict each other. We want to repeat the argument from previous paragraph. First, what is the probability that we shall take γ^{e_1} while all others are still active? It is

$$\frac{\gamma^{e_1}}{2k - \sum_{i=1}^{k} \sum_{j \neq i} \Gamma_{e_i}(\gamma^{e_j})}.$$

Assuming this happened, later the probability that we take γ^{e_2}while $\gamma^{e_3}, ..., \gamma^{e_k}$ are still active is

$$\frac{\gamma^{e_2}}{2(k-1) - \sum_{i=2}^{k} \sum_{j \neq 1,i} \Gamma_{e_i}(\gamma^{e_j})}.$$

And analogically for the rest of the atoms. Therefore the probability that we take all $\gamma^{e_1}, \gamma^{e_2}, ..., \gamma^{e_k}$ in exactly this order is precisely equal to

$$\gamma^{e_1} \cdot \gamma^{e_2} \cdot ... \cdot \gamma^{e_k} \cdot \frac{1}{2k - \sum_{i=1}^{k} \sum_{j \neq i} \Gamma_{e_i}(\gamma^{e_j})} \cdot \frac{1}{2(k-1) - \sum_{i=2}^{k} \sum_{j \neq 1,i} \Gamma_{e_i}(\gamma^{e_j})} \cdot$$
$$\cdot \frac{1}{2(k-2) - \sum_{i=3}^{k} \sum_{j \neq 1,2,i} \Gamma_{e_i}(\gamma^{e_j})} \cdot ... \cdot \frac{1}{4 - \Gamma_{e_{k-1}}(\gamma^{e_k}) - \Gamma_{e_k}(\gamma^{e_{k-1}})} \cdot \frac{1}{2}.$$

Let us denote by $\sim (\gamma^{e_1}, ..., \gamma^{e_k})$ the fact that atoms $\gamma^{e_1}, ..., \gamma^{e_k}$ pairwise not conflict each other. Hence, the total probability of taking $e_1, e_2, ..., e_k$ in exactly this order is equal to

$$\sum_{\gamma^{e_1}} \sum_{\gamma^{e_2} : \sim(\gamma^{e!}, \gamma^{e_2})} \sum_{\gamma^{e_3} : \sim(\gamma^{e_1}, ..., \gamma^{e_3})} ... \sum_{\gamma^{e_k} : \sim(\gamma^{e_1}, ..., \gamma^{e_k})} \frac{\gamma^{e_1}}{2k - \sum_{i=1}^{k} \sum_{j \neq i} \Gamma_{e_i}(\gamma^{e_j})} \cdot$$
$$\cdot \frac{\gamma^{e_2}}{2(k-1) - \sum_{i=2}^{k} \sum_{j \neq 1,i} \Gamma_{e_i}(\gamma^{e_j})} \cdot$$
$$\cdot \frac{\gamma^{e_3}}{2(k-2) - \sum_{i=3}^{k} \sum_{j \neq 1,2,i} \Gamma_{e_i}(\gamma^{e_j})} \cdot$$
$$\cdot ... \cdot$$
$$\cdot \frac{\gamma^{e_{k-1}}}{4 - \Gamma_{e_{k-1}}(\gamma^{e_k}) - \Gamma_{e_k}(\gamma^{e_{k-1}})} \cdot$$
$$\cdot \frac{\gamma^{e_k}}{2}.$$

To each factor we apply AM-HM inequality — $\frac{1}{a_1+...+a_n} \le \frac{1}{n^2}\left(\frac{1}{a_1}+...+\frac{1}{a_n}\right)$— to upperbound it by:

$$\sum_{\gamma^{e_1}} \sum_{\gamma^{e_2}:\sim(\gamma^{e!},\gamma^{e_2})} \sum_{\gamma^{e_3}:\sim(\gamma^{e_1},...,\gamma^{e_3})} \cdots \sum_{\gamma^{e_k}:\sim(\gamma^{e_1},...,\gamma^{e_k})} \frac{\gamma^{e_1}}{k^2}\left(\sum_{j=1}^{k} \frac{1}{2-\sum_{i\ge 1, i\ne j}\Gamma_{e_i}\left(\gamma^{e_j}\right)}\right) \cdot$$
$$\cdot \frac{\gamma^{e_2}}{(k-1)^2}\left(\sum_{j=2}^{k} \frac{1}{2-\sum_{i\ge 2, i\ne j}\Gamma_{e_i}\left(\gamma^{e_j}\right)}\right) \cdot$$
$$\cdot \frac{\gamma^{e_3}}{(k-3)^2}\left(\sum_{j=3}^{k} \frac{1}{2-\sum_{i\ge 3, i\ne j}\Gamma_{e_i}\left(\gamma^{e_j}\right)}\right) \cdot$$
$$\cdot \ldots \cdot$$
$$\cdot \frac{\gamma^{e_{k-1}}}{2^2}\left(\frac{1}{2-\Gamma_{e_k}\left(\gamma^{e_{k-1}}\right)} + \frac{1}{2-\Gamma_{e_{k-1}}\left(\gamma^{e_k}\right)}\right) \cdot$$
$$\cdot \frac{\gamma^{e_k}}{1^2}\left(\frac{1}{2}\right).$$

Note $(k!)^2$ in the denominator. After opening the brackets we will get $k!$ sum of products, each product corresponding to some permutation of set $\{1, ..., k\}$. The first (lexicographically) sum being:

$$\frac{1}{(k!)^2} \sum_{\gamma^{e_1}} \sum_{\gamma^{e_2}:\sim(\gamma^{e!},\gamma^{e_2})} \sum_{\gamma^{e_3}:\sim(\gamma^{e_1},...,\gamma^{e_3})} \cdots \sum_{\gamma^{e_k}:\sim(\gamma^{e_1},...,\gamma^{e_k})}$$
$$\frac{\gamma^{e_1}}{2-\sum_{i=2}^{k}\Gamma_{e_i}\left(\gamma^{e_1}\right)} \cdot \frac{\gamma^{e_2}}{2-\sum_{i=3}^{k}\Gamma_{e_i}\left(\gamma^{e_2}\right)} \cdot \ldots \cdot \frac{\gamma^{e_{k-1}}}{2-\Gamma_{e_k}\left(\gamma^{e_{k-1}}\right)} \frac{\gamma^{e_k}}{2}.$$

We can start wrapping this from the last atom. Note that none of the denominators depends on γ^{e_k}, so the above can be written as:

$$\frac{1}{2}\frac{1}{(k!)^2} \sum_{\gamma^{e_1}} \sum_{\gamma^{e_2}:\sim(\gamma^{e!},\gamma^{e_2})} \sum_{\gamma^{e_3}:\sim(\gamma^{e_1},...,\gamma^{e_3})} \cdots \sum_{\gamma^{e_{k-1}}:\sim(\gamma^{e_1},...,\gamma^{e_{k-1}})} \cdot$$
$$\cdot \frac{\gamma^{e_1}}{2-\sum_{i=2}^{k}\Gamma_{e_i}\left(\gamma^{e_1}\right)} \cdot \frac{\gamma^{e_2}}{2-\sum_{i=3}^{k}\Gamma_{e_i}\left(\gamma^{e_2}\right)} \cdot \ldots \cdot \frac{\gamma^{e_{k-1}}}{2-\Gamma_{e_k}\left(\gamma^{e_{k-1}}\right)} \cdot \sum_{\gamma^{e_k}:\sim(\gamma^{e_1},...,\gamma^{e_k})} \gamma^{e_k}.$$

Since we sum over γ^{e_k} that do not conflict with $\gamma^{e_1}, ..., \gamma^{e_{k-1}}$, then it means that

$$\sum_{\gamma^{e_k}:\sim(\gamma^{e_1},...,\gamma^{e_k})} \gamma^{e_k} = x_{e_k} - \sum_{j\le k-1} \Gamma_{e_k}\left(\gamma^{e_j}\right).$$

Inequality $\frac{x-a}{y-a} \leq x$, for $a \leq x \leq y$, allows us get rid of all $\Gamma_{e_k}(\cdot)$ in the denominators, after applying it $k-1$ to each denominator, i.e.,

$$\begin{aligned}
&\frac{1}{2}\frac{1}{(k!)^2}\sum_{\gamma^{e_1}}\sum_{\gamma^{e_2}:\sim(\gamma^{e!},\gamma^{e_2})}\sum_{\gamma^{e_3}:\sim(\gamma^{e_1},\ldots,\gamma^{e_3})}\cdots\sum_{\gamma^{e_{k-1}}:\sim(\gamma^{e_1},\ldots,\gamma^{e_{k-1}})}\\
&\quad\cdot\frac{\gamma^{e_1}}{2-\sum_{i=2}^{k}\Gamma_{e_i}(\gamma^{e_1})}\cdot\frac{\gamma^{e_2}}{2-\sum_{i=3}^{k}\Gamma_{e_i}(\gamma^{e_2})}\cdot\ldots\cdot\frac{\gamma^{e_{k-1}}}{2-\Gamma_{e_k}(\gamma^{e_{k-1}})}\cdot\left(x_{e_k}-\sum_{j\leq k-1}\Gamma_{e_k}(\gamma^{e_j})\right)\\
\leq\ &\frac{1}{2}\frac{1}{(k!)^2}\sum_{\gamma^{e_1}}\sum_{\gamma^{e_2}:\sim(\gamma^{e!},\gamma^{e_2})}\sum_{\gamma^{e_3}:\sim(\gamma^{e_1},\ldots,\gamma^{e_3})}\cdots\sum_{\gamma^{e_{k-1}}:\sim(\gamma^{e_1},\ldots,\gamma^{e_{k-1}})}\\
&\quad\cdot\frac{\gamma^{e_1}}{2-\sum_{i=2}^{k}\Gamma_{e_i}(\gamma^{e_1})}\cdot\frac{\gamma^{e_2}}{2-\sum_{i=3}^{k}\Gamma_{e_i}(\gamma^{e_2})}\cdot\ldots\cdot\frac{\gamma^{e_{k-1}}}{2-\Gamma_{e_k}(\gamma^{e_{k-1}})}\cdot\frac{x_{e_k}}{2}.
\end{aligned}$$

Now we note that the expression no denominator depends on $\gamma^{e_{k-1}}$, and so we can wrap this up just like we did with γ^{e_k}, and obtain extra factor of $\frac{x_{e_{k-1}}}{2}$ instead of the sum over $\gamma^{e_{k-1}}$. Repeating this up to γ^{e_1} will leave us with just $\frac{1}{(k!)^2}\frac{1}{2^k}x_{e_1}\cdot x_{e_2}\cdot\ldots\cdot x_{e_k}$. Doing so with each of the $k!$ sums will yield an upperbound on them of $\frac{1}{k!}\frac{1}{2^k}x_{e_1}\cdot x_{e_2}\cdot\ldots\cdot x_{e_k}$ — an upperbound on the probability that $e_1, e_2, \ldots, e_k$ will be all taken exactly in this order. Taking also into account that we need to consider all $k!$ possible orderings of $e_1, e_2, \ldots, e_k$ gives us finally that

$$\mathbb{P}\left[\bigwedge_{i=1}^{k}\hat{x}_{e_i}=1\right]\leq\frac{1}{2^k}x_{e_1}\cdot x_{e_2}\cdot\ldots\cdot x_{e_k}=\prod_{i=1}^{k}\mathbb{P}\left[\hat{x}_{e_i}=1\right].$$

Chapter 8

Stochastic Universal Optimization

8.1 Set cover

We are given a universe $E = \{1, ..., n\}$ and collection of E's subsets $\mathcal{S} = \{S_1, S_2, ..., S_m\}$ with weight function $c : \mathcal{S} \to \mathbb{R}_+$. We are given a distribution π on E's subsets. We assuming that the support of π is over a polynomial number of scenarios. If we would like handle distributions given by a black-box, then we can resort to two-possible techniques. If we want to use primal-dual scheme, then we can use sampling average approximation technique of Charikar et al. [14] that approximates within factor $(1 + \varepsilon)$ the proper convex programs with black-box distributions with convex programs using a distribution of polynomial support. Second option is to just estimate the objective functions if we would like to solve corresponding convex programs.

In the problem we need to find a mapping $\phi : E \to \mathcal{S}$. Once we choose the mapping, a set $A \leftarrow \pi$ is drawn, and we cover set A with subsets $\{S_j\}_{j \in \phi(A)}$ (no set is repeated). The goal is to find a mapping that minimizes the expected cost of a cover constructed. We compare ourselves with an optimal such assignment ϕ^{OPT}.

Therefore, the following integer program solves the problem:

$$
\begin{array}{lll}
\min & \sum_{A \in \pi} \pi(A) \left(\sum_S c(S) \cdot \mathbf{1}\left[\exists_{e \in A} z_{e,S} = 1 \right] \right) & \\
\text{s.t.} & \sum_{S : e \in S} x_{e,S} = 1 & \forall_{e \in E} \\
 & x_{e,S} \in \{0, 1\} & \forall_e \forall_{S \ni e},
\end{array}
$$

where variable $x_{e,S}$ represents if $\phi(e) = S$. We do not use the above LP, it just serves as a formal description of the problem. We shall use a bit more involved program than that.

The main result of this section is the following.

Theorem 65. *There exists an algorithm which approximates the optimal universal solution within a factor of $O(\lg n)$.*

We are able to give actually two algorithms with this approximation ratio, one based on randomized rounding of linear programs, and another based on a primal-dual scheme. Both however, use the same linear program that we introduce below.

Configuration LP Consider variable y_B^S which we interpret as $y_B^S = 1 \leftrightarrow B = \{e \in S \,|\, \phi(e) = S\}$, i..e, B are all the elements that map to S. The following is the relaxation of the integer program that solves our problem.

$$\begin{array}{lllr} \min & \sum_S c(S) \sum_{B \subseteq S} y_B^S \cdot \left(\sum_{A\in\pi} \pi(A) \mathbf{1}[A \cap B \neq \emptyset]\right) & \text{(CONF-LP)} & \\ \text{s.t.} & \sum_{B \ni e} \sum_{S \supseteq B} y_B^S \geq 1 & \forall_e / \alpha_e & (8.1) \\ & \sum_{B \subseteq S} y_B^S = 1 & \forall_S & \\ & y_B^S \geq 0 & \forall_S \forall_{B\subseteq S}. & \end{array}$$

How to round the LP Suppose we have a solution for the CONF-LP, let it be just $\left(y_B^S\right)$. Each set S chooses at random one subset $B \subseteq S$ with probability y_B^S (empty set is also possible).

Consider element e. What is the probability that it will not be covered in the above phase? It's exactly

$$\prod_{S: e \in S} \left(1 - \sum_{B \subseteq S: e \in B} y_B^S\right) \leq \exp\left(- \sum_{S: e\in S} \sum_{B\subseteq S: e \in B} y_B^S\right) \leq \frac{1}{e},$$

where the last inequality follows from the LP constraint.

What is the cost of sets taken in this phase? On expectation it's exactly equal to the value of the LP solution.

Therefore, repeat this phase $O(\lg n)$ times, and we get a log-approximation with high probability.

How to solve the LP We can write the objective of CONF-LP in a concise way. Define

$$f(B) = \sum_{A \in \pi} \pi(A) \mathbf{1}[A \cap B \neq \emptyset].$$

Lemma 66. *f is a submodular function.*

Proof. For fixed A function $B \longrightarrow \mathbf{1}[A \cap B \neq \emptyset]$ is submodular. Observe that f is a convex combination of such functions. □

Lemma 67. *We can compute f efficiently.*

Proof. If we have polynomial number of scenarios, then it's trivial. □

Now we have that

$$\sum_{B \subseteq S} y_B^S \cdot \left(\sum_{A \in \pi} \pi(A) \mathbf{1}[A \cap B \neq \emptyset]\right) = \sum_{B\subseteq S} y_B^S \cdot f(B).$$

Constraint 8.1 can be written in the following way

$$\sum_{S \ni e} \sum_{B \subseteq S: e \in B} y_B^S \geq 1.$$

Therefore if we would define $z_e^S = \sum_{B\subseteq S: e\in B} y_B^S$ then we could say that our LP is equal to

$$\begin{array}{lccc} \min & \sum_S c(S) \sum_{B\subseteq S} y_B^S \cdot f(B) & \text{(CONF-LP)} \\ \text{s.t.} & \sum_{S\ni e} z_e^S \geq 1 & \forall_e \\ & \sum_{B\subseteq S: e\in B} y_B^S = z_e^S & \forall_e \forall_{S\ni e} \\ & \sum_{B\subseteq S} y_B^S = 1 & \forall_S \forall_{B\subseteq S.} \end{array}$$

Given the definition $z_e^S = \sum_{B\subseteq S: e\in B} y_B^S$ and constraint $\sum_{B\subseteq S} y_B^S = 1$ we could say that

$$\begin{aligned} & \sum_{B\subseteq S} y_B^S \cdot f(B) \\ \geq \; & \min \left\{ \sum_{B\subseteq S} \alpha_B f(B) \middle| \forall_{e\in S} \sum_{B\ni e} \alpha_B^S = z_e^S ; \sum_{B\subseteq S} \alpha_B = 1 \right\} \qquad (8.2) \\ = \; & f_S^- \left(\left(z_e^S \right)_{e\in S} \right). \end{aligned}$$

the dependence on S in f_S^- is only to stress that coordinates are for $e \in S$.

Therefore we can say that CONF-LP can be written as

$$\begin{array}{lcc} \min & \sum_S c(S) f_S^- \left(\left(z_e^S \right)_{e\in S} \right) & \text{(CONC-LP)} \\ \text{s.t.} & \sum_{S\ni e} z_e^S \geq 1 & \forall_e. \end{array}$$

It happens that f_S^- is called a convex closure of submodular function f. It can be shown (via LP duality) that despite of exponential number of variables, function f_S^- is equal to the so called Lovasz's extension $f_S^L(x) = \mathbb{E}_{\lambda\sim U[0,1]} \left[f_S \left(\{ i : x_i \geq \lambda \} \right) \right]$. Lovasz's extension can be computed efficiently in polynomial time ($(1+\varepsilon)$-approx via sampling). Also f_S^- is a convex function. Therefore in the mathematical program CONC-LP we are to minimize a convex function over a convex set, and this can be done efficiently, for example by the ellipsoid method [32].

8.2 Constrained set multicover

In the Constrained Set Multicover we have a function $r : E \longrightarrow \mathbb{N}$specifying how many times each element needs to be covered. We can take each set from $\mathcal{S}$ at most once. In the stochastic setting each element has to pick a family of $r(e)$ distinct subsets covering him. This variant of set cover can model the design of efficient query cache [5].

On intuitive level, we treat the stochastic version of the problem as an instance of the deterministic version but with exponential number of covering sets and with an additional constraint that no element can be covered with two subsets of some S. Due to the subadditivity of g this condition is equivalent to picking at most one subset of each S at all. For this variant the approach with randomized rounding does not work, and therefore we shall use a primal dual scheme based on the one for deterministic counterpart of the problem, which can be found in, e.g., [50].

We shall use the following linear program

$$\begin{array}{lll} \min & \sum_S c\,(S) \sum_{B \subseteq S} y_B^S \cdot g\,(B) & \text{(MCOV-LP)} \\ \text{s.t.} & \sum_{B \subseteq S: e \in B} y_B^S \geq r(e) & \forall_e \\ & \sum_{B \subseteq S} y_B^S \leq 1 & \forall_S \\ & y_B^S \geq 0 & \forall_{S,\, B \subseteq S}, \end{array}$$

and its dual

$$\begin{array}{lll} \max & \sum_e r(e) w_e - \sum_S z_S & \text{(MCOV-DUAL)} \\ \text{s.t.} & \sum_{e \in B} w_e - z_S \leq c(S) g(B) & \forall_{S,\, B \subseteq S} \\ & w_e \geq 0 & \forall_e \\ & z_S \geq 0 & \forall_S. \end{array}$$

The primal-dual scheme Let us keep a family of sets $(R_s)_{S\in\mathcal{S}}$initialized to $R_s = S$. As long as there are some not sufficiently covered elements we pick a subset of some $R_S$minimizing cost-effectiveness $\hat{c}$.For $B \subseteq S$ it is defined to be $\hat{c}_S(B) = \frac{c(S)g(B)}{|B|}$. When set $B \subseteq S$ is picked we modify $R_s := R_s - B$. Moreover, if an element$e$ gets covered $r(e)$ times we erase it from all $R_S$sets. Note that all used subsets of S are disjoint so no element would be covered by the same S twice.

Let us define $price(e, i)$ to be the cost-effectiveness of set that covered the i-th copy of$e$. We define $j_e^S$to be a number of copy of e covered by a subset of S or$r(e)$ if$e$ has not been covered by any subset of S. We also define

$$\begin{aligned} \alpha_e &= price(e, r(e)) \\ \beta_S &= \sum_e price(e, r(e)) - price(e, j_e^S) \end{aligned}$$

We have $price(e, i) \leq price(e, i+1)$ so $\beta_S \geq 0$. Note that the cost of the solution is$\sum_e \sum_{i=1}^{r(e)} price(e, i) = \sum_e r(e)\alpha_e - \sum_S \beta_S$which is equivalent to the objective function of MCOV-DUAL. We want to show that $(\frac{\alpha}{H_n}, \frac{\beta}{H_n})$ is a feasible solution of this LP.

We need to prove that

$$\sum_{e \in B} \alpha_e - \beta_S = \sum_e price(e, j_e^S) \leq c(S)g(B) \cdot H_n.$$

The summand $price(e, j_e^S)$ is the cost-effectiveness of the set covering$e$ in the moment it got removed from R_S. Let us order the elements of$B$ in the order they were removed from R_S: $e_1, e_2, \ldots, e_k$. Observe that e_i could be covered at that moment by $B_i = e_i, \ldots, e_k$ so $price(e_i, j_{e_i}^S) \leq \frac{c(S)g(B_i)}{|B_i|} \leq \frac{c(S)g(B)}{k-i+1}$. Summing these inequalities proves that $(\frac{\alpha}{H_n}, \frac{\beta}{H_n})$ is a feasible solution of MCOV-DUAL. Therefore the cost of the solution is at most $H_n \cdot OPT_{MCOV-DUAL} \leq H_n \cdot OPT$.

The last thing to deal with is finding subset of R_S minimizing $\hat{c}_S(B) = \frac{c(S)g(B)}{|B|}$. This can be done efficiently due to the following Lemma.

Lemma 68. *If functionh is submodular and $h(\emptyset) \geq 0$ then minimum of$\hat{h}(X) = \frac{h(X)}{|X|}$(defined for non-empty sets) can be found in polynomial time.*

Proof. We will use binary search - it suffices to check if there is a non-empty set satisfying $\frac{h(X)}{|X|} < c$ for given real number$c$. The equivalent question is: if minimum of function $\tilde{h}(X) = h(X) - c|X|$ is negative (note that $\tilde{h}(\emptyset) = h(\emptyset) \geq 0$ so it does not influence the answer). Function $\tilde{h}$ is submodular so this can be answered in polynomial time [32]. □

Theorem 69. *The stochastic universal Constrained Set Multicover admits an approximation of factor $H_n \approx ln(n)$.*

8.3 Metric facility location

We are given clients $E = \{1, ..., n\}$ and facilities $F = \{f_1, f_2, ..., f_m\}$. We need to open subset $I \in [m]$ of facilities and connect each client j to a facility $f = \phi(j)$. Cost of connection (distance) is $d(j, f)$ and we assume that the connection cost satisfies a triangle inequality, i.e., for any two clients j_1, j_2 and any two facilities f_1, f_2 we have $d(j_1, f_2) \leq d(j_1, f_1) + d(j_2, f_1) + d(j_2, f_2)$. Opening facility f costs o_f. We are given a distribution π on the set E of clients; we assume polynomial number of scenarios. After realization of a scenario $A \in \pi$ we open all facilities $\bigcup_{j\in A} \phi(j)$.

The integer program that solves the problem is:

$$\begin{array}{lll} \min & \sum_{A\in\pi} \pi(A) \left(\sum_{f\in F} o_f \cdot \mathbf{1}[\exists_{j\in A} x_{j,f} = 1] + \sum_{j\in A} d(\phi(j), j) \right) & \\ \text{s.t.} & \sum_{f\in F} x_{j,f} = 1 & \forall_{j\in C} \\ & x_{j,f} \in \{0, 1\} & \forall_{j\in C} \forall_{f\in F}. \end{array}$$

Again, the above program just serves as a very formal description of the problem.

The result of this Section is the following Theorem.

Theorem 70. *There exists a 4-approximation algorithm for the stochastic universal facility location, where we compare ourselves to the optimum universal solution.*

Again, in the algorithm we shall use again a configuration LP:

$$\min \sum_{f\in F} o_f \sum_{B\subseteq C} y_B^f \cdot g(B) + \sum_{c\in C} \mathbb{P}[c \in A] \cdot \left(\sum_{f\in F} d(c, f) \sum_{B\ni j} y_B^f \right) \quad \text{(FL-CONF-LP)}$$

$$\text{s.t.} \sum_{i\in F} \sum_{B\ni c} y_B^f \geq 1 \qquad \forall_{c\in C} \, / \cdot \alpha_c$$

$$y_B^f \geq 0 \qquad \forall_{f\in F} \forall_{B\subseteq C}.$$

The dual of this LP is:

$$\max \sum_{c\in C} \alpha_c \quad \text{(FL-CONF-DUAL)}$$

$$\text{s.t.} \sum_{j\in B} \alpha_c \leq o_f \cdot \mathbb{P}[B \cap A \neq \emptyset] + \sum_{c\in B} d(c, f) \, \mathbb{P}[c \in A] \quad \forall_{f\in F} \forall_{B\subseteq C} \Big/ \cdot y_B^f . \tag{8.3}$$

Primal dual-scheme In the algorithm every client c raises its dual variable with speed equal to $\mathbb{P}[c \in A]$. Once a constraint 8.3 becomes tight for a facility f and subset B we assign these subset of clients to f, and we freeze all clients in B. Facility f becomes tentatively open. We continue this process until all clients are tentatively assigned to some facility. Let us for a moment skip the details of how to detect a tight constraint among exponential number of them. ometimes it might be the case that we already have a tight constraint for B and f, but after some time another subset B' of clients gathers a credit of value $o_f \cdot (\mathbb{P}[(B + B') \cap A \neq \emptyset] - \mathbb{P}[B \cap A \neq \emptyset]) + \sum_{c \in B} d(c, f) \mathbb{P}[c \in A]$. In which case we connect tentatively B' to f as well.

S

After we have opened facilities tentatively we need to perform a rerouting step. We consider tentatively open facilities in order of the smallest costs o_f. More precisely, suppose we consider facility f_1, then we assign to f_1 every client c such that $\alpha_c > d(c, f) \mathbb{P}[c \in A]$ — this means that c was contributing towards opening of a facility. Also, we close every facility $\bar{f}$ such that $\alpha_c > d\left(c, \bar{f}\right) \mathbb{P}[c \in A]$, and all clients $N(\bar{f})$ assigned to $\bar{f}$ are getting rerouted to f_1.

Lemma 71. *After rerouting the cost of connection of client j is at most $3\alpha_j$.*

Proof. Consider client j connected to $\bar{f}$. We have closed $\bar{f}$ because there exists c. Let t_c be the time when c became frozen — credit gathered by c is exactly $\mathbb{P}[c \in C] \cdot t_c$. And since $\alpha_c > d\left(c, \bar{f}\right) \mathbb{P}[c \in A]$ it means that $t_c > d\left(c, \bar{f}\right)$ and also that $t_c \geq d(c, f)$. Since $\bar{f}$ was opened after t_c(otherwise c would be assigned to $\bar{f}$), also j was frozen at t_j after t_c, so $t_j \geq t_c$. Therefore, we have that the credit of j is $\alpha_j = t_j \mathbb{P}[j \in C] \geq d\left(j, \bar{f}\right) \mathbb{P}[j \in C]$, so $t_j \geq d\left(j, \bar{f}\right)$. So now the total cost of rerouting j is at most $\mathbb{P}[j \in C]\left(d\left(j, \bar{f}\right) + d\left(\bar{f}, c\right) + d(c, f)\right) \leq \mathbb{P}[j \in C](t_j + t_c + t_c) \leq 3t_j \cdot \mathbb{P}[j \in C] = 3\alpha_j$. □

Lemma 72. *Cost of opening the facilities does not increase.*

Proof. We've opened f. Clients $N(f)$ of f paid their connection cost and the cost of opening f which is $\mathbb{P}[N(f) \cap A \neq \emptyset] \cdot o_f$. Opening of f required to close $\bar{f}$. We rerouted $N(\bar{f})$ to f, and the connection cost was not much greater (the Lemma above). Clients $N(\bar{f})$ were paying $\mathbb{P}\left[N(\bar{f}) \cap A \neq \emptyset\right] \cdot o_{\bar{f}}$ to open facility $\bar{f}$. Now they go to f, so they need to provide the difference in the opening cost which is $\left(\mathbb{P}\left[\left(N(f) + N(\bar{f})\right) \cap A \neq \emptyset\right] - \mathbb{P}[N(f) \cap A \neq \emptyset]\right) \cdot o_f$. But this cost is not greater than the initial cost of opening $\bar{f}$. First,

$$\mathbb{P}\left[N(\bar{f}) \cap A \neq \emptyset\right] \geq \mathbb{P}\left[\left(N(f) + N(\bar{f})\right) \cap A \neq \emptyset\right] - \mathbb{P}[N(f) \cap A \neq \emptyset],$$

since function $S \mapsto \mathbb{P}[S \cap A \neq \emptyset]$ is submodular, and also $o_{\bar{f}} \geq o_f$ because we were scanning the facilities in order of increasing o_f. Thus

$$\begin{aligned}\mathbb{P}\left[N(\bar{f}) \cap A \neq \emptyset\right] \cdot o_{\bar{f}} \quad &\geq \quad \left(\mathbb{P}\left[\left(N(f) + N(\bar{f})\right) \cap A \neq \emptyset\right] - \mathbb{P}[N(f) \cap A \neq \emptyset]\right) \cdot o_{\bar{f}} \\ &\left(\mathbb{P}\left[\left(N(f) + N(\bar{f})\right) \cap A \neq \emptyset\right] - \mathbb{P}[N(f) \cap A \neq \emptyset]\right) \cdot o_f.\end{aligned}$$

□

Now we can conclude the proof of the Theorem. Total cost of the tentative solution, i.e., before rerouting, is $\sum_c \alpha_c$ and this is equal to the cost opening the tentative facilities, and connecting clients to them. From Lemma 71 we know that the total cost of connection after rerouting is at most $3 \cdot \sum_c \alpha_c$. The cost of opening the facilities does not increase, so is still at most $\sum_c \alpha_c$. Therefore, the total cost is at most $4 \sum_c \alpha_c$ which finishes the proof.

Finding a tight constraint in the scheme It is not clear if upfront, i.e., before restarting growing the variables of clients after assigning some clients tentatively, we can indicate which constraint will be tight. It is quite likely that we are able to do so, but there is a simple way to deal with that. We can discretize the time, into intervals of ε. After each timestep we can see if any of the constraints $\sum_{j \in B} \alpha_c \leq o_f \cdot \mathbb{P}\left[B \cap A \neq \emptyset\right] + \sum_{c \in B} d\left(c, f\right) \mathbb{P}\left[c \in A\right]$ became violated. To do so we need to see whether there exists a subset B such that $o_f \cdot \mathbb{P}\left[B \cap A \neq \emptyset\right] + \sum_{c \in B} d\left(c, f\right) \mathbb{P}\left[c \in A\right] - \sum_{j \in B} \alpha_c < 0$. This can be done in polynomial time, because this requires minimizing the function $h\left(B\right) = o_f \cdot \mathbb{P}\left[B \cap A \neq \emptyset\right] + \sum_{c \in B} d\left(c, f\right) \mathbb{P}\left[c \in A\right] - \sum_{j \in B} \alpha_c$, and looking whether the minimum is negative. Function $h(B)$ is submodular as a sum of a submodular and two linear functions, and therefore it can be done in polynomial time.

Bibliography

[1] Marek Adamczyk. Improved analysis of the greedy algorithm for stochastic matching. *Inf. Process. Lett.*, 111:731–737, August 2011. 1.2.2

[2] Marek Adamczyk, Maxim Sviridenko, and Justin Ward. Submodular stochastic probing on matroids. In *STACS*, 2014. 3.1.1, 5, 5.2.4

[3] Alexander A. Ageev and Maxim Sviridenko. Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *J. Comb. Optim.*, 8(3):307–328, 2004. 4.3.1

[4] Noga Alon and Joel Spencer. *The Probabilistic Method*. John Wiley, 1992. 5.2.4

[5] Aris Anagnostopoulos, Luca Becchetti, Ilaria Bordino, Stefano Leonardi, Ida Mele, and Piotr Sankowski. Stochastic query covering for fast approximate document retrieval. *ACM Trans. Inf. Syst.*, 33(3):11:1–11:35, 2015. 8.2

[6] Arash Asadpour, Hamid Nazerzadeh, and Amin Saberi. Stochastic submodular maximization. In *WINE*, pages 477–489, 2008. 1.1.1

[7] Moshe Babaioff, Nicole Immorlica, and Robert Kleinberg. Matroids, secretary problems, and online mechanisms. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2007, New Orleans, Louisiana, USA, January 7-9, 2007*, pages 434–443, 2007. 2.1

[8] Nikhil Bansal, Anupam Gupta, Jian Li, Julián Mestre, Viswanath Nagarajan, and Atri Rudra. When LP is the cure for your matching woes: Improved bounds for stochastic matchings. *Algorithmica*, 63(4):733–762, 2012. 1.1.1.1, 1, 1.1.1.2, 1.2.2, 1.2.2, 1.2.3, 3, 24, 3.1.1, 3.1.1, 3.1.1, 3.1.2, 3.1.3, 29, 5

[9] Alok Baveja, Amit Chavan, Andrei Nikiforov, Aravind Srinivasan, and Pan Xu. Improved bounds in stochastic matching and optimization. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2015, August 24-26, 2015, Princeton, NJ, USA*, pages 124–134, 2015. 1.2.3

[10] E. M. L. Beale. Linear programming under uncertainty. *Journal of the Royal Statistical Society. Series B*, 17:173–184, 1955. 1

[11] Anand Bhalgat. *Optimization under uncertainty*. PhD thesis, 2013. 1

[12] Anand Bhalgat, Ashish Goel, and Sanjeev Khanna. Improved approximation results for stochastic knapsack problems. In *SODA*, pages 1647–1665, 2011. 1

[13] Gruia Calinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a Submodular Set Function Subject to a Matroid Constraint. *SIAM Journal on Computing*, 40(6):1740–1766, 2011. 2.2, 2.2, 4, 4.3.1, 4.3.1, 37, 38, 4.3.1, 4.3.1, 5.2.4

[14] Moses Charikar, Chandra Chekuri, and Martin Pál. Sampling bounds for stochastic optimization. In *Proc. 9th RANDOM*, pages 257–269. Springer, 2005. 8.1

[15] Ioannis Chatzigeorgiou. Bounds on the lambert function and their application to the outage analysis of user cooperation. *IEEE Communications Letters*, 17(8):1505–1508, 2013. 4.3.2

[16] Shuchi Chawla, Jason D. Hartline, David L. Malec, and Balasubramanian Sivan. Multi-parameter mechanism design and sequential posted pricing. In *STOC*, 2010. 1.1.1, 1.1.1, 5.2.3

[17] Chandra Chekuri, Jan Vondrák, and Rico Zenklusen. Dependent randomized rounding via exchange properties of combinatorial structures. In *51th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2010, October 23-26, 2010, Las Vegas, Nevada, USA*, pages 575–584, 2010. 1.1.3

[18] Ning Chen, Nicole Immorlica, Anna R. Karlin, Mohammad Mahdian, and Atri Rudra. Approximating matches made in heaven. In *ICALP*, 2009. 1.1.1, 1.1.1.1, 1.2.2

[19] Robert M. Corless, Gaston H. Gonnet, D. E. G. Hare, David J. Jeffrey, and Donald E. Knuth. On the lambert$w$ function. *Adv. Comput. Math.*, 5(1):329–359, 1996. 4.3.2, 4.3.3

[20] William H Cunningham. Testing membership in matroid polyhedra. *Journal of Combinatorial Theory, Series B*, 36(2):161 – 188, 1984. 7

[21] G.B. Dantzig. Linear programming under uncertainty. *Management Science*, 1:197–206, 1955. 1

[22] Brian C. Dean, Michel X. Goemans, and Jan Vondrák. Adaptivity and approximation for stochastic packing problems. In *SODA*, pages 395–404, 2005. 1.1.1

[23] Brian C. Dean, Michel X. Goemans, and Jan Vondrák. Approximating the stochastic knapsack problem: The benefit of adaptivity. *Math. Oper. Res.*, 33(4):945–964, 2008. 1, 1.1.1, 3.1.1

[24] Devdatt Dubhashi and Desh Ranjan. Balls and bins: A study in negative dependence. *Random Structures and Algorithms*, 13(2):99–124, 1998. 1.1.3

[25] Jittat Fakcharoenphol, Satish Rao, and Kunal Talwar. A tight bound on approximating arbitrary metrics by tree metrics. *J. Comput. Syst. Sci.*, 69(3):485–497, 2004. 1.2.4

[26] Moran Feldman, Joseph Naor, and Roy Schwartz. A unified continuous greedy algorithm for submodular maximization. In *FOCS*, 2011. 4.3.1, 4.3.1, 5, 5.1, 5.1, 5.1, 5.1, 5.1, 42, 43, 44

[27] Moran Feldman, Ola Svensson, and Rico Zenklusen. Online contention resolution schemes. *CoRR*, abs/1508.00142, 2015. 5, 5.2.3, 5.2.4

[28] Esteban Feuerstein, Alberto Marchetti-Spaccamela, Frans Schalekamp, René Sitters, Suzanne van der Ster, Leen Stougie, and Anke van Zuylen. Scheduling over scenarios on two machines. In *Computing and Combinatorics - 20th International Conference, COCOON 2014, Atlanta, GA, USA, August 4-6, 2014. Proceedings*, pages 559–571, 2014. 1

[29] Rajiv Gandhi, Samir Khuller, Srinivasan Parthasarathy, and Aravind Srinivasan. Dependent rounding and its applications to approximation algorithms. *J. ACM*, 53(3):324–360, 2006. 1.1.3, 1.2.2, 3.1.1, 3.2

[30] Michel X. Goemans and Jan Vondrák. Stochastic covering and adaptivity. In *LATIN*, pages 532–543, 2006. 1.1.1

[31] Fabrizio Grandoni, Anupam Gupta, Stefano Leonardi, Pauli Miettinen, Piotr Sankowski, and Mohit Singh. Set covering with our eyes closed. In *FOCS*, pages 347–356, 2008. 1, 1.1.2, 1.2.4

[32] Martin Grötschel, László Lovász, and Alexander Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1(2):169–197, 1981. 8.1, 8.2

[33] Sudipto Guha and Kamesh Munagala. Approximation algorithms for budgeted learning problems. In *STOC*, pages 104–113, 2007. 1, 1.1.1

[34] Sudipto Guha and Kamesh Munagala. Model-driven optimization using adaptive probes. In *SODA*, pages 308–317, 2007. 1.1.1

[35] Anupam Gupta, Ravishankar Krishnaswamy, Marco Molinaro, and R. Ravi. Approximation algorithms for correlated knapsacks and non-martingale bandits. In *FOCS*, pages 827–836, 2011. 1, 1.1.1

[36] Anupam Gupta and Viswanath Nagarajan. A stochastic probing problem with applications. In *IPCO*, 2013. 1.1.1, 1.1.1, 1.2.1, 4, 5, 5.2.4

[37] Anupam Gupta, Martin Pál, R. Ravi, and Amitabh Sinha. Boosted sampling: approximation algorithms for stochastic optimization. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, STOC '04, pages 417–426, New York, NY, USA, 2004. ACM. 1

[38] Nicole Immorlica, David Karger, Maria Minkoff, and Vahab S. Mirrokni. On the costs and benefits of procrastination: approximation algorithms for stochastic combinatorial optimization problems. In *Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms*, SODA '04, pages 691–700, Philadelphia, PA, USA, 2004. Society for Industrial and Applied Mathematics. 1

[39] Lujun Jia, Guolong Lin, Guevara Noubir, Rajmohan Rajaraman, and Ravi Sundaram. Universal approximations for tsp, steiner tree, and set cover. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005*, pages 386–395, 2005. 1.1.2

[40] David R. Karger and Maria Minkoff. Building steiner trees with incomplete global knowledge. In *FOCS'00*, pages 613–623, 2000. 1, 1.1.2, 1.2.4

[41] Jian Li. *Decision making under uncertainty*. PhD thesis, University of Maryland, 2011. 1.2.2

[42] Jian Li and Wen Yuan. Stochastic combinatorial optimization via poisson approximation. In *STOC*, pages 971–980, 2013. 1

[43] Will Ma. Improvements and generalizations of stochastic knapsack and multi-armed bandit approximation algorithms: Extended abstract. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, pages 1154–1163, 2014. 1, 1.2.2

[44] Rolf H. Möhring, Andreas S. Schulz, and Marc Uetz. Approximation in stochastic scheduling: the power of lp-based priority policies. *J. ACM*, 46(6):924–942, 1999. 1

[45] Alessandro Panconesi and Aravind Srinivasan. Fast randomized algorithms for distributed edge coloring. In *Proceedings of the Eleventh Annual ACM Symposium on Principles of Distributed Computing*, PODC '92, pages 251–262, New York, NY, USA, 1992. ACM. 1.1.3

[46] R. Ravi and Amitabh Sinha. Hedging uncertainty: Approximation algorithms for stochastic optimization problems. *Math. Program.*, 108:97–114, August 2006. 1

[47] A. Schrijver. *Combinatorial Optimization - Polyhedra and Efficiency*. Springer, 2003. 1.1.1, 2.1, 2.1, 3.1.2

[48] David B. Shmoys and Chaitanya Swamy. Stochastic optimization is (almost) as easy as deterministic optimization. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 228–237, Washington, DC, USA, 2004. IEEE Computer Society. 1

[49] Aravind Srinivasan. Distributions on level-sets with applications to approximation algorithms. In *42nd Annual Symposium on Foundations of Computer Science, FOCS 2001, 14-17 October 2001, Las Vegas, Nevada, USA*, pages 588–597, 2001. 1.1.3, 2

[50] Vijay V. Vazirani. *Approximation algorithms*. Springer, 2001. 1.1.1, 8.2

[51] Jan Vondrák. Personal correspondence. 4.3.1

[52] Jan Vondrák. *Submodularity in combinatorial optimization*. PhD thesis, Charles University, 2007. 2.2, 4.3.2

[53] Jan Vondràk. Lectures on polyhedral techniques in combinatorial optimization, 2010. 2.2

[54] Jan Vondrák, Chandra Chekuri, and Rico Zenklusen. Submodular function maximization via the multilinear relaxation and contention resolution schemes. In *STOC*, 2011. 1.2.1, 5, 5.1, 5.2.1, 5.2.3, 5.2.3, 5.2.4, 5.2.4, 5.2.4, 5.3

[55] David Williams. *Probability with Martingales*. Cambridge University Press, 1991. 2.3