

# Online Dependent Rounding Schemes for Bipartite Matchings, with Applications

Joseph (Seffi) Naor<sup>\*1</sup>, Aravind Srinivasan<sup>†2</sup>, and David Wajc<sup>‡1</sup>

<sup>1</sup>Technion – Israel Institute of Technology

<sup>2</sup>University of Maryland, College Park

## Abstract

We introduce the abstract problem of rounding an unknown fractional bipartite  $b$ -matching  $\mathbf{x}$  revealed online (e.g., output by an online fractional algorithm), exposed node-by-node on one side. The objective is to maximize the *rounding ratio* of the output matching  $\mathcal{M}$ , which is the minimum over all fractional  $b$ -matchings  $\mathbf{x}$ , and edges  $e$ , of the ratio  $\Pr[e \in \mathcal{M}]/x_e$ . In analogy with the highly influential offline dependent rounding schemes of Gandhi et al. (FOCS’02, J.ACM’06), we refer to such algorithms as *online dependent rounding schemes* (ODRSes). This problem, with additional restrictions on the possible inputs  $\mathbf{x}$ , has played a key role in recent developments in online computing.

We provide the first generic  $b$ -matching ODRSes that impose no restrictions on  $\mathbf{x}$ . Specifically, we provide ODRSes with rounding ratios of 0.646 and 0.652 for  $b$ -matchings and simple matchings, respectively. This breaks the natural barrier of  $1 - 1/e$ , prevalent for online matching problems, and numerous online problems more broadly. Using our ODRSes, we provide a number of algorithms with similar better-than- $(1 - 1/e)$  ratios for several problems in online edge coloring, stochastic optimization, and more.

Our techniques, which have already found applications in several follow-up works (Patel and Wajc SODA’24, Blikstad et al. SODA’25, Braverman et al. SODA’25, and Aouad et al. 2024), include periodic use of *offline* contention resolution schemes (in online algorithm design), grouping nodes, and a new scaling method which we call *group discount and individual markup*.

---

<sup>\*</sup>Supported in part by ISF grant 3001/24 and United States - Israel BSF grant 2022418.

<sup>†</sup>Supported in part by NSF award number CCF-1918749, and by research awards from Amazon and Google.

<sup>‡</sup>Supported in part by a Taub Family Foundation “Leader in Science and Technology” fellowship, and by ISF grant 3200/24, “Prophets, Philosophers, and Online Algorithms”. Work done in part while at Stanford University.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Extensions and Applications . . . . .	2
1.2	Techniques . . . . .	3
1.3	Further related work . . . . .	4
<b>2</b>	<b>Preliminaries</b>	<b>4</b>
2.1	Negative association and strong Rayleigh properties . . . . .	5
<b>3</b>	<b>Rounding Matchings Online</b>	<b>6</b>
3.1	The improved ODRS: Overview and intuition . . . . .	6
3.2	The core of the improved ODRS . . . . .	7
3.3	The improved ODRS . . . . .	9
<b>4</b>	<b>Online Level-Set Rounding</b>	<b>11</b>
4.1	Offline level-set rounding . . . . .	12
4.2	The online level-set rounding algorithm . . . . .	13
4.3	Coupling the online and offline algorithms . . . . .	14
<b>5</b>	<b>Applications</b>	<b>14</b>
5.1	Applications and extensions of the matching ODRS . . . . .	14
5.1.1	Online edge coloring (multigraphs) . . . . .	14
5.1.2	Stochastic extension . . . . .	16
5.1.3	A simple direct application: Algorithms with predictions . . . . .	21
5.2	Application of the level-set ODRS . . . . .	22
5.2.1	Multi-stage stochastic optimization . . . . .	22
5.2.2	Negative association, fairness, diversity, and streaming . . . . .	23
<b>6</b>	<b>Lower Bounds</b>	<b>25</b>
<b>7</b>	<b>Summary and Open Questions</b>	<b>27</b>
<b>A</b>	<b>Additional Preliminaries</b>	<b>27</b>
A.1	Contention Resolution Schemes . . . . .	27
A.2	Negative correlation properties . . . . .	28
A.3	Odds and Ends . . . . .	29
<b>B</b>	<b>Deferred Proofs of Section 3</b>	<b>29</b>
B.1	Polytime implementation . . . . .	30
B.2	The $b$ -matching ODRS . . . . .	31
B.3	Deferred proofs of the $b$ -matching ODRS . . . . .	33
<b>C</b>	<b>Deferred Proofs of Section 4</b>	<b>34</b>
C.1	Direct (partial) analysis of Algorithm 5 . . . . .	34
C.2	Coupling the offline and online algorithms . . . . .	35
<b>D</b>	<b>Deferred Proofs of Section 6</b>	<b>37</b>

# 1 Introduction

We initiate the study of the abstract problem of online *randomized rounding of fractional  $b$ -matchings*. Here, nodes on one side of an unknown bipartite graph  $G = (V, E)$  are revealed over time; each such *online node* and its edges' associated fractions in a fractional  $b$ -matching  $\mathbf{x}$  are revealed online (e.g., provided by some online fractional algorithm). An online dependent rounding scheme (ODRS) must decide immediately and irrevocably, upon a node's arrival, which of its edges to match.<sup>1</sup> The objective is to match each with probability close to the value associated with it in  $\mathbf{x}$ .

**Definition 1.1.** *An ODRS  $\mathcal{A}$  for online matching has (oblivious) rounding ratio  $\alpha \in [0, 1]$  if, when a vertex with edges  $e_1, \dots, e_n$  are revealed online with their associated fractions  $x_{e_1}, \dots, x_{e_n}$ , with the guarantee that  $\mathbf{x} \in \{\mathbf{x} \in \mathbb{R}_{\geq 0}^{|E|} \mid x(\delta(v)) \leq 1 \ \forall v \in V\}$  is a fractional matching,  $\mathcal{A}$  outputs an integral matching  $\mathcal{M}$  satisfying*

$$\Pr[e \in \mathcal{M}] \geq \alpha \cdot x_e \quad \forall e \in E.$$

In offline settings a rounding ratio of one is achievable, by the integrality of the bipartite matching polytope. In contrast, in online settings, this is provably impossible [24, 32, 34]. Nonetheless, as illustrated by the FOCS 2023 workshop on online algorithms and online rounding,<sup>2</sup> algorithms for rounding fractional matchings online are central to numerous breakthroughs in online computing in recent years, for problems such as online bipartite edge-weighted matching [13, 44, 52], stochastic online bipartite matching [55, 83], and online edge coloring [14, 31, 63], among others. Despite the impossibility of rounding ratios of one or  $1 - o(1)$  in general, many of the above results follow from ODRS with such high rounding ratios for *structured* fractional matchings. Echoing these recent developments, Buchbinder et al. [24] asked what additional constraints can be imposed on fractional matchings to allow for lossless (i.e., rounding ratio of one) online rounding.

Breaking from prior work, we explicitly study the problem of designing ODRSes for *unrestricted* fractional ( $b$ -)matchings, and their applications. While recent work of [78] implies an ODRS with non-trivial rounding ratio of 0.526 for matchings (but not for  $b$ -matchings), our understanding of such rounding schemes is nascent, both on the techniques side, as well as the applications side.

**Our ODRSes.** We provide a number of ODRSes, for both structured and unrestricted fractional  $b$ -matchings, as well as a host of applications.

For star graphs, or equivalently rounding in a uniform matroid (i.e., fractions summing up to at most some  $b$  are revealed online), we provide two simple ODRSes with rounding ratio of one respecting prefix constraints, thus providing a counterpart to the classic pivotal sampling of [79]. Indeed, as we show (in Theorem 4.5), one of our algorithm's output distribution is *identical* to the offline algorithm's output distribution, and thus inherits the latter's known strong negative correlation properties, which we leverage later.

For ODRS for unrestricted fractional matchings, we rule out any better than  $2\sqrt{2} - 2 \approx 0.828$  rounding ratio. On the positive side, we show that combining ODRS for star graphs with the single-item *offline* contention resolution scheme (CRS) of Feige and Vondrák [45] yields a simple ODRS with rounding ratio of  $1 - 1/e \approx 0.632$ . Our main result breaks this ubiquitous bound.

**Theorem 1.2.** *There exist bipartite matching ( $b$ -matching) ODRSes with rounding ratio 0.652 (0.646). No bipartite matching ODRS has rounding ratio greater than  $2\sqrt{2} - 2 \approx 0.828$ .*

<sup>1</sup>As in the offline dependent rounding schemes of [51], the term *dependent* here underscores the need for random choices to depend on each other, since independent coin tosses do not yield a valid high-value ( $b$ -)matching.

<sup>2</sup>Online Algorithms and Online Rounding: Recent Progress @ FOCS 2023.

## 1.1 Extensions and Applications

Our ODRSes and the techniques underlying them have a number of applications. We briefly discuss our three main applications here, deferring detailed discussions to Section 5, where we also present other applications of our ODRS to fairness in machine learning and to algorithms with predictions.

**Online edge coloring.** Our ability to round fractional matchings has implications to the recent active line of work on online edge coloring [3, 6, 11, 14, 15, 16, 31, 37, 63, 78]. Here, a graph is revealed online (either vertex-by-vertex or edge-by-edge), and we must decompose this graph into color classes that form matchings (i.e., no vertex has two edges of the same color). A reduction due to [31] shows that online  $(\alpha + o(1))\Delta$ -edge-coloring can be reduced to online matching algorithms that match each edge with probability  $1/(\alpha\Delta)$ . Ignoring the  $o(1)$  term, these problems are in fact equivalent, since sampling a color at random among the  $\alpha\Delta$  color yields an online matching algorithm matching each edge with probability  $1/(\alpha\Delta)$ . This reduction has motivated the interest in online rounding a particular structured fractional matching: the uniform solution,  $\mathbf{x} = \frac{1}{\Delta} \cdot \mathbf{1}$  [14, 31, 32, 63], culminating in a competitive ratio (and rounding ratio for such “spread out” fractional matchings) of  $(1 + o(1))$  for general (simple) graphs under edge arrivals [14]. For *multigraphs*, in contrast, positive results were only known under *random-order arrivals* [3]. Addressing this gap between simple graphs and multigraphs, in Section 5.1.1 we extend the above reduction from  $\alpha$ -competitive edge coloring to computing online matchings that match each (in this case, parallel) edge with probability  $\frac{1}{\alpha\Delta}$ . We thus show that a rounding ratio of  $1/\alpha$  implies a competitive ratio of  $\alpha + o(1)$  for edge coloring multigraphs (and is, in fact, equivalent to it). Our positive and negative results for ODRS yield the first positive and negative results for online edge-coloring in multigraphs under adversarial arrivals. For example, we achieve a competitive factor of  $e/(e - 1) - \Omega(1)$  in bipartite multigraphs, answering a question of Schrijver [31, acknowledgements].

**Stochastic optimization.** Another application of our techniques concerns the online stochastic bipartite weighted matching problem, much studied in the online Prophet-Inequality literature [41, 43, 47, 62]. (See Section 5.1.2 for formal definition.) Recently, Papadimitriou et al. [73] initiated the study of efficiently approximating the optimum *online* algorithm for this problem. They showed that it is PSPACE-hard to approximate the optimum algorithm within some universal constant  $\beta < 1$ , and provided a polynomial-time online algorithm that 0.51-approximates this online optimum. (In contrast, the best competitive ratio, or approximation of the *offline* optimum, is 0.5 for this problem [43].) The bound of [73] can be improved to 0.526 using ideas in [78] and was significantly improved to  $1 - 1/e \approx 0.632$  by [20]. All three results are achieved via online rounding. Extending our online rounding approach of Theorem 1.2 to such stochastic settings, we improve on the recent bound of [20] and break the ubiquitous bound of  $1 - 1/e$ , providing an online polytime 0.652-approximation of the optimal (computationally-unbounded) online algorithm.

**Multi-Stage stochastic optimization.** Our ODRS for star graphs and its strong negative correlations allows us to generalize a result of [80] for multi-stage stochastic optimization. Here, uncertain parameters are stochastic but their distributions get refined over time: actions in earlier stages are cheaper but perhaps less accurate due to the stochasticity, while actions in later stages are more expensive but more accurate. (See the survey [81].) A basic example is the (hyper-)graph covering problem, where sets (hyperedges) are revealed in  $k$  stages, and must be covered by bought elements (nodes). Swamy and Shmoys [82] presented a  $2k$ -approximation for this problem, later improved to 2, in [80], matching the offline hardness of the non-stochastic single-stage problem. Building on the framework of [80, 82], in Section 5.2.1 we generalize this result to allow for sets with *multi*-coverage demands. Our algorithms have approximation ratio of 2 (and tending to *one* in some settings), and these ratios hold w.h.p.

## 1.2 Techniques

In this section we highlight some of the key ideas that allow us to achieve our main results, namely Theorems 1.2 and 4.5, deferring detailed discussions of applications to later sections.

For level-set rounding, we provide two simple ODRS that achieves a rounding ratio of one. Surprisingly, we show via a coupling argument that for this problem, our main online algorithm’s distribution is *identical* to that of (offline) pivotal sampling, a.k.a. Srinivasan sampling [79]. This allows us to inherit this well-studied algorithm’s concentration properties [19, 35, 79]. In particular, this implies that our online algorithm’s output distribution satisfies the *Strong Rayleigh* property (see Section 2), which in turn implies a slew of negative correlation properties, useful for our applications in later sections.

For our impossibility result, we rely on a Ramsey-theoretic probabilistic lemma, whereby in any sufficiently large set of binary variables, some (near-)positive correlation is inevitable. An extension of this lemma (Lemma 6.3), of possible independent interest, rules out algorithmic approaches based on guaranteeing strict negative correlation between offline nodes’ matched statuses.

**ODRSes for  $b$ -matching.** Our first ODRS for  $b$ -matchings more broadly is obtained by interleaving level-set ODRSes with repeated invocations of the *offline* single-item contention resolution scheme (CRS) of Feige and Vondrák [45]. (This is, to the best of our knowledge, the first online algorithm to use offline CRSes in such a black-box manner.) Specifically, we run independent copies of our level-set algorithm, one per offline node, thus having each offline node  $i$  “bid” for arriving online node  $t$  independently of other offline nodes with probability  $x_{i,t}$ , while preserving the  $b$ -matching constraints of the offline nodes. To preserve online nodes’ matching constraints, we then use the single-item offline CRS of [45] at each time  $t$ : this CRS guarantees for such product distributions that  $t$  is allocated to at most one buyer, with every buyer  $i$  being allocated item  $t$  (i.e.,  $(i, t)$  being matched) with probability at least  $(1 - 1/e) \cdot x_{i,t}$ , thus yielding a rounding ratio of  $1 - 1/e$ .

To improve on the above, we first note that the  $(1 - 1/e) \cdot x_{i,t}$  bound is loose if any of the  $x_{i,t}$  fractions are bounded away from 0. Indeed, this  $1 - 1/e$  factor is of the form

$$\left(1 - \prod_i (1 - x_{i,t})\right) / \sum_i x_{i,t} \geq 1 - 1/e,$$

where the inequality is loose if any  $x_{i,t}$  is large. To beat this  $1 - 1/e$  bound, we therefore simulate the existence of a large  $x_{i,t}$ , by *grouping* offline nodes via a bin-packing algorithm, and letting at most one node per group (bin)  $B$  bid for  $t$ . The obtained distribution over bidding offline nodes is no longer a product distribution, but it is strongly negatively correlated, and even negatively associated (see Section 2). This allows us to achieve the same  $1 - 1/e$  bound using CRSes for negatively correlated distributions of Bansal and Cohen [7]. Moreover, this grouping allows us to beat the  $1 - 1/e$  ratio in some cases: In a very concrete sense, it results in effectively a single aggregated offline node bidding with large probability  $\sum_{i \in B} x_{i,t}$ , thus allowing us to beat the bound of  $1 - 1/e$  if much grouping occurs. Furthermore, thinking of bids as “costing” the offline node future matching opportunities, we can even beat the bound of  $1 - 1/e$  after giving nodes in a group a “group discount”, and having them bid with lower probability than  $x_{i,t}$ . This discounting then leaves offline nodes with more budget (probability of not being matched), thus allowing us to charge these nodes more at later times: in particular, when these offline nodes are not grouped, we charge them an “individual markup”, by having them bid with higher probability than  $x_{i,t}$ . This then increases the matching probability of those  $t$  for which little grouping occurs to beyond  $1 - \frac{1}{e}$ , and the improved rounding ratio follows.

### 1.3 Further related work

Online bipartite  $b$ -matching is a prototypical online problem, tracing its origin to the seminal work of Karp et al. [59]. Here, nodes on one side of a bipartite graph are revealed over time, and each node  $v$  can be matched at most  $b_v$  times, with arriving nodes matched immediately and irrevocably. This influential problem has been widely studied in various settings (e.g., fractional/integral, simple/capacitated, unweighted/vertex-weighted/edge-weighted) [2, 12, 13, 34, 42, 44, 46, 58, 59, 70] and has been a cornerstone of the online algorithms literature. (See the surveys [33, 56, 69].)

A natural approach to round a fractional matching  $\mathbf{x}$  is to *activate* each edge  $e$  independently w.p.  $x_e$  and then pick a feasible subset  $\mathcal{I}$  of active edges, such that  $\Pr[e \in \mathcal{I}] \geq \alpha \cdot x_e$ . This is obtained by *contention resolution schemes* (CRSes), first studied in offline settings for single-item problems (rank-one matroids) by Feige and Vondrák [45] (see Section 2), and later for arbitrary matroids by Chekuri et al. [28]. This was extended to online settings by Feldman et al. [48], and has since become a bedrock of stochastic online algorithms, for prophet inequalities [48], secretary problems [38, 39], posted-price mechanisms [27, 60], sequential pricing [76], algorithmic contract design [9] and more. CRSes and OCRSes have been intensely studied for numerous constraints, inspired by [60]; most related to our work is the active line of work on (O)CRSes for matchings [22, 48, 50, 53], where the optimal balance ratio (the counterpart of rounding ratio for ODRSes) is still not fully understood, despite recent exciting progress [23, 67, 68, 72, 76]. While this is an exciting and active area, such OCRS-based ODRSes cannot have rounding ratio better than 0.5, even for rank-one matroids, by connections to the prophet-inequality problem [43, 62], or even  $1/e \approx 0.362$ , as we assume  $\mathbf{x}$  is unknown upfront, and would therefore require *oblivious* OCRS [49]. Going beyond independent bids for each pair  $(i, t)$  is therefore crucial in order to obtain matching ODRSes with high oblivious rounding ratios.

Another online matching rounding scheme is provided by the recent exciting literature on Online Correlated Selection (OCS), first introduced in the groundbreaking paper of Fahrback et al. [44]. Specifically, the OCSes of [52] underlie many rounding-based online algorithms for edge-weighted matching [52], fair matching [54] and i.i.d matching [55, 83]. This powerful machinery can be interpreted as providing oblivious rounding ratios of one for some particular structured solutions (see [24] for discussion). For unrestricted fractional matching, in contrast, this machinery provides useful per-offline-vertex guarantees, but not per-edge oblivious rounding guarantees, as we require.

**Follow-up work.** Following the posting of our paper online, its techniques have found applications in subsequent work. Patel and Wajc [74] and subsequently Aouad et al. [5] use our approach of combining proposals from one side of a bipartite graph with contention resolution schemes on the other, for an infinite-horizon Markovian setting. Blikstad et al. [16] similarly used offline CRS and the derandomization approach of [10] to obtain *deterministic* online bipartite edge coloring algorithms. Finally, [21] build on the discount and individual markup technique together with pivotal sampling [79] to improve on our results of Section 5.1.2 for approximating the optimal stochastic online bipartite matching algorithm (which they refer to as a *philosopher*). We anticipate further applications of our techniques, in particular the use of offline contention resolution schemes for online algorithm design.

## 2 Preliminaries

**Problem Definition.** In the online  $b$ -matching rounding problem, an a priori unknown bipartite graph  $G = (L, R, E)$  is revealed online, together with fractions on the edges incident to the arriving edge's vertices that satisfy the  $b$ -matching constraints. In more detail, at each time  $t$  an online vertex  $t \in L$  arrives, together with fractions  $x_{i,t}$  assigned to its edges to *offline neighbors*  $i \in R = [n]$ . The

fractions  $x_{i,t} \in [0, 1]$ , are promised to satisfy the (fractional)  $b$ -matching constraints, namely each vertex  $v \in (L \cup R)$  has fractional degree  $\sum_{e \ni v} x_e$  at most  $b_v$  (the integer  $b_v$  is revealed when  $v$  is revealed). By splitting each online node  $t$  with capacity  $b_t$  into  $b_t$  online unit-capacity nodes with identical edge sets and fractions  $x_{it}/b_t$  for each copy of edge  $(i, t)$ , we can assume WLOG that all online nodes have unit capacity. Following arrival  $t$ , we must decide, immediately and irrevocably, which edges of  $t$  to add to the output  $b$ -matching  $\mathcal{M}$ , while satisfying the (integral)  $b$ -matching constraints of matching each vertex  $v$  no more than  $b_v$  times, and striving for a large oblivious rounding ratio.

Our ODRSes for  $b$ -matchings repeatedly invoke offline *contention resolution schemes* (CRSes), whose guarantees for product distributions, due to Feige and Vondrák [45], and arbitrary distributions, due to Bansal and Cohen [7], are given below. (For completeness, we provide a self-contained proof for the general case in Appendix A.)

**Lemma 2.1.** *Let  $\mathcal{D} : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$  be a distribution over subsets of  $[n]$ , and denote this distribution's support by  $\text{supp}(\mathcal{D}) := \{S \subseteq [n] \mid \Pr_{R \sim \mathcal{D}}[R = S] \neq 0\}$ . Then, there exists a randomized algorithm  $\text{CRS}(R, \mathbf{v})$  which on input set  $R \sim \mathcal{D}$  and vector  $\mathbf{v} \in \mathbb{R}^n$ , outputs a subset  $O \subseteq R$  of size  $|O| \leq 1$  satisfying*

$$\Pr[i \in O] = v_i \cdot \min_{\substack{S \subseteq [n] \\ \sum_{i \in S} v_i \neq 0}} \frac{\Pr[R \cap S \neq \emptyset]}{\sum_{i \in S} v_i} \quad \forall i \in [n].$$

The algorithm runs in time polynomial in  $n$  if  $\mathcal{D}$  is a product distribution. Otherwise, it runs in time  $\text{poly}(n, T)$ , where  $T \geq |\text{supp}(\mathcal{D})|$  is the time to compute and write down the support of  $\mathcal{D}$ .

## 2.1 Negative association and strong Rayleigh properties

**Definition 2.2.** *Random real-valued variables  $X_1, \dots, X_n$  are negatively associated (NA) if for any two disjoint index sets  $I, J \subseteq [n]$ ,  $I \cap J = \emptyset$ , and two functions  $f : \mathbb{R}^{|I|} \rightarrow \mathbb{R}$  and  $g : \mathbb{R}^{|J|} \rightarrow \mathbb{R}$  both non-decreasing,*

$$\mathbb{E}[f(X_i : i \in I) \cdot g(X_j : j \in J)] \leq \mathbb{E}[f(X_i : i \in I)] \cdot \mathbb{E}[g(X_j : j \in J)].$$

By a simple inductive argument, negative association implies negative cylinder dependence [57].

**Lemma 2.3.** *For any real NA r.v.s  $X_1, \dots, X_n$  and reals  $x_1, \dots, x_n$ , it holds that*

$$\Pr \left[ \bigwedge_i (X_i \geq x_i) \right] \leq \prod_i \Pr[X_i \geq x_i] \quad \text{and} \quad \Pr \left[ \bigwedge_i (X_i \leq x_i) \right] \leq \prod_i \Pr[X_i \leq x_i].$$

The following family of NA distributions is due to Dubhashi and Ranjan [36].

**Lemma 2.4.** *If  $X_1, \dots, X_n$  are binary r.v.s with  $\sum_i X_i \leq 1$  always, then they are NA.*

More elaborate NA distributions can be obtained via simple NA-preserving operations [57].

**Lemma 2.5.** *NA is closed under products and under disjoint non-decreasing function composition. That is, if  $\mathbf{X} = (\mathbf{X}_1, \dots, \mathbf{X}_n)$  is NA, then:*

1. *if  $\mathbf{Y} = (\mathbf{Y}_1, \dots, \mathbf{Y}_m)$  is NA and  $\mathbf{Y}$  is independent of  $\mathbf{X}$ , then so is  $(X_1, \dots, X_n, Y_1, \dots, Y_m)$ ;*
2.  *$(f_1(X_i : i \in I_1), \dots, f_k(X_i : i \in I_k))$  are NA, for any concordant functions (i.e., all non-decreasing or all non-increasing)  $f_1, \dots, f_k$  and disjoint sets  $I_1, \dots, I_k \subseteq [n]$ ,  $I_j \cap I_k = \emptyset \forall j \neq k$ .*



Negative association implies many more useful properties, most notably the applicability of Chernoff-Hoeffding type tail bounds [36] and submodular stochastic dominance (see Appendix A).

An even stronger negative correlation property than NA is the *strong Rayleigh property (SRP)* (see Appendix A for a precise definition). This property implies NA (even under conditioning), as well as concentration of any Lipschitz function [75], and numerous other useful properties.

### 3 Rounding Matchings Online

In this section we mostly focus on the ODRS problem for bipartite matching, deferring a generalization to  $b$ -matchings to Appendix B.3.

We start with a simple  $b$ -matching algorithm, combining our ODRS for uniform matroids (Algorithm 5) with repeated invocations of an *offline* CRS (Lemma 2.1).

**Warm-up:  $1 - 1/e$  rounding ratio.** We consider the following simple algorithm: Run, for each offline node  $i$ , an independent instance of the online level-set rounding Algorithm 5, applied to  $x_{i1}, x_{i2}, \dots$ . When the  $i$ -th copy of Algorithm 5 fixes  $X_{it} = 1$ , we say that  $i$  *bids* for online node  $t$ . Out of the set of bidders (potential buyers)  $P_t$ , we then pick at most one  $i \in P_t$  to match  $t$  to, using the offline contention resolution scheme of Lemma 2.1, guaranteeing each node  $i$  a probability of  $x_{i,t} \cdot \min_{S \subseteq [n]} \frac{\Pr[S \cap P_t \neq \emptyset]}{\sum_{i \in S} x_{i,t}}$  of being matched. This step runs in polytime, by independence of the bids. Since Algorithm 5 is an online algorithm, so is the resultant algorithm. Moreover, since the output of Algorithm 5 satisfies the  $b$ -matching constraints, and  $t$  is matched to at most one neighbor by the CRS, this algorithm's output satisfies the  $b$ -matching constraints. Finally, for each offline node set  $S \subseteq [n]$ , by independence of the bids, the Taylor expansion of  $\exp(-x)$  and convexity (see Claim A.6), we have

$$\Pr[S \cap P_t \neq \emptyset] = 1 - \prod_{i \in S} (1 - x_{it}) \geq 1 - \prod_{i \in S} \exp(-x_{it}) \geq \left(1 - \frac{1}{e}\right) \cdot \sum_{i \in S} x_{it}. \quad (1)$$

Therefore, by Lemma 2.1, this ODRS matches each edge  $(i, t)$  with probability at least  $(1 - 1/e) \cdot x_{i,t}$ .

**Lemma 3.1.** *There exists a polytime ODRS for bipartite  $b$ -matching with rounding ratio  $1 - 1/e$ .*

One natural approach to improve the above algorithm's rounding ratio would be to attempt to negatively correlate the probability of different offline nodes to have previously bid, thus increasing  $\Pr[S \cap P_t \neq \emptyset]$  for all offline sets  $S \subseteq [n]$ . Unfortunately, for a sufficiently large number of offline nodes, (near-)positive correlation is, generally, unavoidable: see Lemma 6.3 for a proof of such a Ramsey-theoretic statement. Instead, to achieve a better rounding ratio, we explicitly negatively correlate *the bids* of previously-non-bidding offline nodes when they do bid.

#### 3.1 The improved ODRS: Overview and intuition

In this section we outline our improved ODRS. We focus here and in most subsequent sections on simple matchings, for clarity's sake, deferring an extension to  $b$ -matchings to Appendix B.2.

**A bad example, and a false start.** Consider a graph with a single online node  $t$  with  $x_{it} = \frac{1}{n}$  for all  $n$  offline nodes. In this case, under independent bids,  $t$  gets no bids with probability  $(1 - \frac{1}{n})^n \approx \frac{1}{e}$ , and so one of the  $n$  offline nodes must get matched with probability no greater than  $(1 - \frac{1}{e}) \cdot \frac{1}{n}$ . For this simple example a rounding ratio of one is possible, by correlating bids for  $t$ : if we pick exactly one neighbor  $i$  to bid with probability  $x_{it}$ , then each offline node both bids and buys (is matched to)  $t$  with probability  $x_{it}$ . Unfortunately, this approach is impossible to implement in general: if



we let each offline node  $i$  bid for at most one time  $t$  (recall that this is a simple matching instance) with probability  $x_{i,t}$ , then, for  $s_{it} := \sum_{t' < t} x_{it'}$  the fractional degree of  $i$  at time  $t$ , we must have  $i$  bid for  $t$  with probability  $p_{it} := x_{it}/(1 - s_{it})$ , and yet  $\sum_i p_{it}$  may exceed one.

**Bin-packing low-degree nodes.** Our first step is to adopt a *partial* implementation of the above approach, by grouping offline nodes into groups  $B$  for which  $\sum_{i \in B} p_{it} \leq 1$ , and using one  $\text{Uni}(0, 1)$  variable to offer at most one bid per group. Specifically, if we denote the set of low-degree neighbors of  $t$  by  $L_t := \{i \in N(t) \mid s_{it} \leq \theta\}$  for some threshold  $\theta \in [0, 1]$  that we will optimize later, we will pack low-degree neighbors into bins using a greedy bin-packing algorithm such that each bin  $B$  has  $\sum_{i \in B} p_{i,t} \leq 1$ . Moreover, each such bin  $B$  (but one) has  $\sum_{i \in B} p_{i,t} \geq 1/2$  and hence has high total  $x$ -value of  $\sum_{i \in B} x_{i,t} \geq \frac{1-\theta}{2}$ . (Here we use that  $p_{i,t} = x_{i,t}/(1 - s_{i,t}) \leq x_{i,t}/(1 - \theta)$ .) Using negative association arguments (see Section 2), we can show that this approach essentially replaces multiple bids of small  $x$ -value with bids of  $x$ -value equal to the small bids' sum. In particular, if the total  $x$ -value of low-degree neighbors  $i \in L_t$  is high, this results in  $\min_{S \subseteq [n]} \frac{\Pr[S \cap P_t \neq \emptyset]}{\sum_{i \in S} x_{i,t}} \geq 1 - \frac{1}{e} + \Omega(1)$ , and a rounding ratio greater than  $1 - \frac{1}{e}$ , by Lemma 2.1.

**Group discounting and individual markup.** It remains to address the challenge of increasing  $\Pr[S \cap P_t \neq \emptyset]$  if the total  $x$ -value of low-degree nodes  $L_t$  is small, and so very little grouping occurs. For this, we note that from the perspective of every offline node  $i$ , a time step  $t$  has: (1) a value (matching probability), and (2) a cost (bidding probability). The latter is indeed a cost, since we do not allow offline nodes to bid more than once. The value at each time is monotone increasing in offline nodes' costs. Now, when we use grouping we decrease the contention faced by the CRS, as it now faces fewer bidding nodes, due to each bin only providing one bid (or none). We can therefore safely decrease the bidding probability (cost) in the case that much grouping occurs, giving the grouped nodes a *group discount*, while still obtaining a rounding ratio greater than  $1 - 1/e$ . The upshot of the group discount is that offline nodes now have a higher chance of not having bid before later time steps where they are not grouped, in which case such individual bidders can pay an *individual markup*, allowing us to guarantee that in this case too  $\min_{S \subseteq [n]} \frac{\Pr[S \cap P_t \neq \emptyset]}{\sum_{i \in S} x_{i,t}} \geq 1 - \frac{1}{e} + \Omega(1)$ , with the ensuing  $1 - \frac{1}{e} + \Omega(1)$  rounding ratio following from the offline CRS of Lemma 2.1.

### 3.2 The core of the improved ODRS

In this section we introduce our improved ODRS' core subroutine, Algorithm 1. (The parameter  $\theta \in [0, 1]$  will be optimized later, and for now it is safe to think of the input vectors  $\mathbf{x}, \mathbf{v}$  as both equaling the input fractional matching.) At each time step  $t$ , the algorithm groups low-fractional-degree offline nodes, and then all remaining nodes, into buckets, using the classic first-fit bin-packing algorithm,<sup>3</sup> with each offline node  $i$  having a size of  $\frac{x_{i,t}}{1-s_{i,t}}$ . This is the conditional probability with which  $i$  should bid for  $t$  if  $i$  is *free*, i.e., has not bid before, to guarantee a marginal bidding probability of  $x_{i,t}$ . (This follows from our online level-set rounding algorithm's probabilities.) Items are grouped into bins of size one, and then at most one offline node bids per bin (lines 9-12), resulting in the set of *candidates*,  $C_t$ . The free candidates  $P_t = C_t \cap F$  then *bid* for  $t$ , who is then matched to (at most) one bidding neighbor  $i \in P_t$ , chosen by a CRS.

First, we note that Algorithm 1—including the bin-packing steps that require  $\frac{x_{i,t}}{1-s_{i,t}} \leq 1$ —is well-defined provided  $\sum_t x_{i,t} \leq 1$  for all  $i, t$ , since this implies that  $x_{i,t} \leq 1 - \sum_{t' < t} x_{i,t'} = 1 - s_{i,t}$ . Next, by construction, each offline node bids (and is thus matched) at most once, while the CRS guarantees that each online node is matched at most once. To summarize, we have the following.

<sup>3</sup>This algorithm maintains a sorted list of bins with each bin  $B$  containing items of overall size at most one,  $\sum_{i \in B} s_i \leq 1$ , and for each item  $i$  in order, adds  $i$  to the first (possibly newly-opened) bin that has enough room left.

---

**Algorithm 1** ODRS-core( $\theta, \mathbf{x}, \mathbf{v}$ )

---

```
1:  $\mathcal{M} \leftarrow \emptyset$ 
2:  $F \leftarrow [n]$  ▷ Free offline nodes
3: for each time  $t$  do
4:   for each  $i \in [n]$  do
5:      $s_{i,t} \leftarrow \sum_{t' < t} x_{i,t'}$ 
6:      $\mathcal{B}_t \leftarrow \text{FirstFit} \left( \left\{ \left( i, \frac{x_{i,t}}{1-s_{i,t}} \right) \mid i \in [n], s_{i,t} \leq \theta \right\} \right)$  ▷ Bucketing low-degree nodes
7:      $\mathcal{B}_t \leftarrow \mathcal{B}_t \cup \text{FirstFit} \left( \left\{ \left( i, \frac{x_{i,t}}{1-s_{i,t}} \right) \mid i \in [n], s_{i,t} > \theta \right\} \right)$  ▷ Bucketing high-degree nodes
8:      $C_t \leftarrow \emptyset$  ▷ Candidates
9:     for each  $B \in \mathcal{B}_t$  do
10:       $U \sim \text{Uni}[0, 1]$ 
11:      if  $U \leq \sum_{i \in B} \frac{x_{i,t}}{1-s_{i,t}}$  then
12:         $C_t \leftarrow C_t \cup \min_i \left\{ i \in [k] \mid U \leq \sum_{\substack{j \in B \\ j \leq i}} \frac{x_{j,t}}{1-s_{j,t}} \right\}$ 
13:       $P_t \leftarrow F \cap C_t$  ▷ Bidders = free candidates
14:       $F \leftarrow F \setminus P_t$  ▷ Bidders cease being free
15:       $O \leftarrow \text{CRS}(P_t, \{v_{i,t}\}_i)$  ▷ Contention Resolution
16:      if  $|O| = 1$  then
17:         $\mathcal{M} \leftarrow \mathcal{M} \cup \{(i, t) \mid i \in O\}$ 
18: Output  $\mathcal{M}$ 
```

---

**Observation 3.2.** *Algorithm 1 is well-defined and outputs a matching if  $\sum_t x_{i,t} \leq 1 \ \forall i \in [n]$ .*

We turn to analyzing the algorithm's rounding ratio, starting with the following fact implying that if much grouping occurs, then most bins have high  $x$ -value.

**Fact 3.3.** *For each time  $t$ , at most one bin  $B \in \mathcal{B}_t$  at Line 6 has  $\sum_{i \in B} x_{i,t} < \frac{1-\theta}{2}$ .*

*Proof.* A well-known property of First-Fit is that at most one bin in the output of this algorithm is less than  $\frac{1}{2}$  full (see, e.g., [85, 86]). Therefore, since  $s_{i,t} \leq \theta$  for each offline node  $i$  in a bin  $B \in \mathcal{B}_t$  computed in Line 6, each such bin  $B$  (barring perhaps one) has

$$\frac{1}{2} \leq \sum_{i \in B} \frac{x_{i,t}}{1-s_{i,t}} \leq \sum_{i \in B} \frac{x_{i,t}}{1-\theta}. \quad \square$$

The following lemma, combined with Fact 3.3, provides a guarantee no worse than the independent-proposals approach. Indeed, as we will show shortly, it implies a better rounding ratio if much grouping of low-degree neighbors occurs.

**Lemma 3.4.** *For any time  $t$  and set of offline nodes  $S \subseteq [n]$ , Algorithm 1 satisfies*

$$\Pr[S \cap P_t \neq \emptyset] \geq 1 - \prod_{B \in \mathcal{B}_t} \left( 1 - \sum_{i \in B \cap S} x_{i,t} \right).$$

*Proof.* Let  $F_t$  denote  $F$  at time  $t$ , and let  $C_t$  and  $P_t = C_t \cap F_t$  be as in Algorithm 1. Next, let  $C_{i,t} = \mathbb{1}[i \in C_t]$  indicate whether  $i$  is a candidate at time  $t$ , and define  $F_{i,t}$  and  $P_{i,t}$  similarly. By independence of  $C_{i,t}$  and  $F_{i,t}$  and since  $\Pr[C_{i,t}] = \frac{x_{i,t}}{1-s_{i,t}}$  and  $\Pr[F_{i,t}] = 1 - s_{i,t}$  (provable by a

simple induction on  $t \geq 0$ ), we have that  $\Pr[P_{i,t}] = x_{i,t}$ . Therefore, by linearity of expectation and  $\sum_{i \in B} C_{i,t} \leq 1$ , we have that  $\Pr[P_t \cap S \cap B = \emptyset] = 1 - \sum_{i \in B \cap S} x_{i,t}$ . It remains to show that the events  $[P_t \cap S \cap B = \emptyset]$  for different  $B$  are negative cylinder dependent, giving the desired inequality,

$$\Pr[P_t \cap S = \emptyset] = \Pr \left[ \bigwedge_{B \in \mathcal{B}_t} (P_t \cap S \cap B = \emptyset) \right] \leq \prod_{B \in \mathcal{B}_t} \Pr[P_t \cap S \cap B = \emptyset] = \prod_{B \in \mathcal{B}_t} \left( 1 - \sum_{i \in B \cap S} x_{i,t} \right).$$

To this end, we will show that the events  $[P_t \cap S \cap B = \emptyset]$  are negatively associated (NA), which by Lemma 2.3 yields the required above inequality, and hence yields the lemma. First, by Lemma 2.4, for any bin  $B'$  at time  $t' < t$ , the binary variables  $C_{i,t'}$  for all nodes  $i \in B' \cap S$ , whose sum is at most one, are NA. Moreover, the distributions over different bins and different time steps are independent, and so by closure of NA under products (Lemma 2.5), all  $C_{i,t'}$  are NA. Therefore, the variables  $F_{i,t} = 1 - \max_{t' < t} C_{i,t'}$  are non-increasing functions of disjoint NA variables  $C_{i,t'}$ , and so by closure of NA under such functions (Lemma 2.5), the variables  $F_{i,t}$  are NA. Moreover, the variables  $C_{i,t}$  are independent from the variables  $F_{i,t}$  (who are functions only of  $C_{i,t'}$  for all  $t' < t$ ), and consequently, by another application of closure of NA under products (Lemma 2.5), all variables  $\{F_{i,t}, C_{i,t} \mid i \in S\}$  are NA. Finally, since  $\mathbb{1}[P_t \cap S \cap B = \emptyset] = 1 - \sum_{i \in B \cap S} C_{i,t} \cdot F_{i,t}$  are monotone decreasing functions of disjoint NA variables (different bins  $B$ ), then by closure of NA under such functions (Lemma 2.5), the variables  $\mathbb{1}[P_t \cap S \cap B = \emptyset]$  are NA, as desired.  $\square$

### 3.3 The improved ODRS

Lemma 3.4 and Fact 3.3 allow us to argue that if much of  $t$ 's fractional degree is contributed by neighbors  $i$  with low fractional degree,  $s_{i,t} \leq \theta$ , then  $\Pr[S \cap P_t \neq \emptyset] \geq (1 - \frac{1}{e} + \Omega(1)) \cdot \sum_{i \in S} x_{i,t}$ . By Lemma 2.1, this allows to match edges of such nodes  $t$  with probability greater than  $(1 - \frac{1}{e}) \cdot x_{i,t}$ . Indeed, we can afford to decrease the fractions  $x_{i,t}$  of low-degree nodes (that may be grouped effectively) and still retain a similar  $1 - \frac{1}{e} + \Omega(1)$  rounding ratio for such  $t$ . In return, we can afford to increase the fractions  $x_{i,t}$  of nodes of high degree (who might not be grouped). More precisely, we consider the following transformation to  $\mathbf{x}$ , capturing this aforementioned *group discount and individual markup*. For  $\theta := \frac{\delta}{\epsilon + \delta}$ , with  $\epsilon, \delta \in [0, 1]$  chosen later, we use the assignment  $\hat{\mathbf{x}} : E \rightarrow \mathbb{R}_{\geq 0}$ :

$$\hat{x}_{i,t} = x_{i,t} \cdot (1 - \epsilon) + \int_{z=\max(\theta, s_{i,t})}^{\max(\theta, s_{i,t} + x_{i,t})} (\epsilon + \delta) dz. \quad (2)$$

Intuitively,  $\hat{\mathbf{x}}$  decreases/increases the  $x$ -value of (parts of) edges of  $i$  until/after  $i$  has fractional degree  $\theta$ , by a  $(1 - \epsilon)$  and  $(1 + \delta)$  factor, respectively. So, for example,  $\hat{x}_{i,t} \geq x_{i,t} \cdot (1 - \epsilon)$  if  $s_{i,t} \leq \theta$  and  $\hat{x}_{i,t} = x_{i,t} \cdot (1 + \delta)$  if  $s_{i,t} > \theta$ . While  $\hat{\mathbf{x}}$  is not a fractional matching, since online nodes  $t$  may have  $\sum_i \hat{x}_{i,t} > 1$ , this assignment does satisfy the fractional degree constraints of *offline* nodes.

**Fact 3.5.** *For every offline node  $i \in [n]$  and time  $t$ , we have that  $\hat{s}_{i,t} := \sum_{t' < t} \hat{x}_{i,t'} \leq s_{i,t} \leq 1$ .*

*Proof.* If  $s_{i,t} \leq \theta$ , trivially  $\hat{s}_{i,t} = s_{i,t} \cdot (1 - \epsilon) \leq s_{i,t}$ . Otherwise, by our choice of  $\theta = \frac{\delta}{\epsilon + \delta}$ , we have

$$\hat{s}_{i,t} = s_{i,t} \cdot (1 - \epsilon) + (s_{i,t} - \theta)^+ \cdot (\epsilon + \delta) = s_{i,t} \cdot (1 + \delta) - \delta \leq s_{i,t}. \quad \square$$

Our ODRS, given in Algorithm 2, first scales the fractions as in Equation (2) (easily computable online, as these  $\hat{x}_{i,t}$  only depend on information known by time  $t$ ) and feeds the resultant vectors  $\hat{\mathbf{x}}, \mathbf{x}$  (playing the roles of  $\mathbf{x}$  and  $\mathbf{v}$ , respectively) into our core ODRS subroutine, Algorithm 1. By Fact 3.5 and Observation 3.2, we have that Algorithm 2 is a well-defined online algorithm, outputting a valid matching. It remains to analyze this algorithm's rounding ratio, starting with the following lemma.

---

**Algorithm 2** ODRS( $\epsilon, \delta, \mathbf{x}$ )

---

```

1:  $\theta \leftarrow \frac{\delta}{\epsilon + \delta}$ 
2: Init ODRS-core( $\theta \cdot (1 - \epsilon), \hat{\mathbf{x}}, \mathbf{x}$ )
3: for each time  $t$  do
4:   for each  $i \in [n]$  do
5:      $s_{i,t} \leftarrow \sum_{t' < t} x_{i,t'}$ 
6:     Compute  $\hat{x}_{i,t}$  from  $(x_{i,t}, s_{i,t}, \epsilon, \delta)$  as in Equation (2)
7:   Run next step of ODRS-core( $\theta \cdot (1 - \epsilon), \hat{\mathbf{x}}, \mathbf{x}$ )
8: Output  $\mathcal{M}$  computed by ODRS-core( $\theta \cdot (1 - \epsilon), \hat{\mathbf{x}}, \mathbf{x}$ )

```

---

**Lemma 3.6.** *Let  $\epsilon, \delta \in [0, 1]$  be such that the function  $f_{\epsilon, \delta}(z) := \exp(-z \cdot (1 + \delta)) - (1 - z \cdot (1 - \epsilon))$  satisfies:*

$$f_{\epsilon, \delta}(z) \geq 0 \quad \text{for all } z \geq \frac{1 - \theta}{2} = \frac{\epsilon}{2(\epsilon + \delta)}. \quad (3)$$

Then Algorithm 2 with parameters  $\epsilon$  and  $\delta$  has rounding ratio at least

$$1 - \exp\left(-1 - \delta + \frac{\epsilon + \delta}{1 - \epsilon}\right) \cdot \frac{1 - \epsilon}{1 + \delta}.$$

*Proof.* By construction of  $\hat{x}_{i,t}$ , we have that  $\hat{s}_{i,t} \leq \theta \cdot (1 - \epsilon)$  iff  $s_{i,t} \leq \theta$ . Define  $L_t := \{i \mid s_{i,t} \leq \theta\} = \{i \mid \hat{s}_{i,t} \leq \theta \cdot (1 - \epsilon)\}$ . By Lemma 3.4 and the Taylor expansion of  $1 - \exp(-z)$ , for any offline node set  $S \subseteq [n]$  and time  $t$ , the set of bidders  $P_t$  computed by ODRS-core( $\theta \cdot (1 - \epsilon), \hat{\mathbf{x}}, \mathbf{x}$ ) satisfies

$$\begin{aligned} \Pr[S \cap P_t \neq \emptyset] &\geq 1 - \prod_{B \not\subseteq L_t} \left(1 - \sum_{i \in B \cap S} \hat{x}_{it}\right) \cdot \prod_{B \subseteq L_t} \left(1 - \sum_{i \in B \cap S} \hat{x}_{it}\right) \\ &\geq 1 - \exp\left(-\sum_{i \in S \setminus L_t} x_{it} \cdot (1 + \delta)\right) \cdot \prod_{B \subseteq L_t} \left(1 - \sum_{i \in B \cap S} x_{it} \cdot (1 - \epsilon)\right). \end{aligned}$$

We now lower bound the ratio  $\frac{\Pr[S \cap P_t \neq \emptyset]}{\sum_{i \in S} x_{it}}$ . To do so, we note that our lower bound for  $\Pr[S \cap P_t \neq \emptyset]$  is a concave function in  $\sum_{i \in S} x_{it}$  in the range  $[0, 1]$ , and so our lower bound for the above ratio is minimized by virtually increasing  $\sum_{i \in S} x_{it}$  until  $\sum_{i \in S} x_{it} = 1$ , by adding all  $i \notin S$  to the set and possibly adding dummy nodes  $J$  with  $s_{j,t} = \theta$  for all  $j \in J$  and  $\sum_J x_{jt} = 1 - \sum_i x_{it} \geq 0$  (see Claim A.6).<sup>4</sup> On the other hand, by Fact 3.3, every bin  $B \subseteq L_t$  but at most one bin  $B^*$  has  $\sum_{i \in B} \hat{x}_{i,t} \geq \frac{1 - \theta(1 - \epsilon)}{2} \geq \frac{(1 - \theta)(1 - \epsilon)}{2}$ , and hence  $\sum_{i \in B} x_{i,t} \geq \frac{1 - \theta}{2}$ . Combining the above, we get

$$\begin{aligned} \frac{\Pr[S \cap P_t \neq \emptyset]}{\sum_i x_{it}} &\geq 1 - \exp\left(-\left(1 - \sum_{i \notin L_t} x_{it}\right) \cdot (1 + \delta)\right) \cdot \prod_{B \subseteq L_t} \left(1 - \sum_{i \in B} x_{it} \cdot (1 - \epsilon)\right) \\ &\geq 1 - \exp\left(-\left(1 - \sum_{i \in B^*} x_{it}\right) \cdot (1 + \delta)\right) \cdot \left(1 - \sum_{i \in B^*} x_{it} \cdot (1 - \epsilon)\right) \\ &\geq 1 - \exp\left(-1 - \delta + \frac{\epsilon + \delta}{1 - \epsilon}\right) \cdot \frac{1 - \epsilon}{1 + \delta}. \end{aligned} \quad (4)$$

---

<sup>4</sup>Concretely: adding any value to  $\sum_{i \in S} x_{i,t}$  corresponding to adding (part of) another (dummy) element to  $S$  decreases the value, so concavity of the resulting function together with Claim A.6 completes the proof of this claim.

Above, the second inequality follows from the lemma's hypothesis, in Equation (3), and the last inequality follows from the following claim, whose proof by inspection of  $g$  is deferred to Appendix B.

**Claim 3.7.** *If  $\epsilon, \delta \in [0, 1]$ , the function  $g(y) := 1 - \exp(-(1-y) \cdot (1+\delta)) \cdot (1-y \cdot (1-\epsilon))$  satisfies*

$$\forall y \in \mathbb{R} : \quad g(y) \geq g\left(\frac{\epsilon + \delta}{(1-\epsilon)(1+\delta)}\right) = 1 - \exp\left(-1 - \delta + \frac{\epsilon + \delta}{1-\epsilon}\right) \cdot \frac{1-\epsilon}{1+\delta}.$$

Finally, the rounding ratio of the ODRS then follows from Lemma 2.1 and Equation (4).  $\square$

Lemma 3.6 provides a sufficient condition for  $\epsilon, \delta$  to (possibly) allow for greater-than- $(1-1/e)$  rounding ratios, though this condition requires satisfying infinitely many non-linear constraints. The following lemma, which follows from convexity of  $f_{\epsilon, \delta}$  and is proven in Appendix B, provides a simpler condition that is sufficient to guarantee all the above infinitely-many constraints.

**Lemma 3.8.** *Let  $f_{\epsilon, \delta}(z)$  be as in Lemma 3.6. If  $f_{\epsilon, \delta}(a) \geq 0$  and  $f'_{\epsilon, \delta}(a) \geq 0$ , then  $f_{\epsilon, \delta}(b) \geq 0 \forall b \geq a$ .*

Combining the above, we are finally ready to optimize the rounding ratio of Algorithm 2.

**Theorem 3.9.** *Algorithm 2 run with  $(\epsilon, \delta) \approx (0.0480, 0.0643)$  achieves a rounding ratio of 0.652.*

*Proof.* By lemmas 3.6 and 3.8, the rounding ratio of Algorithm 1 with parameters  $(\epsilon, \delta)$  is at least as high as  $1 - \exp\left(-1 - \delta + \frac{\epsilon + \delta}{1-\epsilon}\right) \cdot \frac{1-\epsilon}{1+\delta}$ , provided that  $f_{\epsilon, \delta}\left(\frac{1-\theta}{2}\right) \geq 0$  and  $f'_{\epsilon, \delta}\left(\frac{1-\theta}{2}\right) \geq 0$ , where  $\frac{1-\theta}{2} = \frac{\epsilon}{2(\epsilon+\delta)}$  and  $f_{\epsilon, \delta}$  is as in Lemma 3.6. Thus, the algorithm's rounding ratio is at least as high as the value of the following program, when run with parameters given by its optimal solution  $(\epsilon, \delta)$ . Off-the-shelf numerical solvers yield a maximum value of  $\alpha = 0.652$  at  $(\epsilon, \delta) \approx (0.0480, 0.0643)$ .

$$\begin{aligned} \max \quad & \alpha = 1 - \exp\left(-1 - \delta + \frac{\epsilon + \delta}{1-\epsilon}\right) \cdot \frac{1-\epsilon}{1+\delta} \\ \text{s.t.} \quad & \exp\left(-\frac{\epsilon}{2(\epsilon+\delta)} \cdot (1+\delta)\right) - 1 + \frac{\epsilon}{2(\epsilon+\delta)} \cdot (1-\epsilon) \geq 0 \\ & -(1+\delta) \cdot \exp\left(-\frac{\epsilon}{2(\epsilon+\delta)} \cdot (1+\delta)\right) + 1 - \epsilon \geq 0 \\ & \epsilon, \delta \in [0, 1]. \end{aligned} \quad \square$$

**Computational considerations.** The algorithm implied by Theorem 3.9, as stated, takes exponential time, due to its reliance on the CRS of Lemma 2.1 for non-independent bid distributions. As we show in Appendix B.1, a slight downscaling of the  $x$ -values allows us to achieve essentially the same rounding ratio, while also guaranteeing that the number of bidders at each time step is small. This allows us to compute the support of the distribution of  $P_t$  in polynomial time, which by Lemma 2.1 then gives us the following polytime counterpart to Theorem 3.9.

**Theorem 3.10.** *For any  $\gamma > 0$ , there exists an  $n^{O(1/\gamma)}$ -time bipartite matching ODRS with rounding ratio of  $0.652 - \gamma$ .*

## 4 Online Level-Set Rounding

In this section we present two online level-set rounding algorithms. The guarantees of such algorithms, first obtained by [79, 80] in *offline settings*, are as follows.

On input  $\mathbf{x} \in [0, 1]^n$  with integer sum  $\sum_i x_i \in \mathbb{Z}$  (possibly by adding a dummy value) and partial sums  $s_j := \sum_{i \leq j} x_i$ , a level-set rounding algorithm must output a set  $\mathcal{S} \subseteq [n]$  with characteristic vector  $\mathbf{X}$  given by  $X_i := \mathbf{1}[i \in \mathcal{S}]$  with partial sums  $S_j := \sum_{i \leq j} X_i = |\mathcal{S} \cap [j]|$  satisfying the following properties for all  $i$ .

(P1) **(Marginals)**  $\mathbb{E}[X_i] = x_i$ .

(P2) **(Rounding)**  $S_i \in \{\lfloor s_i \rfloor, \lceil s_i \rceil\}$ .

Property (P1) and linearity of expectation imply that  $\mathbb{E}[S_i] = s_i$ . Property (P2) requires that  $S_i$  always equals this expectation, “up to rounding”.

In this work, we require *online* counterparts to the above. Here, the input is again a sequence of reals,  $x_1, x_2, \dots \in [0, 1]$ , but now they are revealed in an online fashion. An online level-set rounding algorithm maintains a monotonically increasing set  $\mathcal{S} \subseteq \mathbb{N}$  (initially empty), and decides at time  $t$  (when  $x_t$  is revealed) whether to add  $t$  to its output set  $\mathcal{S}$ , immediately and irrevocably.

As we shall see shortly, obtaining these first two properties is easy to achieve online. However, for several of our applications in Section 5 it will be more useful to additionally guarantee strong negative correlation between the elements of  $\mathbf{X}$ . Specifically, we require the following.

(P3) **(Strongly Rayleigh)** The joint distribution  $\mathbf{X}$  is strongly Rayleigh (and thus, also NA).

**A warm-up:** We start with a simple algorithm that satisfies Properties (P1) and (P2), but not Property (P3).<sup>5</sup> Upon initialization, we draw a uniform threshold  $\tau \sim \text{Uni}[0, 1]$ . Next, for every time step  $j$ , for  $s_j := \sum_{i \leq j} x_i$  the running sum until the  $j^{\text{th}}$  arrival (inclusive), we add element  $j$  to the output set  $\mathcal{S}$  if the interval  $(s_{j-1}, s_j]$  contains a point  $n + \tau$  for some integer  $n$ . Thus,  $\Pr[i \in \mathcal{S}] = \Pr[(s_{i-1}, s_i] \cap (\mathbb{N} + \tau)] = s_i - s_{i-1} = x_i$ , and this algorithm satisfies Property (P1). Moreover, as we output one element for any sum interval  $(n, n + 1]$ , it is easy to see that the output by time  $j$  contains either  $\lfloor s_j \rfloor$  or  $\lceil s_j \rceil$  many elements, and so this algorithm satisfies Property (P2). To see that this algorithm does not satisfy (P3), which implies pairwise negative correlation, we note that for an input  $\mathbf{x} = \frac{1}{2} \cdot \mathbf{1}$  (i.e.,  $x_i = 1/2$  for all  $i$ ), negative correlation is (very much) violated, since  $X_i = X_{i+2}$  for all  $i$ .

The main contribution of this section is an online level-set rounding algorithm satisfying both basic properties of an online level-set algorithm, and the strong negative correlation Property (P3) (necessary for some of our applications in Section 5). As this algorithm’s analysis involves coupling with the *offline* level-set rounding algorithm of Srinivasan [79, 80], we first start by revisiting this offline algorithm and its properties.

## 4.1 Offline level-set rounding

Here we consider an *offline* procedure due to [79]. Recall that our objective will be to design an *online* algorithm satisfying the aforementioned properties, but for now we revisit the offline algorithm of [79] and prove that when run with a certain specific way of choosing the key indices  $i_1$  and  $i_2$  (see Algorithm 3), then it satisfies the above three properties.

As usual, “min( $S$ )” in Algorithm 3 refers to the minimum element of a nonempty set  $S$ ; the index  $i_2$  in Algorithm 3 is always well-defined since  $\sum_i x_i \in \mathbb{Z}$  and, as we shall see,  $\sum_i y_i = \sum_i x_i$  always. We emphasize that the algorithm of [79] allows much liberty in the choice of  $i_1$  and  $i_2$ , and that our specific choice of  $i_1$  and  $i_2$  is required to satisfy Property (P2). Also, the random choice in each call to STEP is independent of the random choices of the past.

---

<sup>5</sup>We thank an anonymous reviewer for this suggestion.



---

**Algorithm 3** Offline Level-Set Rounding

---

```

1: Initialize  $\mathbf{y} \leftarrow \mathbf{x}$ 
2: while  $\text{frac}(y) := \{i \mid y_i \notin \{0, 1\}\} \neq \emptyset$  do
3:   Let  $i_1 \leftarrow \min(\text{frac}(y))$ 
4:   Let  $i_2 \leftarrow \min(\text{frac}(y) \setminus \{i_1\})$ 
5:   STEP( $y_{i_1}, y_{i_2}$ )
6: Output  $\mathcal{S} := \{i \mid y_i = 1\}$ 

```

---



---

**Algorithm 4** STEP( $A, B$ )  $\triangleright$  modifies inputs

---

```

1: if  $A + B < 1$  then
2:    $(A, B) \leftarrow \begin{cases} (A + B, 0) & \text{w.p. } \frac{A}{A+B} \\ (0, A + B) & \text{w.p. } \frac{B}{A+B} \end{cases}$ 
3: else  $\triangleright A + B \in [1, 2)$ 
4:    $(A, B) \leftarrow \begin{cases} (1, A + B - 1) & \text{w.p. } \frac{1-B}{2-A-B} \\ (A + B - 1, 1) & \text{w.p. } \frac{1-A}{2-A-B} \end{cases}$ 

```

---

That Algorithm 3 terminates and outputs a random set satisfying Property (P1) is known [79]. Nonetheless, for completeness, we provide a proof of this fact in Appendix C. In the same appendix we prove by induction on the number of invocations of STEP() that every prefix sum is preserved with probability one, up to rounding, and so Algorithm 3 satisfies Property (P2).

**Lemma 4.1.** *Let  $s_j := \sum_{i \leq j} x_i$  and  $Y_j^t := \sum_{i \leq j} y_i^t$ , where  $y_i^t$  is the value  $y_i$  after  $t$  applications of STEP in Algorithm 3. Then,  $Y_j^t \in \llbracket s_j \rrbracket, \lceil s_j \rceil$  for all  $j$  and  $t$ .*

Strong negative correlation (Property (P3)) was proven in [19] for any instantiation of [79], as long as we use some fixed choice of  $i_1$  and  $i_2$ , as we do in Algorithm 3:

**Lemma 4.2.** *Let  $X_i := \mathbb{1}[i \in \mathcal{S}]$  be the indicators for  $i$  being in the output set of Algorithm 3. Then,  $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_n)$  is a family of strongly Rayleigh random variables.*

To conclude, we have the following.

**Lemma 4.3.** *Algorithm 3 satisfies properties (P1) (P2) and (P3).*

## 4.2 The online level-set rounding algorithm

We now turn to providing an online counterpart to Algorithm 3. Here, the input is a sequence of reals,  $x_1, x_2, \dots \in [0, 1]$ , revealed in an online fashion. Algorithm 5 maintains a set  $\mathcal{S} \subseteq \mathbb{N}$  (initially empty), and decides at time  $t$  (when  $x_t$  is revealed) whether to add  $t$  to its output set  $\mathcal{S}$ , immediately and irrevocably. Let  $X_t := \mathbb{1}[t \in \mathcal{S}]$ , and let  $S_t := \sum_{t' \leq t} X_{t'}$  denote the cardinality of  $\mathcal{S}$  right after time  $t$ , i.e.,  $S_t = |\mathcal{S} \cap [t]|$ . Finally, let  $s_t := \sum_{t' \leq t} x_{t'}$ . We will show that Algorithm 5 satisfies properties (P1), (P2) and (P3) given by its offline counterpart, Algorithm 3.

---

**Algorithm 5** Online Level-Set Rounding

---

```

1: Initialize  $\mathcal{S} \leftarrow \emptyset$ 
2: for arrival  $x_t$  (at time  $t \geq 1$ ) do
3:   add  $t$  to  $\mathcal{S}$  with probability  $p_t := \begin{cases} 0 & |\mathcal{S}| = \lceil s_t \rceil \\ 1 & |\mathcal{S}| < \lfloor s_t \rfloor \\ \frac{x_t}{\lfloor s_{t-1} \rfloor + 1 - s_{t-1}} & |\mathcal{S}| = \lfloor s_t \rfloor = \lfloor s_{t-1} \rfloor \\ \frac{s_t - \lfloor s_t \rfloor}{s_{t-1} - \lfloor s_{t-1} \rfloor} & |\mathcal{S}| = \lfloor s_t \rfloor > \lfloor s_{t-1} \rfloor \text{ and } s_{t-1} \neq \lfloor s_{t-1} \rfloor \\ 0 & \text{else.} \end{cases}$ 
4: Output  $\mathcal{S}$ 

```

---

The random bits used to make the random choice in each step of Algorithm 5 are independent of the random bits used in the past: i.e., while  $p_t$  depends on  $S_{t-1}$ , the random bits used to generate the event of probability  $p_t$  in step  $t$ , are independent of the random bits of past steps.

### 4.3 Coupling the online and offline algorithms

Here we prove that Algorithm 5 satisfies properties (P1), (P2) and (P3). Proving properties (P1), (P2) directly is not too hard (see Lemma C.2).<sup>6</sup> Proving Property (P3), on the other hand, is slightly more involved; we do so now, using that the much-simpler properties (P1) and (P2) hold for both the offline and online algorithms. To prove that the last property holds, we show that both algorithms induce the *same distribution* over outputs. We outline a proof here, deferring a complete proof to Appendix C.

**Theorem 4.4.** *Fix a vector  $\mathbf{x} \in \{0, 1\}^n$ . Let  $\mathcal{D}$  and  $\mathcal{D}'$  denote the probability distributions on  $\{0, 1\}^n$  induced by the online and offline algorithms run on  $\mathbf{x}$ , respectively. Then,  $\mathcal{D} = \mathcal{D}'$ .*

*Proof (Sketch).* We prove by induction on  $t \in [n]$  that the following holds:

$$\forall (b_1, \dots, b_{t-1}) \in \{0, 1\}^{t-1}, \Pr_{\mathcal{D}}[X_t = 1 \mid (\forall i < t, X_i = b_i)] = \Pr_{\mathcal{D}'}[X_t = 1 \mid (\forall i < t, X_i = b_i)]. \quad (5)$$

The (strong) inductive hypothesis, whereby Equation (5) holds for all  $t' < t$ , clearly implies that for all  $(b_1, \dots, b_t) \in \{0, 1\}^t$  we have that  $\Pr_{\mathcal{D}}[(\forall i < t, X_i = b_i)] = \Pr_{\mathcal{D}'}[(\forall i < t, X_i = b_i)]$ . Similarly, Equation (5) holds for all  $t \leq n$  iff the two distributions are equal. The proof then proceeds by considering the different possibilities for the prefix sums  $s_{t-1}$  and  $s_t$ , which for the online algorithm directly determine the above conditional probability. These conditions also determine the offline algorithm's performance up to the  $(t-1)^{st}$  invocation of STEP(), from which a careful analysis implies that the conditional probabilities of the offline and online algorithms are equal.  $\square$

Theorem 4.4 combined with Lemma 4.3 immediately implies the desired properties of our Algorithm 5, as these properties are determined by the distribution over output sets.

**Theorem 4.5.** *Algorithm 5 is an online algorithm satisfying properties (P1), (P2) and (P3).*

## 5 Applications

In this section we provide applications and extensions of our online rounding algorithms. Mirroring the exposition in Section 1.1, we start with the application of our online ( $b$ -)matching algorithms, omitting the edge-weighted matching application with ML predictions, which follows directly from our main result.

### 5.1 Applications and extensions of the matching ODRS

#### 5.1.1 Online edge coloring (multigraphs)

We show here how to obtain the first online algorithm that achieves a better-than-2 competitive factor for online edge-coloring in bipartite *multigraphs* under adversarial arrivals, specifically under one-sided node arrivals, as studied in [31]. We assume that upon arrival of each node, its neighboring edges are revealed and must be colored. As already mentioned, for online edge coloring, it is known that the greedy algorithm's competitive ratio of 2 is tight for graphs of low maximum degree  $\Delta = O(\log n)$  [8], while better competitive ratios are known for high-degree graphs under various arrival models, all for simple graphs [6, 11, 14, 15, 31, 63, 78], with the exception of the random-order result of [3].

---

<sup>6</sup>In fact, Algorithm 5 can be shown to be the *only* online algorithm satisfying both these simple properties if we further restrict it to only storing in memory the partial sums  $s_{t-1}$  and  $S_{t-1}$  at the beginning of step  $t$ .

We obtain our improved bounds for multigraphs by extending a reduction from online edge coloring to “fair” matching, suggested by [31] for simple graphs (see also [78, Appendix B]), given in Lemma 5.2.

**Definition 5.1.** *We say that a random matching  $\mathcal{M}$  in a multigraph with maximum degree  $\Delta$  is  $\alpha$ -fair if for each (parallel) edge  $e$  we have that  $\Pr[e \in \mathcal{M}] \geq \frac{1}{\alpha\Delta}$ .*

An  $\alpha$ -fair algorithm with  $\alpha = 1$  gives the highest uniform lower bound on edges’ matching probability. An  $\alpha$ -fair algorithm with other  $\alpha$  ( $\alpha > 1$ ) therefore  $\alpha$ -approximates this maximally-fair algorithm.

With the above terminology, we can now state the reduction.

**Lemma 5.2.** *An online  $\alpha$ -fair matching algorithm yields an online  $(\alpha + o(1))\Delta$ -edge-coloring algorithm for  $n$ -node bipartite multigraphs with maximum degree  $\Delta = \omega(\log n)$ .*

We defer a brief proof sketch of the above to the end of the section, and turn to discussing its consequences and converses.

**Corollary 5.3.** *An ODRS with rounding ratio  $1/\alpha$  yields an online  $(\alpha + o(1))\Delta$ -edge-coloring algorithm for  $n$ -node multigraphs with maximum degree  $\Delta = \omega(\log n)$ .*

*Proof.* Consider the fractional matching  $\mathbf{x}$  assigning value  $x_e = \frac{\kappa(e)}{\Delta}$  to the simple edge  $e$  with  $\kappa(e)$  parallel copies in the multigraph. (Note that  $\sum_{e \ni v} \kappa(e) \leq \Delta$  by definition, so this is indeed a fractional matching.) Applying an ODRS with rounding ratio  $1/\alpha$  to this fractional matching  $\mathbf{x}$  yields a randomized matching  $\mathcal{M}$  that matches every (parallel) edge in the graph with probability  $\frac{1}{\alpha\Delta}$ , i.e., it yields an  $\alpha$ -fair matching.  $\square$

We note that there is in fact an equivalence between  $\alpha$ -fair matching algorithms and  $\alpha$ -competitive edge-coloring algorithms, as “the converse” also holds.

**Observation 5.4.** *An online  $\alpha\Delta$ -edge-coloring algorithm  $\mathcal{A}$  for  $n$ -node multigraphs with maximum degree  $\Delta$  yields an ODRS with rounding ratio  $1/\alpha$  for fractional matchings  $\mathbf{x}$  with  $x_e \cdot \Delta$  integral for all  $e$ .*

*Proof.* Given a fractional matching  $\mathbf{x} \in \mathbb{Q}^E$  in a simple graph  $G = (V, E)$ , using that  $x_e \cdot \Delta$  is integral for each edge  $e$ , we provide to  $\mathcal{A}$  a multigraph  $\mathcal{H} = (V, F)$  with each edge  $e$  having multiplicity  $\kappa(e) = x_e \cdot \Delta$  in  $\mathcal{H}$ , and then randomly sample one of the  $\alpha\Delta$  edge colors (matchings)  $\mathcal{M}$  computed by  $\mathcal{A}$ . This yields an ODRS with rounding ratio of  $1/\alpha$ , since for each edge  $e \in E$  we have that

$$\Pr[e \in \mathcal{M}] = \kappa(e) \cdot \frac{1}{\alpha\Delta} = \frac{x_e}{\alpha}. \quad \square$$

By the equivalence between edge coloring multigraphs and ODRSes for matchings, together with our (upper and lower) bounds on the rounding ratio of such ODRSes, we obtain the following bounds on the number of colors needed to edge-color multigraphs online.

**Theorem 5.5.** *In  $n$ -node bipartite multigraphs with maximum degree  $\Delta$  under adversarial one-sided vertex arrivals, there exists an online  $1.533\Delta$ -edge-coloring algorithm, assuming  $\Delta = \omega(\log n)$ . In contrast, no  $(1/(2\sqrt{2} - 2 + \epsilon))\Delta \approx (1.207 - O(\epsilon))\Delta$ -edge-coloring exists even for  $\Delta = 2$ .*

We note in passing that all known positive results for online edge coloring under adversarial arrivals follow from Lemma 5.2, and indeed from Corollary 5.3 [31, 63, 78]. The reason most of these results do not extend to multigraphs is that their ODRSes required their input fractional

matchings  $\mathbf{x}$  to be of the form  $x_e = \frac{1}{\Delta}$ , or more generally for  $x_e = o(1)$  for each edge  $e$  (including its multiplicities). The work of Saberi and Wajc [78] is a lone exception to this rule, and combining its ODRS with Lemma 5.2 also yields an algorithm for multigraphs, but using as many as  $1.9\Delta$  colors.

We conclude the section with a brief proof sketch of Lemma 5.2.

*Proof sketch of Lemma 5.2.* We describe the algorithm in an offline setting first. Let  $C$  be such that  $C = \omega(\log n)$  and  $C = o(\Delta)$ . (Such  $C$  exist, by our necessary assumption that  $\Delta = \omega(\log n)$ .) Initially, the entire multigraph is uncolored. For some  $\Delta/C$  many rounds, compute  $\alpha \cdot C$  colors as follows: In the uncolored subgraph  $U$ , compute  $\alpha \cdot C$  many  $\alpha$ -fair matchings  $\mathcal{M}_1, \dots, \mathcal{M}_{\alpha C}$ , and let these occupy the next  $\alpha C$  colors. Since every vertex has at most  $\alpha C = o(\Delta)$  many edges colored during a round, any vertex with degree close to  $\Delta$  at the beginning of the round has degree  $\Delta - o(\Delta)$  by the end of the round. Therefore, since these  $\mathcal{M}_i$  are  $\alpha$ -fair matchings, and so match each (parallel) edge with probability at least  $\frac{1}{\alpha \cdot \Delta}$ , by standard concentration bounds, all vertices that have degree close to  $\Delta$  have some  $C \cdot (1 - o(1))$  many edges colored during a round (with high probability). This allows us to compute a high-probability upper bound of on the resultant uncolored multigraph at the end of the round (possibly useful to compute  $\alpha$ -fair matchings), and moreover implies that after  $\Delta/C$  many rounds (and so with  $(\Delta/C) \cdot \alpha \cdot C = \alpha \cdot \Delta$  colors used), the uncolored multigraph has maximum degree  $\Delta - (\Delta/C) \cdot C \cdot (1 - o(1)) = o(\Delta)$ , and can be colored greedily using a further  $2 \cdot o(\Delta) = o(\Delta)$  many colors, for  $(\alpha + o(1))\Delta$  colors overall.

While the above algorithm was described in an offline setting, if one can compute an  $\alpha$ -fair algorithm online (possibly with information regarding the high-probability upper bound on  $\Delta$  in each round), then the entire algorithm can be made to run online: for each (vertex/edge) arrival, perform the next steps of the  $\alpha\Delta$  many  $\alpha$ -fair matching algorithms outputting color classes  $\mathcal{M}_1, \dots, \mathcal{M}_{\alpha\Delta}$ , where for the  $i$ -th such algorithm, we simulate only the arrival of the parts of the subgraph that are not contained  $\mathcal{M}_1, \dots, \mathcal{M}_{i-1}$ . The same approach is used to simulate the final greedy coloring steps, using the fact that the greedy  $(2\Delta - 1)$ -edge-coloring algorithm is an online algorithm.  $\square$

### 5.1.2 Stochastic extension

In this section, we generalize our matching ODRS and its analysis to the following stochastic online bipartite matching problem. At each time  $t$ , an online node arrives with weight vector  $\mathbf{w}_t \sim \mathcal{D}_t$ , where  $\mathcal{D}_1, \mathcal{D}_2, \dots$  are a priori known independent distributions. A simple example which we focus on for notational simplicity (though our approach generalizes to this problem in full generality) is as follows: each online node  $t$  arrives according to a Bernoulli event  $A_t \sim \text{Ber}(p_t)$ , each edge  $(i, t)$  having some intrinsic value  $w_{it}$ , but realized weight  $v_{it} = w_{it} \cdot A_t$ . This problem (in its full generality) is an online Bayesian selection problem, and a number of algorithms with competitive ratio of  $\frac{1}{2}$  are known for it [41, 43, 47]. By the classic lower bound of Krengel and Sucheston for the single-item prophet inequality problem [62], the above ratio is best-possible when comparing with the offline optimum algorithm.

In [73], Papadimitriou et al. initiate the study of this problem in terms of the (polytime) approximability of the optimum *online* algorithm. Perhaps surprisingly, they show that while for simple cases of the problem (e.g., the single-offline-node problem), the optimum online algorithm is computable using a simple poly-sized dynamic program, in general it is PSPACE-hard to approximate the optimum online algorithm within some absolute constant  $\alpha < 1$  (say,  $\alpha \approx 0.999999$ ), and showed that a better-than-0.5 approximation of the optimum online algorithm can be obtained by polytime online algorithms. This positive result was subsequently improved to 0.526 implicitly [78] and to  $1 - 1/e \approx 0.632$  explicitly [20]. All these results are achieved by rounding the following LP, which

can be shown to upper bound the optimum online algorithm's value [73, 84].

$$\max \sum_{i,t} w_{i,t} \cdot x_{i,t} \quad (\text{LP } OPT_{on})$$

$$\text{s.t. } \sum_{t,r} x_{i,t} \leq 1 \quad \forall i \in [n] \quad (6)$$

$$\sum_i x_{i,t} \leq p_t \quad \forall t \quad (7)$$

$$x_{i,t} \leq p_t \cdot \left(1 - \sum_{t' < t} x_{i,t'}\right) \quad \forall i, t \quad (8)$$

$$x_{i,t} \geq 0 \quad \forall i, t. \quad (9)$$

**Lemma 5.6.** *The optimum of the above LP upper bounds the expected gain of any online algorithm.*

Most of these constraints are evident, and follow from (expected) matching constraints, which apply also to offline algorithms. The one constraint which applies to online algorithms but not to offline ones is Constraint (8). This constraint follows from the fact that if offline node  $i$  is matched to  $t$ , then  $t$  has already arrived, and that  $i$  is unmatched before time  $t$  — two events which are independent for online algorithms.

We note that in their paper's full version, Papadimitriou et al. [73] (private communication) show that the above LP has a “gap” of  $1 - \frac{1}{2e}$ , in the sense that no online algorithm (even computationally-unbounded ones) can get a better-than- $(1 - \frac{1}{2e})$  fraction of this LP's value. So, an approximation ratio greater than  $1 - \frac{1}{2e}$  is unachievable using this LP, and a ratio of  $1 - \frac{1}{e}$  is achievable [20]. Is the latter bound tight? In what follows, we show how the approach underlying our ODRSes, combined with (and simplifying) the high-level analysis of [20], allows one to break this bound.

**ODRS for the stochastic setting.** As in the non-stochastic setting of Section 3, specifically Algorithm 1, we group offline nodes and have them coordinate a single bid per group (bin). Similarly, our stochastic counterpart to Algorithm 2 applies group discounting and individual markup to the target marginal bidding probability, and then calls the (soon-to-be) modified Algorithm 1 with parameter  $\theta \cdot (1 - \epsilon)$  and vector  $\hat{x}_{i,t}$  given by Equation (2), restated below for ease of reference.

$$\hat{x}_{i,t} = x_{i,t} \cdot (1 - \epsilon) + \int_{z=\max(\theta, s_{i,t})}^{\max(\theta, s_{i,t} + x_{i,t})} (\epsilon + \delta) dz.$$

The three main differences compared to Section 3 are as follows: First, to account for the probability  $p_t$  of  $t$  to arrive, the conditional probability for  $i$  to bid if  $i$  is free and  $t$  arrives used in Line 6-Line 12 is now  $\frac{\hat{x}_{i,t}}{p_t \cdot (1 - \hat{s}_{i,t})}$ , for  $\hat{s}_{i,t} := \sum_{t' < t} \hat{x}_{i,t'}$ . Second, we allow offline nodes to bid more than once, and denote them as *free* (i.e., belonging to  $F$ ) as long as they are not matched. This avoids offline nodes becoming positively correlated due to an unlikely online node's arrival. The offline nodes' matching constraints are trivially satisfied. Finally, satisfying the online nodes' matching constraints is even easier than in Algorithm 1 for the non-stochastic setting. Rather than using a CRS, we have an arriving online node  $t$  that receives bids simply greedily match to the bidding node  $i$  maximizing the value  $w_{i,t}$ . See the pseudocode in Algorithm 6.

**Fact 5.7.** *Algorithm 6 is well-defined. In particular,  $\frac{\hat{x}_{i,t}}{p_t \cdot (1 - \hat{s}_{i,t})} \leq 1$  for every pair  $(i, t)$ .*

---

**Algorithm 6** Stochastic Matching Algorithm
 

---

```

1:  $\mathcal{M} \leftarrow \emptyset$ 
2:  $\mathbf{x} \leftarrow$  solution to (LP  $OPT_{on}$ )
3: for each time  $t$  do
4:   for each  $i \in [n]$  do
5:     compute  $\hat{x}_{i,t}$  as in Equation (2), and  $\hat{s}_{i,t} = \sum_{t' < t} \hat{x}_{i,t'}$ .
6:    $\mathcal{B}_t \leftarrow \text{FirstFit} \left( \left\{ \left( i, \frac{\hat{x}_{i,t}}{p_t \cdot (1 - \hat{s}_{i,t})} \right) \mid i \in [n], s_{i,t} \leq \theta \right\} \right)$   $\triangleright$  Bucketing low-degree nodes
7:    $\mathcal{B}_t \leftarrow \mathcal{B}_t \cup \{i\} \mid i \in [n], s_{i,t} > \theta$   $\triangleright$  Trivially bucketing high-degree nodes
8:    $C_t \leftarrow \emptyset$   $\triangleright$  Candidates
9:   for each  $B \in \mathcal{B}_t$  do
10:     $U \sim \text{Uni}[0, 1]$ 
11:    if  $U \leq \sum_{i \in B} \frac{\hat{x}_{i,t}}{p_t \cdot (1 - \hat{s}_{i,t})}$  then
12:       $C_t \leftarrow C_t \cup \min_i \left\{ i \in [k] \mid U \leq \sum_{j \in B, j \leq i} \frac{\hat{x}_{j,t}}{p_t \cdot (1 - \hat{s}_{j,t})} \right\}$ 
13:    $P_t \leftarrow C_t \setminus V(\mathcal{M})$   $\triangleright$  Bidders = free candidates
14:   if  $P_t \neq \emptyset$  and  $t$  arrived then
15:     pick some  $i \in \arg \max \{w_{i,t} \mid i \in P_t\}$ 
16:      $\mathcal{M} \leftarrow \mathcal{M} \cup \{(i, t)\}$ 
17: Output  $\mathcal{M}$ 

```

---

*Proof.* By Fact 3.5, we have that  $\hat{s}_{i,t} = s_{i,t} \cdot (1 - \epsilon) + (s_{i,t} - \theta)^+ \cdot (\epsilon + \delta) \leq s_{i,t}$ . Therefore, by our choice of  $\theta = \frac{\delta}{\epsilon + \delta}$  and LP Constraint (8), we have that, indeed

$$\frac{\hat{x}_{i,t}}{p_t \cdot (1 - \hat{s}_{i,t})} \leq \begin{cases} \frac{x_{i,t} \cdot (1 - \epsilon)}{p_t \cdot (1 - s_{i,t})} \leq \frac{x_{i,t}}{p_t \cdot (1 - s_{i,t})} \leq 1 & s_{i,t} \leq \theta \\ \frac{x_{i,t} \cdot (1 + \delta)}{p_t \cdot (1 - s_{i,t} + \delta)} = \frac{x_{i,t}}{p_t \cdot (1 - s_{i,t})} \leq 1 & s_{i,t} > \theta. \end{cases} \quad \square$$

Algorithm 6 is clearly a polytime algorithm. It remains to analyze its approximation ratio.

The following lemma makes explicit a fact implicit in the proof of the main theorem of [20]: specifically, that a per-online-node “approximation ratio”, relating  $w(\mathcal{M}(t), t)$ —the weight obtained by matching  $t$ —to the LP solution’s value from  $t$ , implies a global approximation ratio of the same value. We provide a short self-contained proof of this fact for completeness.

**Lemma 5.8.** *If for each online node  $t$  and  $z \geq 0$  we have that  $\Pr[w(\mathcal{M}(t), t) \geq z] \geq \sum_{i: w_{i,t} \geq z} \alpha \cdot x_{i,t}$ , then the algorithm is an  $\alpha$ -approximation of the optimum online algorithm.*

*Proof.* Denote by  $w_t := w(\mathcal{M}(t), t)$  the weight of the matched edge of  $t$ , with  $w_t = 0$  if  $t$  is not matched (possibly due to non-arrival). Then, the weight of the output matching  $\mathcal{M}$  satisfies

$$\sum_t \mathbb{E}[w_t] = \sum_t \int_{z=0}^{\infty} \Pr[w_t \geq z] dz \geq \sum_t \int_{z=0}^{\infty} \alpha \cdot \sum_{i: w_{i,t} \geq z} x_{i,t} dz = \alpha \cdot \sum_{i,t} w_{i,t} \cdot x_{i,t}.$$

The proof is then completed by Lemma 5.6, implying that the RHS upper bounds  $\alpha$  times the value of the optimum online algorithm’s gain.  $\square$

The objective of our analysis will therefore be to lower bound  $\Pr[w(\mathcal{M}(t)) \geq z]$  for all  $z \geq 0$ .



In our analysis, we let  $F_t = [n] \setminus V(\mathcal{M})$  be the set of free offline nodes at time  $t$ , and let  $F_{i,t} = \mathbb{1}[i \in F_t] = 1 - \sum_{t' < t} \mathbb{1}[(i, t') \in \mathcal{M}]$  indicate whether  $i$  is free by time  $t$ . Similarly, we let  $E_{i,t} = 1 - F_{i,t}$  indicate whether  $i$  was matched (“engaged”) before time  $t$ .

Our first observation is that our modification to the notion of freedom results in the following lower bound on the probability of a node being free at time  $t$ .

**Observation 5.9.** *For each time  $t$  and offline  $i \in [n]$ , we have that  $\Pr[F_{i,t}] \geq 1 - \hat{s}_{i,t}$ .*

*Proof.* Let  $B_{i,t}^1$  indicate that  $i$  bid for the first time before time  $t$ . We prove that  $\mathbb{E}[B_{i,t}^1] = \hat{s}_{i,t}$ , by induction on  $t \geq 1$ . The base case, is trivial, as  $F_{i,1} \equiv 1$  and  $\hat{s}_{i,1} \equiv 0$ . Now, by induction, we have

$$\mathbb{E}[B_{i,t+1}^1 - B_{i,t}^1] = p_t \cdot \frac{\hat{x}_{i,t}}{p_t \cdot (1 - \hat{s}_{i,t})} \cdot (1 - \mathbb{E}[B_{i,t}^1]) = \hat{x}_{i,t},$$

where the first equality relies on independence of  $t$ ’s arrival,  $i$  becoming a candidate at time  $t$ , and  $i$  not having yet bid, and the second equality follows from the inductive hypothesis. By linearity, we then have that  $\mathbb{E}[B_{i,t+1}^1] = \hat{s}_{i,t+1}$ . Since  $1 - F_{i,t} = E_{i,t} \leq B_{i,t}^1$ , the claim follows.  $\square$

Next, for any time  $t$  and offline node set  $S \subseteq [n]$ , we let  $F_{S,t} = \bigwedge_{i \in S} F_{i,t}$  and  $E_{S,t} = \bigwedge_{i \in S} E_{i,t}$  indicate whether all of  $S$  is free (respectively, engaged) by time  $t$ . So far, we have established that  $\Pr[E_{i,t}] \leq \hat{s}_{i,t}$ . The following lemma, whose statement and proof mirror that of [20, Lemma 3.3] (who use  $x_{i,t}$  and  $s_{i,t}$ , as opposed to  $\hat{x}_{i,t}$  and  $\hat{s}_{i,t}$ ), asserts that this upper bound is “sub-multiplicative”.

**Lemma 5.10.** *For each time  $t$  and set of offline nodes  $S \subseteq [n]$ , we have that*

$$\Pr[E_{S,t}] \leq \prod_{i \in S} \hat{s}_{i,t}.$$

To prove the above, we need the following claim from [20]. For completeness, we provide a short probabilistic proof of this lemma (see [20, Lemma 3.2] for a longer algebraic proof).

**Lemma 5.11.** *For all  $i \in [n]$ , let  $q_i \in [0, 1]$ . Then,*

$$\sum_{A \subseteq S} \Pr[E_{A,t}, F_{S \setminus A,t}] \cdot \prod_{i \in S \setminus A} q_i = \sum_{A \subseteq S} \Pr[E_{A,t}] \cdot \prod_{i \in A} (1 - q_i) \prod_{i \in S \setminus A} q_i.$$

*Proof.* Consider a probability space where at time  $t$  each offline node  $i$  tosses a coin with probability of heads  $q_i$ , independently. The LHS corresponds to all free nodes having their coin come up heads. The RHS corresponds to the same event, i.e., all nodes whose coin came up tails were engaged.  $\square$

*Proof of Lemma 5.10.* The proof is by induction on  $t \geq 1$  for all sets  $S$ . The base case  $t = 1$ , for which  $s_{it} \equiv 0$  for all  $i$ , is trivial. For the inductive step, denote the marginal probability of  $i$  bidding at time  $t$  (requiring  $t$ ’s arrival) by

$$m_{i,t} := \Pr[i \in P_t] = \frac{\hat{x}_{i,t}}{1 - \hat{s}_{i,t}}.$$

We note that with this terminology, we have that

$$\hat{s}_{i,t+1} = \hat{s}_{i,t} + \hat{x}_{i,t} = \hat{s}_{i,t} + m_{i,t} \cdot (1 - \hat{s}_{i,t}) = \hat{s}_{i,t} \cdot (1 - m_{i,t}) + m_{i,t}. \quad (10)$$

The inductive step follows by noting that at most one of the nodes  $i \in S$  may be matched at time  $t$ , and this requires that the single free node  $i \in S$  bid for  $t$ , which happens with probability  $m_{i,t}$

(conditioned on any event determined by randomness up to time  $t$  that implies that the node is free), as follows.

$$\begin{aligned}
\Pr[E_{S,t+1}] &\leq \sum_{\substack{A \subseteq S \\ |S \setminus A| \leq 1}} \Pr[E_{A,t}, F_{S \setminus A,t}] \cdot \prod_{i \in S \setminus A} m_{i,t} \\
&= \sum_{\substack{A \subseteq S \\ |S \setminus A| \leq 1}} \Pr[E_{A,t}] \cdot \prod_{i \in A} (1 - m_{i,t}) \prod_{i \in S \setminus A} m_{i,t} && \text{Lemma 5.11} \\
&\leq \sum_{A \subseteq S} \Pr[E_{A,t}] \cdot \prod_{i \in A} (1 - m_{i,t}) \prod_{i \in S \setminus A} m_{i,t} \\
&\leq \sum_{A \subseteq S} \prod_{i \in A} U_{i,t} \cdot \prod_{i \in A} (1 - m_{i,t}) \prod_{i \in S \setminus A} m_{i,t} && \text{I.H.} \\
&= \prod_{i \in S} (\hat{s}_{i,t} \cdot (1 - m_{i,t}) + m_{i,t}) && \text{binomial expansion} \\
&\leq \prod_{i \in S} \hat{s}_{i,t+1} && \text{Equation (10).} \quad \square
\end{aligned}$$

The preceding lemma allows us to prove the following counterpart to Lemma 3.4, key to our improved approximation ratios.

**Lemma 5.12.** *For any time  $t$  and set of offline nodes  $S \subseteq [n]$ , Algorithm 6 satisfies*

$$\Pr[S \cap P_t \neq \emptyset] \geq 1 - \prod_{B \in \mathcal{B}_t} \left( 1 - \sum_{i \in B \cap S} \hat{x}_{it}/p_t \right).$$

*Proof.* Let  $C_t$  and  $P_t$  be as in Algorithm 6. Since  $P_t = C_t \setminus \{i \mid E_{i,t} = 1\}$ , there are no bids from set  $S$  ( $S \cap P_t = \emptyset$ ) iff all  $i \in A = C_t \cap S$  are no longer free, i.e., iff  $E_{A,t}$  holds. So, by total probability over  $C_t \cap S$ , we have that  $(\star) := \Pr[t \text{ arrives and } S \cap P_t = \emptyset]$  satisfies

$$\begin{aligned}
(\star) &= \sum_{A \subseteq S} \Pr[C_t \cap S = A] \cdot \Pr[E_{A,t}] \\
&= \sum_{A \subseteq S} \prod_{B: |B \cap A| = 0} \Pr[C_t \cap B \cap A = \emptyset] \prod_{B: |B \cap A| = 1} \sum_{i \in B \cap A} \Pr[C_t \cap B \cap A = \{i\}] \cdot \Pr[E_{A,t}] \\
&\leq \sum_{A \subseteq S} \prod_{B: |B \cap A| = 0} \Pr[C_t \cap B \cap A = \emptyset] \prod_{B: |B \cap A| = 1} \sum_{i \in B \cap A} \Pr[C_t \cap B \cap A = \{i\}] \cdot \Pr[E_{i,t}] \\
&= \prod_{B \in \mathcal{B}_t} \left( \Pr[C_t \cap B \cap S = \emptyset] + \sum_{i \in B \cap S} \Pr[C_t \cap B = \{i\}] \cdot \Pr[E_{i,t}] \right) \\
&= \prod_{B \in \mathcal{B}_t} \left( 1 - \sum_{i \in B \cap S} \Pr[C_t \cap B = \{i\}] \cdot \Pr[F_{i,t}] \right) \\
&\leq \prod_{B \in \mathcal{B}} \left( 1 - \sum_{i \in B \cap S} \hat{x}_{i,t}/p_t \right).
\end{aligned}$$

Above, the second and third equality follow from independence of the sets  $\{C_t \cap B \mid B \in \mathcal{B}_t\}$  and from  $|C_t \cap B| \leq 1$  always. The first inequality follows from Lemma 5.10. The fourth equality follows from

$|C_t \cap B| \leq 1$ , and  $F_{i,t} = 1 - E_{i,t}$ . Finally, the last inequality follows from  $\Pr[C_t \cap B = \{i\}] = \frac{\hat{x}_{i,t}}{p_t \cdot (1 - \hat{s}_{i,t})}$  and Observation 5.9, implying that  $\Pr[C_t \cap B = \{i\}] \cdot \Pr[F_{i,t}] \geq \frac{\hat{x}_{i,t}}{p_t \cdot (1 - \hat{s}_{i,t})} \cdot (1 - \hat{s}_{i,t}) \geq \hat{x}_{i,t}/p_t$ .  $\square$

The following lemma, which is a counterpart to Lemma 3.6 for the non-stochastic setting, allows us to obtain a concrete lower bound from Lemma 5.12.

**Lemma 5.13.** *If  $\epsilon, \delta \in [0, 1]$  guarantee that for all  $z \geq \frac{1-\epsilon}{2} = \frac{\epsilon}{2(\epsilon+\delta)}$ ,*

$$f_{\epsilon,\delta}(z) := \exp(-z \cdot (1 + \delta)) - (1 - z \cdot (1 - \epsilon)) \geq 0, \quad (11)$$

*then Algorithm 6 with parameters  $\epsilon$  and  $\delta$  satisfies for each time  $t$  and set  $S \subseteq [n]$*

$$\Pr[S \cap P_t \neq \emptyset] \geq \left(1 - \exp\left(-1 - \delta + \frac{\epsilon + \delta}{1 - \epsilon}\right) \cdot \frac{1 - \epsilon}{1 + \delta}\right) \cdot \sum_{i \in S} x_{i,t}/p_t.$$

The proof is near identical to that of Lemma 3.6, and is therefore omitted.<sup>7</sup>

Finally, the same optimization over  $\epsilon, \delta$  as in Section 3 yields our main result of this section.

**Theorem 5.14.** *Algorithm 6 with parameters  $(\epsilon, \delta) \approx (0.0480, 0.0643)$  is a polytime online stochastic weighted bipartite matching algorithm that 0.652-approximates the optimum online algorithm.*

*Proof.* Let  $\alpha = 1 - \exp\left(-1 - \delta + \frac{\epsilon + \delta}{1 - \epsilon}\right) \cdot \frac{1 - \epsilon}{1 + \delta} \geq 0.652$ . By our algorithm's greedy matching choice,  $t$  gets matched to a node  $i$  with  $w_{i,t} \geq w$  iff  $t$  arrives and gets a bid from the set  $S_{t,w} := \{i \mid w_{i,t} \geq w\}$ . So, by Lemma 5.13, as  $(\epsilon, \delta)$  above satisfy Equation (11) for all  $z \geq 0$  (see the proof of Theorem 3.9),

$$\Pr[w(\mathcal{M}(t), t) \geq w] = \Pr[t \text{ arrives} \wedge P_t \cap S_{t,w} \neq \emptyset] \geq p_t \cdot \alpha \cdot \sum_{i \mid w_{i,t} \geq w} x_{i,t}/p_t = \alpha \cdot \sum_{i \mid w_{i,t} \geq w} x_{i,t}.$$

The approximation ratio of Algorithm 6 then follows from Lemma 5.8, while the algorithm's polynomial running time is immediate.  $\square$

### 5.1.3 A simple direct application: Algorithms with predictions

Our ODRS of Theorem 1.2 finds a direct application in the burgeoning area of *online algorithms with advice* (see the survey [71] and site [1]). Here, an online algorithm is equipped with machine-learned (ML) predictions concerning the input. For example, many works show that for related problems (including bipartite matching), high-quality fractional solutions are efficiently learnable and can guide integral algorithms, provided an effective online rounding scheme is given [64, 65, 66]. We add to this line of work, as follows: By linearity of expectation, our ODRS can round any fractional *edge-weighted* bipartite matching while preserving its value up to a ratio of 0.652. This immediately yields the first online *edge-weighted* bipartite matching algorithm with predictions, getting a competitive ratio of  $(1 - 1/e + \Omega(1))$  with sufficiently good predictions. In contrast, with no predictions, only recently was it shown how to break the barrier of  $1/2$  for this problem (using *free disposal*) [13, 44, 52], and  $1 - 1/e + \Omega(1)$  is impossible to achieve even in unweighted graphs [59].

<sup>7</sup>The only changes to the proof are (1) the syntactic generalization of Fact 3.3 implies that all but one bin  $B \in \mathcal{B}_t$  computed in Line 6 has  $\sum_i \hat{x}_{i,t}/\mathbf{p}_t \geq \frac{1-\epsilon}{2}$ , and (2) the convexity argument requires that we add dummy nodes  $J$  with  $s_{j,t} = \theta$  for all  $j \in J$  and  $\sum_{j \in J} x_{j,t} = p_t - \sum_i x_{i,t}$ , where this sum is non-negative, by LP Constraint 7.

## 5.2 Application of the level-set ODRS

We now turn to discussing applications of our main level-set algorithm, and its strong negative correlation properties.

### 5.2.1 Multi-stage stochastic optimization

We consider an application to multi-stage stochastic optimization, building on the framework of [80]. We recall briefly that in such multi-stage problems, various parameters of an optimization problem materialize stochastically over some number  $k$  of stages, where each stage contains stochastic information about the following stage(s) and wherein we can take actions—with actions cheaper in earlier stages, but possibly inaccurate since we only know the future stochastically.

Motivated by the results of [82], a broad class of multi-stage stochastic covering problems can be cast as the following family of *online* problems: see Section 2 of [80], from which we quote the framework almost verbatim next. There is a hidden covering problem “minimize  $c^T \cdot x$  subject to  $Ax \geq b$  and with all variables in  $x$  being non-negative integers” that is revealed online as follows. We let  $m$  denote the number of rows of  $A$ ; also, the variables in  $x$  are indexed as  $x_{j,\ell}$ , where  $1 \leq j \leq n$  and  $1 \leq \ell \leq k$ . (We interchange  $m$  and  $n$  from [80].) Such a covering problem, as well as a feasible fractional solution  $x^*$  for it (where each  $x_{j,\ell}^*$  is allowed to be a non-negative *real*), are revealed to us in  $k$  stages as follows. In each stage  $\ell$  where  $1 \leq \ell \leq k$ , we are given the  $\ell^{\text{th}}$ -stage fractional values ( $x_{j,\ell}^* : 1 \leq j \leq n$ ) of the variables along with their respective columns in the coefficient matrix  $A$ , and their respective coefficients in the objective-function vector  $c$ . Given this setup, we need to round the variables ( $x_{j,\ell}^* : 1 \leq j \leq n$ ) immediately and irrevocably at stage  $\ell$ , using randomization if necessary. (Note the direct connection to online computation.) The goal is to develop such a rounded vector ( $s_{j,\ell} : 1 \leq j \leq n, 1 \leq \ell \leq k$ ) that satisfies the constraints  $Ay \geq b$ , and whose (expected) approximation ratio  $\mathbb{E}[c^T \cdot y]/c^T \cdot x^*$  is small, where the expectation is only over any random choices made by our online algorithm in this arrival model. This completes the description of the framework from [80].

We make two contributions in this framework. A motivating family of problems to consider is vertex-(multi-)cover on graphs and hypergraphs. The multi-stage stochastic version of the classical vertex-cover problem on a given graph  $G = (V, E)$  is basically as follows: the coverage constraint corresponding to an edge  $e = (u, v) \in E$  is now

$$\left( \sum_{\ell=1}^k x_{u,\ell} \right) + \left( \sum_{\ell=1}^k x_{v,\ell} \right) \geq 1. \quad (12)$$

Improving on the  $2k$ -approximation of Swamy and Shmoys [82] for this problem—in the “online rounding” model mentioned in the previous paragraph—a 2-approximation is developed in [80].

Our *first contribution* is the following generalization of the 2-approximation for multi-stage stochastic vertex cover from [80]. Consider the following more-general problem in  $d$ -regular hypergraphs  $H = (V, E)$  (or hypergraphs with all edge-sizes lower bounded by  $d$ ): for some integer parameter  $t$ , we want the vertices in each edge  $e \in E$  to be covered in total to an extent of at least  $t$ , where vertices can be multi-covered. That is, generalizing (12), the coverage constraint corresponding to an edge  $e = \{v_1, v_2, \dots, v_d\} \in E$  is

$$\sum_{i=1}^d \sum_{\ell=1}^k x_{v_i,\ell} \geq t, \quad (13)$$

where each  $x_{v_i,\ell}$  is allowed to be any non-negative integer; note that the above-seen stochastic vertex cover is the special case where  $d = 2$  and  $t = 1$ . Recalling our online arrival model, we need to

round the LP solution  $(x_{v,\ell}^* : v \in V)$  right away at stage  $\ell$ , for each  $\ell \in [k]$ . We proceed as follows to obtain an  $\alpha := (d + t - 1)/t$ -approximation in this model, generalizing the 2-approximation for  $d = 2$  and  $t = 1$ . In particular, if  $t$  is large, this is essentially a 1-approximation.

Our algorithm works as follows. For each vertex  $v \in V$ , consider the scaled vector  $z(v) := (\alpha \cdot x_{v,\ell}^* : \ell \in [k])$  and independently for all  $v$ , run Algorithm 5 on  $z(v)$  to obtain the final rounded values  $X_{v,\ell} : \ell \in [k]$ . (As usual, we imagine padding this vector with a dummy final element so that the sum of the entries is an integer.) We first verify that (13) is satisfied with probability one for each  $e = \{v_1, v_2, \dots, v_d\} \in E$ :

$$\sum_{i=1}^d \sum_{\ell=1}^k X_{v_i,\ell} \geq \sum_{i=1}^d \left\lfloor \sum_{\ell=1}^k \alpha \cdot x_{v_i,\ell}^* \right\rfloor \quad (\text{by Property (P2)}) \quad (14)$$

$$> \sum_{i=1}^d \left( \left( \sum_{\ell=1}^k \alpha \cdot x_{v_i,\ell}^* \right) - 1 \right) \quad (\lfloor \beta \rfloor > \beta - 1 \text{ for all } \beta) \quad (15)$$

$$\geq \alpha \cdot t - d \quad (\text{since } x^* \text{ satisfies (13)})$$

$$= t - 1,$$

which implies that  $\sum_{i=1}^d \sum_{\ell=1}^k X_{v_i,\ell} \geq t$ , as required, since the LHS here is an integer and because of the strict inequality connecting (14) and (15). We thus obtain an online solution to the rounding problem which satisfies all constraints with probability one, and whose expected objective function value is at most  $\alpha(c^T \cdot x^*)$ . We observe that the work of [80] also runs an online rounding algorithm, but that the problem considered there is much simpler since all we require there (in place of Property (P2)) is that if the entries of the given input vector sum to at least one, then at least one entry is rounded to one with probability one. In particular, this does not address the multi-coverage constraint we satisfy here.

Our *second contribution* is that because of Property (P3), the objective function is strongly concentrated around its mean (given by the Chernoff bound). This is an especially useful property in areas like stochastic optimization, where *the algorithm can be run only once*. This is in comparison to *offline* minimization problems with a non-negative objective where we can repeat the algorithm  $O(1/\epsilon)$  times and choose the best solution obtained, and apply Markov's inequality to each iteration to show that the best solution is at most  $(1 + \epsilon)$  times the expected value with high probability. Thus, the fact that Algorithm 5 yields sharp concentration as implied by Property (P3)—and does not just guarantee Properties (P1) and (P2)—is very useful in the stochastic-optimization context.

Summarizing the preceding discussion (and formalizing the preceding paragraph), we obtain the following result.

**Theorem 5.15.** *For any  $k \geq 1$ , there exists a  $k$ -stage stochastic hypergraph vertex cover algorithm for  $d$ -regular hypergraphs where each hyper-edge must be covered  $t$  times, with approximation ratio that is  $\alpha = \frac{d+t-1}{t}$  in expectation and is  $\alpha(1 + o(1))$  w.h.p. if costs are polynomially bounded and  $OPT = \omega(\log n)$ .*

*Proof.* Our algorithm satisfies the (multi-)coverage constraint, by the above, and has expected approximation ratio  $\alpha$  by construction. The high-probability bound follows from standard Chernoff bounds for negatively-associated variables, relying on Property (P3) and closure of NA under products (Lemma 2.5), relevant since we run independent copies of Algorithm 5 for each vertex.  $\square$

## 5.2.2 Negative association, fairness, diversity, and streaming

While Section 5.2.1 only required the fact that Property (P3) yields sharp tail bounds for sums of the  $X_i$  with non-negative weights, we go further now and use the negative association guaranteed by

Property (P3), which helps us correlate multiple such sums as well as handle submodular objectives well. The submodular-objective application further benefits from the fact that Algorithm 5 can also be viewed as a *data-stream* algorithm: note that at when  $x_t$  arrives, the algorithm need only remember  $s_{t-1}$  and  $S_t$ , and hence can be implemented with  $O(\log n)$  space.

*Intersectionality* is a key notion in the study of fairness, where the intersection of multiple attributes (e.g., race, gender, age) can impact the outcomes of a person in a more fine-grained manner than does standard group-fairness (where we typically take a single attribute). Its study has naturally impacted AI/ML fairness as well—see, e.g., [25] and its followups—and is a topic of much debate in terms of precise definitions (see, e.g., [61]). We make a contribution to this nascent area, while being aware that many such formulations/contributions in the still-early stages of research in AI/ML fairness are speculative.

Consider people arriving online and requesting a resource whose availability expands over time: e.g., vaccines for a new pathogen or a popular new model of car. It is anticipated that by time  $t$ , at most  $a_t$  units of this resource will be available. An ML system that adapts to the changing realities on the ground (e.g., how the pathogen is spreading and which communities seem to be impacted the most) decides online, the probability  $x_t$  with which to allocate the resource to the request arriving at time  $t$ ; we naturally require  $\sum_{i=1}^t x_i \leq a_t$  for all  $t$ . We respond to this in real time by constructing  $X_t \in \{0, 1\}$  using Algorithm 5; Property (P2) ensures that we never exceed the supply-limits  $a_t$ . (In reality, a *batch* of requests will arrive at time  $t$ , which of course Algorithm 5 naturally extends to.) The ML system outputting the vector  $x$  may be (required to be) unaware of protected attributes that characterize our underlying demographic groups and their intersections: given  $x$ , can we show that while intersectional unfairness may be present in  $x$  (in which case the ML system needs to be refined), our rounding algorithm tries to at least *limit the scope* of such intersectional unfairness in some way?

We formulate this problem as follows. Suppose each person requesting the resource has three attributes  $A_1, A_2$ , and  $A_3$ : the “three” here is just for simplicity, and it is easy to see that the framework below generalizes to any number of attributes. Suppose  $U_1, U_2, \dots, U_\alpha$  is some given partition of the population according to their values for  $A_1$  (e.g., if  $A_1$  is age, the  $U_i$ ’s may be disjoint age-intervals). Similarly, suppose we have a partition of the population according to  $A_2$  as  $V_1, V_2, \dots, V_\beta$  and according to  $A_3$  as  $W_1, W_2, \dots, W_\gamma$ . In order to model intersectionality, let us define the subgroup  $G_{i,j,k} := U_i \cap V_j \cap W_k$ . Let  $s_{i,j,k} = \sum_{r \in G_{i,j,k}} X_r$  be the random variable denoting the amount of resource received in total by members of  $G_{i,j,k}$ . We ask: “While intersectional unfairness may be present, say inherently in the vector  $x$ , does our rounding at least limit the extent of it?” That this is indeed true follows from Property (P3) which guarantees negative association, as follows. For any sequence of thresholds  $(t_{i,j,k})$ , the probabilities of different subgroups  $G_{i,j,k}$  receiving intersectional unfairness in the sense of  $s_{i,j,k} \leq t_{i,j,k}$ , can be bounded as well as if the  $X_r$ ’s had been independent:

- $\forall S \subseteq ([\alpha] \times [\beta] \times [\gamma])$ ,  $\Pr[\bigwedge_{(i,j,k) \in S} (s_{i,j,k} \leq t_{i,j,k})] \leq \prod_{(i,j,k) \in S} \Pr[s_{i,j,k} \leq t_{i,j,k}]$ ; and, more generally,
- in disease-spread and meme-spread-like contexts, we are often interested in monotone non-linear functions that model phase transitions: e.g., if a heavily-interacting subgroup receives fewer than a threshold of vaccines, an explosive epidemic could happen within that subgroup—with monotone behavior away from this threshold. Given any sequence of monotone functions (all increasing or all decreasing)  $(f_{i,j,k})$ , we have

$$\forall S \subseteq ([\alpha] \times [\beta] \times [\gamma]), \mathbb{E} \left[ \prod_{(i,j,k) \in S} f_{i,j,k}(s_{i,j,k}) \right] \leq \prod_{(i,j,k) \in S} \mathbb{E} [f_{i,j,k}(s_{i,j,k})].$$



As is well-known, letting  $\chi(\cdot)$  be the indicator function, this implies that  $\sum_{(i,j,k) \in S} \chi(s_{i,j,k} \leq t_{i,j,k})$  has Chernoff-like concentration around its mean; in particular, while some subgroups in  $S$  may face intersectional unfairness (that needs to be separately analyzed by inspecting  $x$ ), it is unlikely that many do.

We next present an application to diversity in search results. In addition to the classical interest in this problem in information retrieval [18], there has been much recent work in search-result diversification, motivated, e.g., by the fact that the document-set retrieved should account for the interests of the user population [26, 30]. By developing a model of knowledge about the topics the query or the documents may refer to, the work of [4] presents an  $(1 - 1/e)$ -approximation for their objective of maximizing average-user satisfaction with the search results. Given a search query  $q$ , a corpus of  $N$  documents  $\mathcal{D}$ , and a bound  $k$  on the number of documents to retrieve from  $\mathcal{D}$  for  $q$ , a monotone submodular function  $f : 2^{\mathcal{D}} \rightarrow [0, 1]$  is developed in [4], where  $f(T)$  is the (approximate) mix of relevance and diversity if the set  $T$  with  $|T| \leq k$  is then returned to the user. The problem is thus to approximately maximize  $f(T)$  subject to  $|T| \leq k$ . Consider the setting where  $\mathcal{D}$  is very large, we only get streaming access to it, and where, after encountering document  $d \in \mathcal{D}$ , we get ML advice on the probability  $x_d$  with which to choose  $d$ . Algorithm 5 is tailor-made for this, as it needs only  $O(\log N)$  space. Furthermore, by the already-known Theorem A.4, we obtain that our final expected value  $\mathbb{E}[f(X_1, X_2, \dots, X_N)]$  is at least as large as if we had instead chosen the  $X_i$ 's independently with marginals  $x_i$ . Moreover, by Theorem A.5, the submodular objective value obtained exhibits strong lower tail bounds (i.e., is unlikely to fall much below its expectation).

## 6 Lower Bounds

In this section we present some impossibility results for ODRSes.

A simple example in [34] rules out a rounding ratio of one, or even just greater than  $\frac{7}{8} = 0.875$ . In this section we improve this bound to  $(2\sqrt{2} - 2) \approx 0.828$  with a similarly simple example.

Before providing our lower bound, we provide an even simpler example ruling out a rounding ratio of one. This example highlights the need of ODRSes with high rounding ratio to create some form of negative correlation between all subsets of offline nodes, motivating our lower bound example. (We are also hopeful that this insight may inform future ODRSes.)

**Example.** Suppose an ODRS  $\mathcal{A}$  with rounding ratio of one exists, and apply it to the following input on  $n = 3$  offline nodes and 4 online nodes: For  $k = 1, 2, 3$ , online node  $k$  only neighbors offline node  $k$ , and  $x_{kk} = 1/2$ . So, each offline node's matched status so far is a  $\text{Ber}(1/2)$  random variable. Finally, online node  $t = 4$  arrives, neighboring some two offline nodes,  $i, j$ , and  $x_{it} = x_{jt} = 1/2$ . The above is clearly a feasible fractional matching, but since we assumed that  $\mathcal{A}$  has a rounding ratio of one, it must match  $t$  with probability one. Since  $t$  cannot be matched if both  $i$  and  $j$  are previously matched, and both are matched with probability  $1/2$  before time  $t$ , this requires  $i$ 's and  $j$ 's matched status before time  $t$  to be perfectly negatively correlated. That is,  $i$  is unmatched iff  $j$  is matched, and vice versa. Since  $i, j$  could be any two offline nodes, this requires perfect negative correlation between any two of these three variables, which is impossible.<sup>8</sup> Thus, we reach a contradiction, and an ODRS with rounding ratio of one is impossible.

To generalize the above and obtain a concrete numeric bound, we need the following fact, whereby (near-)positive pairwise correlation is unavoidable in a large set of Bernoulli random variables. (See Lemma 6.3 for a potentially useful generalization to  $k$ -wise correlations.)

<sup>8</sup>Perfect negative pairwise correlation between three binary variables  $X_1, X_2, X_3 \in \{0, 1\}$  can be stated succinctly as  $X_1 + X_2 = X_1 + X_3 = X_2 + X_3 = 1$ , but this linear system only has a *fractional* solution  $X_1 = X_2 = X_3 = 1/2$ .

**Fact 6.1.** Let  $Y_i \sim \text{Ber}(p)$  for  $i = 1, \dots, n$  be (possibly dependent) Bernoulli variables. Then,

$$\max_{i,j} \text{Cov}(Y_i, Y_j) \geq -2p/(n-1).$$

*Proof.* This bound follows from non-negativity of variance and the following, after rearranging.

$$0 \leq \text{Var} \left( \sum_i Y_i \right) = \sum_i \text{Var}(Y_i) + \sum_{i,j} \text{Cov}(Y_i, Y_j) \leq n \cdot p + \binom{n}{2} \max_{i,j} \text{Cov}(Y_i, Y_j). \quad \square$$

**Lemma 6.2.** Any online rounding algorithm for bipartite fractional matchings has some fractional matching  $\mathbf{x} = \frac{1}{2} \cdot \mathbf{1}^E$  on graph  $G = (V, E)$  such that for some edge  $e \in E$ , the output matching  $\mathcal{M}$  satisfies

$$\Pr[e \in \mathcal{M}] \leq (2\sqrt{2} - 2) \cdot x_e \approx 0.828 \cdot x_e.$$

*Proof.* Consider an input with  $x_e = 1/2$  for each edge, with the first  $n$  online nodes neighboring two distinct offline nodes, with one last online node neighboring two (judiciously chosen) offline nodes, from two prior online nodes' neighborhoods. We wish to show that for some instantiation of this family of  $\mathbf{x}$  and  $G$ , some online node (and hence some edge) is matched with low probability. Specifically, letting  $p = 2\sqrt{2} - 2$ , we wish to show that for some such  $\mathbf{x}$  and  $G$ ,

$$\exists t \in [n+1] \text{ such that } \Pr[t \in V(\mathcal{M})] \leq p + \frac{p}{2(n-1)}, \quad (16)$$

Since each online node  $t$  has  $\sum_i x_{it} = 1$ , this implies that some edge  $e$  is matched with probability at most  $(2\sqrt{2} - 2 + \frac{p}{2(n-1)}) \cdot x_e$ , by linearity of expectation. Taking  $n$  to be sufficiently large rules out any  $(2\sqrt{2} - 2 + \epsilon)$ -approximate rounding. We turn to prove Equation (16).

If any of the first online nodes is matched with probability less than  $p$ , we are done. Otherwise, let  $Y_t = \mathbb{1}[t \in V(\mathcal{M})]$  be indicators for online node  $t$  being matched. We couple these with variables  $Z_t \geq \text{Ber}(p)$  such that  $Y_t \geq Z_t$  always. By Fact 6.1, there exist two online nodes  $t \neq t'$  with  $\text{Cov}(Z_t, Z_{t'}) \geq -2p/(n-1)$ . Let  $N(t) = \{i_1, i_2\}$  and  $N(t') = \{i_3, i_4\}$  be the neighborhoods of  $t$  and  $t'$ , respectively. By a simple averaging argument, some pair of nodes  $(i, j) \in N(t) \times N(t')$  are both matched before the online last arrival with probability at least

$$\frac{\Pr[Y_t, Y_{t'}]}{4} \geq \frac{\Pr[Z_t, Z_{t'}]}{4} \geq \frac{\Pr[Z_t] \cdot \Pr[Z_{t'}] - 2p/(n-1)}{4} \geq \frac{p^2 - 2p/(n-1)}{4}.$$

So, if the last online node neighbors  $\{i, j\}$ , it is matched with probability at most  $1 - \frac{p^2 - 2p/(n-1)}{4}$ . However, as  $p = 2\sqrt{2} - 2$  is a root of  $1 - y - \frac{y^2}{4}$ , this probability is precisely  $1 - \frac{p^2}{4} + \frac{p}{2(n-1)} = p + \frac{p}{2(n-1)}$ , as desired.  $\square$

**Inevitable near-positive cylinder dependence.** A useful extension of Fact 6.1 of possible independent interest argues that for any (sufficiently large) subset of Bernoulli variables, some subset of these variables must be (nearly) positively cylinder dependent. In Appendix D we prove this Ramsey-theoretic lemma.

**Lemma 6.3.** Let  $r \in \mathbb{N}$  and  $0 \leq \epsilon \leq p \leq 1$ . Then, there exists some  $n = n_r(p, \epsilon)$  such that any Bernoulli variables  $Y_1, \dots, Y_n \sim \text{Ber}(p)$  contains some subset  $I \subseteq [n]$  of size  $|I| = 2^r$  such that

$$\mathbb{E} \left[ \prod_i Y_i \right] \geq \prod_i \mathbb{E}[Y_i] - \epsilon.$$

## 7 Summary and Open Questions

In this work we provided a methodical study of online dependent rounding schemes (ODRSes) for bipartite  $b$ -matching. We obtained optimal rounding ratios with strong concentration for star graphs (uniform-matroid rounding) and improved bounds beyond a ratio of  $1 - \frac{1}{e}$  for  $(b)$ -matchings, including the first results for  $b$ -matchings not obtained via OCRSes. (Consequently, ours is the only ratio beyond  $1/2$  known for this problem.) We furthermore provided a number of applications of our ODRSes and their algorithmic approach. Beyond the natural question of improving our rounding ratios for  $b$ -matchings (and the approximation/competitive ratios of their derived applications), our work raises a number of directions for future research.

**Online applications.** We provided a number of online applications of our ODRSes. What further applications do these schemes have? The bipartite setting seems particularly relevant for online scheduling problems, with machines and jobs represented by offline and online nodes, respectively. For minimization problems, it may prove useful to observe that our ODRSes for  $b$ -matching have modest two-sided error: edges  $(i, t)$  are matched with probability at most the probability that  $i$  bids for  $t$ , namely  $\hat{x}_{i,t} \leq (1 + \delta)x_{i,t} \approx 1.064 \cdot x_{i,t}$ . Moreover, these latter events satisfy strong upper-tail bounds, by our ODRS' strong negative correlation properties (see Property (P3) and Lemma B.4).

**Streaming applications.** We note that our ODRSes are not only online algorithms, but moreover are streaming (for uniform matroids) or semi-streaming algorithms (for arbitrary  $b$ -matchings). Indeed, they require us to only remember  $O(\log n)$ -bit partial sums and a single number per offline node. Does this property have further applications in streaming or graph-analytics contexts?

**Beyond matchings.** Finally, we recall that oblivious online contention resolution schemes (OCRSes) yield ODRSes, though this connection does not yield optimal rounding ratios. Recent years have seen an explosion of work on OCRSes for increasingly-rich families of arrival models and combinatorial constraints. These have been fueled in large part by connections to various other online and economic applications, most notably the prophet-inequality problem under more involved combinatorial constraints (see discussion in the influential work of Kleinberg and Weinberg [60]). Can a similarly rich theory of online dependent rounding more broadly be developed, capturing other combinatorial constraints beyond matchings?

### Acknowledgements.

We thank Uri Feige for pointing out the connection of our work to offline CRSes, and thank Sahil Singla for discussions about (lack of prior) examples of black-box application of offline CRSes to online problems. We thank Manish Purohit for asking about the two-sided error of our ODRSes, which we anticipate may be of use in future applications for online minimization problems. Finally, we thank the anonymous reviewers for comments on presentation.

## APPENDIX

### A Additional Preliminaries

In this section we provide omitted proofs and additional preliminaries not covered in Section 2.

#### A.1 Contention Resolution Schemes

We start with a brief, self-contained proof of Lemma 2.1, restated below for ease of reference.

**Lemma 2.1.** Let  $\mathcal{D} : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$  be a distribution over subsets of  $[n]$ , and denote this distribution's support by  $\text{supp}(\mathcal{D}) := \{S \subseteq [n] \mid \Pr_{R \sim \mathcal{D}}[R = S] \neq 0\}$ . Then, there exists a randomized algorithm  $\text{CRS}(R, \mathbf{v})$  which on input set  $R \sim \mathcal{D}$  and vector  $\mathbf{v} \in \mathbb{R}^n$ , outputs a subset  $O \subseteq R$  of size  $|O| \leq 1$  satisfying

$$\Pr[i \in O] = v_i \cdot \min_{\substack{S \subseteq [n] \\ \sum_{i \in S} v_i \neq 0}} \frac{\Pr[R \cap S \neq \emptyset]}{\sum_{i \in S} v_i} \quad \forall i \in [n].$$

The algorithm runs in time polynomial in  $n$  if  $\mathcal{D}$  is a product distribution. Otherwise, it runs in time  $\text{poly}(n, T)$ , where  $T \geq |\text{supp}(\mathcal{D})|$  is the time to compute and write down the support of  $\mathcal{D}$ .

*Proof.* The lemma for product distributions follows from the work of Feige and Vondrák [45]. We focus on the more general case, addressed by Bansal and Cohen [7], but proven here for completeness.

Let  $\alpha := \min_{S \subseteq [n]} \frac{\Pr[R \cap S \neq \emptyset]}{\sum_{i \in S} v_i}$ . We prove the existence of conditional probabilities  $p_{i,S}$  with  $p_{i,S} = 0$  if  $i \notin S$  and  $\sum_i p_{i,S} = 1$  for all  $S$  such that setting  $\Pr[O = \{i\} \mid R = S] = p_{i,S}$  yields  $\Pr[O = \{i\}] = \sum_S \Pr[O = \{i\} \mid R = S] \cdot \Pr[R = S] \geq \alpha \cdot v_i$ . We prove the existence of these probabilities using the max-flow/min-cut theorem applied to the following directed flow network that has a source vertex  $\text{src}$ , a sink vertex  $\text{sink}$ , and two other disjoint sets of vertices  $A$  and  $B$ . Vertices on side  $A$  correspond to subsets of elements in  $[n]$ , with an edge of capacity  $\Pr[R = S]$  from  $\text{src}$  to the node in  $A$  representing set  $S$ . Vertices on side  $B$  correspond to elements  $i \in [n]$ , with an edge of capacity  $\alpha \cdot v_i$  leaving this vertex to  $\text{sink}$ . Infinite-capacity directed edges go from each vertex in  $A$  corresponding to some set  $S$ , to vertices in  $B$  corresponding to each  $i \in S$ . Each vertex  $i \in S$  in  $B$ , has a directed edge to  $\text{sink}$  with capacity  $\alpha \cdot v_i$ . This network is reminiscent of the network used to prove Hall's Theorem via the max-flow/min-cut theorem. Indeed,  $\min_{S \subseteq [n]} \frac{\Pr[R \cap S \neq \emptyset]}{\sum_{i \in S} v_i} \geq \alpha$  is a (scaled) version of Hall's condition, and implies by the same argument the existence of a "perfect" flow  $f$  of value  $\alpha \cdot \sum_i v_i$ , implying that each edge of the form  $(i, \text{sink})$  has its capacity  $\alpha \cdot v_i$  saturated by this flow. This in turn implies the existence of the probabilities  $p_{i,S}$  as desired, where if the flows through the edges  $(\text{src}, S)$  and  $(S, i)$  are  $f_{\text{src}, S}$  and  $f_{S, i}$  respectively, then  $p_{i,S} = f_{S, i} / \Pr[R = S]$  (if  $\Pr[R = S] = 0$ , we take arbitrary non-negative  $p_{i,S}$  with  $p_{i,S} = 0$  if  $i \notin S$  and  $\sum_{i \in S} p_{i,S} = 1$ ). Thus, the above algorithm outputs a set  $O \subseteq R$  of size at most one with each element belonging to  $O$  with the requisite probability. The running time of the algorithm for general distributions follows by polytime solubility of the maximum flow problem.  $\square$

## A.2 Negative correlation properties

Here we formalize the strong notion of negative correlation that our online level-set algorithm enjoys.

**Definition A.1.** A distribution  $\mu : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$  is strongly Rayleigh if its generating polynomial,

$$g_\mu(z) = \sum_{S \subseteq [n]} \mu(S) \prod_{i \in S} z_i,$$

is real stable, i.e., it has no root  $z \in \mathbb{C}^n$  satisfying  $\Im(z_i) > 0$  for all  $i \in [n]$ .

As stated in Section 2 and proven in [17], the strong Rayleigh property implies NA.

**Lemma A.2.** If  $(X_1, \dots, X_n)$  are strongly Rayleigh binary r.v.s, then they are NA.

Moreover, since SRP is closed under conditioning [17], we find that SRP distributions are NA even after conditioning.

The strong Rayleigh property, central to the area of geometry of polynomials, has been highly influential in recent years. To the best of our knowledge, aside from the classic balls-and-bins process, its variants and conditional counterparts (see [19]), our online level-set algorithm is the only online algorithm known to satisfy such a strong negative-correlation property. We believe that this property of our algorithm will find further applications outside of this work.

**Submodular dominance.** As stated in Section 2, negative association implies stochastic dominance in the submodular order. To formally define the above, we briefly recall the definition of submodular functions, which are set functions capturing the notion of diminishing returns.

**Definition A.3.** A set function  $f : 2^{[n]} \rightarrow \mathbb{R}$  is submodular if for all sets  $A \subseteq B \subseteq [n]$  and element  $x \in [n] \setminus B$ , we have that

$$f(A \cup \{x\}) - f(A) \geq f(B \cup \{x\}) - f(B).$$

The following submodular dominance result of Christofides and Vaggelatos [29], later generalized by Singla and Qiu [77] implies that our online level-set algorithm not only preserves weighted objectives losslessly, but also preserves submodular objectives. (See Appendix C.)

**Theorem A.4.** Let  $X_1, \dots, X_n$  be binary NA random variables and let  $X_1^*, \dots, X_n^*$  be independent random variables that are place-wise equal to the  $X_1, \dots, X_n$  respectively in distribution. Then, for any monotone submodular function  $f$ ,

$$\mathbb{E}[f(X_1, \dots, X_n)] \geq \mathbb{E}[f(X_1^*, \dots, X_n^*)].$$

Moreover, the following result of Duppala et al. [40] even implies concentration of the above.

**Theorem A.5.** Let  $X_1, \dots, X_n$  be binary NA random variables and let  $X_1^*, \dots, X_n^*$  be independent random variables that are place-wise equal to the  $X_1, \dots, X_n$  respectively in distribution. Then, for any monotone submodular function  $f$ , with  $\mu := \mathbb{E}[f(X_1^*, \dots, X_n^*)]$  and real value  $\delta > 0$ ,

$$\Pr[f(X_1, \dots, X_n) \leq (1 - \delta) \cdot \mu] \leq \exp\left(-\frac{\delta^2 \mu}{2}\right).$$

### A.3 Odds and Ends

We will also make use of the following direct corollary of convexity.

**Claim A.6.** Let  $f$  be a non-negative concave function. Then, for any  $0 \leq t \leq \alpha$ ,

$$\frac{f(t)}{t} \geq \frac{f(\alpha)}{\alpha}.$$

*Proof.* Writing  $t$  as a convex combination of  $\alpha$  and 0, the claim follows from convexity, as follows.

$$f(t) = f\left(\frac{t}{\alpha} \cdot \alpha + \left(1 - \frac{t}{\alpha}\right) \cdot 0\right) \geq \frac{t}{\alpha} \cdot f(\alpha) + \left(1 - \frac{t}{\alpha}\right) \cdot f(0) \geq \frac{t}{\alpha} \cdot f(\alpha). \quad \square$$

## B Deferred Proofs of Section 3

In this section we provide deferred proofs of claims from Section 3, restated below for ease of reference.

**Claim 3.7.** *If  $\epsilon, \delta \in [0, 1]$ , the function  $g(y) := 1 - \exp(-(1-y) \cdot (1+\delta)) \cdot (1-y \cdot (1-\epsilon))$  satisfies*

$$\forall y \in \mathbb{R} : \quad g(y) \geq g\left(\frac{\epsilon + \delta}{(1-\epsilon)(1+\delta)}\right) = 1 - \exp\left(-1 - \delta + \frac{\epsilon + \delta}{1-\epsilon}\right) \cdot \frac{1-\epsilon}{1+\delta}.$$

*Proof.* Taking the derivative of  $g(y)$ , we get

$$g'(y) = \exp(-(1-y) \cdot (1+\delta)) \cdot ((1-\epsilon) - (1-y \cdot (1-\epsilon)) \cdot (1+\delta)),$$

which vanishes at  $y^* = \frac{\epsilon + \delta}{(1-\epsilon)(1+\delta)}$ . Next, we consider the second derivative of  $g$ .

$$g''(y) = g'(y) \cdot (1+\delta) + \exp(-(1-y) \cdot (1+\delta)) \cdot (1-\epsilon) \cdot (1+\delta).$$

The first summand,  $g'(y) \cdot (1+\delta)$ , vanishes at  $y^*$ , by the above, while the second summand is positive, since  $\epsilon \leq 1$  and  $\delta \geq 0$ . We conclude that  $y^*$  is the global minimum of  $g$ .  $\square$

**Lemma 3.8.** *Let  $f_{\epsilon, \delta}(z)$  be as in Lemma 3.6. If  $f_{\epsilon, \delta}(a) \geq 0$  and  $f'_{\epsilon, \delta}(a) \geq 0$ , then  $f_{\epsilon, \delta}(b) \geq 0 \forall b \geq a$ .*

*Proof.* Fix  $\epsilon, \delta$ . For notational simplicity, let  $f = f_{\epsilon, \delta}$  and let  $z^* = z_{\epsilon, \delta}^*$ . Since  $f$  is the sum of twice-differentiable convex functions (namely an exponential and a linear term), we have that  $f''(z) \geq 0$  for all  $z$ . Therefore, if  $f'(z^*) \geq 0$ , then for all  $z \geq z^*$  we have that

$$f'(z) = f'(z^*) + \int_{z^*}^z f''(x) dx \geq f'(z^*) \geq 0.$$

Thus, if moreover  $f(z^*) \geq 0$ , then the desired statement follows, i.e., for all  $x \geq z^*$  we have that

$$f(z) = f(z^*) + \int_{z^*}^z f'(x) dx \geq f(z^*) \geq 0. \quad \square$$

## B.1 Polytime implementation

So far, we ignored computational aspects of our ODRS of Theorem 3.9. In this section we show how our grouping method used to increase our rounding ratio beyond  $1 - \frac{1}{e}$  moreover allows us to compute  $\Pr[P_t = S]$  for all sets  $S \subseteq [n]$  in polynomial time — and thus obtain a polytime ODRS — while only decreasing our rounding ratio by an arbitrarily small constant  $\gamma > 0$ .

**Theorem 3.10.** *For any  $\gamma > 0$ , there exists an  $n^{O(1/\gamma)}$ -time bipartite matching ODRS with rounding ratio of  $0.652 - \gamma$ .*

*Proof.* First, we scale down all the fractions  $x_{i,t}$  by a  $(1-\gamma)$  factor (this results in a valid fractional matching). As this additional downscaling is linear, this results in each offline node having fractional degree  $\hat{s}_{i,t} \leq 1-\gamma$  in the assignment  $\hat{x}$ , by Fact 3.5. Consequently, by the same argument as Fact 3.3, each bin but at most one in  $B \in \mathcal{B}_t$  has  $\sum_{i \in B} \hat{x}_{i,t} \leq \frac{\gamma}{2}$ . But since  $\sum_i \hat{x}_{i,t} \leq \sum_i x_{i,t} \cdot (1+\delta) \leq (1+\delta)$ , this implies that each bin but one has total  $x$ -value at least  $\sum_{i \in B} x_{i,t} \geq \gamma/2(1+\delta)$ . Consequently, by the fractional matching constraint, whereby  $\sum_i x_{i,t} \leq 1$ , there are at most  $2(1+\delta)/\gamma + 1 = O(1/\gamma)$  such non-empty bins.

The benefit of few bins is that now at each time  $t$  the set of bidders is a set of at most  $O(1/\gamma)$  offline nodes, and so there are at most  $n^{O(1/\epsilon)}$  possible sets  $S \subseteq [n]$  in the support of the distribution over  $P_t$ . Moreover, for each such set  $S \subseteq [n]$ , it is straightforward to compute the probability that  $S$  is the set of first-time proposers at time  $t$  in time  $n^{O(1/\gamma)}$ . (Briefly, for each subset  $S$ , we guess for each of the  $O(1/\gamma)$  nodes in  $S$  at what time they made their first bid. The probability of each of these



$n^{O(1/\gamma)}$  guessed events is then easy to compute in polynomial time, by computing the probability that these nodes did bid in these guessed time steps and no other times.) All in all, this allows to compute, exactly, the support of the distribution of  $P_t$ , in time  $n^{O(1/\gamma)}$ . Therefore, by Lemma 2.1, the CRS invocations take time  $\text{poly}(n^{O(1/\gamma)} = n^{O(1/\gamma)})$  per time step. On the other hand, having scaled down the values  $x_{i,t}$  by a factor of  $(1 - \gamma)$  trivially incurs a multiplicative loss of  $(1 - \gamma)$  in the rounding ratio, and so the obtained ODRS has rounding ratio of  $0.652(1 - \gamma) \geq 0.652 - \gamma$ .  $\square$

## B.2 The $b$ -matching ODRS

In this section we extend our ODRS from matchings to  $b$ -matchings. To avoid repetition, we only outline the changes needed in the algorithm and analysis, and defer the proofs to Appendix B.3.

**The change to the core algorithm.** As we did (implicitly) in Algorithm 1, we let the online level-set algorithm dictate the marginal probabilities with which an offline node  $i$  should bid to  $t$ , based on the number of times  $i$  has bid by time  $t$ , which we denote by  $S_{i,t}$ . (This corresponds to  $S_t$  in Algorithm 5 run from the perspective of the offline node  $i$ .) We process offline nodes at different times  $t$  as follows, with the choice of probabilities guided by Algorithm 5 to guarantee that  $\mathbb{E}[S_{i,t+1} - S_{i,t}] = x_{i,t}$ , by Property (P1), and  $S_{i,t} \in [\lfloor s_{i,t} \rfloor, \lceil s_{i,t} \rceil]$ , by Property (P2).

For nodes with  $\lceil s_{i,t+1} \rceil = \lceil s_{i,t} \rceil$  — whose fractional degrees both before and after time  $t$  lie in the range  $[\lfloor s_{i,t} \rfloor, \lceil s_{i,t} \rceil]$  — we group these offline nodes and have them bid (at most one bid per bin) as in Algorithm 1, with the following changes:  $s_{i,t}$  is replaced by its fractional part,  $s_{i,t} - \lfloor s_{i,t} \rfloor$ : this is done both in the bidding probability, so these items have size  $\frac{x_{i,t}}{1 - (s_{i,t} - \lfloor s_{i,t} \rfloor)}$  when bin packing, and in the condition  $s_{i,t} \leq \theta$ , which are now replaced by the condition  $s_{i,t} - \lfloor s_{i,t} \rfloor \leq \theta$  (and similarly for the condition  $s_{i,t} > \theta$ ), with  $\theta$  to be chosen later. Moreover, we add a candidate offline node  $i \in C_t$  to the list of potential matches  $P_t$  (the bidders) if  $S_{i,t} < \lfloor s_{i,t} \rfloor$  (corresponding to  $i \in F \cap C_t$  in Algorithm 1 for simple matchings).

For all offline nodes with  $\lfloor s_{i,t+1} \rfloor = \lfloor s_{i,t} \rfloor + 1$ , we place these nodes in singleton bins, and we have them bid with probability if  $S_{i,t} = \lfloor s_{i,t} \rfloor$ , or with probability  $\frac{s_{i,t+1} - \lfloor s_{i,t+1} \rfloor}{s_{i,t} - \lfloor s_{i,t} \rfloor}$  if  $S_{i,t} = \lceil s_{i,t} \rceil = \lfloor s_{i,t+1} \rfloor$ .

**Change to core ODRS' analysis:** Fact 3.3 and its proof hold unchanged for our extension of the core ODRS. Lemma 3.4, too, holds unchanged, although the proof, deferred to Appendix B.3, is now slightly more involved. Specifically, the key ingredient, proving that the events  $[P_t \cap S \cap B]$  over different bins  $B$  are negatively associated (NA) requires a proof that the variables  $S_{i,t}$  are NA.

**Change to the improved ODRS.** In our ODRS for  $b$ -matchings, since offline nodes  $i$  for which  $\lfloor s_{i,t+1} \rfloor = \lfloor s_{i,t} \rfloor + 1$  are grouped into singleton bins, we wish to apply a markup to these nodes'  $x_{i,t}$ -values, if  $x_{i,t}$  is small. We therefore consider the following assignment, with  $0 \leq \theta_1 \leq \theta_2 \leq 1$  satisfying  $\theta_2 - \theta_1 = \frac{\delta}{\epsilon + \delta}$ , to be chosen shortly.

$$\hat{x}_{i,t} = \int_{z=s_{i,t}}^{s_{i,t+1}} (1 - \epsilon) \cdot \mathbb{1}[[z] \in [\theta_1, \theta_2]] + (1 + \delta) \cdot \mathbb{1}[[z] \notin [\theta_1, \theta_2]] dz. \quad (17)$$

In words: this assignment decreases the part of  $x_{i,t}$  corresponding to the parts of  $i$ 's fractional degree that are bounded away from 0 and 1 (specifically, those in the range  $[\theta_1, \theta_2]$ ), and increases the part of the fractional degree that is somewhat close to 0 or 1, in the ranges  $[0, \theta_1]$  and  $[\theta_2, 1]$ . Our new ODRS then runs the modified core ODRS with inputs  $\theta := \theta_2 \cdot (1 - \epsilon) + \theta_1 \cdot (\delta + \epsilon)$  and  $\hat{\mathbf{x}}, \mathbf{x}$ . The choice of the parameter  $\theta$  is taken to have  $\hat{s}_{i,t} - \lfloor \hat{s}_{i,t} \rfloor \leq \theta$  if and only if  $s_{i,t} - \lfloor s_{i,t} \rfloor \leq \theta$ .

**Change to the improved ODRS' analysis.** First, Fact 3.5 generalizes as follows, arguing that modified fractional degrees  $\hat{s}_{i,t} = \sum_{t' < t} \hat{x}_{i,t'}$  fall within the same integer range for all  $(i, t)$ .

**Fact B.1.** For every offline node  $i \in [n]$  and time  $t$ , we have that  $\lfloor \hat{s}_{i,t} \rfloor = \lfloor s_{i,t} \rfloor$  and  $\lceil \hat{s}_{i,t} \rceil = \lceil s_{i,t} \rceil$ .

Fact B.1 and Property (P2) of Algorithm 5 imply that offline node  $i$  makes some  $S_{i,t} \in [\lfloor s_{i,t} \rfloor, \lceil s_{i,t} \rceil]$  many bids by time  $t$ . As  $i$  is not matched more time than it bids, this implies that the extended ODRS outputs a valid  $b$ -matching.

Next, the key lemma in the analysis of Algorithm 2, namely Lemma 3.6, needs to be extended slightly, and in particular the condition of Equation (3) needs to hold for all  $z \geq \max \left\{ \frac{1-\theta_2}{2(1+\delta)}, \theta_1 \right\}$ . To make this constraint as less restrictive as possible, we take  $\theta_1$  and  $\theta_2$  to guarantee the needed  $\theta_2 - \theta_1 = \frac{\delta}{\epsilon+\delta}$ , and also satisfy  $\frac{1-\theta_2}{2(1+\delta)} = \theta_1$ , resulting in  $\theta_1 = \frac{\epsilon}{(3+2\delta)(\epsilon+\delta)}$  and  $\theta_2 = \frac{\epsilon+3\delta+2\delta^2}{(3+2\delta)(\epsilon+\delta)}$ . This modification of the range for which we require Condition (3) to hold will again allow us to argue that for all bins  $B$  except for at most one bin  $B^*$  (to be characterized shortly), we have the following:

$$1 - \sum_{i \in B} \hat{x}_{i,t} \leq \exp \left( - \sum_{i \in B} x_{i,t}(1+\delta) \right) \quad (18)$$

Specifically, for bins containing a single offline node  $i$  with  $\lfloor s_{i,t+1} \rfloor = \lfloor s_{i,t} \rfloor + 1$ , we have the following: If  $s_{i,t+1} - \lfloor s_{i,t+1} \rfloor \geq \theta_1$ , then

$$x_{i,t} = s_{i,t+1} - s_{i,t} \geq s_{i,t+1} - \lfloor s_{i,t+1} \rfloor \geq \theta_1.$$

Similarly, if  $s_{i,t} - \lfloor s_{i,t+1} \rfloor \leq \theta_2$ , then

$$x_{i,t} = s_{i,t+1} - s_{i,t} \geq \lfloor s_{i,t+1} \rfloor - s_{i,t} = 1 + \lfloor s_{i,t} \rfloor - s_{i,t} \geq 1 - \theta_2 \geq \frac{1 - \theta_2}{2(1+\delta)}.$$

In both cases, Condition (3) holding for all  $z \geq \max \left\{ \frac{1-\theta_2}{2(1+\delta)}, \theta_1 \right\}$  implies Equation (18). In the alternative scenario where  $s_{i,t+1} - \lfloor s_{i,t+1} \rfloor \leq \theta_1$  and  $s_{i,t} - \lfloor s_{i,t} \rfloor \leq \theta_2$ , we have that  $\hat{x}_{i,t} = (1+\delta) \cdot x_{i,t}$ , and Equation (18) follows from the basic inequality  $1 - z \leq \exp(-z)$ . Finally, by Fact 3.3, for all bins  $B$  containing offline nodes  $i$  with  $\lfloor s_{i,t} \rfloor = \lfloor s_{i,t+1} \rfloor$  and  $\hat{s}_{i,t} \leq \theta$  (i.e.,  $s_{i,t} \leq \theta_2$ ), except for at most one bin  $B^*$ , we have that

$$\sum_{i \in B} x_{i,t} \cdot (1+\delta) \geq \sum_{i \in B} \hat{x}_{i,t} \geq \frac{1-\theta}{2} \geq \frac{1-\theta_2}{2},$$

where the last inequality, whereby  $\theta \leq \theta_2$ , is proven similarly to Fact B.1. So, Condition (3) holding for all  $z \geq \frac{1-\theta_2}{2(1+\delta)}$  implies Equation (18) for all remaining bins other than  $B^*$ . The analysis then proceeds as in Lemma 3.6, yielding the following.

**Lemma B.2.** If  $\epsilon, \delta \in [0, 1]$  guarantee that for all  $z \geq \max \left\{ \frac{1-\theta_2}{2}, \theta_1 \right\} = \frac{\epsilon}{(3+2\delta)(\epsilon+\delta)}$ , the expression

$$f_{\epsilon,\delta}(z) := \exp(-z \cdot (1+\delta)) - (1 - z \cdot (1-\epsilon))$$

is non-negative, then the extension of Algorithm 2 to  $b$ -matchings using scaling Equation (17) and parameters  $\epsilon$  and  $\delta$  has rounding ratio at least

$$1 - \exp \left( -1 - \delta + \frac{\epsilon + \delta}{1 - \epsilon} \right) \cdot \frac{1 - \epsilon}{1 + \delta}.$$

Combined with Lemma 3.8, the above lemma then yields our  $b$ -matching ODRS' rounding ratio, after solving the appropriate modification of the mathematical program used to prove Theorem 3.9.

**Theorem B.3.** The extension of Algorithm 2 to  $b$ -matchings, run with  $(\epsilon, \delta) \approx (0.0347, 0.0425)$ , achieves a rounding ratio of 0.646.

### B.3 Deferred proofs of the $b$ -matching ODRS

In this section we prove that our extension of Algorithm 1 from matching to  $b$ -matchings results in similar benefits when much grouping occurs. We start by introducing some useful notation.

Our online level-set rounding algorithm, Algorithm 5, makes at most one random choice when deciding whether to allocate another online node to the center of the star. Our  $b$ -matching ODRS follows the same logic (from the perspective of the offline node, playing the role of the star's center). Denote by  $C_{i,t}$  an indicator for the relevant coin for  $i$  coming up heads. Our extended core ODRS, Algorithm 1, correlates these coins for different offline nodes  $i$  at time  $t$ . Let  $S_{i,t} \in \{\lfloor s_{i,t} \rfloor, \lceil s_{i,t} \rceil\}$  be the number of times  $i$  bids by time  $t$ . By our algorithm's definition, we have the following expression for  $\mathbb{I}[S_{i,t} \neq \lceil s_{i,t} \rceil]$ .

$$\mathbb{I}[S_{i,t} \neq \lceil s_{i,t} \rceil] = \begin{cases} \mathbb{I}[S_{i,t-1} \neq \lceil s_{i,t-1} \rceil] \cdot (1 - C_{i,t-1}) & \text{if } \lceil s_{i,t} \rceil = \lceil s_{i,t-1} \rceil \\ 1 - \mathbb{I}[S_{i,t-1} = \lceil s_{i,t-1} \rceil] \cdot C_{i,t-1} & \text{if } \lceil s_{i,t} \rceil > \lceil s_{i,t-1} \rceil. \end{cases}$$

**Lemma B.4.** *For any time  $t$ , the random variables  $\{S_{i,t}\}_i$  are NA.*

*Proof.* Proof by induction on  $t \geq 1$ . The base case is trivial, as  $S_{i,1} = 0$  deterministically for all  $i$ . We turn to the inductive step for  $t$ , assuming that the variables  $S_{i,t-1}$  are NA. First, by Lemma 2.4, for any bin  $B \in \mathcal{B}_{t-1}$  at time  $t-1$ , the binary variables  $C_{i,t-1}$  for all nodes  $i \in B$ , whose sum is at most one, are NA. Moreover, the distributions over different bins are independent, and so by closure of NA under products (Lemma 2.5), all  $C_{i,t-1}$  are NA. Moreover, the variables  $C_{i,t-1}$  are independent from the variables  $S_{i,t-1}$  (who are functions only of  $C_{i,t'}$  for all  $t' < t-1$ ), and consequently, by our inductive hypothesis and another application of closure of NA under products (Lemma 2.5), all the variables  $\{S_{i,t-1}, C_{i,t-1}\}_i$  are NA. Therefore, since  $S_{i,t-1} \in [\lfloor s_{i,t-1} \rfloor, \lceil s_{i,t-1} \rceil]$ , we find that  $\mathbb{I}[S_{i,t} \neq \lceil s_{i,t} \rceil]$  is a decreasing function of  $S_{i,t-1}$  and  $C_{i,t-1}$ , i.e., they are decreasing functions of disjoint NA variables. Therefore, by yet another application of closure of NA under products (Lemma 2.5), the events  $\{\mathbb{I}[S_{i,t} \neq \lceil s_{i,t} \rceil]\}_i$  are NA. Finally, since  $S_{i,t} = \lceil s_{i,t} \rceil - \mathbb{I}[S_{i,t} \neq \lceil s_{i,t} \rceil]$  are monotone decreasing functions of NA variables, the variables  $S_{i,t}$  are themselves NA, as desired.  $\square$

We are now ready to prove that our generalization of Algorithm 1 results in bins essentially simulating offline nodes bidding with a probability equal to their constituent nodes' bids. That is, we are ready to prove the following generalization of Lemma 3.4.

**Lemma B.5.** *For any time  $t$  and set of offline nodes  $S \subseteq [n]$ , the  $b$ -matching extension of Algorithm 1 satisfies*

$$\Pr[S \cap P_t \neq \emptyset] \geq 1 - \prod_{B \in \mathcal{B}_t} \left(1 - \sum_{i \in B \cap S} x_{i,t}\right).$$

*Proof.* We denote by  $\mathcal{B}'_t \subseteq \mathcal{B}_t$  the set of singleton bins  $B = \{i\}$  for which  $\lceil s_{i,t+1} \rceil > \lceil s_{i,t} \rceil$ . For such bins  $B \in \mathcal{B}'_t$ ,  $B = \{i\}$ , we have that  $\Pr[P_t \cap S \cap B = \emptyset] = 1 - x_{i,t}$ . For other bins  $B \in \mathcal{B}_t \setminus \mathcal{B}'_t$ , we have that  $\Pr[i \in P_t] = x_{i,t}$ , by independence of  $C_{i,t}$  and  $S_{i,t}$  and since  $\Pr[C_{i,t}] = \frac{x_{i,t}}{1 - (s_{i,t} - \lfloor s_{i,t} \rfloor)}$  and  $\Pr[S_{i,t} = \lfloor s_{i,t} \rfloor] = s_{i,t} - \lfloor s_{i,t} \rfloor$  (the latter following from our online level-set algorithm's analysis). Therefore, by  $\sum_{i \in B} C_{i,t} \leq 1$  implying disjointness of the events  $[i \in P_t]$  for  $i \in B$ , we have that  $\Pr[P_t \cap S \cap B = \emptyset] = 1 - \sum_{i \in B \cap S} x_{i,t}$ . It remains to show that the events  $[P_t \cap S \cap B = \emptyset]$  for different  $B$  are negative cylinder dependent, giving the desired inequality,

$$\Pr[P_t \cap S = \emptyset] = \Pr \left[ \bigwedge_{B \in \mathcal{B}_t} (P_t \cap S \cap B = \emptyset) \right] \leq \prod_{B \in \mathcal{B}_t} \Pr[P_t \cap S \cap B = \emptyset] = \prod_{B \in \mathcal{B}_t} \left(1 - \sum_{i \in B \cap S} x_{i,t}\right).$$

Now, for bins  $B = \{i\} \in \mathcal{B}'_t$ , we have that  $\mathbb{1}[P_t \cap S \cap B = \emptyset] = 1 - [S_{i,t} = \lceil s_{i,t} \rceil] \cdot C_{i,t}$ . For non-singleton bins  $B \in \mathcal{B}_t$ , we have that  $\mathbb{1}[P_t \cap S \cap B = \emptyset] = 1 - \sum_{i \in B \cap S} C_{i,t} \cdot [S_{i,t} = \lfloor s_{i,t} \rfloor]$ . In both cases, the indicators  $\mathbb{1}[P_t \cap S \cap B = \emptyset]$  are NA, as they are monotone decreasing functions of disjoint NA variables (here, we need to first apply decreasing functions that map  $S_{i,t}$  to  $\lceil s_{i,t} \rceil - S_{i,t}$  for all  $i \in B \in \mathcal{B}_t \setminus \mathcal{B}'_t$  and all other variables to themselves). Hence, by closure of NA under such functions (Lemma 2.5), the variables  $\mathbb{1}[P_t \cap S \cap B = \emptyset]$  are NA. Therefore, by Lemma 2.3, these variables are negative cylinder dependent, yielding the required inequality above. The lemma follows.  $\square$

## C Deferred Proofs of Section 4

We start by briefly proving that Algorithm 3 terminates and preserves marginals.

**Fact C.1.** *Algorithm 3 terminates, and satisfies  $\Pr[i \in \mathcal{S}] = x_i$  for all  $i \in [n]$  (Property (P1)).*

*Proof.* It is easy to see that STEP preserves the sum of its inputs/outputs while decreasing the number of fractional such inputs/outputs. Consequently,  $\sum_i y_i$  is integral during the execution of Algorithm 3, while the number of fractional  $y_i$  decreases and cannot be one, so this algorithm terminates with all  $y_i$  binary. Moreover, STEP is easily seen to preserve these values' expectations, and so by induction we have that  $\mathbb{E}[y_i] = x_i$  for all  $i$ , i.e., Algorithm 3 satisfies Property (P1).  $\square$

**Lemma 4.1.** *Let  $s_j := \sum_{i \leq j} x_i$  and  $Y_j^t := \sum_{i \leq j} y_i^t$ , where  $y_i^t$  is the value  $y_i$  after  $t$  applications of STEP in Algorithm 3. Then,  $Y_j^t \in [\lfloor s_j \rfloor, \lceil s_j \rceil]$  for all  $j$  and  $t$ .*

*Proof.* We prove this for all  $j$  by induction on  $t \geq 0$ . The base case is trivial. Fix a time  $t$  and let  $i_1$  and  $i_2$  be as in Algorithm 3 at time  $t + 1$ . If  $i_1, i_2 \notin [j]$  or  $i_1, i_2 \in [j]$ , then, since  $\text{STEP}(y_{i_1}, y_{i_2})$  does not affect  $Y_j^t$  in the former case and preserves the sums  $y_{i_1} + y_{i_2}$  and  $Y_j$  in the latter case, the inductive hypothesis implies  $Y_j^{t+1} = Y_j^t \in [\lfloor s_j \rfloor, \lceil s_j \rceil]$ . Conversely, if  $i_1 \in [j]$  and  $i_2 \notin [j]$  (and therefore  $i_1 = j$ ), we have that  $y_i^{t+1} = y_i^t \in \{0, 1\}$  for all  $i \in [j - 1]$ , while  $y_{i_1}^{t+1} \in [0, 1]$ . Since by the inductive hypothesis we have that  $Y_j^t = Y_{j-1}^t + y_j^t \in [\lfloor s_j \rfloor, \lceil s_j \rceil]$  and  $Y_{j-1}^t = Y_{j-1}^{t+1}$  is the sum of 0-1 terms, while  $y_j^t \in (0, 1)$ , this implies that  $Y_j^{t+1} = Y_{j-1}^{t+1} + y_j^{t+1} = Y_{j-1}^t + y_j^{t+1} \in [\lfloor s_j \rfloor, \lceil s_j \rceil]$ .  $\square$

### C.1 Direct (partial) analysis of Algorithm 5

In this section we provide a simple, self-contained proof that Algorithm 5 satisfies the first two properties we desire of it. In the paper body, we use this direct proof to prove that our algorithm's output distribution is the same as its offline counterpart, from which we also obtain the third desideratum, namely Property (P3).

**Lemma C.2.** *Algorithm 5 is well-defined (i.e.,  $p_t \in [0, 1]$  for all times  $t$  and any realization of the randomness) and it satisfies properties (P1) and (P2).*

*Proof.* We prove the above three properties (including  $p_t \in [0, 1]$ ) by strong induction on  $t \geq 0$ . The base case,  $t = 0$ , is trivial. As our inductive hypothesis (I.H.), we assume that these properties hold for all  $t' \leq t - 1$  and prove them for all  $t' \leq t$ , starting with Property (P2). By the I.H.,  $S_{t-1} \leq \lfloor s_{t-1} \rfloor \leq \lfloor s_t \rfloor$ ; so, since  $p_t = 0$  if  $S_{t-1} = \lfloor s_t \rfloor$ , we trivially have that  $S_t \leq \lfloor s_t \rfloor$ . Moreover, by the I.H.,  $S_{t-1} \geq \lfloor s_{t-1} \rfloor = \lfloor s_t - x_t \rfloor \geq \lfloor s_t \rfloor - 1$ ; so, since  $p_t = 1$  if  $S_{t-1} < \lfloor s_t \rfloor$ , in which case  $S_{t-1} = \lfloor s_t \rfloor - 1$ , we have that  $S_t \geq \lfloor s_t \rfloor$ . Thus,  $S_t \in \{\lfloor s_t \rfloor, \lceil s_t \rceil\}$ , proving Property (P2).

Next we prove that Algorithm 5 is well-defined, as  $p_t \in [0, 1]$  (regardless of prior randomness). The first non-trivial case is when  $S_{t-1} = \lfloor s_t \rfloor = \lfloor s_{t-1} \rfloor$ . Non-negativity of  $p_t$ —a ratio of non-negative terms—is immediate, while

$$p_t = \frac{x_t}{\lfloor s_{t-1} \rfloor + 1 - s_{t-1}} = \frac{s_t - s_{t-1}}{\lfloor s_{t-1} \rfloor + 1 - s_{t-1}} \leq \frac{\lfloor s_t \rfloor + 1 - s_{t-1}}{\lfloor s_{t-1} \rfloor + 1 - s_{t-1}} = \frac{\lfloor s_{t-1} \rfloor + 1 - s_{t-1}}{\lfloor s_{t-1} \rfloor + 1 - s_{t-1}} = 1.$$

The second non-trivial case is when  $S_{t-1} = \lfloor s_t \rfloor > \lfloor s_{t-1} \rfloor$ , in which case  $\lfloor s_t \rfloor = \lfloor s_{t-1} \rfloor + 1$ . Again, non-negativity of  $p_t$  is immediate, while

$$p_t = \frac{s_t - \lfloor s_t \rfloor}{s_{t-1} - \lfloor s_{t-1} \rfloor} = \frac{s_{t-1} + x_t - \lfloor s_{t-1} \rfloor - 1}{s_{t-1} - \lfloor s_{t-1} \rfloor} \leq \frac{s_{t-1} - \lfloor s_{t-1} \rfloor}{s_{t-1} - \lfloor s_{t-1} \rfloor} = 1.$$

To prove Property (P1), we will need a closed form for  $q_t := \Pr[S_t = \lfloor s_t \rfloor]$ , useful when discussing  $\Pr[S_{t-1} = \lfloor s_t \rfloor]$ , the probability that  $t$  may be added to  $\mathcal{S}$ , and  $\Pr[S_{t-1} < \lfloor s_t \rfloor]$ , the probability that  $t$  *must* be added to  $\mathcal{S}$  (to satisfy Property (P2)). Note that  $S_t \sim \lfloor s_t \rfloor + \text{Ber}(1 - q_t)$  by Property (P2), while  $\mathbb{E}[S_t] = \sum_{t' \leq t} \mathbb{E}[X_{t'}] = \sum_{t' \leq t} x_{t'} = s_t$ , by Property (P1) (for all  $t' \leq t$ ) and linearity of expectation. Combining these observations, we obtain  $s_t = \mathbb{E}[S_t] = \lfloor s_t \rfloor + 1 - q_t$ , and therefore,

$$q_t = \lfloor s_t \rfloor + 1 - s_t. \quad (19)$$

We now use the closed form for  $q_{t-1}$  to prove Property (P1) for time  $t$ . The easy case is when  $\lfloor s_t \rfloor = \lfloor s_{t-1} \rfloor$ . Here we have that  $\mathbb{E}[X_t \mid S_{t-1} = \lfloor s_t \rfloor] = \frac{x_t}{q_{t-1}}$ , while trivially  $\mathbb{E}[X_t \mid S_{t-1} = \lceil s_t \rceil] = 0$ . Consequently, since  $S_{t-1} \in \{\lfloor s_{t-1} \rfloor, \lceil s_{t-1} \rceil\} = \{\lfloor s_t \rfloor, \lceil s_t \rceil\}$  in this case, then by total probability we have that  $\mathbb{E}[X_t] = q_{t-1} \cdot \frac{x_t}{q_{t-1}} + 0 = x_t$ , as desired.

Finally, we consider the case where  $\lfloor s_t \rfloor > \lfloor s_{t-1} \rfloor$ , i.e.,  $\lfloor s_t \rfloor = \lfloor s_{t-1} \rfloor + 1$ . If in addition  $s_{t-1} = \lfloor s_{t-1} \rfloor$  (i.e.,  $s_{t-1}$  is integral, implying that  $s_t = s_{t-1} + 1$ , and hence  $x_t = s_t - s_{t-1} = 1$ ), we have by Property (P2) that  $S_{t-1} = s_{t-1} < \lfloor s_t \rfloor$  (with probability one), and so we add  $t$  to  $\mathcal{S}$  with probability one, resulting in  $\mathbb{E}[X_t] = 1 = x_t$ , as desired. If, conversely, we have that  $s_{t-1} \neq \lfloor s_{t-1} \rfloor < \lfloor s_t \rfloor$ , then by total probability, we obtain the desired equality as follows.

$$\begin{aligned} \mathbb{E}[X_t] &= \Pr[S_{t-1} = \lfloor s_{t-1} \rfloor] \cdot 1 + \Pr[S_{t-1} \neq \lfloor s_{t-1} \rfloor] \cdot \frac{s_t - \lfloor s_t \rfloor}{s_{t-1} - \lfloor s_{t-1} \rfloor} \\ &= q_{t-1} + (1 - q_{t-1}) \cdot \frac{x_t + s_{t-1} - (\lfloor s_{t-1} \rfloor + 1)}{s_{t-1} - \lfloor s_{t-1} \rfloor} && \begin{cases} s_t = x_t + s_{t-1} \\ \lfloor s_t \rfloor = \lfloor s_{t-1} \rfloor + 1 \end{cases} \\ &= q_{t-1} + (1 - q_{t-1}) \cdot \frac{x_t - q_{t-1}}{1 - q_{t-1}} && \text{Equation (19)} \\ &= x_t. \end{aligned} \quad \square$$

## C.2 Coupling the offline and online algorithms

Here we prove that our online level-set algorithm, Algorithm 5, induces the same output distribution as the offline algorithm of [79], Algorithm 3

**Theorem 4.4.** *Fix a vector  $\mathbf{x} \in [0, 1]^n$ . Let  $\mathcal{D}$  and  $\mathcal{D}'$  denote the probability distributions on  $\{0, 1\}^n$  induced by the online and offline algorithms run on  $\mathbf{x}$ , respectively. Then,  $\mathcal{D} = \mathcal{D}'$ .*

*Proof.* We prove by induction on  $t \in [n]$  that the following holds:

$$\forall (b_1, \dots, b_{t-1}) \in \{0, 1\}^{t-1}, \quad \Pr_{\mathcal{D}}[X_t = 1 \mid (\forall i < t, X_i = b_i)] = \Pr_{\mathcal{D}'}[X_t = 1 \mid (\forall i < t, X_i = b_i)]. \quad (20)$$

The (strong) inductive hypothesis, whereby Equation (20) holds for all  $t' < t$  clearly implies that for all  $(b_1, \dots, b_t) \in \{0, 1\}^{t-1}$  we have that  $\Pr_{\mathcal{D}}[(\forall i < t, X_i = b_i)] = \Pr_{\mathcal{D}'}[(\forall i < t, X_i = b_i)]$ . Similarly, Equation (20) holds for all  $t \leq n$  iff the two distributions are the same.

The base case  $t = 1$  follows from Property (P1). So suppose  $t > 1$ . Fix  $b_1, \dots, b_{t-1} \in \{0, 1\}$ , and let  $\mathcal{E}$  denote the event “ $\forall i < t, X_i = b_i$ ” (measured w.r.t.  $\mathcal{D}$  or w.r.t.  $\mathcal{D}'$ ). Then, by the strong induction hypothesis,  $\Pr_{\mathcal{D}}[\mathcal{E}] = \Pr_{\mathcal{D}'}[\mathcal{E}]$ , and in particular,  $\Pr_{\mathcal{D}}[\mathcal{E}] \neq 0$  iff  $\Pr_{\mathcal{D}'}[\mathcal{E}] \neq 0$ ; thus we may assume that the events conditioned on in the LHS and in the RHS of (20) both have nonzero (identical) probabilities. This is the only place where we will need the induction hypothesis.

Recall that  $s_{t-1} = \sum_{i=1}^{t-1} x_i$ . Furthermore, although  $S_{t-1}$  technically depends on the rounding algorithm, it is natural to take it to be  $\sum_{i=1}^{t-1} b_i$  here. Note that since both the offline and online algorithms satisfy Property (P2), we have that  $S_{t-1} \in \{\lfloor s_{t-1} \rfloor, \lceil s_{t-1} \rceil\}$ .

**Case I:**  $s_{t-1} \in \mathbb{Z}$ . This is an easy case. It is immediate that  $\Pr_{\mathcal{D}}[X_t = 1 \mid \mathcal{E}] = x_t$  here. This is also not hard to see this for the offline algorithm. Recall that the offline algorithm starts with a copy  $y$  of  $x$ . Since  $s_{t-1} \in \mathbb{Z}$ , the offline algorithm would set  $X_i$  for exactly  $s_{t-1}$  indices  $i \in [t-1]$  to one (and the rest in  $[t-1]$  to zero), and would then recursively run on the suffix  $(y_t, y_{t+1}, \dots, y_n)$  of  $y$  with no correlation with the rounding thus far. Furthermore,  $y_j = x_j$  for all  $j \geq t$ . Thus, by Property (P1) applied to this suffix, we have that  $\Pr_{\mathcal{D}'}[X_t = 1 \mid \mathcal{E}] = x_t$  also.

We may thus assume from now on that  $z \doteq s_{t-1} - \lfloor s_{t-1} \rfloor$  lies in  $(0, 1)$ .

**Case II:**  $s_{t-1} \notin \mathbb{Z}$  and  $z + x_t \leq 1$ . An easy sub-case here is that  $S_{t-1} = \lceil s_{t-1} \rceil$ ; here, since both the offline and online algorithms satisfy Property (P2), it is immediate that in this sub-case

$$\Pr_{\mathcal{D}}[X_t = 1 \mid \mathcal{E}] = \Pr_{\mathcal{D}'}[X_t = 1 \mid \mathcal{E}] = 0.$$

We may thus assume that  $S_{t-1} = \lfloor s_{t-1} \rfloor$ . By the definition of the online algorithm 5 we have that

$$\Pr_{\mathcal{D}}[X_t = 1 \mid \mathcal{E}] = \frac{x_t}{1 - z}. \quad (21)$$

We now analyze  $\Pr_{\mathcal{D}'}[X_t = 1 \mid \mathcal{E}]$ . Consider the offline algorithm *just before* it involves  $x_t$  in STEP: at this point in time, it has:

- set  $\lfloor s_{t-1} \rfloor$  of the variables  $(X_i : i < t)$  to one;
- set  $t - 2 - \lfloor s_{t-1} \rfloor$  of the variables  $(X_i : i < t)$  to zero; and, recalling again that the offline algorithm starts with a copy  $y$  of  $x$ ,
- it has assigned the current value  $z$  to one variable  $y_{i^*}$  where  $i^* \in [t-1]$  is a random variable from some distribution.

At this point, the offline algorithm is basically recursively run on the sequence  $(y_{i^*}, y_t, y_{t+1}, \dots, y_n)$ , where  $y_{i^*} = z$  and  $y_j = x_j$  for all  $j \geq t$ . Crucially, since  $S_{t-1} = \lfloor s_{t-1} \rfloor$ , the conditioning on  $\mathcal{E}$  says that the variable  $y_{i^*}$  is *eventually rounded to 0*. Thus, we can compute  $\Pr_{\mathcal{D}'}[X_t = 1 \mid \mathcal{E}]$  as follows. Suppose the offline algorithm is run on the sequence  $(u_1, u_2, \dots, u_{n-t+2}) \doteq (y_{i^*}, y_t, y_{t+1}, \dots, y_n)$ , to output a random bit-vector  $(U_1, U_2, \dots, U_{n-t+2})$ ; then,

$$\Pr_{\mathcal{D}'}[X_t = 1 \mid \mathcal{E}] = \Pr_{\mathcal{D}'}[U_2 = 1 \mid U_1 = 0]. \quad (22)$$

But since  $u_1 + u_2 = z + x_t \leq 1$ , the first STEP applied to  $u_1$  and  $u_2$  ensures that *at most one of  $u_1$  and  $u_2$  gets rounded to one*. This yields

$$\Pr_{\mathcal{D}'}[U_2 = 1 \mid U_1 = 0] = \frac{\Pr_{\mathcal{D}'}[(U_2 = 1) \wedge (U_1 = 0)]}{\Pr_{\mathcal{D}'}[U_1 = 0]} = \frac{\Pr_{\mathcal{D}'}[U_2 = 1]}{\Pr_{\mathcal{D}'}[U_1 = 0]} = \frac{x_t}{1 - z},$$



where the second equality holds since at most one of  $U_1$  and  $U_2$  is one; we are thus done by (21).

**Case III:**  $s_{t-1} \notin \mathbb{Z}$  and  $z + x_t > 1$ . This is handled similarly. The easy sub-case here is that  $S_{t-1} = \lfloor s_{t-1} \rfloor$ ; here, since both the offline and online algorithms satisfy Property (P2), it is immediate that

$$\Pr_{\mathcal{D}}[X_t = 1 \mid \mathcal{E}] = \Pr_{\mathcal{D}'}[X_t = 1 \mid \mathcal{E}] = 1.$$

We may thus assume that  $S_{t-1} = \lceil s_{t-1} \rceil$ . By the definition of the online algorithm 5 we have that

$$\Pr_{\mathcal{D}}[X_t = 1 \mid \mathcal{E}] = \frac{x_t + z - 1}{z}. \quad (23)$$

Now, the offline algorithm is, like in Case II, essentially run on the sequence  $(u_1, u_2, \dots, u_{n-t+2}) \doteq (y_{i^*}, y_t, y_{t+1}, \dots, y_n)$ —where  $y_{i^*}$  is initially  $z$ —to output a random bit-vector  $(U_1, U_2, \dots, U_{n-t+2})$ , with the key difference from Case II being that  $U_1$  will eventually be set to 1. So,

$$\Pr_{\mathcal{D}'}[X_t = 1 \mid \mathcal{E}] = \Pr_{\mathcal{D}'}[U_2 = 1 \mid U_1 = 1]. \quad (24)$$

Now, since  $u_1 + u_2 = z + x_t > 1$ , the first STEP applied to  $u_1$  and  $u_2$  ensures that *at least* one of  $u_1$  and  $u_2$  gets rounded to one. This implies

$$\begin{aligned} \Pr_{\mathcal{D}'}[U_2 = 1 \mid U_1 = 1] &= \frac{\Pr_{\mathcal{D}'}[U_2 = 1] - \Pr_{\mathcal{D}'}[(U_2 = 1) \wedge (U_1 = 0)]}{\Pr_{\mathcal{D}'}[U_1 = 1]} \\ &= \frac{\Pr_{\mathcal{D}'}[U_2 = 1] - \Pr_{\mathcal{D}'}[U_1 = 0]}{\Pr_{\mathcal{D}'}[U_1 = 1]} \\ &= \frac{x_t + z - 1}{z}, \end{aligned}$$

where the second equality holds since at least one of  $U_1$  and  $U_2$  is one; we are thus done by (23). This concludes the proof of the inductive step.  $\square$

## D Deferred Proofs of Section 6

In this section we prove that any sufficiently many binary variables must contain some  $2^r$  with some (near-)positive  $2^r$ -wise correlation, generalizing the special case of  $r = 1$  given by Fact 6.1.

**Lemma 6.3.** *Let  $r \in \mathbb{N}$  and  $0 \leq \epsilon \leq p \leq 1$ . Then, there exists some  $n = n_r(p, \epsilon)$  such that any Bernoulli variables  $Y_1, \dots, Y_n \sim \text{Ber}(p)$  contains some subset  $I \subseteq [n]$  of size  $|I| = 2^r$  such that*

$$\mathbb{E} \left[ \prod_i Y_i \right] \geq \prod_i \mathbb{E}[Y_i] - \epsilon.$$

*Proof.* We prove this claim for all  $0 \leq \epsilon \leq p \leq 1$  by induction on  $r \geq 1$ , and note that the claim is trivial if  $p^{2^r} \leq \epsilon$ , in which case the RHS is negative, so  $n_r(p, \epsilon) = 2^r$  suffices. Fact 6.1 proves the base case, showing that  $n_1(p, \epsilon) \leq \lceil \frac{2p}{\epsilon} + 1 \rceil$  suffices. To prove the inductive step, consider a set of  $k := n_1(p, \frac{\epsilon}{2^{2^r}}) + 2m$  variables  $Y_1, \dots, Y_k \sim \text{Ber}(p)$ , for  $m$  to be determined shortly. Then, we find a sequence of disjoint pairs  $I_1, I_2, \dots, I_m \subseteq [n]$ , where for each  $I_s = \{i, j\}$ ,  $s \in [m]$ , we have that  $\text{Cov}(Y_i, Y_j) \geq -\frac{\epsilon}{2^{2^r}}$ . The existence of such pairs follows by repeatedly invoking Fact 6.1 applied to the variables  $\{Y_i \mid i \in [n] \setminus \bigcup_{\ell=1}^s I_\ell\}$  to obtain the pair  $I_{s+1}$ . Now, for each pair  $I_s = \{i, j\}$  as above, define a new variable  $Z_s := Y_i \cdot Y_j$ . By  $\text{Cov}(Y_i, Y_j) \geq -\frac{\epsilon}{2^{2^r}}$ , we have that  $\Pr[Z_s] \geq p^2 - \frac{\epsilon}{2^{2^r}}$ . Next, couple these combined variables with equiprobable variables  $A_s \sim \text{Ber}(p^2 - \frac{\epsilon}{2^{2^r}})$  with  $Z_s \geq A_s$

always. Then, if we take  $m = n_{r-1}(p^2 - \frac{\epsilon}{2^{2^r}}, \frac{\epsilon}{2})$ , the inductive hypothesis applied to the  $m$  variables  $A_s$  implies the existence of a subset  $S \subseteq [m]$  of  $2^{r-1}$  of these variables satisfying the following.

$$\mathbb{E} \left[ \prod_{i \in \bigcup_{s \in S} I_s} Y_i \right] = \mathbb{E} \left[ \prod_{s \in S} Z_s \right] \geq \mathbb{E} \left[ \prod_{s \in S} A_s \right] \geq \left( p^2 - \frac{\epsilon}{2^{2^r}} \right)^{2^{r-1}} - \frac{\epsilon}{2} \geq p^{2^r} - 2^{2^{r-1}} \cdot \frac{\epsilon}{2^{2^r}} - \frac{\epsilon}{2} \geq p^{2^r} - \epsilon.$$

Above, the first inequality follows by the coupling  $Z_s \geq A_s$ . The second inequality follows from the inductive hypothesis. The third inequality follows from  $(a-b)^k \geq a^k - 2^k \cdot b$  for any  $a, b \in [0, 1]$  and  $k \in \mathbb{N}$  (as seen by expanding  $(a-b)^k$ ). The final inequality follows from  $2^{2^r-1} \geq 2^{2^{r-1}}$  for  $r \geq 1$ . We conclude that  $\{Y_i \mid i \in \bigcup_{s \in S} I_s\}$  is the desired subset of  $2^r$  variables in  $Y_1, \dots, Y_n$ .  $\square$

**Remark D.1.** We did not attempt to optimize the minimum possible  $n_r(p, \epsilon)$  above. Nonetheless, for concrete bounds, we note that if  $p^{2^r} \geq \epsilon$ , then our proof yields bounds satisfying the following recurrence.

$$n_1(p, \epsilon) \leq \left\lceil \frac{2p}{\epsilon} + 1 \right\rceil \leq \frac{4p}{\epsilon},$$

$$n_r(p, \epsilon) \leq n_1\left(p, \frac{\epsilon}{2^{2^r}}\right) + 2n_{r-1}\left(p^2 - \frac{\epsilon}{2^{2^r}}, \frac{\epsilon}{2}\right) \leq \frac{4 \cdot 2^{2^r} p}{\epsilon} + 2n_{r-1}\left(p^2 - \frac{\epsilon}{2^{2^r}}, \frac{\epsilon}{2}\right).$$

Noting that this recurrence is monotone increasing in  $p$ , we have that  $n_r(p, \epsilon) = O\left(2^{2^r} \cdot \frac{p}{\epsilon}\right)$ .

## References

- [1] Algorithms with predictions (ALPS). <https://algorithms-with-predictions.github.io/>. Accessed: 2024-10-24. pages [↑21](#)
- [2] AGGARWAL, G., GOEL, G., KARANDE, C., AND MEHTA, A. 2011. Online vertex-weighted bipartite matching and single-bid budgeted allocations. In *Proceedings of the 22nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 1253–1264. pages [↑4](#)
- [3] AGGARWAL, G., MOTWANI, R., SHAH, D., AND ZHU, A. 2003. Switch scheduling via randomized edge coloring. In *Proceedings of the 44th Symposium on Foundations of Computer Science (FOCS)*. 502–512. pages [↑2](#), [↑14](#)
- [4] AGRAWAL, R., GOLLAPUDI, S., HALVERSON, A., AND IEONG, S. 2009. Diversifying search results. In *Proceedings of the 2nd Symposium on Web Search and Data Mining (WSDM)*. 5–14. pages [↑25](#)
- [5] AOUD, A., AMANI HAMEDANI, A., AND SABERI, A. 2024. Adaptive policies and approximation schemes for dynamic matching. [Video](#) and [slides](#). accessed: 2024-10-24. pages [↑4](#)
- [6] BAHMANI, B., MEHTA, A., AND MOTWANI, R. 2012. Online graph edge-coloring in the random-order arrival model. *Theory of Computing* 8, 1, 567–595. pages [↑2](#), [↑14](#)
- [7] BANSAL, N. AND COHEN, I. R. 2021. Contention resolution, matrix scaling and fair allocation. In *Proceedings of the 19th Workshop on Approximation and Online Algorithms (WAOA)*. 252–274. pages [↑3](#), [↑5](#), [↑28](#)
- [8] BAR-NOY, A., MOTWANI, R., AND NAOR, J. 1992. The greedy algorithm is optimal for on-line edge coloring. *Information Processing Letters (IPL)* 44, 5, 251–253. pages [↑14](#)

- [9] BECHTEL, C., DUGHMI, S., AND PATEL, N. 2022. Delegated Pandora’s box. 666–693. pages [↑4](#)
- [10] BEN-DAVID, S., BORODIN, A., KARP, R., TARDOS, G., AND WIGDERSON, A. 1994. On the power of randomization in on-line algorithms. *Algorithmica* 11, 1, 2–14. pages [↑4](#)
- [11] BHATTACHARYA, S., GRANDONI, F., AND WAJC, D. 2021. Online edge coloring algorithms via the nibble method. In *Proceedings of the 32nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 2830–2842. pages [↑2](#), [↑14](#)
- [12] BIRNBAUM, B. AND MATHIEU, C. 2008. On-line bipartite matching made simple. *ACM SIGACT News* 39, 1, 80–87. pages [↑4](#)
- [13] BLANC, G. AND CHARIKAR, M. 2021. Multiway online correlated selection. In *Proceedings of the 62nd Symposium on Foundations of Computer Science (FOCS)*. 1277–1284. pages [↑1](#), [↑4](#), [↑21](#)
- [14] BLIKSTAD, J., SVENSSON, O., VINTAN, R., AND WAJC, D. 2024a. Online edge coloring is (nearly) as easy as offline. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing (STOC)*. 36–46. pages [↑1](#), [↑2](#), [↑14](#)
- [15] BLIKSTAD, J., SVENSSON, O., VINTAN, R., AND WAJC, D. 2024b. Simple and asymptotically optimal online bipartite edge coloring. In *Proceedings of the 7th Symposium on Simplicity in Algorithms (SOSA)*. 331–336. pages [↑2](#), [↑14](#)
- [16] BLIKSTAD, J., SVENSSON, O., VINTAN, R., AND WAJC, D. 2025. Deterministic online bipartite edge coloring. In *Proceedings of the 36th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. pages [↑2](#), [↑4](#)
- [17] BORCEA, J., BRÄNDÉN, P., AND LIGGETT, T. 2009. Negative dependence and the geometry of polynomials. *Journal of the American Mathematical Society* 22, 2, 521–567. pages [↑28](#)
- [18] BOYCE, B. 1982. Beyond topicality: A two stage view of relevance and the retrieval process. *Info. Processing and Management* 18, 105–109. pages [↑25](#)
- [19] BRÄNDÉN, P. AND JONASSON, J. 2012. Negative dependence in sampling. *Scandinavian Journal of Statistics* 39, 4, 830–838. pages [↑3](#), [↑13](#), [↑29](#)
- [20] BRAVERMAN, M., DERAKHSHAN, M., AND MOLINA LOVETT, A. 2022. Max-weight online stochastic matching: Improved approximations against the online benchmark. In *Proceedings of the 23rd ACM Conference on Economics and Computation (EC)*. 967–985. pages [↑2](#), [↑16](#), [↑17](#), [↑18](#), [↑19](#)
- [21] BRAVERMAN, M., DERAKHSHAN, M., POLLNER, T., SABERI, A., AND WAJC, D. 2025. New philosopher inequalities for online bayesian matching, via pivotal sampling. In *Proceedings of the 36th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. pages [↑4](#)
- [22] BRUBACH, B., GRAMMEL, N., MA, W., AND SRINIVASAN, A. 2021. Improved guarantees for offline stochastic matching via new ordered contention resolution schemes. *Proceedings of the 55th Annual Conference on Neural Information Processing Systems (NIPS)* 34, 27184–27195. pages [↑4](#)
- [23] BRUGGMANN, S. AND ZENKLUSEN, R. 2022. An optimal monotone contention resolution scheme for bipartite matchings via a polyhedral viewpoint. *Mathematical Programming* 191, 2, 795–845. pages [↑4](#)

- [24] BUCHBINDER, N., NAOR, J. S., AND WAJC, D. 2023. Lossless online rounding for online bipartite matching (despite its impossibility). In *Proceedings of the 34th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 2030–2068. pages [↑1](#), [↑4](#)
- [25] BUOLAMWINI, J. AND GEBRU, T. 2018. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Proceedings of the 1st Conference on Fairness, Accountability and Transparency (FAccT)*. 77–91. pages [↑24](#)
- [26] CARBONELL, J. G. AND GOLDSTEIN, J. 2017. The use of MMR, diversity-based reranking for reordering documents and producing summaries. *SIGIR Forum* 51, 2, 209–210. pages [↑25](#)
- [27] CHAWLA, S., HARTLINE, J. D., MALEC, D. L., AND SIVAN, B. 2010. Multi-parameter mechanism design and sequential posted pricing. In *Proceedings of the 42nd Annual ACM Symposium on Theory of Computing (STOC)*. 311–320. pages [↑4](#)
- [28] CHEKURI, C., VONDRÁK, J., AND ZENKLUSEN, R. 2010. Dependent randomized rounding via exchange properties of combinatorial structures. In *Proceedings of the 51st Symposium on Foundations of Computer Science (FOCS)*. 575–584. pages [↑4](#)
- [29] CHRISTOFIDES, T. C. AND VAGGELATOU, E. 2004. A connection between supermodular ordering and positive/negative association. *Journal of Multivariate analysis* 88, 1, 138–151. pages [↑29](#)
- [30] CLARKE, C. L. A., KOLLA, M., CORMACK, G. V., VECHTOMOVA, O., ASHKAN, A., BÜTTCHER, S., AND MACKINNON, I. 2008. Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 659–666. pages [↑25](#)
- [31] COHEN, I. R., PENG, B., AND WAJC, D. 2019. Tight bounds for online edge coloring. In *Proceedings of the 60th Symposium on Foundations of Computer Science (FOCS)*. 1–25. pages [↑1](#), [↑2](#), [↑14](#), [↑15](#)
- [32] COHEN, I. R. AND WAJC, D. 2018. Randomized online matching in regular graphs. In *Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 960–979. pages [↑1](#), [↑2](#)
- [33] DEVANUR, N. AND MEHTA, A. 2022. Online matching in advertisement auctions. In *Online and Matching-Based Market Design*. Cambridge University Press. pages [↑4](#)
- [34] DEVANUR, N. R., JAIN, K., AND KLEINBERG, R. D. 2013. Randomized primal-dual analysis of ranking for online bipartite matching. In *Proceedings of the 24th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 101–107. pages [↑1](#), [↑4](#), [↑25](#)
- [35] DUBHASHI, D., JONASSON, J., AND RANJAN, D. 2007. Positive influence and negative dependence. *Combinatorics, Probability and Computing* 16, 01, 29–41. pages [↑3](#)
- [36] DUBHASHI, D. AND RANJAN, D. 1996. Balls and bins: A study in negative dependence. *BRICS Report Series* 3, 25. pages [↑5](#), [↑6](#)
- [37] DUDEJA, A., GOSWAMI, R., AND SAKS, M. 2025. Randomized greedy online edge coloring succeeds for dense and randomly-ordered graphs. In *Proceedings of the 36th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. pages [↑2](#)

- [38] DUGHMI, S. 2020. The outer limits of contention resolution on matroids and connections to the secretary problem. In *Proceedings of the 47th International Colloquium on Automata, Languages and Programming (ICALP)*. 42:1–42:18. pages [↑4](#)
- [39] DUGHMI, S. 2022. Matroid secretary is equivalent to contention resolution. In *Proceedings of the 13th Innovations in Theoretical Computer Science Conference (ITCS)*. 58:1–58:23. pages [↑4](#)
- [40] DUPPALA, S., LI, G. Z., LUQUE, J., SRINIVASAN, A., AND VALIEVA, R. 2023. Concentration of submodular functions under negative dependence. *arXiv preprint arXiv:2309.05554*. pages [↑29](#)
- [41] DÜTTING, P., FELDMAN, M., KESSELHEIM, T., AND LUCIER, B. 2020. Prophet inequalities made easy: Stochastic optimization by pricing nonstochastic inputs. *SIAM Journal on Computing (SICOMP)* 49, 3. pages [↑2](#), [↑16](#)
- [42] EDEN, A., FELDMAN, M., FIAT, A., AND SEGAL, K. 2021. An economics-based analysis of ranking for online bipartite matching. In *Proceedings of the 4th Symposium on Simplicity in Algorithms (SOSA)*. 107–110. pages [↑4](#)
- [43] EZRA, T., FELDMAN, M., GRAVIN, N., AND TANG, Z. G. 2020. Online stochastic max-weight matching: prophet inequality for vertex and edge arrival models. In *Proceedings of the 21st ACM Conference on Economics and Computation (EC)*. 769–787. pages [↑2](#), [↑4](#), [↑16](#)
- [44] FAHRBACH, M., HUANG, Z., TAO, R., AND ZADIMOUGHADDAM, M. 2020. Edge-weighted online bipartite matching. In *Proceedings of the 61st Symposium on Foundations of Computer Science (FOCS)*. 412–423. pages [↑1](#), [↑4](#), [↑21](#)
- [45] FEIGE, U. AND VONDRÁK, J. 2006. The allocation problem with submodular utility functions. In *Proceedings of the 47th Symposium on Foundations of Computer Science (FOCS)*. 667–676. pages [↑1](#), [↑3](#), [↑4](#), [↑5](#), [↑28](#)
- [46] FELDMAN, J., KORULA, N., MIRROKNI, V., MUTHUKRISHNAN, S., AND PÁL, M. 2009. Online ad assignment with free disposal. In *Proceedings of the 5th Conference on Web and Internet Economics (WINE)*. 374–385. pages [↑4](#)
- [47] FELDMAN, M., GRAVIN, N., AND LUCIER, B. 2015. Combinatorial auctions via posted prices. In *Proceedings of the 26th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 123–135. pages [↑2](#), [↑16](#)
- [48] FELDMAN, M., SVENSSON, O., AND ZENKLUSEN, R. 2016. Online contention resolution schemes. In *Proceedings of the 27th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 1014–1033. pages [↑4](#)
- [49] FU, H., LU, P., TANG, Z. G., TURKIELTAUB, A., WU, H., WU, J., AND ZHANG, Q. 2022. Oblivious online contention resolution schemes. In *Proceedings of the 5th Symposium on Simplicity in Algorithms (SOSA)*. 268–278. pages [↑4](#)
- [50] FU, H., TANG, Z. G., WU, H., WU, J., AND ZHANG, Q. 2021. Random order vertex arrival contention resolution schemes for matching, with applications. In *Proceedings of the 48th International Colloquium on Automata, Languages and Programming (ICALP)*. 68:1–68:20. pages [↑4](#)

- [51] GANDHI, R., KHULLER, S., PARTHASARATHY, S., AND SRINIVASAN, A. 2006. Dependent rounding and its applications to approximation algorithms. *Journal of the ACM (JACM)* 53, 3, 324–360. pages [↑1](#)
- [52] GAO, R., HE, Z., HUANG, Z., NIE, Z., YUAN, B., AND ZHONG, Y. 2021. Improved online correlated selection. In *Proceedings of the 62nd Symposium on Foundations of Computer Science (FOCS)*. 1265–1276. pages [↑1](#), [↑4](#), [↑21](#)
- [53] GURUGANESH, G. AND LEE, E. 2018. Understanding the correlation gap for matchings. In *Proceedings of the 37th Conference on Foundations of Software Technology and Theoretical Computer Science*. 32:1–32:15. pages [↑4](#)
- [54] HOSSEINI, H., HUANG, Z., IGARASHI, A., AND SHAH, N. 2024. Class fairness in online matching. *Artificial Intelligence* 335, 104177. pages [↑4](#)
- [55] HUANG, Z., SHU, X., AND YAN, S. 2022. The power of multiple choices in online stochastic matching. In *Proceedings of the 54th Annual ACM Symposium on Theory of Computing (STOC)*. 91–103. pages [↑1](#), [↑4](#)
- [56] HUANG, Z., TAND, Z. G., AND WAJC, D. 2024. Online matching: A brief survey. *ACM SIGecom Exchanges* 22, 1, 135–158. pages [↑4](#)
- [57] JOAG-DEV, K. AND PROSCHAN, F. 1983. Negative association of random variables with applications. *The Annals of Statistics*, 286–295. pages [↑5](#)
- [58] KALYANASUNDARAM, B. AND PRUHS, K. R. 2000. An optimal deterministic algorithm for online  $b$ -matching. *Theoretical Computer Science (TCS)* 233, 1, 319–325. pages [↑4](#)
- [59] KARP, R. M., VAZIRANI, U. V., AND VAZIRANI, V. V. 1990. An optimal algorithm for on-line bipartite matching. In *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing (STOC)*. 352–358. pages [↑4](#), [↑21](#)
- [60] KLEINBERG, R. AND WEINBERG, S. M. 2012. Matroid prophet inequalities. In *Proceedings of the 44th Annual ACM Symposium on Theory of Computing (STOC)*. 123–136. pages [↑4](#), [↑27](#)
- [61] KONG, Y. 2022. Are “intersectionally fair” ai algorithms really fair to women of color? a philosophical analysis. In *Proceedings of the 5th Conference on Fairness, Accountability, and Transparency (FAccT)*. 485–494. pages [↑24](#)
- [62] KRENGEL, U. AND SUCHESTON, L. 1978. On semiamarts, amarts, and processes with finite value. *Probability on Banach spaces* 4, 197–266. pages [↑2](#), [↑4](#), [↑16](#)
- [63] KULKARNI, J., LIU, Y. P., SAH, A., SAWHNEY, M., AND TARNAWSKI, J. 2022. Online edge coloring via tree recurrences and correlation decay. In *Proceedings of the 54th Annual ACM Symposium on Theory of Computing (STOC)*. 2958–2977. pages [↑1](#), [↑2](#), [↑14](#), [↑15](#)
- [64] LATTANZI, S., LAVASTIDA, T., MOSELEY, B., AND VASSILVITSKII, S. 2020. Online scheduling via learned weights. In *Proceedings of the 31st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 1859–1877. pages [↑21](#)
- [65] LAVASTIDA, T., MOSELEY, B., RAVI, R., AND XU, C. 2021. Learnable and instance-robust predictions for online matching, flows and load balancing. In *Proceedings of the 29th Annual European Symposium on Algorithms (ESA)*. 59:1–59:17. pages [↑21](#)



- [66] LI, S. AND XIAN, J. 2021. Online unrelated machine load balancing with predictions revisited. In *International Conference on Machine Learning*. 6523–6532. pages [↑21](#)
- [67] MACRURY, C. AND MA, W. 2024. Random-order contention resolution via continuous induction: Tightness for bipartite matching under vertex arrivals. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing (STOC)*. 1629–1640. pages [↑4](#)
- [68] MACRURY, C., MA, W., AND GRAMMEL, N. 2023. On (random-order) online contention resolution schemes for the matching polytope of (bipartite) graphs. In *Proceedings of the 34th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 1995–2014. pages [↑4](#)
- [69] MEHTA, A. 2013. Online matching and ad allocation. *Foundations and Trends® in Theoretical Computer Science* 8, 4, 265–368. pages [↑4](#)
- [70] MEHTA, A., SABERI, A., VAZIRANI, U., AND VAZIRANI, V. 2007. Adwords and generalized online matching. *Journal of the ACM (JACM)* 54, 5, 22. pages [↑4](#)
- [71] MITZENMACHER, M. AND VASSILVITSKII, S. 2020. Algorithms with predictions. *CoRR abs/2006.09123*. pages [↑21](#)
- [72] NUTI, P. AND VONDRÁK, J. 2023. Towards an optimal contention resolution scheme for matchings. In *Proceedings of the 24th Conference on Integer Programming and Combinatorial Optimization (IPCO)*. 378–392. pages [↑4](#)
- [73] PAPADIMITRIOU, C., POLLNER, T., SABERI, A., AND WAJC, D. 2021. Online stochastic max-weight bipartite matching: Beyond prophet inequalities. In *Proceedings of the 22nd ACM Conference on Economics and Computation (EC)*. 763–764. pages [↑2](#), [↑16](#), [↑17](#)
- [74] PATEL, N. AND WAJC, D. 2024. Combinatorial stationary prophet inequalities. In *Proceedings of the 35th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 4605–4630. pages [↑4](#)
- [75] PEMANTLE, R. AND PERES, Y. 2014. Concentration of lipschitz functionals of determinantal and other strong rayleigh measures. *Combinatorics, Probability and Computing* 23, 1, 140–160. pages [↑6](#)
- [76] POLLNER, T., ROGHANI, M., SABERI, A., AND WAJC, D. 2022. Improved online contention resolution for matchings and applications to the gig economy. In *Proceedings of the 23rd ACM Conference on Economics and Computation (EC)*. 321–322. pages [↑4](#)
- [77] QIU, F. AND SINGLA, S. 2022. Submodular dominance and applications. In *Proceedings of the 25th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX)*. 44:1–44:21. pages [↑29](#)
- [78] SABERI, A. AND WAJC, D. 2021. The greedy algorithm is *not* optimal for on-line edge coloring. In *Proceedings of the 48th International Colloquium on Automata, Languages and Programming (ICALP)*. 109:1–109:18. pages [↑1](#), [↑2](#), [↑14](#), [↑15](#), [↑16](#)
- [79] SRINIVASAN, A. 2001. Distributions on level-sets with applications to approximation algorithms. In *Proceedings of the 42nd Symposium on Foundations of Computer Science (FOCS)*. 588–597. pages [↑1](#), [↑3](#), [↑4](#), [↑11](#), [↑12](#), [↑13](#), [↑35](#)

- [80] SRINIVASAN, A. 2007. Approximation algorithms for stochastic and risk-averse optimization. In *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 1305–1313. pages [↑2](#), [↑11](#), [↑12](#), [↑22](#), [↑23](#)
- [81] SWAMY, C. AND SHMOYS, D. B. 2006. Approximation algorithms for 2-stage stochastic optimization problems. *ACM SIGACT News* 37, 1, 33–46. pages [↑2](#)
- [82] SWAMY, C. AND SHMOYS, D. B. 2012. Sampling-based approximation algorithms for multi-stage stochastic optimization. *SIAM Journal on Computing (SICOMP)* 41, 4, 975–1004. pages [↑2](#), [↑22](#)
- [83] TANG, Z. G., WU, J., AND WU, H. 2022. (fractional) online stochastic matching via fine-grained offline statistics. In *Proceedings of the 54th Annual ACM Symposium on Theory of Computing (STOC)*. 77–90. pages [↑1](#), [↑4](#)
- [84] TORRICO, A. AND TORIELLO, A. 2022. Dynamic relaxations for online bipartite matching. *INFORMS Journal on Computing* 34, 4, 1871–1884. pages [↑17](#)
- [85] VAZIRANI, V. V. 2001. *Approximation algorithms*. Vol. 1. Springer. pages [↑8](#)
- [86] WILLIAMSON, D. P. AND SHMOYS, D. B. 2011. *The design of approximation algorithms*. Cambridge university press. pages [↑8](#)