Free-Order Secretary for Two-Sided Independence Systems

Kristóf Bérczi [*]   Vasilis Livanos [†]   José A. Soto [‡]   Victor Verdugo [§]

Abstract

The *Matroid Secretary Problem* is a central question in online optimization, modeling sequential decision-making under combinatorial constraints. Despite significant progress, the *Matroid Secretary Conjecture* – which asks whether every matroid admits a constant-competitive algorithm – remains open. We introduce a *bipartite graph framework* that unifies and extends several known formulations, including the bipartite matching, matroid intersection, and random-order matroid secretary problems. In this model, elements form a bipartite graph between *agents* and *items*, and the objective is to select a matching that satisfies feasibility constraints on both sides, given by two *independence systems*.

We study the *free-order* setting, where the algorithm may adaptively choose the next element to reveal. For *k-matroid intersection*, we leverage a core lemma by (Feldman, Svensson and Zenklusen, 2022) to design an $\Omega(1/k^2)$-competitive algorithm, extending known results for single matroids. Building on this, we identify the structural property underlying our approach and introduce *k-growth systems* – a new class of independence systems that generalize *k*-matchoids and may be of independent combinatorial interest. We establish a generalized core lemma for *k*-growth systems, showing that a suitably defined set of *critical elements* retains a $\Omega(1/k^2)$ fraction of the optimal weight. Using this lemma, we extend our $\Omega(1/k^2)$-competitive algorithm to *k*-growth systems for the edge-arrival model.

We then study the *agent-arrival* model, which presents unique challenges to our framework. We extend the core lemma to this model and then apply it to obtain an $\Omega(\beta/k^2)$-competitive algorithm for *k*-growth systems, where β denotes the competitiveness of a special type of order-oblivious algorithm for the item-side constraint. Finally, we relax the matching assumption and extend our results to the case of *multiple item selection*, where agents have individual independence systems coupled by a global item-side constraint. We obtain constant-competitive algorithms for fundamental cases such as partition matroids and *k*-matching constraints.

We also study the structural role of *k-growth systems* within the hierarchy of *k-systems*. We analyze their closure under key operations, such as parallel extension, restriction and contraction, and derive a new characterization of *k-extendible systems* in terms of contractions of *k-systems*.

---

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

[†]Center for Mathematical Modeling, Universidad de Chile, Santiago, Chile (`vas.livanos@gmail.com`)

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

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

Contents

1   Introduction

The *secretary problem* is one of the classical examples in optimal stopping theory. In the original setup, one must select the best among n candidates arriving in a uniformly random order. No information about the candidates is known in advance. Still, the decision-maker can rank the candidates observed so far and make an immediate and irrevocable decision to accept or reject the current one. The well-known strategy of observing the first $1/e$ fraction of candidates without accepting any, and then selecting the first one who is better than all previously seen candidates, yields an optimal $1/e$ probability of choosing the best one [27, 35, 53].

Recent years have seen a renewed surge of interest in the secretary problem and its extensions, mainly driven by their deep connections to problems in online mechanism design, where agents arrive sequentially and seek to acquire items or services. Beyond this setting, the secretary problem is also closely related to online matching, which encompasses numerous real-world applications, including ride-sharing platforms, online labor markets, advertising auctions, and cloud resource allocation. In most applications, the original setup is generalized to a combinatorial framework, where each arriving element e reveals a nonnegative weight $w(e)$, and the objective is to maximize the expected total weight of the selected elements, subject to the chosen set being independent with respect to a given feasibility constraint.

Typical constraints that appear frequently, both in applications and in the literature, include uniform matroids[1] (also known as cardinality constraints) and partition matroids [44], matching constraints [29, 42, 48, 57], and knapsack constraints [2, 7, 43, 46, 60], among others. Perhaps due to their rich combinatorial structure and the strong performance guarantees they provide for greedy algorithms, matroid constraints have received particular attention in the literature on the secretary problem. Although the *Matroid Secretary Conjecture* [8] – which asks whether there exists a constant-competitive algorithm for every matroid – remains unsolved (the best known competitive ratio is $\Omega(1/\log\log r)$ where r is the rank of the underlying matroid [32, 51]), several important subclasses of matroids admit constant-factor competitive ratios, see, e.g., [9, 42, 48, 69, 70].

In many important practical settings, however, the feasibility constraint cannot be captured by a single matroid. Instead, the chosen set must be independent with respect to multiple matroids defined over the same ground set – in other words, a *matroid intersection* constraint. In this model, Feldman, Svensson, and Zenklusen [34] showed that, if each individual matroid admits an *order-oblivious*[2] constant-competitive algorithm, then one can combine these to obtain a constant-competitive algorithm for the intersection, provided the number of matroids is constant. Moving beyond the classical random-order formulation – where the general matroid secretary conjecture remains unresolved – one interesting variant is the *free-order* model. In this setting, the algorithm is allowed to adaptively choose the next element to reveal its weight at each step. For this model, Jaillet, Soto, and Zenklusen [37, 38] devised an algorithm that selects every element in the optimal-weight basis of a matroid with probability 1/4.

Motivated by both theoretical considerations and real-world applications, we introduce a new *bipartite graph model* for the free-order setting. In this model, we are given a bipartite graph consisting of *agents* A and *items* B, and the objective is to select a matching that satisfies two feasibility constraints: the set of matched agents must belong to a given family $\mathcal{F}_A$, and the

---

[1] For a formal description of matroids see Definition 2.

[2] An order-oblivious secretary algorithm proceeds in two phases: a sampling phase that draws a random subset of elements, observes their weights, and selects none; and a selection phase that processes all remaining elements, with an arrival order that may be adversarial and even adaptive to the observed sample.

set of matched items must belong to a given family $\mathcal{F}_B$. This formulation generalizes bipartite matching secretary, which corresponds to the case where both $\mathcal{F}_A$ and $\mathcal{F}_B$ are free matroids[3], but is considerably more expressive. By varying $\mathcal{F}_A$ and $\mathcal{F}_B$, the model is able to capture a broad range of constraint families, including matroid and matroid intersection constraints, and even the random-order matroid secretary problem within a free-order framework. Moreover, approaching free-order selection through this two-sided matching perspective reveals a close relationship with the classical *online bipartite matching* setting since, for instance, our model can naturally accommodate *agent-arrival* settings.

Running example. To motivate our bipartite graph model, consider the following example. Agents represent facilities providing different services, such as water, electricity, or internet, while items correspond to clients. The weight of an edge $\{a, b\}$ represents the utility obtained if facility a services client b. When both $\mathcal{F}_A$ and $\mathcal{F}_B$ are free, the model reduces to a standard bipartite matching problem; later we show how our framework allows us to handle more complex variants of this setting.

1.1 Overview of our Contributions and Techniques

In this paper, we leverage our new *bipartite graph model* to investigate the free-order secretary problem under both the *edge-arrival* and *agent-arrival* settings. Our results consist of one conceptual and three algorithmic contributions.

***k*-matroid intersection.** We begin with k-matroid intersection. The algorithm of Jaillet, Soto, and Zenklusen [37] provides a natural starting point: sample half of the elements, order the optimal solution within the sample by decreasing weight, and then process the remaining elements in an order according to the span of the prefixes of this sampled optimum. Unfortunately, this approach cannot be generalized to matroid intersection directly due to the following two obstacles: first, the spans of the individual matroids may require completely different orders and second, even with a favorable order, it is a priori unclear why the intersection of the individual optimal sets retains enough weight compared to the global optimum. The latter problem can be resolved using a lemma by Feldman, Svensson and Zenklusen [34]: by sampling a large fraction of the elements, they show that the so-called *greedy-relevant* elements of the second phase retain a decent fraction of the optimum weight – we refer to this as the *core lemma* for matroid intersection (Lemma 1). Equipped with this structural insight, we design an algorithm that extends the approach of [37], by using the union of the individual spans of the matroids. Our algorithm selects every element in the intersection of the optimal sets of the greedy-relevant elements with probability $1/4$ (Theorem 3). More importantly, this algorithm serves as a versatile black-box that applies beyond k-matroid intersection, whenever the underlying constraint family admits an appropriate analogue of the core lemma.

Running example. Suppose now that each facility requires certain raw materials or resources supplied by a central authority in order to operate. For each resource, the subset of facilities that can be served simultaneously forms a gammoid.[4] Hence, the global constraint on the agents can be viewed as a combination of gammoids. Notably, although no constant-factor guarantees are known for the secretary problem with gammoid constraints in the random-order setting, our bipartite graph framework yields constant-factor guarantees even when multiple resource constraints are present.

---

[3]In a free matroid, all subsets of the ground set are independent.

[4]A *gammoid* is a special type of matroid in which a set S of vertices in a given graph is independent if there exist vertex-disjoint paths from S to a designated sink.

k-growth systems. We then generalize this result in two different directions. First, we revisit the core lemma by Feldman, Svensson and Zenklusen. We identify the key structural characteristic that underpins the lemma and, motivated by this property, we introduce a new family of independence systems which we call *k-growth systems*. This new family fits nicely in the existing hierarchy of k-systems: it contains k-matchoids and, in turn, is contained within the class of k-extendible systems. Intuitively, k-growth systems require a stronger k-extendability property, not just for independent sets but for arbitrary sets. Beyond their theoretical appeal, k-growth systems capture several additional constraint families that appear in practice, including bounded-ratio knapsack constraints and stable sets of intersection graphs, thereby providing a unifying abstraction for a wide range of combinatorial settings.

To generalize the core lemma to k-growth systems, we first need to generalize the notion of span from matroids to k-growth systems. We introduce a new "span-like" operator, the *primitive hull*, which, for any set X, consists of X together with all elements e that form a circuit with X. We then generalize our core lemma to k-growth systems: we identify a set of *critical* elements, those elements e that do not lie in the primitive hull of the greedy-relevant elements of higher weight. We prove that, for every k-growth system, the set of critical elements preserves a $\Omega\left(1/k^2\right)$ fraction of the weight of the optimal basis (Lemma 5, Theorem 4). Substituting our new core lemma into the black-box algorithm from Theorem 3 immediately yields a $\Omega\left(1/k^2\right)$-competitive algorithm for the free-order secretary problem over any k-growth system (Theorem 5). In fact, we establish a stronger result; we introduce the notion of *combination* of systems, which generalizes the classical k-matchoid construction, allowing the individual components of the overall system to be arbitrary constraint families rather than only matroids. We prove that any combination of growth systems remains a k-growth system for an appropriate k, and show that both our generalized core lemma and the resulting algorithm extend seamlessly to combinations of growth systems, further broadening the applicability of our framework.

Running example. In addition to resource constraints, each facility may also incur an operational cost, while the government operates under a fixed budget for servicing clients. This naturally induces a knapsack constraint on the set of facilities. The class of growth systems is particularly appealing here, as it accommodates the combined feasibility constraint of (bounded-ratio) knapsack and matroid constraints.

Agent-arrival setting. The strength of the bipartite graph model becomes evident when we turn to the *agent-arrival* setting. Unfortunately, the results established for edge arrivals do not translate directly, as the agent-arrival setting introduces three main challenges. The first difficulty arises from the loss of partial observability. In the edge-arrival model, sampling a fraction of edges reveals information about some edges incident to each agent. Under agent arrivals, "calling" an agent reveals all of its incident edges simultaneously, and the agent cannot be matched later on. To address this, we switch from reasoning about all *greedy-relevant edges* to a single *top-relevant edge* per agent. This is the heaviest greedy relevant edge incident to that agent. Since the matching constraint allows at most one edge per agent, we can canonically associate each agent with its top-relevant edge. This perspective lets us extend the core lemma for k-growth systems to the agent-arrival setting, showing that the set of top-relevant edges still retains an $\Omega\left(1/k^2\right)$ fraction of the optimal basis weight (Lemma 6).

The second obstacle arises when attempting to use our general algorithm from Theorem 3 in a black-box manner. The algorithm requires querying agents in an order determined by the primitive

hull on the agent side, ensuring that agents whose greedy-relevant edges are critical with respect to $\mathcal{F}_A$ are considered with constant probability. This order, however, provides no guarantee on the item side – in fact, the selected edges may form a dependent set in $\mathcal{F}_B$. To address this, we restrict $\mathcal{F}_B$ to systems that admit a special *core-selecting order-oblivious* algorithm, which preserves its competitive guarantee even when, after a random sampling phase, the remaining elements are presented in an arbitrary order.

This, however, reveals the third, and most subtle, obstacle in our approach: When we sample agents, we simultaneously observe all edges incident to those agents. From the perspective of the item side, this does not induce a uniformly random sample of items or even of item-adjacent edges, since edges corresponding to the same agent are inherently correlated. Notice, however, that after our first sample, each agent contributes at most one greedy-relevant edge – their top greedy-relevant edge, if one exists. Therefore, to overcome this final obstacle, after the first sample, we restrict our ground set to the (unknown) set of greedy-relevant edges. We can then perform a second, independent sampling phase, ensuring that the sampled greedy-relevant edges form a uniform random subset of the new ground set; for more details on this, see Section 5. Finally, applying the algorithm from Theorem 3 as a black box (which introduces a third and final sampling phase) yields an $\Omega\left(\beta/k^2\right)$-competitive algorithm for the agent-arrival model, where β is the competitive ratio of the order-oblivious algorithm for $\mathcal{F}_B$ (Theorem 7). This result combines the structural insights of the edge-arrival model with the algorithmic challenges of agent arrivals, and is thus the main algorithmic result of the paper.

Running example. Suppose that the items actually represent contracts between agencies and clients, with each facility responsible for servicing one contract. The selected contracts thus have to form a matching between agencies and clients, corresponding to the item-side constraint $\mathcal{F}_B$. Because this constraint admits a core-selecting, order-oblivious algorithm, our bipartite graph framework yields constant-factor guarantees for free-order agent arrivals even in this setting.

Beyond matching constraints. We next consider settings in which each agent may select multiple items. Extending the core lemma to this setting remains challenging – even for the intersection of two matroids – so we take a different route and focus on the case of *unrelated agents*. Here, each agent possesses their own, separate independence system governing the feasible subsets of its incident edges, and feasibility across agents is coupled only through a global item-side constraint. In this setting, agents arrive sequentially, reveal the weights of all adjacent edges, and the algorithm must irrevocably select a feasible subset for each agent. Interestingly, this model captures a rich class of problems in online selection, including the *secretary problem with groups* of Korula and Pál [48], as well as variants where items represent edges in low-degree graphs or hypergraphs and feasible sets correspond to matchings [42, 57]. We design a constant-competitive algorithm (Theorem 8) for the agent-arrival setting when the item-side constraint is sufficiently simple, for example, a partition matroid or a k-matching constraint, demonstrating that our bipartite free-order framework extends naturally to heterogeneous and structurally diverse environments. It is worth noting that this approach leads to an $\Omega(1/k^2)$-competitive algorithm for combinations of systems admitting a so-called k-directed certifiers [57], which includes combinations of several classes of matroids.

Running example. Finally, when the constraints governing different facilities are independent of one another, we can allow each facility to serve multiple contracts while the contract side still satisfies a matching constraint. Our framework continues to provide constant-factor guarantees in this more general setting.

6
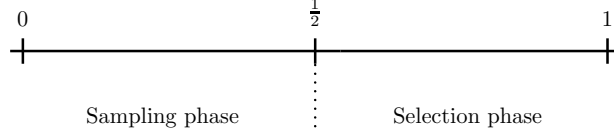
(a) Illustration of the algorithm of Jaillet, Soto, and Zenklusen for a single matroid (Theorem 2), with a single sampling phase in $[0, 1/2)$ and a selection phase in $[1/2, 1]$.
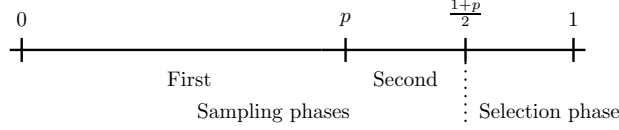
(b) Illustration of our algorithm for combinations of k-growth systems in the edge-arrival setting (Theorem 5), with a first sampling phase in $[0, p)$ to define the greedy-relevant elements in subsequent phases, a second sampling phase in $[p, (1 + p)/2)$ used by our black-box algorithm (Theorem 2) to decide the order in which to "call" the greedy-relevant elements, and a selection phase in $[(1 + p)/2, 1]$. Here, $p = \sqrt{1 - 1/(k + 1)}$ for arbitrary k-growth systems and $p = 1 - 1/(2k)$ for k-matchoids.
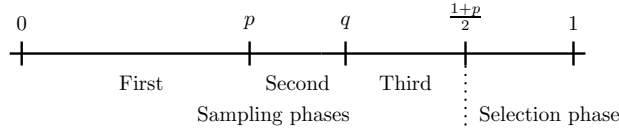
(c) Illustration of our algorithm for combinations of k-growth systems in the agent-arrival setting (Theorem 7), with a first sampling phase in $[0, p)$ to define the greedy-relevant elements in subsequent phases, a second sampling phase in $[p, q)$ required by the corresponding order-oblivious algorithm for $\mathcal{F}_B$, a third sampling phase in $[q, (1 + q)/2)$ used by our black-box algorithm (Theorem 2) to decide the order in which to "call" the greedy-relevant agents, and a selection phase in $[(1 + q)/2, 1]$. Here, $p = \sqrt{1 - 1/(k + 1)}$ for arbitrary k-growth systems and $p = 1 - 1/(2k)$ for k-matchoids, and q depends on the sampling probability of the order-oblivious algorithm for $\mathcal{F}_B$.

Figure 1: A visual explanation of the sampling and selection phases of each of our algorithms.

The k-systems hierarchy. Finally, we examine the position of k-growth systems within the broader k-systems hierarchy and study their relationship to other well-known classes. As discussed earlier, we show that every k-matchoid is a k-growth system and present an example of a k-growth system that is not k-circuit bounded. We also show that the class of k-growth systems is, in turn, contained within the k-extendible systems. We further conjecture that these two classes actually coincide, which, if true, would yield a new structural characterization of k-extendible systems and potentially enable new algorithmic techniques for this important class across a range of online and combinatorial optimization settings. We illustrate the expressiveness of k-growth systems through several examples, including bounded-ratio knapsack constraints and stable sets of interval graphs. Moreover, we show that all these classes, except for k-systems, are closed under parallel extension, restriction, and contraction. More importantly, we provide a new characterization of k-extendible systems as exactly those k-systems whose contractions remain k-systems.

1.2 Related Work

Online Matchings and Combinatorial Assignment. The baseline is unweighted one-sided bipartite online matching, where the *Ranking* algorithm achieves the optimal ratio $1 - 1/e$ [41]. Secretary-style generalizations introduce weights and random arrival, leading to two standard models in general graphs: *vertex arrival* (upon a vertex's arrival, incident edges to previously seen vertices are revealed and must be acted on immediately) and *edge arrival* (edges appear one by one, each

requiring an irrevocable decision). Early secretary formulations on graphs and hypergraphs are due to Korula and Pál [48]. For general graphs, Ezra, Feldman, Gravin, and Tang [29] established tight bounds of 5/12 for vertex arrival and an exponential-time 1/4 for edge arrival, with the latter later made polynomial-time in [57]. In bipartite weighted settings with budgets (AdWords), online allocation under random order was modeled and analyzed by Mehta, Saberi, Vazirani, and Vazirani [58], while related secretary-style approaches for weighted bipartite matching and its extensions were developed by Kesselheim, Radke, Tönnis, and Vöcking [42]. Beyond graphs, *online submodular welfare* captures diminishing-returns valuations over arriving items and admits a 1/2-competitive greedy algorithm, which is optimal among polynomial-time algorithms under standard assumptions [40]. A unifying abstraction for these problems is *online combinatorial assignment* over independence systems, where agents arrive with (possibly multi-weight) preferences over a fixed feasibility family (e.g., k-hypergraph matchings, matroids, matroid intersections, matchoids), and the algorithm assigns at most one feasible element per agent while maintaining independence [57].

Prophet Inequalities and Online Contention Resolution Schemes. Secretary problems are closely connected to *prophet inequalities*, a model that has attracted significant attention due to its applications in mechanism design and auction theory [3, 4, 5, 6, 11, 12, 16, 17, 19, 20, 21, 25, 26, 30, 36, 71]. In the prophet inequality framework, the element weights $w(e)$ are drawn independently from known distributions, and the arrival order is typically adversarial. The classical single-item result of Krengel, Sucheston, and Garling [49, 50, 67] established the optimal 1/2-competitive ratio, later extended to matroid constraints by Kleinberg and Weinberg [45], who also obtained a $1/(4k - 2)$-competitive ratio for intersections of k matroids. The best current upper bound was recently improved to slightly better than $\Theta(1/\sqrt{k})$ [68]. Subsequent work has explored various arrival models, usually achieving stronger guarantees. In the *free-order* setting, where the algorithm can choose the order of arrivals, early progress was made by Beyhaghi, Golrezaei, Paes Leme, Pál and Sivan [10] and Peng and Tang [63], with the current best competitive ratio of 0.7258 due to Bubna and Chiplunkar [14]. Determining the optimal competitive ratio in this setting remains a central open question in optimal stopping theory.

Secretary problems are also closely related to *online contention resolution schemes* (OCRSs), which play a central role in Bayesian mechanism design and posted-price mechanisms [18, 33, 65, 66]. OCRSs share many conceptual similarities with prophet inequalities and can be combined in a black-box way, both properties that align with several ideas developed in this paper. Feldman, Svensson, and Zenklusen [33] introduced the first OCRSs for matroid, matching, and knapsack constraints, achieving a $1/(e(k + 1))$ guarantee for k-matroid intersections. Lee and Singla [52] later obtained optimal guarantees for matroids under a slightly stronger assumption of a weaker adversary. Despite this progress, determining the optimal OCRS guarantees for matroids, matchings, and knapsacks remains an active area of research [13, 28, 29, 54, 55, 56, 61, 64]. Recently, Dughmi [22, 23, 24] established a deep and beautiful equivalence: obtaining constant-factor OCRSs for matroid constraints in the random-order setting – for any correlated distribution for which such guarantees exist offline – would imply the matroid secretary conjecture.

The k-system Hierarchy. The notion of a k-system goes back to the classical analysis of greedy algorithms for independence systems by Jenkyns [39] and by Korte and Hausmann [47], who characterized approximation guarantees in terms of the rank quotient, that is, the ratio of the largest and smallest bases of a subset. This concept was later adopted under the name p-system in submodular maximization [15]. Jenkyns [39] also implicitly introduced the notion later referred

to as k-circuit-bounded systems, although no explicit name was proposed in that work. The class of k-extendible systems was introduced by Mestre [59], who identified it as a natural subclass of k-systems for which the greedy algorithm achieves a $1/k$-approximation for linear objectives, thereby fitting into the hierarchy

$$k\text{-matroid intersection} \subseteq k\text{-matchoid} \subseteq k\text{-circuit bounded} \subseteq k\text{-extendible} \subseteq k\text{-system}.$$

However, most previous work on k-systems has focused on their algorithmic applications, mainly in the analysis of greedy and submodular maximization algorithms. As a result, these systems are well understood from a performance-guarantee perspective but much less from a combinatorial one. In particular, their basic structural properties have not been studied in a systematic way.

1.3   Organization

We start with Section 2, where we describe our bipartite graph model and present some necessary preliminaries and definitions. In Section 3 we describe our algorithm for k-matroid intersection, presented in a black-box way as it will be used for our later results as well. In Section 4 we define our new independence class of k-growth systems and present our generalized Core lemma for combinations of k-growth systems. Section 5 contains our results for the agent-arrival model. We then extend the agent-arrival setting to allow multiple items per agent when the agents' constraints are unrelated in Section 6. We conclude in Section 7 by relating k-growth systems to the other classes in the k-systems hierarchy and describe several interesting properties of the hierarchy classes.

2   The Bipartite Graph Model

In this section, we define our proposed bipartite graph model for free-order secretary problems. We begin with the basic notation; all further terminology is introduced as needed. We denote the sets of *reals* and *integers* by $\mathbb{R}$ and $\mathbb{Z}$, and add the subscripts $+$ when referring to nonnegative values. For a positive integer k, we write $[k] := \{1, \ldots, k\}$. Given a ground set S, a subset $X \subseteq S$, and an element $y \in S$, we abbreviate $X \setminus \{y\}$ and $X \cup \{y\}$ as $X - y$ and $X + y$, respectively, and write $\{x\}$ simply as x when no confusion arises. For a $p \in [0, 1]$ and a set S of elements, we use $S(p)$ to denote a random subset of S where every element appears independently with probability p. For a function $w \colon S \to \mathbb{R}_+$, we use $w(X) := \sum_{x \in X} w(x)$. If $G = (V, E)$ is a graph, and $S \subseteq V$ is a subset of vertices, we use $\delta(S)$ to denote the set of edges with one endpoint in S and one endpoint outside S. Let us first recall the definition of independence systems and matroids.

Definition 1 (Independence System). An *independence system* $\mathcal{F} = (S, \mathcal{I})$ consists of a finite set S and a family of *independent sets* $\mathcal{I} \subseteq 2^S$ satisfying the following two properties:

(I1) $\mathcal{I}$ is non-empty,

(I2) $\mathcal{I}$ is downward closed, that is, $I \in \mathcal{I}$ implies $J \in \mathcal{I}$ for all $J \subseteq I$.

Note that conditions (I1) and (I2) together implies $\emptyset \in \mathcal{I}$. Any set $X \notin \mathcal{I}$ is called *dependent*. The inclusion-wise minimal dependent sets are called *circuits*, and a circuit of size 1 is called a *loop*. For any subset $X \subseteq S$, the inclusion-wise maximal independent subsets of X are called *bases* of X.

Examples of independence systems include matroids, matchings in hypergraphs, matchoids, k-extendible systems, and k-systems (see Section 7 for details and definitions). Below, we provide a few basic definitions for matroids and refer the reader to [62] for an extensive treatment.

Definition 2 (Matroid). A matroid $M = (S, \mathcal{I})$ is an independence system that satisfies the following *augmentation* property:

$$\forall\, I, J \in \mathcal{I},\; |I| < |J| \implies \exists\, e \in J \setminus I \text{ such that } I \cup \{e\} \in \mathcal{I}.$$

This property is equivalent to the statement that, for every $X \subseteq S$, all maximal independent subsets of X (the bases of X) have the same cardinality. With this, one defines the rank of X, denoted by $r(X)$, as the common size of the bases of X. The closure (or span) of X is

$$\mathrm{span}(X) \;=\; X \;\cup\; \{e \in S \setminus X :\; r(X \cup \{e\}) = r(X)\}.$$

Throughout the paper, we consider independence systems equipped with a nonnegative weight function $w : S \to \mathbb{R}_+$ on the ground set, with distinct values. This standard assumption avoids tie-breaking issues and lets us focus on structural and algorithmic aspects. For $X \subseteq S$, let $\mathrm{OPT}(X)$ be a maximum-weight basis of X with respect to w. In general independence systems, $\mathrm{OPT}(X)$ need not be unique even under distinct weights; one can fix uniqueness via a lexicographic rule. In matroids, however, the maximum-weight basis under distinct weights is unique.

Definition 3 (Weighted greedy on X). Given $X \subseteq S$, the *weighted greedy algorithm* processes the elements of X in non-increasing order of w and builds a set G starting from $\emptyset$, adding an element e when $G \cup \{e\} \in \mathcal{I}$. We denote its output by $\mathrm{Greedy}(X)$.

The following proposition, by Korte and Hausmann [47], relates the weighted greedy algorithm to k-systems.

Proposition 1. *For matroids,* $\mathrm{Greedy}(X)$ *is the unique maximum-weight basis of X under distinct weights, i.e.,* $\mathrm{Greedy}(X) = \mathrm{OPT}(X)$. *More generally, an independence system is a k-system if and only if, for every $X \subseteq S$ and every nonnegative weight function w, the greedy algorithm yields a k-approximate maximum-weight basis, i.e.,*

$$w(\mathrm{Greedy}(X)) \geq \frac{1}{k} w(\mathrm{OPT}(X)).$$

2.1 Free-order Bipartite Secretary Model

Let $G = (A \cup B, E)$ be an undirected bipartite graph. We refer to A as the "agent" side and B as the "item" side. Let $w \colon E \to \mathbb{R}_+$ be a nonnegative injective weight function over the edge set. Furthermore, let $\mathcal{F}_A = (A, \mathcal{I}_A)$ and $\mathcal{F}_B = (B, \mathcal{I}_B)$ denote independence systems over the vertices on the agent and item side, respectively. In the *bipartite secretary model*, we are given G, $\mathcal{F}_A$, and $\mathcal{F}_B$ as input.[5] We say that a set of edges $F \subseteq E$ is a *feasible matching* if

(*i*) F is a matching in G.

(*ii*) The set of agents incident to F, $\{a \in A \mid \delta(a) \cap F \neq \emptyset\}$, is independent in $\mathcal{F}_A$.

(*iii*) The set of items incident to F, $\{b \in B \mid \delta(b) \cap F \neq \emptyset\}$ is independent in $\mathcal{F}_B$.

---

[5]Similarly to matroid algorithms, we assume that independence systems are given via an independence oracle that, for a given set X, answers yes or no depending on whether X is independent or not.

If an edge $\{a, b\}$ is in F we say that item b is assigned to agent a. The weight function w over the edges is initially unknown and revealed over sequential rounds. An algorithm for this problem starts with an empty set of edges ALG. In round i, one or more edges may be revealed, and at the end of the round, the algorithm must immediately and irrevocably decide to add a subset F_i of the edges revealed in that round, to the current solution ALG , under the condition that ALG is always a feasible matching. The goal is to select a maximum-weight feasible matching. We focus on two versions of the model, differing in how rounds are defined:

(EA) **Edge Arrival:** at each round, a single new edge reveals its weight.

(AA) **Agent Arrival:** at each round, the set $\delta(a)$ of all edges incident to some agent $a \in A$ reveal their weights simultaneously.

In the *free-order* setting, the algorithm adaptively chooses the next object to be revealed: the next edge in (EA) or the next agent in (AA). This formulation is remarkably powerful: it generalizes several settings previously studied in the literature, even when $\mathcal{F}_A$ and $\mathcal{F}_B$ are simple constraints such as matroids. We briefly discuss these connections below.

2.2 Relation to Existing Models

We begin by noting that the free-order edge-arrival variant is exactly the *free-order secretary problem* over the independence system of feasible matchings induced by $(G, \mathcal{I}_A, \mathcal{I}_B)$. Consequently, rather than analyzing this specific instance, we develop our results (and state existing ones) for the free-order model on general independence systems. Let us focus on the case when both $\mathcal{F}_A$ and $\mathcal{F}_B$ are matroids. We distinguish between different cases according to their respective matroid classes.

$\mathcal{F}_A$ **and** $\mathcal{F}_B$ **are both free matroids.** In this setting, the edge arrival model coincides with the bipartite matching secretary problem, for which there exists a 1/4-competitive (polynomial-time) algorithm even in random-order [57]. For the agent arrival model, on there is an optimal $1/e$-competitive algorithm in random-order [42].

$\mathcal{F}_A$ **is a general matroid and** $\mathcal{F}_B$ **is a free matroid.** For the specific case in which the graph G itself is a matching, both the edge and the agent-arrival model coincide with the free-order matroid secretary problem, for which there exists a 1/4-competitive algorithm [37, Theorem 2]. For general graphs, neither version has been studied before.

$\mathcal{F}_A$ **is a free matroid and** $\mathcal{F}_B$ **is a general matroid.** Here, the edge arrival model is the same as in the above setting where $\mathcal{F}_A$ is a general matroid and $\mathcal{F}_B$ is the free matroid. Surprisingly, the agent arrival model is equivalent to the classical matroid secretary problem under a uniformly random arrival order. To see this, let G be a complete bipartite graph. The adversary can choose a random perfect matching X of the graph and assign non-zero weights only to edges in X. Every time we observe the weights of edges incident to an agent in any order, we only observe one edge with positive weight, and we can only select it at that moment. Since all agents look the same before they arrive, the order in which they arrive does not matter, and we are forced to select it uniformly at random. Therefore, we recover the classical matroid secretary problem.

$\mathcal{F}_A$ **and** $\mathcal{F}_B$ **are both general matroids.** Note that the set of edges we can choose at the same time actually forms an independent set in the intersection of $\mathcal{F}_A$ and $\mathcal{F}_B$. To see this, consider the following standard construction. Let $M_A = (E, \mathcal{J}_A)$ be the matroid where, for each agent

$a \in A$, all edges in $\delta(a)$ are treated as parallel copies of the same element in $\mathcal{F}_A$. This way, each agent a can be matched to at most one item e and the constraint for $\mathcal{F}_A$ is satisfied. Using the same approach, define another matroid $M_B = (E, \mathcal{J}_B)$ using the item side. By construction, any feasible matching is an independent set that belongs to both M_A and M_B. In the edge arrival model, no constant-competitive guarantee was known for this setting prior to our work. We obtain a 1/64-competitive algorithm (Corollary 1) for this setting.

In the agent-arrival model, this framework generalizes the case where $\mathcal{F}_A$ is the free matroid and $\mathcal{F}_B$ is an arbitrary matroid, and is therefore at least as hard as the classical matroid secretary problem under a uniformly random order. Nevertheless, we will see that if the matroid constraint is simple enough (see Section 5) we can achieve a constant competitive algorithm for this setting too.

2.3 Additional definitions

Competitive ratio. Let $(S, \mathcal{I})$ be an independence system and let $w : S \to \mathbb{R}_+$ be injective (all weights distinct and positive). Fix a maximum-weight basis $\mathrm{OPT} = \mathrm{OPT}(S) \in \mathcal{I}$ (for matroids with injective weights, this basis is unique, so the choice for that case is canonical). We say an algorithm for any of the considered variants of the secretary problem is α-*probability-competitive* if, for this fixed OPT, every $e \in \mathrm{OPT}$ is included in the algorithm's output $\mathrm{ALG} \in \mathcal{I}$ with probability at least α. We say the algorithm is α-*utility-competitive* if the output satisfies $\mathbb{E}[w(\mathrm{ALG})] \geq \alpha\, w(\mathrm{OPT})$. Note that α-probability-competitiveness implies α-utility-competitiveness, since

$$\mathbb{E}[w(\mathrm{ALG} \cap \mathrm{OPT})] = \sum_{e \in \mathrm{OPT}} w(e) \Pr[e \in \mathrm{ALG}] \geq \alpha \cdot w(\mathrm{OPT}).$$

Combinations of independence systems. In most of our results we work with independence systems obtained from simpler ones via an operation we call *combination*, which strictly extends the usual *intersection*. Let $\{(S_j, \mathcal{I}_j)\}_{j=1}^M$ be independence systems. Their *combination* is the system $(S, \mathcal{I})$ where

$$S = \bigcup_{j=1}^M S_j \qquad \text{and} \qquad \mathcal{I} = \{\, I \subseteq S : \forall j \in [M],\ I \cap S_j \in \mathcal{I}_j \,\}.$$

Note that if all S_j coincide, the combination reduces to the intersection $\bigcap_{j=1}^M \mathcal{I}_j$ on that common ground set.

Example: Matchings and k-matchoids. For a graph $G = (V, E)$, the family of matchings $\mathcal{M} = (E, \mathcal{I})$ with $\mathcal{I} = \{\, F \subseteq E : \forall v \in V,\ |\delta(v) \cap F| \leq 1 \,\}$ is exactly the combination of the rank-1 uniform matroids at each vertex, $\mathcal{M}_v = (\delta(v), \mathcal{I}_v)$ with $\mathcal{I}_v = \{\, F_v \subseteq \delta(v) : |F_v| \leq 1 \,\}$. Each edge of E belongs to exactly two components (those of its endpoints). More generally, k-matchoids are exactly combinations of matroids in which each element belongs to at most k components (see Section 7); these strictly generalize k-matroid intersections.

Example: Generalized Assignment Problem (GAP). Let A be agents, B jobs, and $E = A \times B$, where (a, b) means assigning job b to agent a. Each agent a has time budget $T_a \geq 0$, and processing b on a consumes $t_a(b) \geq 0$. For each $a \in A$, define a knapsack system on $\delta(a) = \{(a, b) : b \in B\}$ by $\mathcal{I}_a = \{F \subseteq \delta(a) : \sum_{(a,b) \in F} t_a(b) \leq T_a\}$. For each $b \in B$, define a rank-1 uniform matroid on $\delta(b) = \{(a, b) : a \in A\}$ by $\mathcal{I}_b = \{F \subseteq \delta(b) : |F| \leq 1\}$. The feasible assignments for the GAP problem are exactly those in the *combination* of the per-agent knapsacks $(\delta(a), \mathcal{I}_a)$ and the per-job rank-1 uniform matroids $(\delta(b), \mathcal{I}_b)$.

Algorithm 1: Free-order Matroid Secretary [37]

> **Input:** A matroid $M = (S, \mathcal{I})$.
> **Output:** An independent set ALG $\in \mathcal{I}$.

1 ALG $\leftarrow \emptyset$
2 for *each $e \in S$* **do** Choose t_e independently and uniformly from $(0, 1)$
3 $S_1 \leftarrow \{e \mid t_e \in (0, 1/2)\}$
4 $S_2 \leftarrow \{e \mid t_e \in [1/2, 1)]\}$
5 Observe (without accepting) all elements in S_1
6 unseen $\leftarrow S_2$ // The set of elements not yet revealed
7 Sort the elements of S_1 in decreasing order of weights as $e_1, \ldots, e_m$
8 for $j = 1$ *to* m **do**
9 | Let $Q = \mathbf{unseen} \cap \mathrm{span}(\{e_1, \ldots, e_j\})$
10 | **for** *each $f \in Q$ in uniformly random order* **do**
11 | | **unseen** $\leftarrow$ **unseen** $- f$ // Reveal f
12 | | **if** $w(f) > w(a_j)$ *and* ALG $+ f \in \mathcal{I}$ **then** ALG $\leftarrow$ ALG $+ f$
13 | **end**
14 end
15 $Q \leftarrow$ **unseen**
16 for *each $f \in Q$ in uniformly random order* **do**
17 | **if** ALG $+ f \in \mathcal{I}$ **then** ALG $\leftarrow$ ALG $+ f$
18 end
19 return ALG

---

3 Free-Order Secretary Problem on k-Matroid Intersection

In this section, we present our algorithm for k-matroid intersection. Before that, we briefly recall the 1/4-competitive free-order matroid secretary algorithm of Jaillet, Soto, and Zenklusen [37, 38].

3.1 Warm-up: Free-Order Matroid Secretary

Let $M = (S, \mathcal{I})$ be a matroid given in advance, and let $w \colon S \to \mathbb{R}_+$ be an unknown positive weight function, where, without loss of generality, all elements have distinct weights. For the sake of completeness and to motivate future extensions, we include a proof of the following result that appeared in [37, 38]; the algorithm is presented as Algorithm 1.

Theorem 2 (Jaillet, Soto, Zenklusen [37, 38])**.** *Algorithm 1 is 1/4-probability competitive for the free-order matroid secretary problem.*

Proof. Let OPT be the unique maximum weight basis of S and fix $f \in \mathrm{OPT}$. Let h_1 be the heaviest element in $S_1 - f$ such that f is spanned by the elements of higher weight than h_1 in $S_1 - f$. In other words,

$$h_1 \coloneqq \arg\max \left\{ w(h) \mid h \in S_1 - f \text{ such that } f \in \mathrm{span}(\{g \in S_1 - f \mid w(g) > w(h)\}) \right\}.$$

Similarly, let h_2 be the heaviest element in $S_2 - f$ such that f is spanned by the elements of higher weight than h_2 in $S_2 - f$. In other words,

$$h_2 \coloneqq \arg\max \left\{ w(h) \mid h \in S_2 - f \text{ such that } f \in \mathrm{span}(\{g \in S_2 - f \mid w(g) > w(h)\}) \right\}.$$

13

If there are no such elements, we set $h_1 = \perp$ and $h_2 = \perp$, respectively, with $w(\perp) = 0$.

Since $S_1 - f$ and $S_2 - f$ are identically distributed, we have that $\Pr[w(h_1) \geq w(h_2)] = 1/2$, regardless of whether one or both of h_1 and h_2 are $\perp$. Next, we condition on the independent event that $f \in S_2$, and conclude that both $w(h_1) \geq w(h_2)$ and $f \in S_2$ occur together with probability at least $1/4$. In what follows, we condition on these two events occurring. Since $f \in \text{OPT}$, we know that $f \notin \text{span}(\{g \in S \mid w(g) > w(f)\})$. Thus, $w(f) > w(h_1) \geq w(h_2)$.

Recall from the description of the algorithm that S_1 is ordered in decreasing order of weights as $e_1, \ldots, e_m$. Assume that h_1 is the j^*-th highest weight element of S_1, that is, $h_1 = e_{j^*}$, where we set $j^* = m + 1$ if $h_1 = \perp$. By construction, f is revealed by the algorithm precisely in iteration j^*, when it first enters the set Q. Therefore, at the moment when f is revealed, f satisfies both $w(f) > w(e_{j^*}) = w(h_1)$ and $\text{ALG} + f \in \mathcal{I}$, and thus f is included in ALG. We thus conclude that for every $f \in \text{OPT}$, $\Pr[f \in \text{ALG}] \geq 1/4$. □

Theorem 2 shows that, for a single matroid, it suffices to use half of the elements as a sample to determine both an order in which to reveal the remaining elements and thresholds $w(e_j)$ for those called in the j-th group, in order to obtain a constant-factor competitive algorithm. The algorithm then processes the elements in this order and greedily selects those whose value exceeds their threshold. Interestingly, the order in which elements are revealed depends on the *span* of prefixes of the sampled set, that is, $\{e_1, \ldots, e_j\}$, when the elements are sorted by weight. Our generalization to k-matroid intersection systems builds directly on this observation.

3.2   Obstacles and Workarounds

Next, we consider the k-matroid intersection setting. Let $M_i = (S, \mathcal{I}_i)$ be a matroid for $i \in [k]$, and let $M = (S, \mathcal{I})$, with $\mathcal{I} = \cap_{i=1}^k \mathcal{I}_i$, be their intersection. When we try to generalize Algorithm 1 to the intersection of k matroids, two main obstacles arise. First, the spans of the prefixes $\{e_1, \ldots, e_j\}$ across the matroids can be completely unrelated. In fact, the best order to reveal the non-sampled elements for one matroid may even be the reverse of the best order for another. Second, even if we had a favorable order, there is no hope for a constant-factor probability-competitive algorithm. This is because matroids satisfy a minor-optimality condition that does not extend to matroid intersection. Specifically, for a single matroid, if $f \in \text{OPT}(S)$, then $f \in \text{OPT}(X)$ for any subset $X \subseteq S$ containing f. For matroid intersection, however, this monotonicity condition can fail; $\text{OPT}(S)$ may contain f, but removing a single element $e \in S - f$ can lead to $\text{OPT}(S - e)$ omitting f entirely. Thus, we have to rely on utility-competitiveness instead.

Union-Span and Critical Elements. To circumvent the first difficulty, a natural approach is to use the *union-span*: for each prefix $\{e_1, \ldots, e_j\}$, we "call" all unseen elements that are in the union of all individual matroid span functions. Notice that the elements that are in the intersection of the optimum matroid bases of all individual matroids are exactly those elements not union-spanned by elements in S of higher weight; we call these the *critical* elements.

A natural first idea is to use Algorithm 1 to select each critical element with probability $1/4$. However, even for simple matroid intersection systems, such as bipartite matchings, the set of critical elements may have insufficient weight relative to the optimum independent set. To see this, consider the following bipartite matching system over a simple path $e_1, \ldots, e_\ell$ with weights $1 + \varepsilon > w(e_1) > \cdots > w(e_\ell) = 1$. Note that, for each $i \geq 2$, every element e_i is *union-spanned* by e_{i-1}, so e_1 is the only critical element. Therefore, even though the weight of a maximum matching is roughly $\ell/2$, the only critical element has weight 1. This shows that, in general, the set of critical

elements may have arbitrary low weight compared to the optimum basis. Given this, one might expect that the issue could be resolved by considering the set of critical elements with respect to a random subset of S instead of S itself. This approach fails as well, however: for instance, there may be many parallel elements with the same weight, or elements that are loops in some matroids but not in others.

Greedy-Relevant Elements. To address the second issue mentioned above, we use a different set, first defined by Feldman, Svensson and Zenklusen [34].

Definition 4 (Greedy-Relevant Elements). *Let $Y \subseteq S$. We say that an element $y \in S \setminus Y$ is greedy-relevant with respect to Y if y belongs to the output of the weighted-greedy algorithm on $Y + y$. We use $\mathrm{Rel}(Y)$ to denote the set of greedy-relevant elements of Y. In other words,*

$$\mathrm{Rel}(Y) = \{y \in S \setminus Y \mid y \in \mathrm{Greedy}(Y + y)\}.$$

Feldman, Svensson and Zenklusen [34] showed that if every element of S is contained in Y with probability p for some large enough $p > 0$, then the set of critical elements of $\mathrm{Rel}(Y)$ has high enough weight compared to the weight of the set $\mathrm{Greedy}(Y)$ that the weighted-greedy algorithm selects on Y. More precisely, they showed the following lemma.

Lemma 1 (Core Lemma for Matroid Intersection). *Let $(S, \mathcal{I})$ be a k-matroid intersection system and $Y = Y(p)$ be a random set that contains each element of S independently with probability p for some $p \in [0, 1]$. Then*

$$\mathbb{E}\left[w\left(\bigcap_{i=1}^{k}(\mathrm{OPT}_i(\mathrm{Rel}(Y)))\right)\right] \geq \mathbb{E}\left[\frac{(1 - (1 - p)k))(1 - p)}{p} w(\mathrm{Greedy}(Y))\right]$$

$$\geq \frac{(1 - (1 - p)k)(1 - p)}{k} w(\mathrm{OPT}(S)),$$

where OPT_i represents the optimum weight basis in the i-th matroid, and OPT is the optimum weight basis of the k-matroid intersection system. Choosing $p = 1 - \frac{1}{2k}$ yields

$$\mathbb{E}\left[w\left(\bigcap_{i=1}^{k}(\mathrm{OPT}_i(\mathrm{Rel}(Y)))\right)\right] \geq \frac{1}{4k^2} w(\mathrm{OPT}(S)).$$

Hull and Core. Given Lemma 1, we are almost ready to describe our algorithm for k-matroid intersection. The main difference from the algorithm for a single matroid is that we first sample a relatively large set of elements Y in order to identify a good subset $\mathrm{Rel}(Y)$ of greedy-relevant elements, and then execute the free-order algorithm on $\mathrm{Rel}(Y)$. We note that this algorithm can be applied in a black-box manner to more general independence systems, assuming that one has appropriate notions of *critical* elements and what it means for an element to be *spanned* by a set. We will make use of this observation in later sections; for this reason, the algorithm is presented in its general form.

To describe our algorithm in its general form, we need to generalize the notion of *span* from matroids and also describe a set of *core* elements that will play the role of OPT for more general independence systems.

Definition 5 (Primitive Hull). Let $(S, \mathcal{I})$ be an independence system. We define the *primitive hull* function PHull: $2^S \to 2^S$ by

$$\text{PHull}(X) \coloneqq X \cup \{e \in S \mid e \text{ is in a circuit of } X + e\}.$$

Notice that, for a single matroid constraint, PHull is equivalent to the matroid span function. In fact, PHull satisfies the following useful properties.

(H1) (Inclusion) For all $X \subseteq S$, $X \subseteq \text{PHull}(X)$.

(H2) (Monotonicity) For all $X, Y \subseteq S$ with $X \subseteq Y$, we have $\text{PHull}(X) \subseteq \text{PHull}(Y)$.

(H3) (Basis Extension) For all $X \subseteq S$ and $e \in S \setminus X$, if $e \notin \text{PHull}(X)$ then for any $I \subseteq X$ such that $I \in \mathcal{I}$, we have $I + e \in \mathcal{I}$. In particular, if $X \in \mathcal{I}$, then $X + e \in \mathcal{I}$.

On the other hand, PHull does not necessarily satisfy some properties of the matroid span function, such as $\text{span}(\text{span}(X)) = \text{span}(X)$.

For combinations of systems, however, we use the union of the primitive hulls of each system. We call this the *hull*.

Definition 6 (Hull). Let $\mathcal{F} = (S, \mathcal{I})$ be a combination of systems $\{(S_j, \mathcal{I}_j)\}_{j=1}^M$. We define the *hull* function Hull: $2^S \to 2^S$ of $\mathcal{F}$ by

$$\text{Hull}(X) \coloneqq \bigcup_{j=1}^M \text{PHull}_j(X \cap S_j),$$

where PHull_j is the primitive hull of $(S_j, \mathcal{I}_j)$.

It is worth mentioning that, if $\mathcal{F}$ is a combination of two or more systems, $\text{PHull} \subseteq \text{Hull}$ but the reverse is not true, even for the intersection of two matroids. To see this, suppose that $\mathcal{F}$ is the intersection of two matroids M_1 and M_2 and consider three elements x, y and a such that $\{x, y, a\}$ is a circuit in M_1 and $\{x, y\}$ is a circuit in M_2. Let $X = \{x, y\}$. Then, $a \in \text{PHull}_{M_1}(X)$, which implies that $a \in \text{Hull}_{\mathcal{F}}(X)$, but $a \notin \text{PHull}_{\mathcal{F}}(X)$ since $X = \{x, y\}$ is itself a circuit in $\mathcal{F}$ and thus adding a to X does not create any new circuits. However, notice that, for k-matroid intersection, we have $\text{PHull}_j = \text{span}_j$ for every component system $(S_j, \mathcal{I}_j)$, and thus Hull is equivalent to the union-span.

We notice that the function Hull just defined satisfies all (H1), (H2) and (H3). The inclusion and monotonicity properties of Hull follow directly from the corresponding properties for the individual primitive hulls. To see the basis extension property, observe that, for $X \subseteq S$ and $e \in S \setminus X$, we have

$$\begin{aligned}
e \notin \text{Hull}(X) &\iff \forall j, \; e \notin \text{PHull}_j(X \cap S_j) \\
&\iff \forall j, \; \forall I_j \subseteq X \cap S_j \text{ s.t. } I_j \in \mathcal{I}_j, \; I_j + e \in \mathcal{I}_j \\
&\iff \forall I \subseteq X \text{ s.t. } I \in \mathcal{I}, \; I + e \in \mathcal{I}
\end{aligned}$$

Next, we define the set of *critical* elements for larger families of systems beyond matroids. We call this the set of *core* elements.

Definition 7 (Primitive Core). Let $(S, \mathcal{I})$ denote an independence system and let $w \colon S \to \mathbb{R}_+$ denote an arbitrary injective weight function. For any set $X \subseteq S$ and element e, let $A_e(X) = \{f \in X \mid w(f) > w(e)\}$ denote the set of elements with weight higher than X. We define the set of *primitive core* elements as

$$\text{PCore}(X) = \{e \in X \mid e \notin \text{PHull}(A_e(X))\}.$$

In other words, an element $x \in X$ that is in the primitive core of X cannot be the smallest element of any circuit in X. Equivalently, x is in the primitive core of X if, for every independent set $I \in \mathcal{I}$ with $I \subseteq \{y \in X \mid w(y) > w(x)\}$, we have $I + x \in \mathcal{I}$. Note that the last formulation implies that x belongs to every basis of $\{y \in X \mid w(y) \geq w(x)\}$. Recall that $\mathrm{Greedy}(X)$ denotes the set of elements selected by the weighted-greedy algorithm on X.

Lemma 2. *For all $X \subseteq Y \subseteq S$, we have $X \cap \mathrm{PCore}(Y) \subseteq \mathrm{PCore}(X) \subseteq \mathrm{Greedy}(X)$.*

Proof. Let $e \in X \cap \mathrm{PCore}(Y)$. For any circuit C with $C \subseteq Y$, e cannot be the smallest element of C. In particular, the same holds for every circuit contained in X, and so $e \in \mathrm{PCore}(X)$. Next, let $e \in \mathrm{PCore}(X)$, and suppose we run the weighted-greedy algorithm on X. Let H be the set constructed by greedy just before e is considered. Note that H is an independent set contained in $\{y \in X \mid w(y) > w(x)\}$. Since $e \in \mathrm{PCore}(X)$, we have that $H + e \in \mathcal{I}$, and so e is added to the set constructed by the weighted-greedy algorithm. $\qquad\square$

One consequence of Lemma 2 is that $\mathrm{PCore}(X)$ is always independent for any X. However, if $\mathcal{F}$ is a matroid, we have a stronger property.

Lemma 3. *Let $\mathcal{M} = (S, \mathcal{I})$ be a matroid. For any $X \subseteq S$, we have*

$$\mathrm{PCore}(X) = \mathrm{Greedy}(X) = \mathrm{OPT}(X),$$

where $\mathrm{OPT}(X)$ denotes the unique maximum weight basis of X.

Proof. The first equality follows since, for a matroid with distinct weights, $e \in \mathrm{Greedy}(X)$ if and only if e is not spanned by elements of higher weight, which exactly corresponds to the definition of $e \in \mathrm{PCore}(X)$. The second equality holds since, for matroids, the greedy algorithm on X returns the optimum basis of X. $\qquad\square$

This equivalence fails to hold for more general independence systems. In fact, even for the intersection of two matroids, the sets $\mathrm{PCore}(X)$, $\mathrm{Greedy}(X)$, and $\mathrm{OPT}(X)$ may be all different. For combinations of systems, we use the intersection of the primitive cores of each system. We call this set the *core* and its elements the *critical* elements.

Definition 8 (Core and Critical Elements). Let $\mathcal{F} = (S, \mathcal{I})$ be a combination of systems $\{(S_j, \mathcal{I}_j)\}_{j=1}^{M}$ and $w \colon S \to \mathbb{R}_+$ be an arbitrary weight function. We define the *core* of a set $X \subseteq S$ as

$$\mathrm{Core}(X) := \bigcap_{j=1}^{M} \left((X \setminus S_j) \cup \mathrm{PCore}_j(X \cap S_j) \right),$$

where PCore_j is the primitive core of $(S_j, \mathcal{I}_j)$. We call the elements in $\mathrm{Core}(X)$ the *critical* elements of X.

Notice that, for the case of k-matroid intersection, we have

$$e \in \mathrm{Core}(X) \iff \forall j, \; e \in \mathrm{PCore}_j(X) \iff \forall j, \; e \notin \mathrm{PHull}_j(\{f \in X \mid w(f) > w(e)\})$$
$$\iff \forall j, \; e \notin \mathrm{span}_j(\{f \in X \mid w(f) > w(e)\})$$
$$\iff \forall j, \; e \in \mathrm{OPT}_j(X).$$

In other words, for the intersection of k matroids, $\mathrm{Core}(X)$ is exactly $\bigcap_{j=1}^{k} \mathrm{OPT}_j(X)$. It is easy to see that, for a combination of systems, the relationship between the core and the hull from the component systems extends to the entire combination.

Observation 1. *Let $\mathcal{F} = (S, \mathcal{I})$ be a combination of systems $\{(S_j, \mathcal{I}_j)\}_{j=1}^M$ and $w \colon S \to \mathbb{R}_+$ be an arbitrary weight function. Then, for all $X \subseteq S$ and $e \in X$,*

$$e \in \mathrm{Core}(X) \iff e \notin \mathrm{Hull}(\{f \in X \mid w(f) > w(e)\}).$$

Proof. Let PHull_j and PCore_j denote the primitive hull and primitive core of $(S_j, \mathcal{I}_j)$, respectively. We have

$$
\begin{aligned}
e \in \mathrm{Core}(X) &\iff \forall j, \ (e \notin S_j) \vee (e \in \mathrm{PCore}_j(X \cap S_j)) \\
&\iff \forall j, \ (e \notin S_j) \vee (e \notin \mathrm{PHull}_j(\{f \in X \cap S_j \mid w(f) > w(e)\})) \\
&\iff e \notin \mathrm{Hull}(\{f \in X \mid w(f) > w(e)\}). \qquad \square
\end{aligned}
$$

Algorithmic Aspects. Our black-box algorithm depends on the ability to solve the following problem for the given feasibility constraint: given a set X and an element e, is $e \in \mathrm{Hull}(X)$? Since Hull is defined as the union of PHull_j over all system components $\mathcal{F}_j$, this problem reduces to testing whether $e \in \mathrm{PHull}_j(X)$ for some set X and e.

Fix a system component $\mathcal{F}_j$. Whenever $\mathcal{F}_j$ is a matroid, computing $\mathrm{PHull}_j(X) = \mathrm{span}_j(X)$ can be done in polynomial time. For the case of knapsack constraints, we can compute $\mathrm{PHull}_j(X)$ by solving a knapsack subproblem, which can be done in pseudopolynomial time. However, for general systems, computing $\mathrm{Hull}(X)$ can be arbitrarily hard; we do not pursue this direction here, as it lies outside the scope of the present work.

3.3 Free-Order Secretary Algorithm for k-Matroid Intersection

We now have all the necessary ingredients to design a general free-order algorithm that applies to any system admitting a well-defined Hull function, presented as Algorithm 2. The following theorem relates the expected weight of the independent set returned by our algorithm to that of $\mathrm{Core}(\mathrm{Rel}(Y))$, where Y is a random subset of S.

Theorem 3. *Let $(S, \mathcal{I})$ be an independence system and $Y = Y(p)$ be a random set where every element of S is in Y independently with probability p for some $p \in [0, 1]$. Then, Algorithm 2 returns an independent set ALG such that, for every $f \in \mathrm{Core}(\mathrm{Rel}(Y))$,*

$$\Pr[f \in \mathrm{ALG}] \geq \frac{1}{4}.$$

Proof. First, note that the algorithm always returns an independent set, since before adding any element e to ALG, it explicitly checks whether $\mathrm{ALG} + e \in \mathcal{I}$. We now condition on the set Y. Observe that $\mathrm{Rel}(Y) \subseteq S_1 \cup S_2$, and let $B := \mathrm{Core}(\mathrm{Rel}(Y))$. We next show that each element of B is added to ALG with probability $1/4$. The proof of this fact follows the same argument as in Algorithm 1, but we include it here for completeness.

Fix $f \in B$ and let h_1 be the heaviest element in $(S_1 - f) \cap \mathrm{Rel}(Y)$ such that f is in the hull of the elements of higher weight than h_1 in $(S_1 - f) \cap \mathrm{Rel}(Y)$. In other words,

$$h_1 := \arg\max \left\{ w(h) \mid h \in (S_1 - f) \cap \mathrm{Rel}(Y) \text{ s. t. } f \in \mathrm{Hull}(\{g \in (S_1 - f) \cap \mathrm{Rel}(Y) \mid w(g) \geq w(h)\}) \right\}.$$

Similarly, let h_2 be the heaviest element in $(S_2 - f) \cap \mathrm{Rel}(Y)$ such that f is spanned by the elements of higher weight than h_2 in $(S_2 - f) \cap \mathrm{Rel}(Y)$. In other words,

$$h_2 := \arg\max \left\{ w(h) \mid h \in (S_2 - f) \cap \mathrm{Rel}(Y) \text{ s. t. } f \in \mathrm{Hull}(\{g \in (S_2 - f) \cap \mathrm{Rel}(Y) \mid w(g) \geq w(h)\}) \right\}.$$

18

Algorithm 2: Free-Order General Secretary

Input: An independence system $(S, \mathcal{I})$, a parameter p.
Output: An independent set ALG.

1 ALG $\leftarrow \emptyset$ and $q \leftarrow (1+p)/2$
2 for *each $e \in S$* **do** Choose t_e independently and uniformly from $(0,1)$
3 $Y \leftarrow \{e \mid t_e \in (0, p)\}$
4 $S_1 \leftarrow \{e \mid t_e \in [p, (1+p)/2)\}$
5 $S_2 \leftarrow \{e \mid t_e \in [(1+p)/2, 1]\}$
6 Observe (without accepting) all elements in $Y \cup S_1$
7 unseen $\leftarrow S_2$ // The set of elements not yet revealed
8 Sort the elements of $\mathrm{Rel}(Y) \cap S_1$ in decreasing order of weights as $e_1, \ldots, e_m$
9 for $j = 1$ *to* m **do**
10 $\quad$ Let $Q = \mathbf{unseen} \cap \mathrm{Hull}(\{e_1, \ldots, e_j\})$
11 $\quad$ **for** *each $f \in Q$ in uniformly random order* **do**
12 $\quad\quad$ **unseen** $\leftarrow$ **unseen** $- f$ // Reveal f
13 $\quad\quad$ **if** $f \in \mathrm{Rel}(Y)$ *and* $w(f) > w(e_j)$ *and* ALG $+f \in \mathcal{I}$ **then**
14 $\quad\quad\quad$ ALG $\leftarrow$ ALG $+f$
15 $\quad\quad$ **end**
16 $\quad$ **end**
17 end
18 $Q \leftarrow$ **unseen**
19 for *each $f \in Q$ in uniformly random order* **do**
20 $\quad$ **if** $f \in \mathrm{Rel}(Y)$ *and* ALG $+f \in \mathcal{I}$ **then** ALG $\leftarrow$ ALG $+f$
21 end
22 return ALG

If there are no such elements, we set $h_1 = \perp$ and $h_2 = \perp$, respectively, with $w(\perp) = 0$.

Since, conditioned on Y, the sets $(S_1 - f) \cap \mathrm{Rel}(Y)$ and $(S_2 - f) \cap \mathrm{Rel}(Y)$ are identically distributed, we have that $\Pr[w(h_1) \geq w(h_2)] = 1/2$, regardless of whether one or both of h_1 and h_2 are $\perp$. Next, we condition on the independent event that $f \in S_2$, and conclude that both $w(h_1) \geq w(h_2)$ and $f \in S_2$ occur together with probability at least $1/4$. In what follows, we condition on these two events occurring.

We now verify that f satisfies all the conditions for inclusion in ALG in lines 13 and 20 of Algorithm 2. Since $f \in \mathrm{Rel}(Y)$, the first condition is satisfied. Next, recall that, $S_1 \cap \mathrm{Rel}(Y)$ is ordered in decreasing order of weights as $e_1, \ldots, e_m$. Assume that h_1 is the j^*-th highest weight element of $S_1 \cap \mathrm{Rel}(Y)$, that is, $h_1 = e_{j^*}$, where $j^* = m + 1$ if $h_1 = \perp$. By construction, f is revealed by the algorithm precisely in iteration j^*, when it first enters the set Q. Furthermore, since $f \in \mathrm{Core}(\mathrm{Rel}(Y))$, by the monotonicity of Hull we conclude that

$$f \notin \mathrm{Hull}(\{g \in \mathrm{Rel}(Y) \mid w(g) > w(f)\}) \supseteq \mathrm{Hull}(\{g \in \mathrm{Rel}(Y) \cap S_1 \mid w(g) > w(f)\}).$$

Thus, $w(f) > w(h_1) \geq w(h_2)$, so we satisfy the second condition of line 13.

Let ALG$'$ denote the solution immediately before f is revealed, and note that all elements in ALG$'$ are in $\mathrm{Rel}(Y) \cap S_2$ and have weights larger than $w(e_{j^*})$. There are two possibilities. First, if

$h_2 \neq \perp$, since $w(e_{j*}) = w(h_1) \geq w(h_2)$ by the definition of h_1 and h_2, we have

$$f \notin \mathrm{Hull}(\{g \in (S_2 - f) \cap \mathrm{Rel}(Y) \colon w(g) \geq w(e_{j*})\}) \supseteq \mathrm{Hull}(\mathrm{ALG}').$$

Here, the last containment follows by the monotonicity of Hull, since all elements considered by ALG′ for addition had weights larger than $w(e_{j*})$. Second, if $h_2 = \perp$ then, again by the monotonicity of Hull, we have

$$f \notin \mathrm{Hull}((S_2 - f) \cap \mathrm{Rel}(Y)) \supseteq \mathrm{Hull}(\mathrm{ALG}').$$

It follows that at the moment when f is revealed, $f \notin \mathrm{Hull}(\mathrm{ALG}')$, and thus $\mathrm{ALG}' + f \in \mathcal{I}$ by the properties of Hull. We conclude that all conditions hold, and therefore $f \in \mathrm{ALG}$. We have shown that for every realization of Y and every $f \in B = \bigcap_{i=1}^{k} \mathrm{Core}(\mathrm{Rel}(Y))$, $\Pr[f \in \mathrm{ALG} \mid Y] \geq \frac{1}{4}$. Since this holds for every realization of Y, by unconditioning, we get that $\Pr[f \in \mathrm{ALG}] \geq \frac{1}{4}$ as stated. $\qquad \square$

We now combine Lemma 1 and Theorem 3 to get the following corollary.

Corollary 1. *Let $(S, \mathcal{I})$ be a k-matroid-intersection system. For $p = 1 - 1/2k$, Algorithm 2 returns a set* ALG *such that*

$$\mathbb{E}[w(\mathrm{ALG})] \geq \frac{1}{16k^2} w(\mathrm{OPT}).$$

Next Steps. In the following sections, we extend both Lemma 1 and Theorem 3 in two different directions. First, we define and explore a new class of independence systems, called k-growth systems, which includes k-matchoids as well as knapsack constraints with bounded size ratios; this new class is already of independent combinatorial interest on its own. We then generalize the Core Lemma (Lemma 1), first to k-matchoids and then to general k-growth systems. Using appropriate notions of Hull for both systems, we recover $\Omega\left(1/k^2\right)$-utility competitive guarantees for both, in the free-order setting.

Second, we turn to the agent-arrival bipartite graph model. All the results above apply directly to the edge-arrival case, provided the edge independence system belongs to one of the studied classes. However, these results do not naturally extend to the agent-arrival setting, where multiple edges are revealed simultaneously. We overcome this difficulty by establishing a new version of the Core Lemma that holds for a random subset Y of agents rather than edges. Equipped with this tool, we design a new algorithm for the agent-arrival model in settings where the independence system on the *item side* admits an order-oblivious algorithm.

4   Free-Order Secretary on Combinations of Growth Systems

In this section, we extend Lemma 1 to a new and broader class of independence systems that contains k-matchoids and knapsack constraints with bounded size ratios.

4.1   Growth Systems and Combinations

The goal of this section is to introduce a novel class of independence systems with interesting structural properties. We call this the class of k-*growth systems*, and present the definition below; a more detailed discussion of its properties and its relationships to other classes of independence systems is provided in Section 7.

Definition 9 (*k*-growth system). Let $\mathcal{F} = (S, \mathcal{I})$ be an independence system and k be a positive integer. We say that $\mathcal{F}$ is a *k-growth system* if it satisfies the following *k*-Basis-Growth (*k*BG) axiom.

(kBG) $\quad \forall X \subseteq S, \forall I \in \mathcal{I}$, there exists a partition (Q, Z) of $I \setminus X$ with $|Z| \leq k\,|X \setminus I|$ such that
$\qquad\qquad$ ($\star$1) for every basis P of X, $P \cup Q \in \mathcal{I}$.

Many independence systems that arise in applications can be viewed as combinations of simpler component systems. For example, k-matchoids are formed as the intersection of n matroids – where n may be much larger than k – under the condition that each element is a not a co-loop in at most k of them. As it turns out, k-growth systems enjoy a similar combination property that will be useful for our analysis: the combination of k_i-growth systems is itself a growth system, with a parameter k determined by the k_i's of the individual components. More importantly, the union of the individual Hull functions of each component is the Hull function for the combined system. We exploit this latter property to obtain an improved result for combinations of systems whose components are matroids. The proof of the following lemma can be found in Section 7.

Lemma 4. Let $\mathcal{F} = (S, \mathcal{I})$ be a combination of systems $\{(S_j, \mathcal{I}_j)\}_{j=1}^{M}$, where each $(S_j, \mathcal{I}_j)$ is a k_j-growth system. Then, $\mathcal{F}$ is a k-growth system, where

$$k = \max_{e \in S} \sum_{j\,:\, e \in S_j} k_j.$$

In particular, for each $X \subseteq S$ and $I \in \mathcal{I}$, there exists a collection of sets $Z_j \subseteq (I \setminus X) \cap S_j$ such that, for every basis P of X, we have $P \cup \left((I \setminus X) \setminus \bigcup_{j=1}^{M} Z_j \right) \in \mathcal{I}$, and, for all $j \in [M]$,

$$|Z_j| \leq \begin{cases} r_j((X \setminus I) \cap S_j) - |(X \cap I) \cap S_j| & \text{if } k_j = 1, \\ k_j |(X \setminus I) \cap S_j| & \text{if } k_j > 1, \end{cases}$$

where r_j denotes the rank function of the j-th component $(S_j, \mathcal{I}_j)$. In particular, $|Z_j| \leq k_j |(X \setminus I) \cap S_j|$ for $j \in [M]$, since r_j is a matroid rank function when $k_j = 1$.

4.2   The Core Lemma for Combinations of Growth Systems

We are now ready to describe the main result of this section: our generalization of the Core Lemma of [34] from intersections of matroids to combinations of growth systems.

Lemma 5 (Core Lemma for Growth Systems). *Let $\mathcal{F} = (S, \mathcal{I})$ be a combination of systems $\{(S_j, \mathcal{I}_j)\}_{j=1}^{M}$ where each $(S_j, \mathcal{I}_j)$ is a k_j-growth system, and let* Core *be the Core of $\mathcal{F}$. Also, let $k = \max_{e \in E} \sum_{j\,:\, e \in S_j} k_j$, and $Y = Y(p)$ be a random set where every element of S is in Y independently with probability p for some $p \in [0, 1]$. Then,*

$$\mathbb{E}\left[w\left(\mathrm{Core}\left(\mathrm{Rel}(Y)\right)\right)\right] \geq \frac{(p - (1-p)k)(1-p)}{p^2}\ \mathbb{E}\left[w\left(\mathrm{Greedy}(Y)\right)\right].$$

Furthermore, if all components are matroids, i.e. $\mathcal{F}$ is a k-matchoid, then

$$\mathbb{E}\left[w\left(\mathrm{Core}\left(\mathrm{Rel}(Y)\right)\right)\right] \geq \frac{(p - (1-p)(k-1))(1-p)}{p}\ \mathbb{E}\left[w\left(\mathrm{Greedy}(Y)\right)\right].$$

21

Algorithm 3: Simulation algorithm

---

1 $R, N, R', N' \leftarrow \emptyset$

2 **for** $i = 1$ *to* m **do**

3   Toss a coin c_i that is heads with probability p `// If heads,` e_i `is in the sampling phase`

4   **if** $(R' \cup N') + e_i \in \mathcal{I}$ **then**

5   **if** $e_i \in \mathrm{Core}((R \cup N) + e_i)$ **then**

6   **if** c_i *is heads* **then** $R' \leftarrow R' + e_i$ **else** $R \leftarrow R + e_i$

7   **else**

8   **if** c_i *is heads* **then** $N' \leftarrow N' + e_i$ **else** $N \leftarrow N + e_i$

9   **end**

10   **end**

11 **end**

Proof. Recall that, by Lemma 4, $\mathcal{F}$ is a k-growth system, where $k = \max_{e \in E} \sum_{j\,:\,e \in S_j} k_j$. Let $e_1, \ldots, e_m$ be the elements of S sorted in decreasing order of weights. The idea is to split the elements into sets R, N, R', and N', where $R' \cup N' = \mathrm{Greedy}(Y)$ and $R \cup N = \mathrm{Rel}(Y)$, such that $R' = \mathrm{Greedy}(Y) \cap \mathrm{Core}(\mathrm{Rel}(Y))$, $R = \mathrm{Core}(\mathrm{Rel}(Y))$ and $N' = \mathrm{Greedy}(Y) \backslash R'$, $N = \mathrm{Rel}(Y) \backslash \mathrm{Core}(\mathrm{Rel}(Y))$. Then, the proof of the lemma will follow from the following four inequalities:

- $\mathbb{E}\big[|N'|\big] \geq \frac{p}{1-p}\,\mathbb{E}[|N|]$,   (1)

- $\mathbb{E}[|R|] + \mathbb{E}[|N|] \geq \frac{1-p}{p}(\mathbb{E}[|R'|] + \mathbb{E}[|N'|])$,   (2)

- $\mathbb{E}[|N'|] \leq k\,(\mathbb{E}[|R|] + \mathbb{E}[|N|])$,   (3)

- $\mathbb{E}[|N'|] \leq k\,\mathbb{E}[|R|] + (k-1)\,\mathbb{E}[|N|]$ if all components of $\mathcal{F}$ are matroids.   (4)

To construct R, N, R' and N', we use Algorithm 3 which is an offline simulation algorithm.

Let R_i, N_i, R'_i and N'_i be the sets R, N, R' and N', respectively, at the end of the i-th iteration, where we set $R_0 = N_0 = R'_0 = N'_0 = \emptyset$. Note that Y has the same distribution as the set of edges e_i for which c_i is heads, so we let Y be that set. Let $E_i := \{e_1, \ldots, e_i\}$ and $Y_i := Y \cap E_i$. Also note that $R'_i \cup N'_i = \mathrm{Greedy}(Y_i)$ and $R_i \cup N_i = \mathrm{Rel}(Y_i) \cap E_i = \mathrm{Rel}(Y) \cap E_i$, and that N_i, R_i, N'_i and R'_i are pairwise disjoint. Let $I_i := R'_i \cup N'_i$ and $X_i := R_i \cup N_i$.

First, notice that, at each iteration, every e_i that satisfies both $(R' \cup N') + e_i \in \mathcal{I}$ and $e_i \in \mathrm{Core}((R \cup N) + e_i)$ is added to R' with probability p and to R with probability $1 - p$. Similarly, every e_i that satisfies both $(R' \cup N') + e_i \in \mathcal{I}$ and $e_i \notin \mathrm{Core}((R \cup N) + e_i)$ is added to N' with probability p and to N with probability. $1 - p$. From these, it follows that, for every i,

$$(1-p)\,\mathbb{E}[|R'_i|] = p\,\mathbb{E}[|R_i|] \quad \text{and} \quad (1-p)\,\mathbb{E}[|N'_i|] = p\,\mathbb{E}[|N_i|],$$

and thus (1) and (2) follow directly from the above. Next, we show that (3) holds as well. First, we verify that R, N, R', and N' are the sets as claimed.

Claim 1. *For all $i \in [m]$, $R'_i \cup N'_i = \mathrm{Greedy}(Y_i)$.*

Proof. Observe that, for $\ell \leq i$, $R'_i \cup N'_i$ corresponds to the set of elements e_ℓ from E_i which satisfied $(R'_{\ell-1} \cup N'_{\ell-1}) + e_\ell \in \mathcal{I}$ and for which the coin c_ℓ came up heads. In other words, $R'_i \cup N'_i$ is the subset of Y_i of elements e_ℓ which satisfied $(R'_{\ell-1} \cup N'_{\ell-1}) + e_\ell \in \mathcal{I}$, which is exactly $\mathrm{Greedy}(Y_i)$. $\square$

Claim 2. *For all $i \in [m]$, $R_i \cup N_i = \mathrm{Rel}(Y) \cap E_i$.*

Proof. Similarly to the previous claim, for $\ell \leq i$, $R_i \cup N_i$ corresponds to the set of elements e_ℓ from E_i which satisfied $(R'_{\ell-1} \cup N'_{\ell-1}) + e_\ell \in \mathcal{I}$ and for which the coin c_ℓ came up tails, and thus, by the definition of greedy-relevant elements, we have $R_i \cup N_i = \mathrm{Rel}(Y) \cap E_i$. $\qquad\square$

Claim 3. *For all $i \in [m]$, $R_i = \mathrm{Core}(\mathrm{Rel}(Y)) \cap E_i$.*

Proof. Observe that, for $\ell \leq i$, R_i corresponds to the set of elements e_ℓ from E_i which satisfied $(R'_{\ell-1} \cup N'_{\ell-1}) + e_\ell \in \mathcal{I}$, $e_\ell \in \mathrm{Core}((R_{\ell-1} \cup N_{\ell-1}) + e_\ell)$ and for which the coin c_ℓ came up tails. By Claim 2, this is equivalent to $e_\ell \in \mathrm{Core}((\mathrm{Rel}(Y) \cap E_{\ell-1}) + e_\ell)$ and, by Observation 1, to $e_\ell \notin \mathrm{Hull}((\mathrm{Rel}(Y) \cap E_{\ell-1}) + e_\ell) = \mathrm{Hull}(\mathrm{Rel}(Y))$, where the equality follows from the fact that $E_{\ell-1}$ contains the elements of S with weight higher than that of e_ℓ and from the definition of Hull, as well as from the fact that $e_\ell \in \mathrm{Rel}(Y)$. Thus, we get that $e_\ell \in \mathrm{Core}(\mathrm{Rel}(Y))$. Since $e_\ell \in E_i$, the claim follows. $\qquad\square$

Claim 4. *For all $i \in [m]$, $R'_i = \mathrm{Greedy}(Y_i) \cap \mathrm{Core}(\mathrm{Rel}(Y))$.*

Proof. By Claim 1, we have that $R'_i \subseteq \mathrm{Greedy}(Y_i)$. In particular, for $\ell \leq i$, R'_i corresponds to the set of elements e_ℓ from $\mathrm{Greedy}(Y_i)$ which also satisfied $e_\ell \in \mathrm{Core}((R_{\ell-1} \cup N_{\ell-1}) + e_\ell)$. By Claim 2, this is equivalent to $e_\ell \in \mathrm{Core}((\mathrm{Rel}(Y) \cap E_{\ell-1}) + e_\ell)$ and, by Observation 1, to $e_\ell \notin \mathrm{Hull}((\mathrm{Rel}(Y) \cap E_{\ell-1}) + e_\ell) = \mathrm{Hull}(\mathrm{Rel}(Y))$, where the equality follows from the fact that $E_{\ell-1}$ contains the elements of S with weight higher than that of e_ℓ and from the definition of Hull, as well as from the fact that $e_\ell \in \mathrm{Rel}(Y)$. Thus, we get that $e_\ell \in \mathrm{Core}(\mathrm{Rel}(Y))$, and the claim follows. $\qquad\square$

Next, we use Lemma 4 on the disjoint sets X_i and I_i to obtain sets $Z_{i,j} \subseteq I_i$ for each $j \in [M]$ with the bounds given by the lemma, and let $Z_i = \bigcup_{j=1}^M Z_{i,j}$ and $Q_i = (I_i \setminus X_i) \setminus Z_i = I_i \setminus Z_i$. We show that $N'_i \subseteq Z_i$ for all i.

To see this, notice that, Claims 1 and 4 and the definition of N', we get that every element e_ℓ with $\ell \leq i$ that enters N'_i satisfies $e_\ell \notin \mathrm{Core}(N_{\ell-1} \cup R_{\ell-1} + e_\ell)$. In particular, there exists an index j such that $e_\ell \in S_j$ and $e_\ell \notin \mathrm{Core}_j((N_{\ell-1} \cup R_{\ell-1}) \cap S_j)$, and thus there exists a circuit C of the j-th component such that $e_\ell \in C \cap S_j$. Therefore, the set $P_j := C - e_\ell \subseteq (N_{\ell-1} \cup R_{\ell-1}) \cap S_j$ satisfies $P_j \in \mathcal{I}_j$ but $P_j + e_\ell \notin \mathcal{I}_j$. This implies that $e_\ell \notin Q_{i,j} := (I_i \setminus Z_{i,j}) \cap S_j$. Therefore, e_ℓ has to belong to $Z_{i,j} \subseteq Z_i$, and thus $N'_i \subseteq Z_i$.

Now, we condition on the coins c_ℓ for $\ell \leq i$ and we apply the basis-growth axiom (kBG) to the disjoint sets X_i and $I_i \in \mathcal{I}$, yielding

$$|N'_i| \leq \left| \bigcup_{j=1}^M Z_{i,j} \right| \leq \sum_{j=1}^M k_j |X_i \cap S_j| = \sum_{e \in X_i} \sum_{j:\, e \in S_j} k_j = k\,|X_i| = k(|R_i| + |N_i|).$$

By unconditioning, we obtain

$$\mathbb{E}[|N'_i|] \leq k\,(\mathbb{E}[|R_i|] + \mathbb{E}[|N_i|]), \tag{5}$$

and thus (3) follows. For the particular case when all components are matroids and so $\mathcal{F}$ is a k-matchoid, we use the following stronger inequality.

$$|N_i'| \leq \left| \bigcup_{j=1}^{M} Z_{i,j} \right| \leq \sum_{j=1}^{M} r_j(X_i \cap S_j) = \sum_{\ell=1}^{i} \sum_{j=1}^{M} r_j(X_\ell \cap S_j) - r_j(X_{\ell-1} \cap S_j)$$

$$\leq \sum_{\ell=1}^{i} \sum_{e_\ell \in R_\ell} k + \sum_{\ell=1}^{i} \sum_{e_\ell \in N_\ell} (k-1) = k\,|R_i| + (k-1)\,|N_i|,$$

where the last inequality follows from the fact that if $e_\ell \in R_\ell$, then for every matroid j in which $e_\ell \in S_j$, e_ℓ does not close any circuits in X_ℓ. Therefore, for all such ℓ, we have $r_j(X_\ell \cap S_j) - r_j(X_{\ell-1} \cap S_j) = 1$. Since e_ℓ participates in at most k matroids, we have $\sum_{j=1}^{M}(r_j(X_\ell \cap S_j) - r_j(X_{\ell-1} \cap S_j)) \leq k$. On the other hand, if $e_\ell \in N_\ell$, then there must be an index j with $e_\ell \in S_j$ such that e_ℓ closes a circuit in X_ℓ. Therefore, the sum above is at most $k-1$. Once again, by unconditioning, we obtain

$$\mathbb{E}[|N_i'|] \leq k\,\mathbb{E}[|R_i|] + (k-1)\,\mathbb{E}[|N_i|], \tag{6}$$

and thus (4) follows.

Putting it all together, we have

$$k\,\mathbb{E}[|R_i|] \geq \mathbb{E}[|N_i'|] - k\,\mathbb{E}[|N_i|]$$

$$\geq \left(\frac{p}{1-p} - k \right) \mathbb{E}[|N_i|]$$

$$\geq \left(\frac{p}{1-p} - k \right) \left(\frac{1-p}{p}(\mathbb{E}[|R_i'|] + \mathbb{E}[|N_i'|]) - \mathbb{E}[|R_i|] \right),$$

yielding

$$\mathbb{E}[|R_i|] \geq \left(\frac{p}{1-p} - k \right) \frac{(1-p)^2}{p^2}(\mathbb{E}[|R_i'|] + \mathbb{E}[|N_i'|])$$

$$= \frac{(p - (1-p)k)(1-p)}{p^2}(\mathbb{E}[|R_i'|] + \mathbb{E}[|N_i'|]).$$

Thus, by Claims 1 and 3, we have

$$\mathbb{E}\left[|\mathrm{Core}(\mathrm{Rel}(Y)) \cap E_i|\right] \geq \frac{(p - (1-p)k)(1-p)}{p^2} \mathbb{E}\left[|\mathrm{Greedy}(Y_i)|\right].$$

Since this is true for every i, applying linearity of expectation and the definition of the Core function yields

$$\mathbb{E}\left[w\left(\mathrm{Core}(\mathrm{Rel}(Y))\right)\right] \geq \frac{(p - (1-p)k)(1-p)}{p^2} \mathbb{E}\left[w\left(\mathrm{Greedy}(Y)\right)\right].$$

For the particular case when all components are matroids, we use (6) instead of (5) to get

$$k\,\mathbb{E}[|R_i|] \geq \mathbb{E}[|N_i'|] - (k-1)\,\mathbb{E}[|N_i|]$$

$$\geq \left(\frac{p}{1-p} - (k-1) \right) \mathbb{E}[|N_i|]$$

$$\geq \left(\frac{p}{1-p} - (k-1) \right) \left(\frac{1-p}{p}(\mathbb{E}[|R_i'|] + \mathbb{E}[|N_i'|]) - \mathbb{E}[|R_i|] \right).$$

After rearranging the terms, we get

$$\mathbb{E}[|R_i|] \geq (p - (1-p)(k-1)) \frac{1-p}{p} (\mathbb{E}[|R_i'|] + \mathbb{E}[|N_i'|]).$$

Similarly as before, by Claims 1 and 3, we have

$$\mathbb{E}\left[|\text{Core}(\text{Rel}(Y)) \cap E_i|\right] \geq \frac{(p - (1-p)(k-1))(1-p)}{p} \mathbb{E}\left[|\text{Greedy}(Y_i)|\right].$$

Since this is true for every i, applying linearity of expectation and the definition of the Core function yields

$$\mathbb{E}\left[w\left(\text{Core}(\text{Rel}(Y))\right)\right] \geq \frac{(p - (1-p)(k-1))(1-p)}{p} \mathbb{E}\left[w\left(\text{Greedy}(Y)\right)\right]. \qquad \square$$

4.3 Free-Order Secretary Algorithm for Edge Arrivals

Next, we summarize the guarantees of our Core lemma with respect to $w(\text{OPT})$. Note that our guarantee for k-matchoids is exactly the same as the previous guarantee for k-matroid intersection, see Lemma 1.

Theorem 4. *Let $\mathcal{F} = (S, \mathcal{I})$ be a combination of systems $\{(S_j, \mathcal{I}_j)\}_{j=1}^{M}$ where each $(S_j, \mathcal{I}_j)$ is a k_j-growth system. Let $\text{Core}_j(X)$ denote the set of critical elements with respect to X in $(S_j, \mathcal{I}_j)$. Also, let $k = \max_{e \in E} \sum_{j : e \in S_j} k_j$, and $Y = Y(p)$ be a random set that contains each element of S independently with probability p for some $p \in [0, 1]$. Then, for $p = \sqrt{1 - 1/(k+1)}$, we have*

$$\mathbb{E}\left[w\left(\text{Core}\left(\text{Rel}(Y)\right)\right)\right] \geq \left(2 + \frac{1}{k} - 2\sqrt{1 + \frac{1}{k}}\right) w(\text{OPT}(S)) > \left(\frac{1}{4k^2} - \frac{1}{8k^3}\right) w(\text{OPT}(S)).$$

Furthermore, if all components are matroids, i.e., $\mathcal{F}$ is a k-matchoid, setting $p = 0$ for $k = 1$ and $p = 1 - 1/(2k)$ for $k \geq 2$ yields

$$\mathbb{E}\left[w\left(\text{Core}\left(\text{Rel}(Y)\right)\right)\right] \geq \begin{cases} w(\text{OPT}(S)) & \text{if } k = 1, \\ \frac{1}{4k^2} w(\text{OPT}(S)) & \text{if } k \geq 2. \end{cases}$$

Proof. Since the system $(S, \mathcal{I})$ is a k-system the weighted-greedy algorithm yields a $1/k$-approximation with respect to the weight of $\text{OPT}(S)$. Therefore,

$$\mathbb{E}[w(\text{Greedy}(Y))] \geq \frac{1}{k} \mathbb{E}[w(\text{OPT}(Y))] \geq \frac{1}{k} \mathbb{E}[w(\text{OPT}(S)) \cap Y] = \frac{p}{k} w(\text{OPT}(S)).$$

Thus, by Lemma 5,

$$\mathbb{E}\left[w\left(\text{Core}\left(\text{Rel}(Y)\right)\right)\right] \geq \frac{(p - (1-p)k)(1-p)}{pk} w(\text{OPT}(S)).$$

Optimizing p yields $p = \sqrt{1 - \frac{1}{k+1}}$ and

$$\mathbb{E}\left[w\left(\text{Core}\left(\text{Rel}(Y)\right)\right)\right] \geq \left(2 + \frac{1}{k} - 2\sqrt{1 + \frac{1}{k}}\right) w(\text{OPT}(S)).$$

Now consider the case where all components are matroids. If $k = 1$, then the components of $\mathcal{F}$ are pairwise disjoint and thus the system is a direct sum of matroids. In this case, for $p = 0$ we have $\text{Rel}(Y) = S$ and, for all $j \in [M]$, $\text{Core}_j(S) = \text{OPT}(S)$. Thus,

$$\mathbb{E}\left[w\left(\text{Core}\left(\text{Rel}(Y)\right)\right)\right] = w\left(\text{Core}(S)\right) = w(\text{OPT}(S)).$$

If $k \geq 2$, by Lemma 5,

$$\mathbb{E}\left[w\left(\text{Core}\left(\text{Rel}(Y)\right)\right)\right] \geq \frac{(p - (1-p)(k-1))(1-p)}{k} w(\text{OPT}(S)).$$

Optimizing p yields $p = 1 - \frac{1}{2k}$ and

$$\mathbb{E}\left[w\left(\text{Core}\left(\text{Rel}(Y)\right)\right)\right] \geq \frac{1}{4k^2} w(\text{OPT}(S)). \qquad \square$$

Finally, using Algorithm 2 and Theorem 3, we obtain our main result for the free-order secretary problem on combinations of growth systems.

Theorem 5. *Let $\mathcal{F} = (S, \mathcal{I})$ be a combination of systems $\{(S_j, \mathcal{I}_j)\}_{j=1}^M$ where each $(S_i, \mathcal{I}_i)$ is a k_j-growth system. Let $k = \max_{e \in E} \sum_{j: e \in S_j} k_j$. Then, for $p = \sqrt{1 - 1/(k+1)}$, Algorithm 2 has a utility-competitive ratio at least*

$$\frac{1}{2} + \frac{1}{4k} - \frac{1}{2}\sqrt{1 + \frac{1}{k}} > \frac{1}{16k^2} - \frac{1}{32k^3}.$$

Furthermore, if all components are matroids, i.e. $\mathcal{F}$ is a k-matchoid, Algorithm 2 has a utility-competitive ratio at least $\frac{1}{4}$ if $k = 1$, and at least $\frac{1}{16k^2}$ if $k \geq 2$.

5   The Agent-Arrival Setting

In this section, we turn our attention to the agent-arrival setting. As discussed in the introduction, the Core lemma (Lemma 5) and thus our competitive ratio guarantees (Theorem 5) unfortunately do not generalize directly from edge arrivals to this setting.

First, we need to "transfer" the constraints from the sets of agents and items to the set of edges. Let $\mathcal{F}_A = (A, \mathcal{J}_A)$ be a combination of systems $\{(A_j, \mathcal{J}_j)\}_{j=1}^{M_A}$ such that, for all $j \in [M_A]$, $(A_j, \mathcal{J}_j)$ is a k_j-growth system. Furthermore, let $\mathcal{F}_B = (B, \mathcal{L}_B)$ be a combination of systems $\{(B_j, \mathcal{L}_j)\}_{j=1}^{M_B}$ such that, for all $j \in [M_B]$, $(B_j, \mathcal{L}_j)$ is an ℓ_j-growth system.

Before we "transfer" $\mathcal{F}_A$ and $\mathcal{F}_B$ from the set of agents and items, respectively, to the edges, we need to take care of the matching constraint inherent in our model. This is easy to do however, since k-growth systems are closed under parallel-extension (for a proof and more information see Lemma 9). Therefore, for each system $(A_j, \mathcal{J}_j)$ for $j \in [M_A]$ (resp. $(B_j, \mathcal{L}_j)$ for $j \in [M_B]$) and each agent $a \in A$ (resp. item $b \in B$) we create $|\delta(a)|$ (resp. $|\delta(b)|$) many copies of a (resp. b). Since all copies are parallel, we associate each edge in $\delta(a)$ (resp. $\delta(b)$) with a unique copy of a (resp. b). Since $(A_j, \mathcal{J}_j)$ (resp. $(B_j, \mathcal{L}_j)$) is a k_j-growth (resp. ℓ_j-growth) system, by Lemma 9, the resulting set system is still a k_j-growth (resp. ℓ_j-growth) system. In addition, for every $(A_j, \mathcal{J}_j)$ (resp. $(B_j, \mathcal{L}_j)$), no feasible solution includes two edges incident to the same original agent a (resp. item b), and is thus still a matching.

Next, for each $(A_j, \mathcal{J}_j) \in \mathcal{F}_A$, we create an independence system $(E_{A,j}, \mathcal{I}_{A,j})$ on the set of edges. Then, the ground set $E_{A,j} = \delta(A_j)$ are the edges adjacent to the agents in A_j. Next, for a set $X \subseteq E_{A,j}$ of edges, let $V_A(X) = \{a \in A_j \mid \delta(a) \cap X \neq \emptyset\}$ denote the set of agents incident to some edge in X. Then, for the feasible sets, we have $\mathcal{I}_{A,j} = \{X \subseteq E_{A,j} \mid V_A(X) \in \mathcal{J}_j\}$. Let $\mathcal{F}_{A,E} = (E, \mathcal{I}_A)$ denote the combination of $\{(E_{A,j}, \mathcal{I}_{A,j})\}_{j=1}^{M_A}$. Similarly, for each $(B_j, \mathcal{L}_j) \in \mathcal{F}_B$, we create an independence system $(E_{B,j}, \mathcal{I}_{B,j})$ on the set of edges. Its ground set $E_{B,j} = \delta(B_j)$ are the edges adjacent to the items in B_j. Next, for a set $X \subseteq E_{B,j}$ of edges, let $V_B(X) = \{b \in B_j \mid \delta(b) \cap X \neq \emptyset\}$ denote the set of items incident to some edge in X. Then, for the feasible sets, we have $\mathcal{I}_{B,j} = \{X \subseteq E_{B,j} \mid V_B(X) \in \mathcal{L}_j\}$. Let $\mathcal{F}_{B,E} = (E, \mathcal{I}_B)$ denote the combination of $\{(E_{B,j}, \mathcal{I}_{B,j})\}_{j=1}^{M_B}$. Our global constraint $\mathcal{F} = (E, \mathcal{I})$ on the set of edges can now be obtained as a combination of the systems $\{(E_{A,j}, \mathcal{I}_{A,j})\}_{j=1}^{M_A}$ and $\{(E_{B,j}, \mathcal{I}_{B,j})\}_{j=1}^{M_B}$. In other words, a set X of edges is feasible for our global constraint if and only if $X \in \mathcal{I}_A \cap \mathcal{I}_B$.

Top-Relevant Edges. Similar to the edge-arrival setting, we begin by observing the edges adjacent to a sampled fraction of *agents*. We then use this sampled set of agents to define a set of *greedy-relevant edges* in subsequent phases of our algorithm.

Given a set $Y \subseteq A$ of agents, we apply the weighted-greedy algorithm for the system $(E, \mathcal{I})$ to the set of edges incident to Y to obtain a set $\text{Greedy}(\delta(Y)) \in \mathcal{I}$. Recall that, by Definition 4, for every set X of edges, we say that an edge $e \in E \setminus X$ is greedy-relevant with respect to X if e belongs to the output of the weighted-greedy algorithm on $X + e$. Then, for every agent $a \in A \setminus Y$, we associate the edge of highest weight in $\delta(a)$ that is greedy-relevant with respect to $\delta(Y)$, call it the *top-relevant edge of a with respect to Y*, and denote it by $\text{top-rel}(a, Y)$. In other words, $\text{top-rel}(a, Y) := \arg\max\{w(e) \mid e \in \delta(a), \text{ and } e \text{ is greedy-relevant with respect to } \delta(Y)\}$. If a has no greedy-relevant edges, we write $\text{top-rel}(a) = \bot$.

In what follows, our aim is to show that, for a sufficiently large set of sampled agents Y, the *set of top-relevant edges with respect to Y* which contains every top-relevant edge for every agent $a \in A \setminus Y$, retains a decent fraction of the weight of a maximum-weight basis. To keep our notation consistent with the Core lemma for edge arrivals, we abuse notation and use $\text{Rel}(Y)$ to denote the set of top-relevant edges with respect to Y, i.e.

$$\text{Rel}(Y) := \{\text{top-rel}(a, Y) \in E \mid a \in A \setminus Y\}.$$

5.1 The Core Lemma for Agent Arrivals

We are now ready to describe the main result of this section: our generalization of the Core Lemma for combinations of growth systems, from the edge-arrival setting to the agent-arrival setting.

Lemma 6. *Let $\mathcal{F}_{A,E} = (E, \mathcal{I}_A)$ be a combination of $\{(E_{A,j}, \mathcal{I}_{A,j})\}_{j=1}^{M_A}$, where each $(E_{A,j}, \mathcal{I}_{A,j})$ is a k_j-growth system, and $\mathcal{F}_{B,E} = (E, \mathcal{I}_B)$ be a combination of $\{(E_{B,j}, \mathcal{I}_{B,j})\}_{j=1}^{M_B}$, where each $(E_{B,j}, \mathcal{I}_{B,j})$ is an ℓ_j-growth system. Also, let $\mathcal{F} = (E, \mathcal{I})$ denote the intersection of $\mathcal{F}_{A,E}$ and $\mathcal{F}_{B,E}$, let $\text{Core}(X)$ denote the set of critical elements with respect to X in $\mathcal{F}$, i.e.,*

$$\text{Core}(X) = \bigcap_{j=1}^{M_A}((X \setminus E_{A,j}) \cup \text{PCore}_{j,A}(X \cap E_{A,j})) \cap \bigcap_{j=1}^{M_B}((X \setminus E_{B,j}) \cup \text{PCore}_{j,B}(X \cap E_{B,j})),$$

where $\mathrm{PCore}_{j,A}$ is the primitive core function of $(E_{A,j}, \mathcal{I}_{A,j})$ and $\mathrm{PCore}_{j,B}$ is the primitive core function of $(E_{B,j}, \mathcal{I}_{B,j})$. Let

$$k := \max_{e \in E} \sum_{j:\, e \in E_{A,j}} k_j + \sum_{j:\, e \in E_{B,j}} \ell_j,$$

and $Y = Y(p)$ be a random set where every agent $a \in A$ is in Y independently with probability p for some $p \in [0,1]$. Then,

$$\mathbb{E}\left[w\left(\mathrm{Core}\left(\mathrm{Rel}(Y)\right)\right)\right] \geq \frac{(p - (1-p)k)(1-p)}{p^2} \; \mathbb{E}\left[w\left(\mathrm{Greedy}(\delta(Y))\right)\right].$$

Furthermore, if all components are matroids, i.e. $\mathcal{F}$ is a k-matchoid, then

$$\mathbb{E}\left[w\left(\mathrm{Core}\left(\mathrm{Rel}(Y)\right)\right)\right] \geq \frac{(p - (1-p)(k-1))(1-p)}{p} \; \mathbb{E}\left[w\left(\mathrm{Greedy}(\delta(Y))\right)\right].$$

Proof. Let $e_1, \ldots, e_m$ be the elements of E sorted in decreasing order of weights. We will define disjoint sets R, N, R', and N', where $R' \cup N' = \mathrm{Greedy}(\delta(Y))$ and $R \cup N = \mathrm{Rel}(Y)$, such that $R' = \mathrm{Greedy}(\delta(Y)) \cap \mathrm{Core}(\mathrm{Rel}(Y))$, $R = \mathrm{Core}(\mathrm{Rel}(Y))$ and $N' = \mathrm{Greedy}(\delta(Y)) \setminus R'$, $N = \mathrm{Rel}(Y) \setminus \mathrm{Core}(\mathrm{Rel}(Y))$. To construct R, N, R' and N', we use Algorithm 4 which is an offline simulation algorithm. Notice that Algorithm 4 is very similar to Algorithm 3, with the exception that, when we consider an edge e_i, if we have already processed the agent incident to e_i, this implies that e_i is not the top-relative edge of its incident agent, and we ignore it. In Line 17, we have added the instruction "Do nothing" for clarity. Let us start by proving that the sets R, N, R' and N' satisfy the claimed conditions.

Let $R_i, N_i, R'_i, N'_i, \overline{A}_i$ be the sets $R, N, R', N', \overline{A}$ respectively, at the end of the i-th iteration, where $R_0 = N_0 = R'_0 = N'_0 = \overline{A}_0 = \emptyset$. Let $E_i = \{e_1, \ldots, e_i\}$ and also say that an edge e_i is a *candidate for greedy* if, on iteration i, $R'_{i-1} \cup N'_{i-1} + e_i \in \mathcal{I}$. Finally, let Y be the set of agents $a \in A$ for which the coin c_a is heads. Thus, we assume that they are the same set, and furthermore, we let $Y_i = \overline{A}_i \cap Y$. Also note N_i, R_i, N'_i and R'_i are pairwise disjoint. Let $I_i := R'_i \cup N'_i$ and $X_i := R_i \cup N_i$.

Claim 5. *For all $i \in [m]$, $R'_i \cup N'_i = \mathrm{Greedy}(\delta(Y_i))$.*

Proof. Consider the i-th step of our algorithm and let $a = V_A(e_i)$. If e_i is not a candidate for greedy or if c_a is tails, then in our algorithm $Y_{i+1} = Y_i$, $R'_i = R'_{i-1}$ and $N'_i = N'_{i-1}$ so the claim holds. Thus, assume that e_i is a candidate for greedy and that c_a is heads. Note that in this case, it must be that $a \notin \overline{A}_{i-1}$; if not, there exists an index $j < i$ such that e_j is the first edge incident to a that is a candidate for greedy. But since c_a is heads, we have $a \in Y_j \subseteq Y_{i-1} \subseteq \overline{A}_{i-1}$, which is a contradiction. Thus, $a \notin \overline{A}_{i-1}$.

In that case, $Y_i = Y_i + a$ and since e_i is a candidate for greedy,

$$\mathrm{Greedy}(\delta(Y_i)) = \mathrm{Greedy}(\delta(Y_{i-1}) \cup \delta(a)) = \mathrm{Greedy}(\delta(Y_{i-1}) + e_i) = \mathrm{Greedy}(\delta(Y_{i-1})) + e_i,$$

since e_i is the first edge incident to a that is a candidate for greedy. Indeed, all edges in $\delta(a)$ with higher weight than e_i were discarded on previous iterations (i.e. they were not candidates for greedy) and all edges in $\delta(a)$ with lower weight than e_i won't enter $\mathrm{Greedy}(\delta(Y_j))$ for any $j \geq i$, since e_i is has entered $\mathrm{Greedy}(\delta(Y_i))$ and Greedy cannot contain two edges incident to the same agent.

Furthermore, by the description of the algorithm, e_i will enter either R' or N' so,

$$R'_i \cup N'_i = R'_{i-1} \cup N'_{i-1} + e_i. \qquad \square$$

28

Algorithm 4: Simulation algorithm for agent arrivals

---

1 $R, N, R', N' \leftarrow \emptyset$

2 $\overline{A} \leftarrow \emptyset$ `// The set of processed agents`

3 **for** $a \in A$ **do**

                                  `// If heads, a is in the sampling phase`

4   Toss a coin c_a that is heads with probability p

5 **end**

6 **for** $i = 1$ *to* m **do**

7   Let $V_A(e_i)$ be the agent incident to e_i

8   **if** $(R' \cup N') + e_i \in \mathcal{I}$ **then**

9   **if** $V_A(e_i) \notin \overline{A}$ **then**

10 $\overline{A} \leftarrow \overline{A} + V_A(e_i)$

11   **if** $e_i \in \mathrm{Core}((R \cup N) + e_i)$ **then**

12   **if** $c_{V_A(e_i)}$ *is head* **then** $R' \leftarrow R' + e_i$ **else** $R \leftarrow R + e_i$

13   **else**

14   **if** $c_{V_A(e_i)}$ *is head* **then** $N' \leftarrow N' + e_i$ **else** $N \leftarrow N + e_i$

15   **end**

16   **else**

                     `// `e_i` wasn't the first edge of `$V_A(e_i)$` that could enter `$\mathrm{Greedy}(E_i)$

17   Do nothing

18   **end**

19   **end**

20 **end**

Claim 6. *For all* $i \in [m]$, $R_i \cup N_i = \mathrm{Rel}(Y) \cap E_i$.

Proof. Consider the i-th step of our algorithm and let $a = V_A(e_i)$. First, note that if e_i is not a candidate for greedy or if c_a is heads, then in our algorithm $R_i \cup N_i = R_{i-1} \cup N_{i-1}$ so the claim holds. Thus, assume that e_i is a candidate for greedy and that c_a is tails.

Now, if e_i is the top-relevant edge of a with respect to Y, i.e. if $e_i = \text{top-rel}(a, Y)$, then we have that $e_i \in \mathrm{Rel}(Y) \cap E_i$ and also that $a \notin \overline{A}_{i-1}$, and thus, by the description of the algorithm, e_i will enter either R or N so,

$$R_i \cup N_i = R_{i-1} \cup N_{i-1} + e_i.$$

On the other hand, if $e_i \neq \text{top-rel}(a, Y)$, then $e_i \notin \mathrm{Rel}(Y) \cap E_i$. This, however, implies that there exists a $j < i$ such that $e_j = \text{top-rel}(a, Y)$ and, since e_j has higher weight than e_i, this means that we have processed a earlier and $a \in \overline{A}_{i-1}$. In either case, the claim holds. $\square$

Claim 7. *For all* $i \in [m]$, $R_i = \mathrm{Core}(\mathrm{Rel}(Y)) \cap E_i$.

Proof. We show both inclusions. Let $e \in R_i$. Then, there exists $j < i$ such that $e = e_j$, $e_j \in R_j$ and

$$e_j \in \mathrm{Core}(R_{j-1} \cup N_{j-1} + e_j) = \mathrm{Core}(R_j \cup N_j) = \mathrm{Core}(\mathrm{Rel}(Y) \cap E_j).$$

Since all the elements in $\mathrm{Rel}(Y) \setminus E_j$ have weight smaller than that of e_j we also get that $e_j \in \mathrm{Core}(\mathrm{Rel}(Y))$, and therefore $e_j \in \mathrm{Core}(\mathrm{Rel}(Y)) \cap E_i$.

For the other direction, let $e \in \text{Core}(\text{Rel}(Y)) \cap E_i \subseteq \text{Rel}(Y) \cap E_i$. This means that $e = e_j$ for some $j \leq i$ and $e_j \in \text{Core}(\text{Rel}(Y)) \cap E_j$. Since $\text{Core}(\text{Rel}(Y)) \subseteq \text{Rel}(Y)$, by Claim 6, we know that $e_j \in R_j \cup N_j$. However, since $e_j \in \text{Core}(\text{Rel}(Y)) \cap E_j$ we also have that $e_j \in \text{Core}(\text{Rel}(Y_{j-1})) \cap E_j$ and thus $e_j \in \text{Core}((R_{j-1} \cup N_{j-1}) + e_j)$, which implies that $e_j \in R_j \subseteq R_i$. □

Claim 8. *For all $i \in [m]$, $R'_i = \text{Greedy}(\delta(Y_i)) \cap \text{Core}(\text{Rel}(Y))$.*

Proof. By Claim 5, we have that $R'_i \subseteq \text{Greedy}(\delta(Y_i))$. In particular, for $j \leq i$, R'_i corresponds to the set of edges e_j from $\text{Greedy}(\delta(Y_i))$ which also satisfied $e_j \in \text{Core}((R_{j-1} \cup N_{j-1}) + e_j)$ and for which the agent a incident to e_j is not processed before step j. By Claim 6, the former is equivalent to $e_j \in \text{Core}((\text{Rel}(Y) \cap E_{j-1}) + e_j)$, which implies that $e_j \in \text{Core}(\text{Rel}(Y) \cap E_j) \subseteq \text{Core}(\text{Rel}(Y))$. For the latter, every edge $e_j \in \text{Greedy}(\delta(Y_i)) \cap \text{Core}(\text{Rel}(Y))$ is the top-relevant edge of its incident agent a and thus we have that $V_A(e_j) \notin \overline{A}_{j-1}$. Thus, the claim holds. □

Now we will prove the following four inequalities about the sizes of the constructed sets.

- $\mathbb{E}[|N'|] \geq \frac{p}{1-p} \mathbb{E}[|N|]$, $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ (7)
- $\mathbb{E}[|R|] + \mathbb{E}[|N|] \geq \frac{1-p}{p}(\mathbb{E}[|R'|] + \mathbb{E}[|N'|])$, $\qquad\qquad\qquad\qquad$ (8)
- $\mathbb{E}[|N'|] \leq k\,(\mathbb{E}[|R|] + \mathbb{E}[|N|])$, $\qquad\qquad\qquad\qquad\qquad\qquad\quad$ (9)
- $\mathbb{E}[|N'|] \leq k\,\mathbb{E}[|R|] + (k-1)\,\mathbb{E}[|N|]$ if all components of $\mathcal{F}$ are matroids. (10)

The proofs of (7) and (8) are a bit more subtle in the agent arrival case than in the edge arrival case.

Claim 9. *For all $i \in [m]$,*

$$(1-p)\mathbb{E}[|R'_i|] = p\,\mathbb{E}[|R_i|] \quad \text{and} \quad (1-p)\mathbb{E}[|N'_i|] = p\,\mathbb{E}[|N_i|].$$

Proof. Fix an agent a. Observe that the first time an edge $e_i \in \delta(a)$ is about to enter $R' \cup R$ (resp. $N' \cup N$), the result of c_a is used for the first time. If the coin is head, which happens with probability p, then e_i is added to R' (resp. N'), otherwise, to R (resp. N). Since $a \in \overline{A}_i$, no other edge $e_j \in \delta(a)$ with $j > i$ will ever be about to enter $R' \cup R$ (resp. $N' \cup N$) and thus c_a will not be used again. □

With (7), (8) and Claims 5, 6, 7 and 8 established, the rest of the proof follows exactly like the proof of Lemma 5. We first use the fact that $\mathcal{F}$ is a k-growth system to establish (9) and (10), and then we manipulate the four established inequalities in the same way as in Lemma 5 to obtain the desired conclusion. □

We summarize the guarantees of our Core lemma with respect to $w(\text{OPT})$. Note that the guarantees we obtain are identical to those of Theorem 4 for the edge-arrival setting.

Theorem 6. *Let $\mathcal{F}_{A,E} = (E, \mathcal{I}_A)$ be a combination of $\{(E_{A,j}, \mathcal{I}_{A,j})\}_{j=1}^{M_A}$, where each $(E_{A,j}, \mathcal{I}_{A,j})$ is a k_j-growth system, and $\mathcal{F}_{B,E} = (E, \mathcal{I}_B)$ be a combination of $\{(E_{B,j}, \mathcal{I}_{B,j})\}_{j=1}^{M_B}$, where each $(E_{B,j}, \mathcal{I}_{B,j})$ is an ℓ_j-growth system. Also, let $\mathcal{F} = (E, \mathcal{I})$ denote the combination of $\mathcal{F}_{A,E}$ and $\mathcal{F}_{B,E}$, let $\text{Core}(X)$ denote the set of critical elements with respect to X in $\mathcal{F}$, let $\text{Core}(X)$ denote the set of critical elements with respect to X in $\mathcal{F}$, i.e.,*

$$\text{Core}(X) = \bigcap_{j=1}^{M_A}((X \setminus E_{A,j}) \cup \text{PCore}_{j,A}(X \cap E_{A,j})) \cap \bigcap_{j=1}^{M_B}((X \setminus E_{B,j}) \cup \text{PCore}_{j,B}(X \cap E_{B,j})),$$

where $\mathrm{PCore}_{j,A}$ is the primitive core function of $(E_{A,j}, \mathcal{I}_{A,j})$ and $\mathrm{PCore}_{j,B}$ is the primitive core function of $(E_{B,j}, \mathcal{I}_{B,j})$, and let $Y = Y(p)$ be a random set where every agent $a \in A$ is in Y independently with probability p for some $p \in [0,1]$. Then, for $p = \sqrt{1 - \frac{1}{k+1}}$, we have

$$\mathbb{E}\left[w\left(\mathrm{Core}\left(\mathrm{Rel}(Y)\right)\right)\right] \geq \left(2 + \frac{1}{k} - 2\sqrt{1 + \frac{1}{k}}\right) w(\mathrm{OPT}(E)) > \left(\frac{1}{4k^2} - \frac{1}{8k^3}\right) w(\mathrm{OPT}(E)).$$

Furthermore, if all components are matroids, i.e., $\mathcal{F}$ is a k-matchoid, setting $p = 0$ for $k = 1$ and $p = 1 - 1/(2k)$ for $k \geq 2$ yields

$$\mathbb{E}\left[w\left(\mathrm{Core}\left(\mathrm{Rel}(Y)\right)\right)\right] \geq \begin{cases} w(\mathrm{OPT}(E)) & \text{if } k = 1, \\ \frac{1}{4k^2} \, w(\mathrm{OPT}(E)) & \text{if } k \geq 2. \end{cases}$$

Proof. Since the system $(E, \mathcal{I})$ is a k-system the weighted-greedy algorithm yields a $1/k$-approximation with respect to the weight of $\mathrm{OPT}(E)$. Therefore,

$$\mathbb{E}[w(\mathrm{Greedy}(\delta(Y)))] \geq \frac{1}{k}\mathbb{E}[w(\mathrm{OPT}(\delta(Y)))] \geq \frac{1}{k}\mathbb{E}[w(\mathrm{OPT}(E)) \cap \delta(Y)] = \frac{p}{k}w(\mathrm{OPT}(E)).$$

Thus, by Lemma 6,

$$\mathbb{E}\left[w\left(\mathrm{Core}\left(\mathrm{Rel}(Y)\right)\right)\right] \geq \frac{(p - (1-p)k)(1-p)}{pk}w(\mathrm{OPT}(E)).$$

Optimizing p yields $p = \sqrt{1 - \frac{1}{k+1}}$ and

$$\mathbb{E}\left[w\left(\mathrm{Core}\left(\mathrm{Rel}(Y)\right)\right)\right] \geq \left(2 + \frac{1}{k} - 2\sqrt{1 + \frac{1}{k}}\right) w(\mathrm{OPT}(E)).$$

Now, consider the case where all components are matroids. If $k = 1$, then the components of $\mathcal{F}$ are pairwise disjoint and thus the system is a direct sum of matroids, so it is itself a matroid. In this case, for $p = 0$ we have $\mathrm{Rel}(Y) = E$ and, for all $j \in [M]$, $\mathrm{Core}_j(E) = \mathrm{OPT}(E)$. Thus,

$$\mathbb{E}\left[w\left(\mathrm{Core}\left(\mathrm{Rel}(Y)\right)\right)\right] = w\left(\mathrm{Core}(E)\right) = w(\mathrm{OPT}(E)).$$

If $k \geq 2$, by Lemma 6,

$$\mathbb{E}\left[w\left(\mathrm{Core}\left(\mathrm{Rel}(Y)\right)\right)\right] \geq \frac{(p - (1-p)(k-1))(1-p)}{k}w(\mathrm{OPT}(E)).$$

Optimizing p yields $p = 1 - \frac{1}{2k}$ and

$$\mathbb{E}\left[w\left(\mathrm{Core}\left(\mathrm{Rel}(Y)\right)\right)\right] \geq \frac{1}{4k^2}w(\mathrm{OPT}(E)). \qquad \square$$

5.2 Order-Oblivious Core-Selecting Algorithms

As discussed previously, for edge arrivals, our guarantees follow directly by using the Core lemma in conjunction with our black-box algorithm from Theorem 3. For agent arrivals, this is no longer the case, for two main reasons.

Problems with the Edge-Arrival Approach. First, suppose we try to repeat the approach of our black-box algorithm from Theorem 3 for edge-arrival: split the remaining agents into two parts S_1 and S_2, use S_1 to define a calling order, and then process S_2 in that order. Of course, we cannot "call" individual edges and when we "call" an agent, we observe all their incident edges simultaneously. The problem arises from the fact that we cannot "call" all edges incident to a single item. This asymmetry implies that the order in which our algorithm should "call" the agents should depend on the Hull of $\mathcal{F}_{A,E}$, i.e. the system from the agent side, to guarantee that any agent whose top-relevant edge lies in the core of $\mathcal{F}_{A,E}$ has a constant probability of being considered for selection.

However, even if we call the agents in the order specified by the Hull of $\mathcal{F}_{A,E}$, this still does not resolve all our problems, as we still might not be able to select a decent fraction of the common core of $\mathcal{F}_{A,E}$ and $\mathcal{F}_{B,E}$. This is because the edges that our algorithm selects (that are in the core of the top-relevant edges of the sample with respect to $\mathcal{F}_{A,E}$) may be dependent in the item-side system $\mathcal{F}_{B,E}$, since we do not have any guarantees that these edges are also in the core of the top-relevant edges of the sample with respect to $\mathcal{F}_{B,E}$. Ideally, we would want to select edges that are also in the item-side core, but the arrival order defined by the agent side does not ensure we pick them; indeed, each time we select an edge outside the item core, we may block one or many later edges in the item core.

To circumvent this issue, we focus our attention to systems $\mathcal{F}_{B,E}$ on the item side that admit an *order-oblivious core-selecting algorithm*; one that, after a uniformly random sample phase, ensures that every element in the core of $\mathcal{F}_{B,E}$ that was not sampled is selected with constant probability, regardless of the arrival order of the elements in the sampling phase.

Definition 10. Given an independence system $(S, \mathcal{I})$ with hidden weights on the elements, we say that an algorithm $\mathcal{A}_r$ that uses r internal random bits and selects a set of elements $\mathrm{ALG} \in \mathcal{I}$ is (q, α) *order-oblivious core-selecting* (OOCS) if, for any $X \subseteq S$ and any $e \in \mathrm{Core}(X)$, if Y is a random set where every element of $X - e$ appears independently with probability q, then

$$\mathbb{E}_{Y \sim X - e} \left[\min_{\sigma} \Pr_r \left[e \in \mathrm{ALG} \mid e \notin Y \right] \right] \geq \alpha,$$

where the minimum is taken over all orderings σ of the elements in $X \setminus Y$.

To better explain the above definition, we describe how an (q, α)-OOCS algorithm $\mathcal{A}_r$ works.

1. Before the process begins, $\mathcal{A}_r$ is presented with $(S, \mathcal{I})$.

2. An adversary is then allowed to delete an arbitrary set of elements from S and restrict the true ground set of the process to an arbitrary $X \subseteq S$ that is *unknown* to $\mathcal{A}_r$.

3. The process begins and $\mathcal{A}_r$ observes a random subset $Y \subseteq X$, which we call the sample, that includes each element of X independently with probability q. $\mathcal{A}_r$ learns the identities and weights of the elements in Y, and does not select any of them.

4. $\mathcal{A}_r$ then observes the remaining elements $\overline{Y} = X \setminus Y$ in an adversarial order that may depend on the sample Y, but not on $\mathcal{A}_r$'s internal random bits r.

5. Upon observing an element of $\overline{Y}$, $\mathcal{A}_r$ must immediately and irrevocably decide whether to select it – adding it to ALG – subject to maintaining $\mathrm{ALG} \in \mathcal{I}$.

6. For any arbitrary restriction $X \subseteq S$, and conditioning only on the event that an element $e \in \mathrm{Core}(X)$ is not sampled, i.e. on $e \in \overline{Y}$, we require that the expected probability that $\mathcal{A}_r$

selects e, regardless of the arrival order of the elements in $\overline{Y}$ in the selection phase, is at least α. The expectation is over the random sample Y and the probability is over the algorithm's internal random bits r.

Notice that Definition 10 is similar to the notion of a c-OPT competitive algorithm for random-order matroid intersection secretary problems, introduced by [34], except that here we require selecting *core elements* rather than *elements in the optimal basis*. However, it is actually a slightly stronger requirement since an order-oblivious core-selecting algorithm has to work for any restriction of the original system, without knowing the true ground set, and give the same guarantees. This stronger requirement is necessary for our algorithm, in order to resolve our final obstacle, which is that our agent-side sample does not induce a uniformly random sample of items or even of item-adjacent edges. We describe our idea to circumvent this last problem in Section 5.3.

Given the above, one may wonder whether (q, α)-OOCS algorithms exist for constant q and α for any feasibility constraints. As it turns out, several matroid classes have known algorithms that are (q, α)-OOCS for constant q and α, including uniform, partition, laminar and co-graphic matroids. In addition, even though no order-oblivious core-selecting algorithm with good guarantees is known for transversal matroids, graphic matroids and knapsack systems, we discuss in Section 5.4 how to transform them into systems that we can use for our algorithms with minimal losses in the competitive ratio guarantee.

Setting aside briefly the requirement for the algorithm to work for arbitrary restrictions, notice that, for matroids, every order-oblivious algorithm also selects elements in the core of any ground set, since the elements of the core are exactly the elements of the optimal basis. However, this need not be the case for general independence systems.

Next, we present an example of a simple $(1/2, 1/2)$-OOCS algorithm for rank-1 matroids.

Example 1. Let $(S, \mathcal{I})$ be a uniform rank-1 matroid. The algorithm takes the sample, computes the maximum-weight sampled element e^* and, in the non-sampled phase, selects the first observed element f with $w(f) > w(e^*)$. If the sample is empty, it selects the first non-sampled element it observes. Notice that this algorithm does not require knowing $|S|$ a priori. To see that it is $(1/2, 1/2)$-OOCS, fix any $X \subseteq S$ and let $e \in \mathrm{Core}(X) = \mathrm{OPT}(X)$ be the maximum-weight element of X. For $Y = S(q)$ and $q = 1/2$, we condition on $e \notin Y$. Feed $Y \cap X$ to the algorithm as the sample, and then present $X \setminus Y$ adversarially. Let e_2 be the element in X with second-highest weight. With probability $1/2$, we have $e_2 \in Y$. In this case, $w(e^*) \geq w(e_2)$ and the only element f that satisfies $w(f) > w(e^*)$ is e, so e is selected in the second phase. Hence $\Pr\left[e \in \mathrm{ALG} \mid e \in \overline{Y}\right] \geq 1/2$.

Next, we show that, for matroids, random-order constant-probability-competitive algorithms can be transformed to order-oblivious core-selecting algorithms.

Observation 2. *For any matroid $M = (S, \mathcal{I})$, if $\mathcal{A}$ is a (q, α)-OOCS algorithm for M, there exists a $(1-q)\alpha$-probability competitive algorithm for M, when the elements in S arrive in uniformly random order*

Proof. Let ALG denote the set of elements selected by $\mathcal{A}$ in the random-order setting, Y denote the sample and $\overline{Y} = S \setminus Y$. For every $e \in \mathrm{OPT}$,

$$\Pr\left[e \in \mathrm{ALG}\right] = \Pr\left[e \in \overline{Y}\right] \, \mathbb{E}_{Y-e}\left[\Pr\left[e \in \mathrm{ALG} \mid e \in \overline{Y}\right]\right]$$
$$\geq (1-q) \, \mathbb{E}_{Y-e}\left[\min_{\sigma} \Pr\left[e \in \mathrm{ALG} \mid e \in \overline{Y}, \sigma\right]\right] \geq (1-q)\alpha. \qquad \square$$

With regard to our model, when the given item-side constraint is a combination of individual systems, having an OOCS algorithm for the individual components suffices to obtain an OOCS algorithm for the overall constraint, as the following lemma shows. Recall that, for a combination of M systems, PCore_j denotes the primitive core of the j-th individual system for $j \in [M]$, and by Definition 8, for any $X \subseteq S$,

$$\mathrm{Core}(X) := \bigcap_{j=1}^{M} \left((X \setminus S_j) \cup \mathrm{PCore}_j(X \cap S_j) \right).$$

Lemma 7. *Let $\mathcal{F} = (S, \mathcal{I})$ be a combination of systems $\{(S_j, \mathcal{I}_j)\}_{j=1}^{M}$ where each $(S_j, \mathcal{I}_j)$ admits a (q_j, α_j)-OOCS algorithm $\mathcal{A}_j$. Let $k = \max_{e \in S} |\{j : e \in S_j\}|$ denote the maximum number of individual systems any element participates in and $q^\star = \max_{j=1}^{M} q_j$. Then, for*

$$q := 1 - (1 - q^\star)^k \quad and \quad \alpha := \min_{e \in S} \prod_{j \,:\, e \in S_j} \alpha_j \,,$$

there exists a (q, α)-OOCS algorithm $\mathcal{A}$ for the combination $(S, \mathcal{I})$ of $\{(S_j, \mathcal{I}_j)\}_{j=1}^{M}$.

Proof. For every element e and every $j \in [M]$, we independently draw $t'_j(e) \sim U[0, 1]$ from the uniform distribution in the interval $[0, 1]$. Then, for every $e \in S$ we denote by $j_\ell(e)$ the index of the ℓ-th individual system in the ordering $(S_1, \mathcal{I}_1), (S_2, \mathcal{I}_2), \ldots, (S_M, \mathcal{I}_M)$ that e participates in. For $\ell = 1, 2, \ldots, k$, let $t_\ell(e)$ be equal to $t'_{j_\ell(e)}$. If e participates in $h < k$ systems, we let $t_{h+1}(e), \ldots, t_k(e) \sim U[0, 1]$ be drawn independently from all other random variables.

Consider the following sets:

$$\forall j \in [M], \; Y_j := \{e \in S_j \mid t_j(e) \le q_j\}, \quad and \quad Y := \{e \in S \mid \exists \ell \in [k] \text{ s.t. } t_\ell(e) \le q^\star\}.$$

Notice that for every $j \in [M]$ we have $Y \supseteq Y_j$, since $q_j \le q^\star$. Furthermore, for every element $e \in S$, we have $\Pr[e \in Y] = 1 - (1 - q^\star)^k = q$, since even if e participates in less than k systems, we still toss k coins for e and observe whether or not one of them is less than $q^\star$ to decide whether $e \in Y$ or not. Furthermore, the events $(e \in Y)_{e \in S}$ are mutually independent by construction. Therefore, Y is a truly random sample of S with inclusion probability q. Similarly, the random sets $(Y_j)_{j \in [M]}$ are also mutually independent random variables since they depend on the result of different random coins.

Let $X \subseteq S$ denote the unknown arbitrary set that the adversary restricts our ground set to before the process begins. Our algorithm will receive a sample in which each element of X is independently selected with probability q. Notice that $Y \cap X$ is a random set distributed exactly according to the sample of our distribution, so in what follows we assume that this set is really $Y \cap X$. Observe also that Y_j is a random subset of S_j such that for every element $e \in S_j$, we have $e \in Y_j$ independently with probability q_j.

By the same logic as before, we have that $Y_j \cap X$ is a random set distributed exactly according to the sample expected by $\mathcal{A}_j$, restricted to X. Since $Y_j \subseteq Y$ for all j, our algorithm $\mathcal{A}$ proceeds as follows:

1. $\mathcal{A}$ receives the set of sampled elements which, as discussed, is identically distributed to $Y \cap X$ so we assume it is really that set.

2. Next, $\mathcal{A}$ will create simulated realizations of the random variables $t'_\ell(e)$ for $\ell = 1, 2, \ldots, k$ and $e \in X \cap Y$ that are compatible with the definition of Y, so they are identically distributed. To do that each element $e \in Y \cap X$ and $j \in [M]$, $\mathcal{A}$ will draw $t'_j(e) \sim U[0,1]$, as described above. It will repeat this process until one of the $t_\ell(e)$'s is less than q, for $\ell = 1, 2, \ldots, k$. When this happens, the conditioning is correct and we assume that these are the correct draws of the $t_\ell(e)$'s.

3. Using the previous step, for every $j \in [M]$, $\mathcal{A}$ creates a simulated set by including every element $e \in Y \cap X \cap S_j$ for which the random variable $t_\ell(e)$, for the value of ℓ that corresponds to system $(S_j, \mathcal{I}_j)$, is less than q_j. The resulting set is now distributed identically to $Y_j \cap X$, so in what follows we assume that this set is really $Y_j \cap X$.

4. For every individual system $(S_j, \mathcal{I}_j)$, $\mathcal{A}$ feeds $Y_j \cap X$ to $\mathcal{A}_j$ as its sample. Since $Y_j \cap X$ is identically distributed to the sample expected by $\mathcal{A}_j$ restricted to X, we now start the second, selection phase for every system $j \in [M]$.

5. Let $\overline{Y} = X \setminus Y$ denote the set of elements in the selection phase. Assume that $\mathcal{A}$ observes them in an adversarial arrival order π.

6. At every step i, $\mathcal{A}$ observes an element e_i. For every j such that $e_i \in S_j$, $\mathcal{A}$ "feeds" e_i to $\mathcal{A}_j$ in the selection phase. If all such $\mathcal{A}_j$ accept e_i, $\mathcal{A}$ accepts e_i as well; otherwise it rejects e_i.

7. After all elements of $\overline{Y}$ end, $\mathcal{A}$ feeds all the elements of $(Y \setminus Y_j) \cap S_j$ to $\mathcal{A}_j$, for each j in arbitrary order. Nothing is really selected here, but the process is done anyway so that each $\mathcal{A}_j$ receives all the elements from $\overline{Y} \cap S_j$ on its selection phase. Then, the process ends.

We now show that $\mathcal{A}$ is (q, α)-OOCS. Previously, we showed that the sample of $\mathcal{A}$ is distributed identically to $Y \cap X$, which is a random subset of X where each element appears independently with probability q. Fix an element $e \in \mathrm{Core}(X)$ and let ALG denote the set selected by $\mathcal{A}$. Next, we show that

$$\mathbb{E}_{Y \sim X - e}\left[\min_\sigma \Pr_r\left[e \in \mathrm{ALG} \mid e \notin Y\right]\right] \geq \alpha.$$

We have

$$\mathbb{E}_{Y \sim X - e}\left[\min_\sigma \Pr_r\left[e \in \mathrm{ALG} \mid e \notin Y\right]\right]$$

$$= \mathbb{E}_{Y \sim X - e}\left[\min_\sigma \Pr_r\left[\bigwedge_{j:\, e \in S_j} \mathcal{A}_j \text{ accepts } e \;\middle|\; e \notin Y\right]\right]$$

$$= \mathbb{E}_{Y \sim X - e}\left[\min_\sigma \Pr_r\left[\bigwedge_{j:\, e \in S_j} \mathcal{A}_j \text{ accepts } e \;\middle|\; \forall j \text{ s.t. } e \in S_j,\ e \notin Y_j\right]\right]$$

$$= \mathbb{E}_{Y_1 \overset{q_1}{\sim} (X-e) \cap S_1, \ldots, Y_M \overset{q_M}{\sim} (X-e) \cap S_M}\left[\min_\sigma \prod_{j:\, e \in S_j} \Pr_r\left[\mathcal{A}_j \text{ accepts } e \mid e \notin Y_j\right]\right]$$

$$= \prod_{j:\, e \in S_j} \mathbb{E}_{Y_1 \overset{q_1}{\sim} (X-e) \cap S_1, \ldots, Y_M \overset{q_M}{\sim} (X-e) \cap S_M}\left[\min_\sigma \Pr_r\left[\mathcal{A}_j \text{ accepts } e \mid e \notin Y_j\right]\right]$$

$$\geq \prod_{j:\, e \in S_j} \alpha_j \geq \min_{e' \in S} \prod_{j:\, e' \in S_j} \alpha_j = \alpha,$$

35

where the second equality follows from the fact that, for any two systems $(S_j, \mathcal{I}_j)$ and $(S_{j'}, \mathcal{I}_{j'})$ for which $e \in S_j$ and $e \notin S_{j'}$, it is irrelevant to $\mathcal{A}_j$ whether or not $e \in Y_{j'}$, given that $e \notin Y_j$, the third equality follows from the fact that the Y_j's are independent subsets of $(X - e) \cap S_j$, conditioned on $e \notin Y$, the fourth equality follows from the independence of the decisions of all $\mathcal{A}_j$'s and the fifth inequality follows from the fact that every $\mathcal{A}_j$ is a (q_j, α_j)-OOCS algorithm. $\qquad\square$

5.3 Free-Order Secretary Algorithm for Agent Arrivals

As discussed earlier, we are now faced with our final obstacle. Even if we have order-oblivious core-selecting algorithms for the constraints on the item side, they still require a uniformly random sample to work. However, when we sample agents, we simultaneously observe all edges incident to those agents. From the perspective of the item side, this set is neither a uniformly random sample of item-adjacent edges, nor does it induce a uniformly random sample of the corresponding items. This is easy to see: consider an extreme case where, originally, there are only two items, one connected to all agents and the other connected only to a single agent. After our parallel extension construction and our transfer of $\mathcal{F}_B$ from the items to the edges, every edge is connected to a unique agent and a unique item, but we have many copies of the first item and only a single copy of the second one. So a uniformly random sample of edges does not correspond to a uniformly random sample of items, and thus no such sample can be used an an input to an order-oblivious core-selecting algorithm.

However, we are able to get around this issue by observing that, after our very first sampling phase, each agent contributes at most one greedy-relevant edge – their top greedy-relevant edge, if one exists – to the set of greedy-relevant edges. Now the idea is to do the following: after the first sample, we restrict our ground set for the item side to the (unknown) set of top-relevant edges. Let's call this set X. Essentially, X forms the true ground set of our order-oblivious core-selecting algorithms for the item side. Now, even though we cannot "call" items, we can perform another sampling phase, independently from our first sampling phase, to ensure that the sampled top-relevant edges in this second sampling phase forms a uniformly random subset of the new ground set X.

This insight motivates the stronger requirement we impose on order-oblivious core-selecting algorithms: they must maintain their guarantees even if, at the beginning of the process, an arbitrary and unknown subset of the ground set is removed (corresponding here to the edges that are not top-relevant from the agent side). In other words, the algorithm must perform equally well under any adversarial restriction of the ground set applied before the process begins.

We finally have all the ingredients we need to design an algorithm for the agent-arrival setting. Note that, for the Hull function, we use the Hull of $\mathcal{F}_A$, the system on the agent side. Algorithm 5 receives as input the following:

1. $\mathcal{F}_{A,E} = (E, \mathcal{I}_A)$ is the (edge-induced) agent-system.

2. $\mathcal{F}_{B,E} = (E, \mathcal{I}_B)$ is the (edge-induced) item-system.

3. $\mathcal{A}$ is a (q, α)-OOCS algorithm for $\mathcal{F}_{B,E}$.

4. A parameter p.

Algorithm 5: Agent-arrival Free-Order Secretary

Output: An independent set ALG

1 ALG $\leftarrow \emptyset$

2 $p_1 \leftarrow p$

3 $p_2 \leftarrow p_1 + (1 - p_1)q$

4 $p_3 \leftarrow (1 + p_2)/2$

5 **for** *each* $a \in A$ **do** Choose t_a independently and uniformly from $(0, 1)$

6 $Y \leftarrow \{a \mid t_a \in (0, p_1)\}$

7 $S_0 \leftarrow \{a \mid t_a \in [p_1, p_2)\}$

8 $S_1 \leftarrow \{a \mid t_a \in [p_2, p_3)\}$

9 $S_2 \leftarrow \{a \mid t_a \in [p_3, 1]\}$

10 Observe (without accepting) all agents in $Y \cup S_0 \cup S_1$

11 **unseen** $\leftarrow S_2$ `// The set of agents not yet revealed`

12 Compute the edge-sets $E_0 = \text{Rel}(Y) \cap S_0$ and $E_1 = \text{Rel}(Y) \cap S_1$

13 Run $\mathcal{A}$ on $\mathcal{F}_B|_{\text{Rel}(Y)} = (E, \mathcal{I}_B)|_{\text{Rel}(Y)}$ using E_0 as the sample

14 Sort the elements of E_1 in decreasing order of weights as $e_1, \ldots, e_m$. Let $a_1, \ldots, a_m$ be the respective agents such that e_i is incident to a_i

15 **for** $j = 1$ *to* m **do**

16 $\quad$ Let $Q = \textbf{unseen} \cap \text{Hull}_A(\{a_1, \ldots, a_j\})$

17 $\quad$ **for** *each* $a \in Q$ *in uniformly random order* **do**

18 $\quad\quad$ **unseen** $\leftarrow$ **unseen** $- a$ `// Reveal a`

19 $\quad\quad$ **if** top-rel$(a, Y) = \perp$ **then** skip to the next $a \in Q$

20 $\quad\quad$ $f \leftarrow$ top-rel(a, Y)

21 $\quad\quad$ **if** $w(f) > w(e_j)$ *and* $\mathcal{A}$ *selects* f *and* ALG $+f \in \mathcal{I}_A \cap \mathcal{I}_B$ **then**

22 $\quad\quad\quad$ ALG $\leftarrow$ ALG $+f$

23 $\quad\quad$ **end**

24 $\quad$ **end**

25 **end**

26 $Q \leftarrow$ **unseen**

27 **for** *each* $a \in Q$ *in uniformly random order* **do**

28 $\quad$ **if** top-rel$(a, Y) = \perp$ **then** skip to the next $a \in Q$

29 $\quad$ $f \leftarrow$ top-rel(a, Y)

30 $\quad$ **if** $\mathcal{A}$ *selects* f *and* ALG $+f \in \mathcal{I}_A \cap \mathcal{I}_B$ **then**

31 $\quad\quad$ ALG $\leftarrow$ ALG $+f$

32 $\quad$ **end**

33 **end**

34 **return** ALG

Theorem 7. *Let $\mathcal{F}_{A,E} = (E, \mathcal{I}_A)$ be a combination of $\{(E_{A,j}, \mathcal{I}_{A,j})\}_{j=1}^{M_A}$, where each $(E_{A,j}, \mathcal{I}_{A,j})$ is a k_j-growth system, and $\mathcal{F}_{B,E} = (E, \mathcal{I}_B)$ be a combination of $\{(E_{B,j}, \mathcal{I}_{B,j})\}_{j=1}^{M_B}$, where each $(E_{B,j}, \mathcal{I}_{B,j})$ is an ℓ_j-growth system. Also, let $\mathcal{F} = (E, \mathcal{I})$ denote the intersection of $\mathcal{F}_{A,E}$ and $\mathcal{F}_{B,E}$ and $\mathcal{A}$ be a (q, α)-OOCS algorithm for $\mathcal{F}_{B,E}$. Then, for $p = \sqrt{1 - 1/(k+1)}$, Algorithm 5 has a*

utility-competitive ratio at least

$$\frac{\alpha(1-q)}{4} \left(2 - 2\sqrt{1 + \frac{1}{k}} + \frac{1}{k} \right) > \alpha(1-q) \left(\frac{1}{16k^2} - \frac{1}{32k^3} \right).$$

Furthermore, if all components are matroids, i.e. if $(E, \mathcal{I}_A \cap \mathcal{I}_B)$ is a k-matchoid with $k_{j,A} = 1$ and $\ell_{j,B} = 1$ for every j and $|\{j : e \in E_{A,j}\}| + |\{j : e \in E_{B,j}\}| \leq k$ for every $e \in E$, for $p = 1 - \frac{1}{2k}$, Algorithm 5 has a utility-competitive ratio at least

$$\frac{\alpha(1-q)}{16k^2}.$$

Proof. First, it is clear that the algorithm returns an independent set since, before adding any edge e to ALG, the algorithm checks whether ALG $+e \in \mathcal{I}$. We first condition on a fixed realization of the sampled set Y of agents. Let

$$\text{Core}_{\text{Common}}(\text{Rel}(Y)) = \text{Core}_A(\text{Rel}(Y)) \cap \text{Core}_B(\text{Rel}(Y)),$$

where

$$\text{Core}_A(\text{Rel}(Y)) = \bigcap_{j=1}^{M_A} \text{Core}_{A,j}(\text{Rel}(Y)) \text{ and } \text{Core}_B(\text{Rel}(Y)) = \bigcap_{j=1}^{M_B} \text{Core}_{B,j}(\text{Rel}(Y)),$$

where $\text{Core}_{A,j}$ ($\text{Core}_{B,j}$ respectively) corresponds to the individual core function of the system component $(E_{A,j}, \mathcal{I}_{A,j})$ (resp. $(E_{B,j}, \mathcal{I}_{B,j})$). We show that each element in $\text{Core}_{\text{Common}}(\text{Rel}(Y))$ is selected with probability at least $\alpha(1-q)/4$. Let $e \in \text{Core}_{\text{Common}}(\text{Rel}(Y)) \subseteq \text{Rel}(Y)$. Condition now on the event that $e \notin S_0$ (i.e. $e \in S_1 \cup S_2$).

Recall that, $e \in \text{Core}_B(\text{Rel}(Y))$, and observe that $E_0 = \text{Rel}(Y) \cap S_0$ is a random subset of $\text{Rel}(Y)$ obtained in such a way that every element of $\text{Rel}(Y) - e$ is in E_0 independently with probability

$$\frac{p_2 - p_1}{1 - p_1} = q.$$

Since $\mathcal{A}$ is (q, α)-OOCS for $\mathcal{F}_{B,E}$, if we "feed" E_0 as the sample of $\mathcal{A}$, we conclude that, for every possible ordering σ of $\text{Rel}(Y) \setminus E_0$, the event

$$\mathcal{E}_0 : e \text{ is accepted by } \mathcal{A} \text{ when revealed,}$$

which depends only on the internal random coins of $\mathcal{A}$ (recall that the random coins r of $\mathcal{A}$ are independent from the internal arrival order of the elements), satisfies

$$\mathbb{E}_{S_0 - e} \left[\Pr_r [\mathcal{E}_0] \,\Big|\, e \notin S_0 \right] \geq \alpha.$$

Here we are crucially using the fact that $\mathcal{A}$ is order-oblivious, since the order in which $\text{Rel}(Y) \setminus E_0$ is presented to $\mathcal{A}$ is chosen by the instructions of our algorithm. Next, condition on the realization of $S_0 - e$ – in other words, $S_1 - e$ and $S_2 - e$ are still undecided – and let $E_2 = \text{Rel}(Y) \cap S_2$.

Let h_1 as the heaviest element in $E_1 - e = \text{Rel}(Y) - e$ such that e is in the agent-hull of the elements with a weight greater than h_1 in $E_1 - e$. In other words,

$$h_1 := \arg\max \{w(h) \mid h \in E_1 - e, \text{ and } e \in \text{Hull}_A(\{g \in E_1 - e \mid w(g) \geq w(h)\})\}.$$

Similarly for h_2 in $E_2 - f$:

$$h_2 := \arg\max\{w(h) \mid h \in E_2 - e, \text{ and } e \in \mathrm{Hull}_A(\{g \in E_2 - e \mid w(g) \geq w(h)\})\}.$$

If there are no such elements, we set $h_1 = \bot$ and $h_2 = \bot$, respectively, with $w(\bot) = 0$.

Since, conditioned on Y and $S_0 - e$, the sets $(E_1 - e)$ and $(E_2 - e)$ are identically distributed, we have that $\Pr[w(h_1) \geq w(h_2)] = 1/2$, regardless of whether one or both of h_1 and h_2 are $\bot$. Next, we condition on the independent event that $f \in S_2$, and conclude that both $\mathcal{E}_1 : w(h_1) \geq w(h_2)$ and $\mathcal{E}_2 : e \in S_2$ occur together with probability at least $1/4$ given the previous conditions on Y and $S_0 - e$. So, in what follows, we also suppose that the two events $\mathcal{E}_1$ and $\mathcal{E}_2$ occur.

Next, recall that, $E_1 = \mathrm{Rel}(Y) \cap S_1$ is ordered in decreasing order of weights as $e_1, \ldots, e_m$. Let j^* be such that $h_1 = e_{j^*}$ (where $j^* = m + 1$ if $h_1 = \bot$). By construction, e is revealed by the algorithm precisely in iteration j^*, when it first enters the set Q. Furthermore, since $e \in \mathrm{Core}_A(\mathrm{Rel}(Y)) = \bigcap_{j=1}^{M_A} \mathrm{Core}_{A,j}(\mathrm{Rel}(Y))$ we conclude, by the relations between Hull_A and Core_A and the monotonicity of Hull_A that

$$f \notin \mathrm{Hull}_A(\{g \in \mathrm{Rel}(Y) \mid w(g) > w(e)\}) \supseteq \mathrm{Hull}_A(\{g \in \mathrm{Rel}(Y) \cap S_1 \mid w(g) > w(e)\}).$$

Thus, $w(e) > w(h_1) \geq w(h_2)$, so upon the arrival of e, we satisfy that $w(e) > w(e_{j^*})$.

Let ALG' denote the solution immediately before e is revealed and note that all elements in ALG' are in $\mathrm{Rel}(Y) \cap S_2 = E_2$ and have weights larger than $w(e_{j^*})$. There are two possibilities. First, if $h_2 \neq \bot$, since $w(e_{j^*}) = w(h_1) \geq w(h_2)$, by the definition of h_1 and h_2, we have

$$f \notin \mathrm{Hull}_A(\{g \in E_2 \mid w(g) \geq w(e_{j^*})\}) \supseteq \mathrm{Hull}_A(\mathrm{ALG}'),$$

where the last inequality follows by the monotonicity of Hull, and since all elements considered by ALG' for addition had weights larger than $w(e_{j^*})$.

Second, if $h_2 = \bot$ then, again by the monotonicity of Hull_A, we have

$$f \notin \mathrm{Hull}_A(E_2) \supseteq \mathrm{Hull}_A(\mathrm{ALG}').$$

It follows that at the moment in which f is revealed, $f \notin \mathrm{Hull}_A(\mathrm{ALG}')$, and thus $\mathrm{ALG}' + f \in \mathcal{I}_A$ by the properties of Hull_A.

Finally, let $\mathcal{A}_0$ be the set of elements that $\mathcal{A}$ selects before e's arrival. We know that $\mathrm{ALG}' \subseteq \mathcal{A}_0$ by the description of our algorithm. So, if $\mathcal{A}$ selects e upon arrival we would immediately have that $\mathrm{ALG}' + e \in \mathcal{I}_A \cap \mathcal{I}_B$ since the set of elements that $\mathcal{A}$ selects are independent in $\mathcal{I}_B$. Therefore, so long as $\mathcal{A}$ selects e (i.e., as long as $\mathcal{E}_0$ occurs), e will be added to ALG. Putting it all together:

$$\begin{aligned}
\Pr[e \in \mathrm{ALG} \mid Y] &= \Pr[e \notin S_0] \cdot \mathbb{E}_{S_0 - e} \Pr[e \in \mathrm{ALG} \mid Y, e \notin S_0] \\
&\geq \frac{1 - p_2}{1 - p_1} \mathbb{E}_{S_0 - e} \Pr[\mathcal{E}_0 \wedge \mathcal{E}_1 \wedge \mathcal{E}_2 \mid Y, e \notin S_0] \\
&= (1 - q)\mathbb{E}_{S_0 - e}[\Pr[\mathcal{E}_0 \mid Y, e \notin S_0] \cdot \Pr[\mathcal{E}_1 \wedge \mathcal{E}_2 \mid Y, e \notin S_0]] \\
&\geq (1 - q)\mathbb{E}_{S_0 - e}[\Pr[\mathcal{E}_0 \mid Y, e \notin S_0]] \cdot \frac{1}{4} \\
&\geq \frac{\alpha(1 - q)}{4}.
\end{aligned}$$

Since this holds for every realization of Y and every $e \in \text{Core}_{\text{Common}}(\text{Rel}(Y)))$, for the case of general combinations of growth-systems and for $p = \sqrt{1 - \frac{1}{k+1}}$, we conclude that

$$\mathbb{E}[w(\text{ALG})] \geq \frac{\alpha(1-q)}{4} \mathbb{E}\left[w(\text{Core}_{\text{Common}}(\text{Rel}(Y)))\right]$$

$$\geq \frac{\alpha(1-q)}{4} \left(2 - 2\sqrt{1 - \frac{1}{k}} + \frac{1}{k}\right) w(\text{OPT}(E)),$$

where the last inequality follows from Lemma 6.

Similarly, for the case of k-matchoids and $p = 1 - \frac{1}{2k}$, we conclude that

$$\mathbb{E}[w(\text{ALG})] \geq \frac{\alpha(1-q)}{4} \mathbb{E}\left[w(\text{Core}_{\text{Common}}(\text{Rel}(Y)))\right]$$

$$\geq \frac{\alpha(1-q)}{16k^2} w(\text{OPT}(E)),$$

where again the last inequality follows from Lemma 6. $\qquad\square$

5.4   Remarks

It is worth noting that our free-order agent-arrival technique also applies, even when some item-side components do not admit an order-oblivious core-selecting (OOCS) algorithm. In particular, by using techniques similar to the reduce-and-solve construction of Feldman, Svensson, and Zenklusen [34] for random-order matroid intersection, we can reduce those components to instances that do admit an OOCS and then apply our method unchanged, incurring only the loss dictated by the reduction. Without going into full generality, we briefly explain how to deal with some specific examples.

Graphic Matroids.   We can reduce graphic matroids to a (random) partition matroid using an idea of Korula and Pál [48]. Given a graph $G = (V, E)$, before the process begins, let π denote a uniformly random permutation of the vertices V. We create a partition matroid where every vertex u is associated with a partition class. For any edge $e = \{u, v\}$, we place e in the partition class of u if and only if $\pi(u) < \pi(v)$. Now we can use an $(1/2, 1/2)$-OOCS for the resulting partition matroid, incurring only an extra loss of $1/2$ by the reduction. What is actually done in the agent-arrival model is to replace the graphic matroid component by the partition matroid just defined, even before starting the agent-arrival algorithm (in particular, greedy, and the greedy-relevant edges are computed with respect to this new combination of systems). Note that in the transformed instance we are not enforcing selection from the core of the original graphic; instead, we select from the core of this new partition matroid. Nevertheless, the expected weight guarantee remains sufficiently high because the analysis applies to the greedy-relevant edges of this new combined system.

Interestingly, reductions like the one described above can be designed for any matroid class that has the α-partition property [1], so long as the randomized partition can be created a priori, before observing any elements. For example, this is the case for k-column sparse matroids (see [69]).

Transversal Matroids The idea is a little different here. Assume the item-side constraint is a combination in which one component is a transversal matroid T. Represent T by its standard bipartite presentation $H = (L, R; E_H)$: the left vertices L are the original ground elements, the right vertices R are auxiliaries, and a set $F \subseteq L$ is independent in T iff there exists a matching in H that saturates F.

Duplication and partition intersection. For each $e \in L$ and each neighbor $r \in N_H(e)$, create a copy e_r (think of it as the edge (e, r)), and let $U = \{e_r : e \in L,\ r \in N_H(e)\}$ be the new ground set. Define two partition matroids on U:

$$\mathcal{P}_{\text{left}} : \text{ at most one element from each block } \{e_r : r \in N_H(e)\} \text{ for every } e \in L,$$
$$\mathcal{P}_{\text{right}} : \text{ at most one element from each block } \{e_r : e \in L\} \text{ for every } r \in R.$$

Then $\mathcal{P}_{\text{left}} \cap \mathcal{P}_{\text{right}}$ has independent sets exactly the matchings of H. Projecting an independent set $I \subseteq U$ to $\{e \in L : \exists r \text{ with } e_r \in I\}$ yields precisely the L-vertices that can be simultaneously matched in H, i.e., an independent set of T.

Incorporation to the agent-arrival algorithm. Similar to what we did for graphic matroids, we will replace the component T on the original combination of systems by the two components $\mathcal{P}_{\text{left}}$ and $\mathcal{P}_{\text{right}}$. This is done at the very beginning, so both greedy and the greedy-relevant elements are computed with respect to the new combinations. Whenever the agent-arrival algorithm decides to reveal an agent that owns a *copy* of an original element $e \in L$ of the transversal in its neighborhood, actually all of its copies $\{e_r : r \in N_H(e)\}$ are simultaneously revealed (the weight that is revealed is actually $w(e)$ on all of them; break ties arbitrarily to maintain injectivity if needed.) Accepting any copy e_r is interpreted as accepting e. In the transformed system we run an order-oblivious core-selecting (OOCS) procedure, since $\mathcal{P}$ is an intersection of partition matroids.

Matchoid parameter. This reduction is not preserving for the growth parameter: one original matroidal component is replaced by two. In particular, if the original item-side constraint was a k-matchoid, the transformed constraint is a $(k+1)$-matchoid, so the competitive ratio by this reduction we obtain is slightly decreased..

Core and guarantee. Similar to the graphic case in the transformed instance, we are *not* enforcing selection from the core of the original transversal matroid T, but from the core of the new components $\mathcal{P}_{\text{left}} \cap \mathcal{P}_{\text{right}}$. This is still fine for the analysis to work.

Knapsack with integer sizes. We give a small construction that transforms knapsack constraints into a hypergraph matching, for the particular case in which all item sizes are integer numbers between 1 and k.

Essentially, for a knapsack of total capacity C and n elements, we create a hypergraph $H = (V, E)$ with $C + n$ vertices. The first C vertices correspond to unit slots in the knapsack, while the remaining n vertices each correspond to an original element of the knapsack constraint. Order the elements of the original knapsack constraint as $e_1, \ldots, e_n$. For every e_i with size $s_i \in \mathbb{N}$, we create $C - s_i + 1$ hyperedges. Each "copy" hyperedge $e_{i,j}$ (for $j = 1, \ldots, C - s_i + 1$) contains the vertices $\{j, \ldots, j + s_i - 1\} \cup \{C + i\}$. The last vertex ensures we pick at most one copy of e_i; the chosen copy encodes the position of e_i in the knapsack, ensuring that the final set of selected hyperedges is feasible. Furthermore, this hypergraph matching constraint is represented by a $(k+1)$-matchoid in which each component is a rank-1 uniform matroid, and each *copy* hyperedge participates in at most $k+1$ of these matroids.

We replace the knapsack constraint with this $(k+1)$-matchoid (each piece of this matchoid admits an OOCS algorithm since it is uniform), and by using the same idea described for transversals, we can effectively replace the original elements with their copies in the agent-arrival model. If instead of having sizes 1 to k we have a k-ratio-bounded knapsack (so the sizes range between some base size m and $k \cdot m$), and if the capacity of the knapsack is sufficiently large compared to m, then we can still carry out a similar construction with only a slight loss in the approximation factor by

choosing a sufficiently small ε and rounding up each item size to the nearest integer multiple of $m\varepsilon$. By standard techniques, every independent set feasible for the original constraint contains a subset that is feasible in the new, rounded knapsack with almost the same total weight, so we use this rounded system instead and apply the previous construction. This incurs a constant blow-up (on the order of $1/\varepsilon$) in the number of matroids used to describe the system.

6   Multiple Item Selection

Our techniques for the free-order agent-arrival model on bipartite graphs require that each agent selects at most one incident edge. In this section, we consider a different extension of the bipartite-graph model, which we call the *unrelated-agent* model.

Definition 11 (Unrelated-agent model). Let $G = (A \cup B, E)$ be a bipartite graph, and for $a \in A$ let $\delta(a) \subseteq E$ denote the set of edges incident to a. Each agent $a \in A$ has an independence system $\mathcal{F}_a = (\delta(a), \mathcal{I}_a)$ over its incident edges. There is also an independence system $\mathcal{F}_B = (B, \mathcal{I}_B)$ over the set of items B. A subset $F \subseteq E$ of edges is *feasible* if:

(i) for every agent a, the set of edges of F incident to a is independent, i.e., $\delta(a) \cap F \in \mathcal{I}_a$,

(ii) the set of assigned items is independent, i.e., $\{\, b \in B : \delta(b) \cap F \neq \emptyset \,\} \in \mathcal{I}_B$.

We say that agents are *unrelated* in this model because each has its own independence system, and there is no global system that constrains whether one agent can select elements based on the actions of another. Our goal is to study the agent-arrival model in this setting, namely, upon arrival, each agent a reveals the weights of all edges incident to it, and we must irrevocably select a subset $F_a \subseteq \delta(a)$ such that the union $\bigcup_{a \in A} F_a$ is feasible.

Although, in general, the entire problem can be encoded within the item system by treating each edge as an item – so that conditions (i) and (ii) are captured by a single system $(B, \mathcal{I}_B)$ – our purpose is to understand what can be achieved in the agent-arrival setting when the item system is simple. A canonical example is the case where the only constraint is that each item is assigned to at most one agent (i.e., the item system is a partition matroid).

This setting can be used to model problems such as the *secretary problem with groups* proposed by Korula and Pál [48]. In this problem, there is a unique independence system $(B, \mathcal{I}_B)$ whose ground set is partitioned (arbitrarily) into a known number m of *groups* $B_1, \ldots, B_m$ (but the partition is unknown). Once the groups have been formed, the groups arrive in random order, and upon arrival, the identity and weight of the elements are revealed and the algorithm must select a subset from each arriving group in such a way that the total set of selected elements is feasible for the system, and it has maximum weight. We note that this model falls within our framework, as it can be represented by a bipartite graph between a set $A = \{a_1, \ldots, a_m\}$ of m agents and the set B of items. The adversary will assign each group B_i to agent a_i by assigning $w(\{a_i, x\}) = w(x)$ for all $x \in B_i$ and $w(\{a_i, x\}) = 0$ (or an infinitesimal small number) if $x \notin B_i$.

6.1   The Case of Hypergraph Matching

Consider the special case of the unrelated-agent model in which the item constraint is a hypergraph matching constraint. This is, the items are hyperedges of an auxiliary hypergraph H, and the set of items (i.e., hyperedges) that can be used simultaneously must form a matching in H. In

the *combinatorial assignment problem with agent arrivals*, we are given a bipartite hypergraph $H = (A \cup R, E)$ where each hyperedge e contains exactly one node $A(e)$ in A and a subset $R(e) \subseteq R$ of nodes in R, so we can always denote them as $e = (a, R(e))$, with $a = A(e) \in A$. The set A is regarded as the agents. And the nodes in R are the *goods*. Each agent $a \in A$ owns a private independence system $\mathcal{F}_a = (\delta(a), \mathcal{I}_a)$ over the set $\delta(a)$ of hyperedges that contain a. Also, there is a global function $w \colon E \to \mathbb{R}_+$ where $w(e) = w(a, R(e))$ is interpreted as the weight that agent a assigns to $R(e)$. We say that a set of hyperedges $F \subseteq E$ is feasible if:

(i) for all $a \in A$, $\delta(a) \cap F \in \mathcal{I}_a$;

(ii) every individual node of b is in at most one hyperedge of F, i.e. for all $b \in R$, $|\delta(b) \cap F| \leq 1$.

An interpretation of this model comes from the combinatorial auction setting, where each hyperedge can be understood as a *bundle* of goods. Each agent a has a hidden valuation $w(e) = w(a, R(e))$ over bundles e of goods, and each agent a can receive not just one bundle, but a collection of them, under the conditions that the assignment to a must be feasible for the private independence system of agent a, i.e., $(\delta(a), \mathcal{I}_a)$, and each node in R must be in at most one of the assigned hyperedges F. The simplest example is when the hyperedges are actually edges: i.e., they consist of a single agent and a single item. In this example, upon arrival, agent a would reveal the value of all edges in $\delta(a)$, and then an algorithm must immediately assign a subset of it to a in such a way that this subset is feasible for a's own independence system, and each item can be assigned to at most one edge. When the bundles have size at most k, we call this setting the k-combinatorial assignment with agent arrivals.

Relation to Previous Bipartite Models. We remark that the combinatorial assignment problem with agent arrivals is a direct extension of the bipartite vertex-at-a-time hypergraph matching problem studied by Korula and Pál [48] and Kesselheim, Radke, Tönnis, and Vöcking [42], and the combinatorial assignment model by Marinkovic, Soto and Verdugo [57]. The main difference is that, rather than limiting each agent to a single hyperedge, agent a may receive any subset of $\delta(a)$ that is feasible in $\mathcal{F}_a$. In the prior works, this corresponds to the special case where each $\mathcal{F}_a$ is the rank-1 uniform matroid on $\delta(a)$.

The Algorithm. Each agent selects a random arrival time uniformly and independently on the interval $(0, 1)$ and uses it to represent the random-order arrival of the agents. We also assume that, for any set of agents A', we can compute the optimum allocation $\mathrm{OPT}(A')$, i.e., the maximum weight feasible set of hyperedges in the hypergraph induced by A' and R. We need some notation. For any set $S \subseteq A$ of agents, any agent $a \in S$, let $F(S, a) = \mathrm{OPT}(S) \cap \delta(a)$ be the set of hyperedges of $\mathrm{OPT}(S)$ that are incident to a, and let $N(S, a) = \bigcup_{e \in F(S,a)} R(e)$ be the goods in R that are assigned to a. For every $t > p$, let $X(S, a, t)$ be the subset of goods in $N(S, a)$ that are still unassigned by the algorithm by time t (or more precisely, immediately before time t).

Algorithm 6: BASIC ONLINE ASSIGNMENT ALGORITHM
---
1 ALG $\leftarrow \emptyset$; Ignore all agents arriving before time $p \in (0, 1)$
2 for *each agent a arriving at time $t_a \in [p, 1]$, in arrival order* **do**
3 $\quad$ Let A_a be the set of agents that have arrived up to time t_a
4 $\quad$ Compute $\text{OPT}(A_a)$, and let $F(a) = \text{OPT}(A_a) \cap \delta(a)$ denote the edges of $\text{OPT}(A_a)$
$\quad\quad$ incident to the new agent a.
5 $\quad$ Let $F'(a) = \{e \in F(a) \colon \forall f \in \text{ALG}, R(e) \cap R(f) = \emptyset\}$, i.e., the subset of edges e of $F(a)$
$\quad\quad$ such that $R(e)$ is composed of goods that have not been assigned by the algorithm yet.
6 $\quad$ Assign to a all edges in $F'(a)$, that is: $\text{ALG} \leftarrow \text{ALG} \cup F'(a)$
7 end
8 return ALG.

Lemma 8. *Let $S \subseteq A$ be a fixed set of at least $k + 1$ agents. Let $i = |S|$ and suppose that S is the set of the first i arriving agents on the algorithm. Let a be the last arriving agent in S, and $t = t(a) > p$ be its time of arrival. Let also $D \in F(S, a)$ be a subset of goods among the ones assigned to a in $\text{OPT}(S)$, with $|D| \leq k$. Then the probability that $D \subseteq X(S, a, t)$, i.e., that all goods in D are unassigned at time t, is at least $(p/t)^k$.*

We defer the proof of this lemma and show how to get the following theorem.

Theorem 8. *ALG is $\int_p^1 (p/t)^k dt$-competitive for the k-combinatorial assignment with agent arrivals. By choosing appropriate values of p for each k, the competitive ratio is $1/e$ for $k = 1$ and*

$$k^{-k/(k-1)} = \frac{1}{k} - \frac{\ln(k)}{k^2} + \mathrm{O}\left(\frac{\ln^2(k)}{k^3}\right) \ \text{for } k \geq 2$$

Proof. Let n be the number of agents. Let ALG_i be the contribution to the solution at the i-th arrival, that is, if $F \subseteq E$ is the final assignment and a is the i-th arriving agent, then $\text{ALG}_i := \sum_{e \in \delta(a) \cap \text{ALG}} w(e)$. For any set S of agents, let $\ell(S)$ be the last one to arrive inside S in the random order. Let A_i be the set of the first i agents to arrive and let $q_i(t)$ be the probability density function for the arrival time of the i-th agent among n uniformly arriving agents. Then:

$$\mathbb{E}[\text{ALG}_i] = \frac{1}{\binom{n}{i}} \sum_{\substack{S \subseteq A \\ |S|=i}} \frac{1}{i} \sum_{a \in S} \int_0^1 \mathbb{E}[\text{ALG}_i \,|\, A_i = S, a = \ell(S), t(a) = t] q_i(t) \, dt$$

$$= \frac{1}{\binom{n}{i}} \sum_{\substack{S \subseteq A \\ |S|=i}} \frac{1}{i} \sum_{a \in S} \int_p^1 \mathbb{E}[\text{ALG}_i \,|\, A_i = S, a = \ell(S), t(a) = t] q_i(t) \, dt, \tag{11}$$

where the last equality holds since, before time p, we do not select any hyperedge. Note that for fixed S, $a = \ell(S)$ and $t = t(a)$, Lemma 8 implies that

$$\mathbb{E}[\text{ALG}_i \,|\, A_i = S, a = \ell(S), t(a) = t] = \sum_{e \in F(S,a)} w(e) \Pr(R(e) \subseteq X(S, a, t))$$

$$\geq \sum_{e \in F(S,a)} w(e)(p/t)^k,$$

44

since for all e, we assumed that $|R(e)| \leq k$. The assumption that S has at least $k+1$ agents can be easily satisfied by adding to our original instance enough dummy agents without incident edges. Therefore, summing (11) over all i, and changing the summation order, we have

$$\sum_{i=1}^{n} \mathbb{E}[\mathrm{ALG}_i] \geq \int_{p}^{1} (p/t)^k \sum_{i=1}^{n} \frac{q_i(t)}{i} \frac{1}{\binom{n}{i}} \sum_{\substack{S \subseteq A \\ |S|=i}} \sum_{a \in S} \sum_{e \in F(S,a)} w(e) \, dt$$

$$= \int_{p}^{1} (p/t)^k \sum_{i=1}^{n} \frac{q_i(t)}{i} \frac{1}{\binom{n}{i}} \sum_{\substack{S \subseteq A \\ |S|=i}} w(\mathrm{OPT}(S)) \, dt$$

$$= \int_{p}^{1} (p/t)^k \sum_{i=1}^{n} \frac{q_i(t)}{i} \mathbb{E}_{\{S \colon |S|=i\}}[w(\mathrm{OPT}(S))] \, dt$$

where the expectation is taken over all sets S of exactly i agents (uniformly). Now observe that:

$$\mathbb{E}_{\{S \colon |S|=i\}}[w(\mathrm{OPT}(S))] \geq \mathbb{E}_{\{S \colon |S|=i\}}[w(\mathrm{OPT}(A) \cap \delta(S))] \geq \frac{i}{n} \cdot w(\mathrm{OPT}(A)),$$

since each agent in A is included in S with probability i/n, and the restriction $\mathrm{OPT}(A) \cap \delta(S)$ is a feasible solution over S. Putting it all together:

$$\mathbb{E}[w(\mathrm{ALG})] = \sum_{i=1}^{n} \mathbb{E}[\mathrm{ALG}_i]$$

$$\geq \int_{p}^{1} (p/t)^k \sum_{i=1}^{n} \frac{q_i(t)}{i} \frac{i}{n} w(\mathrm{OPT}(A)) \, dt$$

$$= w(\mathrm{OPT}(A)) \int_{p}^{1} (p/t)^k \cdot \frac{1}{n} \sum_{i=1}^{n} q_i(t) \, dt$$

$$= w(\mathrm{OPT}(A)) \int_{p}^{1} (p/t)^k \, dt = \begin{cases} w(\mathrm{OPT}(A)) \cdot p \ln(1/p) & \text{if } k = 1, \\ w(\mathrm{OPT}(A)) \cdot \frac{p - p^k}{k-1} & \text{if } k \geq 1. \end{cases}$$

For $k = 1$, we set $p = 1/e$, to obtain a competitive ratio of $p \ln(1/p) = 1/e$. For $k \geq 2$, we set $p = k^{-1/(k-1)}$ to obtain a competitive ratio of $k^{-k/(k-1)}$ $\qquad\square$

We now show the proof of the missing lemma above.

Proof of Lemma 8. Let ℓ be the number of agents of S arriving before time p. Label all the arriving times of the agents increasingly as $\tau_1 < \cdots < \tau_\ell < p < \tau_{\ell+1} < \tau_{\ell+2} < \cdots < \tau_{i-1} < t = \tau_i$. For convenience, denote a^q to the q-th arriving agent so that $a^i = a$ and let $A_q = \{a^1, \ldots, a^q\}$ be the set of the q first agents. By the statement of the lemma, $A_{i-1} = S \setminus \{a\}$. Recall that $D \subseteq F(S,a)$ is a fixed set of items.

Note that the optimum assignment $\mathrm{OPT}(A_{i-1})$ at time τ_{i-1} assigns each good $b \in D$ to at most one agent in A_{i-1}; it follows that the number of agents that could get a good from D at time $i-1$ is at most $|D| \leq k$. Since the identity of the agent arriving at time τ_{i-1} is chosen uniformly at random from the set A_{i-1}, with probability at least $(1 - k/(i-1))_+ = (i-1-k)_+/(i-1)$, agent a^{i-1} does not receive any good from D at time τ_{i-1}. Here $x_+ = \max(x, 0)$ is the positive part of

45

x. Now, suppose that we have already revealed the identities of $a_{q+1}, a_{q+2}, \ldots, a_{i-1}, a_i = a$ and that none of them received goods from B. Then $A_q = S \setminus \{a_{q+1}, \ldots, a_{i-1}, a_i\}$. Once again, we note that the optimum assignment at time τ_q assigns each good $b \in D$ to at most one agent in A_q and that the identity of agent a_q is uniformly chosen at random from A_q. Then, with probability at least $(1 - k/q)_+ = (q - k)_+/q$, agent a_q does not receive any good from D. By repeating this until $q = \ell + 1$, we conclude that

$$\Pr[D \in X(S, a, t) \mid \ell, \tau_{\ell+1}, \ldots, \tau_{i-1}, t = \tau_i] \geq \frac{(i-1-k)_+}{i-1} \cdot \frac{(i-2-k)_+}{i-2} \cdot \ldots \cdot \frac{(\ell+1-k)_+}{\ell+1}$$
$$= \frac{\binom{\ell}{k}}{\binom{i-1}{k}}.$$

Therefore,

$$\Pr[D \in X(S, a, t) | t = \tau_i] = \frac{\mathbb{E}[\binom{\ell}{k}]}{\binom{i-1}{k}},$$

where the expectation is over the random variable ℓ that counts the number of the agents in $S \setminus \{a\}$ that arrive in the interval $[0, p]$. Since all agents in $S \setminus \{a\}$ chose their arrival time in $[0, t]$ uniformly at random, we get $\ell \sim \mathrm{Bin}(i-1, p/t)$, and therefore, using known properties of the factorial moments of binomial variables we get $\Pr[D \in X(S, a, t) | t = \tau_i] = (p/t)^k$. $\qquad \square$

Extensions. We note that the same analysis used for the hypergraph matching case can be applied to more general independence systems on the item side. In fact, following Marinkovic, Soto and Verdugo [57], we can replace the hypergraph matching system (which is the intersection of k unitary partition matroids) by the combination of matroids admitting a so-called *directed certifier* [57, Definition 3]. Our approach can also be extended to the case of more general weight functions for the agents, e.g., *fractionally subadditive*.[6] Namely, for each agent a, consider the function $v_a(F) = \max_{I \subseteq F, I \in \mathcal{I}_a} \sum_{e \in I} w(e)$, that is, if F is a set of edges incident to a, then $v_a(F)$ is the maximum weight of a subset of F that can be assigned to a. For each a, the function v_a is fractionally subadditive. In this variant, the combinatorial assignment problem with agent arrival becomes an online combinatorial allocation problem with XOS valuations over *hyperedges*. Kesselheim, Radke, Tönnis, and Vöcking [42] gave an algorithm that works for the specific problem of XOS valuations when all $R(e)$ are singletons, achieving a $1/e$ competitive algorithm.[7] Our approach applies to the case of fractionally subadditive weights over hyperedges as well, generalizing the result by Kesselheim et al [42].

7 Properties of the k-systems Hierarchy

The study of k-systems and their subclasses plays a key role in understanding the structural and algorithmic limits of independence systems beyond matroids. These classes capture different levels of combinatorial flexibility and have been useful for understanding the trade-offs between structural generality and algorithmic tractability in submodular optimization, secretary problems, and related

---

[6]v is fractionally subadditive (a.k.a., XOS) if there is a fixed family $w_1, \ldots, w_k$ of additive weight functions over R such that for all S we have $v(S) = \max_{i \in [k]} w_i(S)$.

[7]In [42] the algorithm is stated for submodular valuations, but the proof also works for XOS valuations.

areas. The introduction of k-growth systems offers a promising new class that likely refines this hierarchy, although its exact relation to previously studied systems remains to be fully understood. We believe that the proposed class is of combinatorial interest on its own, and one of the implicit goals of this paper is to initiate its study.

In this section, we give an overview of k-systems, present examples of k-growth systems, and explain how k-growth systems fit in the hierarchy. We also study the closure properties of several subclasses of k-systems under fundamental operations such as restriction and contraction. In particular, we give a characterization of k-extendible systems in terms of contractions of k-systems.

7.1 Overview of Existing Classes

Matroids are characterized by their defining exchange properties. Since their introduction, several related classes have been proposed, with each subsequent class assuming a weaker form of these properties. We give a brief overview of the most important subclasses of k-systems.

Definition 12 (k-matroid intersection). An independence system $(S, \mathcal{I})$ is a *k-matroid intersection* system if there exist matroids $\mathcal{M}_i = (S, \mathcal{I}_i)$ for $i \in [k]$, such that

$$\mathcal{I} = \{F \subseteq S \colon F \in \mathcal{I}_i \text{ for all } i \in [k]\}.$$

Definition 13 (k-matchoid). An independence system $(S, \mathcal{I})$ is a *k-matchoid* if there exist matroids $\mathcal{M}_i = (S_i, \mathcal{I}_i)$ for $i \in [N]$, where the ground sets S_i may intersect, such that $S = \bigcup_{i=1}^{N} S_i$, each $e \in S$ belongs to at most k of the ground sets $(S_i)_{i=1}^{N}$, and

$$\mathcal{I} = \{F \subseteq S \colon F \cap S_i \in \mathcal{I}_i \text{ for all } i \in [N]\}.$$

Definition 14 (k-circuit bounded). An independence system $(S, \mathcal{I})$ is *k-circuit bounded* if for every independent $I \in \mathcal{I}$ and every $e \notin I$, $I + e$ contains at most k distinct circuits.

Definition 15 (k-extendible). An independence system $(S, \mathcal{I})$ is *k-extendible* if, for every $A \subseteq B \in \mathcal{I}$ and every $e \in S \setminus B$, we have the following k-extension property:

$$A + e \in \mathcal{I} \implies \exists Z \subseteq B \setminus A, \text{such that } |Z| \leq k \text{ and } (B + e) \setminus Z \in \mathcal{I}.$$

Definition 16 (k-system). An independence system $\mathcal{F} = (S, \mathcal{I})$ is a *k-system* if, for every subset $X \subseteq S$ and every pair P, Q of bases of X, we have $|P| \leq k \, |Q|$.

The following hierarchy of independence systems for $k \in \mathbb{N}$ is well-known, see e.g. [31].

$$k\text{-matroid intersection} \subseteq k\text{-matchoid} \subseteq k\text{-circuit bounded} \subseteq k\text{-extendible} \subseteq k\text{-system}.$$

It is easy to see that 1-systems coincide with matroids, and therefore, for $k = 1$, all classes are the same. For $k \geq 2$, however, all containments in the hierarchy are known to be strict. Furthermore, as Proposition 1 states, an independence system is a k-system if and only if, for every nonnegative weight function, the weighted greedy algorithm returns a $1/k$-approximation to the maximum-weight basis.

7.2 Growth Systems: A New Class of Independence Systems

Recall the definition of k-growth systems, presented here again for convenience.

Definition 9 (k-growth system). Let $\mathcal{F} = (S, \mathcal{I})$ be an independence system and k be a positive integer. We say that $\mathcal{F}$ is a *k-growth system* if it satisfies the following k-Basis-Growth (kBG) axiom.

(kBG) $\forall X \subseteq S, \forall I \in \mathcal{I}$, there exists a partition (Q, Z) of $I \setminus X$ with $|Z| \leq k \, |X \setminus I|$ such that
       ($\star$1) for every basis P of X, $P \cup Q \in \mathcal{I}$.

It is easy to see that condition ($\star$1) can be replaced by any of the following equivalent ones.

($\star$2) For every independent set $P \in \mathcal{I}$ with $P \subseteq X$, $P \cup Q \in \mathcal{I}$.

($\star$3) Q is a subset of every basis of $X \cup Q$.

($\star$4) For every circuit C, if $C \subseteq X \cup Q$ then $C \subseteq X$.

We remark that to check whether an independence system satisfies the (kBG) axiom, it is enough to check it for all sets X and bases B of the independence system.

Relations to the k-systems hierarchy. Next, we describe how k-growth systems are related to other classes in the hierarchy of independence systems.

Theorem 9. *The 1-growth systems are precisely matroids. Moreover, in this case the set Z guaranteed by the axiom can be chosen to satisfy the stronger bound $|Z| \leq r(X) - |X \cap I|$.*

Proof. Let $\mathcal{F} = (S, \mathcal{I})$ be a 1-growth system, and let J and I be two independent sets with $|J| < |I|$. Apply the 1-growth axiom (1BG) with $X = J$ to obtain a partition (Q, Z) of $I \setminus J$ such that $|Z| \leq |J \setminus I|$. Since $|I| > |J|$, we have $|I \setminus J| > |J \setminus I|$, hence $|Z| < |I \setminus J|$ and so $Q = (I \setminus J) \setminus Z \neq \emptyset$. For any $e \in Q$, the axiom guarantees $J + e \in \mathcal{I}$. Thus the augmentation property holds, and $\mathcal{F}$ is a matroid.

Now let $\mathcal{M} = (S, \mathcal{I})$ be a matroid, and let $X \subseteq S$ and $I \in \mathcal{I}$. Take Q to be any basis of $I \setminus X$ in the contracted matroid $\mathcal{M}/X$. Then, by the properties of contraction, we know that for every basis P of X in $\mathcal{M}$, the set $P \cup Q$ forms a basis of $X \cup I$, and therefore is independent. Define $Z = (I \setminus (X \cup Q))$. Then (Q, Z) is a partition of $I \setminus X$, and it only remains to show that $|Z| \leq |X \setminus I|$. To see this, let R be a basis of $X \setminus I$ in $\mathcal{M}/(X \cap I)$. Since $X \cap I \in \mathcal{I}$, this means that $R \cup (X \cap I)$ is a basis of X in $\mathcal{M}$. By the properties of Q discussed above, $R \cup (X \cap I) \cup Q$ is a basis of $X \cup I$ in $\mathcal{M}$. Moreover, since $I \in \mathcal{I}$, it follows that $|R \cup (X \cap I) \cup Q| \geq |I| = |X \cap I| + |Q| + |Z|$. In particular, $|Z| = |R| = r(X) - |X \cap I| \leq |X| - |X \cap I| = |X \setminus I|$, as needed. $\qquad \square$

As it turns out, k-growth systems fit nicely in the hierarchy between k-matchoids and k-extendible systems, for any $k \in \mathbb{N}$. In fact, for the former containment, we show a stronger result.

Lemma 4. Let $\mathcal{F} = (S, \mathcal{I})$ be a combination of systems $\{(S_j, \mathcal{I}_j)\}_{j=1}^{M}$, where each $(S_j, \mathcal{I}_j)$ is a k_j-growth system. Then, $\mathcal{F}$ is a k-growth system, where

$$k = \max_{e \in S} \sum_{j \, : \, e \in S_j} k_j.$$

In particular, for each $X \subseteq S$ and $I \in \mathcal{I}$, there exists a collection of sets $Z_j \subseteq (I \setminus X) \cap S_j$ such that, for every basis P of X, we have $P \cup \left((I \setminus X) \setminus \bigcup_{j=1}^{M} Z_j\right) \in \mathcal{I}$, and, for all $j \in [M]$,

$$|Z_j| \leq \begin{cases} r_j((X \setminus I) \cap S_j) - |(X \cap I) \cap S_j| & \text{if } k_j = 1, \\ k_j|(X \setminus I) \cap S_j| & \text{if } k_j > 1, \end{cases}$$

where r_j denotes the rank function of the j-th component $(S_j, \mathcal{I}_j)$. In particular, $|Z_j| \leq k_j|(X \setminus I) \cap S_j|$ for $j \in [M]$, since r_j is a matroid rank function when $k_j = 1$.

Proof. Let $X \subseteq S$ and $I \in \mathcal{I}$. Fix a $j \in [M]$. Since $(S_j, \mathcal{I}_j)$ is a k_j growth system, there exist sets $Z_j \subseteq (I \setminus X) \cap S_j$ and $Q_j = ((I \setminus X) \cap S_j) \setminus Z_j$ with $|Z_j| \leq k_j|(X \setminus I) \cap S_j)|$ such that for every independent set P_j of $X \cap S_j$ in $(S_j, \mathcal{I}_j)$, we have $P_j \cup Q_j \in \mathcal{I}_j$. If $k_j = 1$, the j-th component is a matroid and the same holds with the stronger bound $|Z_j| \leq r_j((X \setminus I) \cap S_j) - |(X \cap S_j) \cap I|$.

Now let

$$Z := \bigcup_{i=1}^{M} Z_j, \quad Q := (I \setminus X) \setminus Z.$$

For every $P \in \mathcal{I}$ and $j \in [M]$, we have $(P \cap S_j) \cup Q_j \in \mathcal{I}_j$. Since $Q \cap S_j \subseteq Q_j$, it follows that $(P \cup Q) \cap S_j = (P \cap S_j) \cup (Q \cap S_j) \subseteq (P \cap S_j) \cup Q_j$. Therefore, $(P \cup Q) \cap S_j \in \mathcal{I}_j$, and thus we have $P \cup Q \in \mathcal{I}$.

Finally, note that

$$|Z| = \left|\bigcup_{j=1}^{N} Z_j\right| \leq \sum_{j=1}^{N} |Z_j| \leq \sum_{j=1}^{N} k_j|(X \setminus I) \cap S_j)| = \sum_{e \in X \setminus I} \sum_{j : e \in S_j} k_j \leq k|X \setminus I|,$$

so the system is a k-growth system. $\square$

Theorem 10. *The following statements hold for any $k \in \mathbb{N}$.*

(a) *Every k-growth system is k-extendible.*

(b) *Every k-matchoid is a k-growth system.*

Proof. For (a), let $(S, \mathcal{I})$ be a k-growth system, and let $A \subseteq B \in \mathcal{I}$, $e \in S \setminus B$ such that $A + e \in \mathcal{I}$. Applying (kBG) to $X = A + e$ and $I = B$ we deduce that there exists a set $Z \subseteq B \setminus (A + e) = B \setminus A$ with $|Z| \leq k\ |(A + e) \setminus B| = k$ such that $(A + e) \cup (B \setminus (A \cup Z)) = (B \setminus Z) + e \in \mathcal{I}$. Thus, $(S, \mathcal{I})$ satisfies the k-extension property.

Finally, notice that (b) follows directly from Lemma 4 since, for matroids, we have $k_j = 1$ for all $j \in [M]$. $\square$

Next, we show that the latter containment is strict; in fact, we separate k-growth systems not only from k-matchoids but also from the larger class of k-circuit bounded systems.

Theorem 11. *For any $k \geq 2$, there exist k-growth systems that are not k-circuit bounded.*

Proof. Let $S = \{e_0, e_1, \ldots, e_{k+1}\}$ and consider the family of bases $\mathcal{B} = \{\{e_0, e_i\} : i \in [k+1]\} \cup \{\{e_1, \ldots, e_{k+1}\}\}$. Let $\mathcal{F} = (S, \mathcal{I})$ where $\mathcal{I} = \{F \subseteq B \mid B \in \mathcal{B}\}$.

Claim 10. $\mathcal{F}$ *is a k-growth system.*

Proof. By our previous observation, it suffices to check condition (*k*BG) for all sets X and bases B of the independence system. If $e_0 \in B$, then $|B| = 2$, and the k-growth condition (*k*BG) is trivially satisfied for any X. For $B = \{e_1, \ldots, e_{k+1}\}$, $X \setminus B \neq \emptyset$ only holds if $X \setminus B = \{e_0\}$. In this case, regardless of $B \setminus X$, we can always find a Z with $|Z| \leq k$ such that $|B \setminus (X \cup Z)| \leq 1$, in which case adding it to any basis of X will form an independent set. $\square$

Claim 11. $\mathcal{F}$ *is not* $\left(\binom{k+1}{2} - 1\right)$*-circuit bounded.*

Proof. For $I = \{e_1, \ldots, e_{k+1}\}$, the set $I + e_0$ contains $\binom{k+1}{2}$ circuits of the form $\{e_0, e_i, e_j\}$ for distinct $i, j \in [k+1]$. $\square$

The theorem follows by Claims 10 and 11. $\square$

7.3   Examples of k-growth Systems

Theorem 10 shows that k-growth systems lie between k-matchoids and k-extendible systems in the k-system hierarchy, while Theorem 11 shows that they do not form a subclass of k-circuit-bounded systems. However, the precise relationship between the proposed class and the previous ones remains an interesting open problem. To gain a better understanding of the class, we next present some fundamental examples.

Knapsack systems. We also show that the class of k-growth systems includes specific classes of knapsack systems. Let S be a finite set and $s\colon S \to \mathbb{R}_{\geq 0}$ be a nonnegative function. The associated *knapsack system* is $(S, \mathcal{I})$ where $\mathcal{I} = \{I \subseteq S\colon \sum_{e \in I} s(e) \leq 1\}$. We say that the knapsack system has *ratio bound* ρ if $\max_{e \in E} s(e) = \rho \, \min_{e \in E} s(e)$.

Theorem 12. *Every knapsack system* $(S, \mathcal{I})$ *with ratio bound* ρ *is a* $\lceil \rho \rceil$*-growth system.*

Proof. Let $X \subseteq S$ and $I \in \mathcal{I}$. Let $\alpha = \min_{e \in E} s(e)$, $\beta = \max_{e \in E} s(e)$ and set $k := \lceil \beta/\alpha \rceil$. We distinguish two cases.

Case 1. If $s(X \setminus I) \geq s(I \setminus X)$, then we have

$$\alpha |I \setminus X| \leq s(I \setminus X) \leq s(X \setminus I) \leq \beta |X \setminus I|,$$

yielding $|I \setminus X| \leq k|X \setminus I|$. Then we can set $Z := I \setminus X$ and $Q := \emptyset$.

Case 2. If $s(X \setminus I) < s(I \setminus X)$, then $s(X) < s(I) \leq 1$ and therefore $X \in \mathcal{I}$. If $|I \setminus X| \leq k|X \setminus I|$, then we can simply set $Z := I \setminus X$ and $Q := \emptyset$. Otherwise $|I \setminus X| > k|X \setminus I|$, and set Z to be any set of the largest $k|X \setminus I|$ elements of $I \setminus X$, and $Q := I \setminus (X \cup Z)$. Note that the size bound for Z is attained and

$$s(Z) \geq \alpha |Z| = \alpha k |X \setminus I| \geq \frac{\alpha}{\beta} k s(X \setminus I) \geq s(X \setminus I).$$

Therefore

$$s(X \cup Q) = s(X \setminus I) + s(X \cap I) + s(Q) \leq s(Z) + s(X \cap I) + s(Q) = s(I) \leq 1,$$

thus $X \cup Q \in \mathcal{I}$. $\square$

Stable Sets of Intersection Graphs. Let D be a ground set and $V \subseteq 2^D$ be a family of subsets; we refer to elements of D and V as *points* and *figures*, respectively. The *intersection graph* of the family is an undirected graph $G = (V, E)$ with

$$E = \left\{ uv \in \binom{V}{2} \colon u \cap v \neq \emptyset \right\}.$$

A set $W \subseteq V$ is considered as *stable* or *independent* in G if and only if its figures are pairwise disjoint. Let $(V, \mathcal{I})$ be the independence system of stable sets of G.

When D is finite, computing a stable set is equivalent to choosing, for each point $p \in D$, *at most one* figure that contains p. Thus $(V, \mathcal{I})$ is the intersection of $|D|$ partition matroids, one per point. If every figure contains at most k points, then each figure participates in at most k of those matroids and the system is a k-matchoid. Moreover, the circuits all have size 2: if a set is not stable, it contains two figures meeting at some point, and any proper subset is stable.

We focus on a special family of figures defined by intervals on a line. Let $D = [N]$, and for integers $1 \leq a \leq b \leq N$, let $V_{a,b}$ denote the collection of all intervals whose sizes lie between a and b, where the size of an interval is the number of its points. That is,

$$V_{a,b} = \{[i,j] \colon 1 \leq i \leq j \leq N, \ a \leq j - i + 1 \leq b\}.$$

By the above, $(V_{a,b}, \mathcal{I})$ is a b-matchoid. We show it is also c-circuit-bounded and a c-growth system with $c = \left\lfloor \frac{b-2}{a} \right\rfloor + 2$.

Circuit boundedness. Let $R = \{v_1, \ldots, v_r\} \subseteq V_{a,b}$ be stable and take any interval $v' = [\ell, r] \in V_{a,b}$ such that $R \cup \{v'\} \notin \mathcal{I}$. Let

$$Z(v') = \{ v \in R \colon v \cap v' \neq \emptyset \}$$

be the intervals in R intersecting v'. The number of circuits created by adding v' is exactly $|Z(v')|$; our goal is to bound $|Z(v')|$.

Let $Z_0(v') \subseteq Z(v')$ be those intervals that do not contain either endpoint ℓ or r of v'. Then every $v \in Z_0(v')$ is a subinterval of the interior of v', so the total number of points they cover is at most $|v'| - 2 \leq b - 2$. Since each interval has length at least a, we get

$$a \, |Z_0(v')| \ \leq \ |v'| - 2 \ \leq \ b - 2,$$

yielding $|Z_0(v')| \leq \left\lfloor \frac{b-2}{a} \right\rfloor$. Since at most two additional intervals in $Z(v')$ can contain the endpoints ℓ and r, we obtain

$$|Z(v')| \ \leq \ |Z_0(v')| + 2 \ \leq \ \left\lfloor \tfrac{b-2}{a} \right\rfloor + 2 \ = \ c,$$

proving c-circuit-boundedness.

Being c-growth. Let $X \subseteq V_{a,b}$ be arbitrary, $I \in \mathcal{I}$ be stable, and set $R := I \setminus X$. For each $v' \in X \setminus I$, define $Z(v') \subseteq R$ as above; by the previous bound, we have $|Z(v')| \leq c$. Let

$$Z := \bigcup_{v' \in X \setminus I} Z(v'), \qquad Q := R \setminus Z.$$

Every interval in Q is disjoint from every interval in $X \setminus I$ by construction, and also from every interval in $X \cap I$ by the independence of I. Therefore, for any stable set $P \subseteq X$, we have $P \cup Q \in \mathcal{I}$. Moreover,

$$|Z| \ \leq \ \sum_{v' \in X \setminus I} |Z(v')| \ \leq \ c \, |X \setminus I| \ \leq \ c \, |X|,$$

showing the *c*-growth property (*c*BG).

Not a c-matchoid. Finally, let us show that for certain values of a and b, the system above is not a *c*-matchoid. We give an example below with $N = 8$, $a = b = 3$ and $c = \lfloor (b-2)/a \rfloor + 2 = 2$.

Denote $v_j = \{j, j+1, j+2\}$ for $j \in [6]$ the six intervals of size 3. Suppose that the system of stable sets $(S, \mathcal{I})$ is a 2-matchoid given by a family of matroids. Since every singleton is independent and every circuit of $(S, \mathcal{I})$ has size 2, we conclude that each circuit $\{x, y\}$ of $(S, \mathcal{I})$ is a matroid-circuit in at least one of the matroids.

The four pairs

$$\{v_3, v_1\}, \ \{v_3, v_2\}, \ \{v_3, v_4\}, \ \{v_3, v_5\}$$

Using the same reasoning for the circuits

$$\{v_4, v_2\}, \ \{v_4, v_3\}, \ \{v_4, v_5\}, \ \{v_4, v_6\},$$

we get that v_4 is contained in two distinct matroids M_C and M_D such that $\{v_4, v_2\}$ and $\{v_4, v_3\}$ are circuits of M_C, while $\{v_4, v_5\}$ and $\{v_4, v_6\}$ are circuits of M_D.

Furthermore, by circuit elimination, $\{v_1, v_2\}$ is also a circuit in M_A, and $\{v_4, v_5\}$ is a circuit in M_B. Observe that v_3 belongs to the ground sets of M_A, M_B, and M_C, and since $M_A \neq M_B$, it follows that $M_C \in \{M_A, M_B\}$. If $M_C = M_A$, then M_A contains the circuits $\{v_1, v_2\}$ and $\{v_2, v_4\}$. By circuit elimination, $\{v_1, v_4\}$ would also be a circuit, contradicting that $\{v_1, v_4\} \in \mathcal{I}$ is independent. Similarly, if $M_C = M_B$, then M_B contains the circuits $\{v_4, v_5\}$ and $\{v_2, v_4\}$, and circuit elimination would imply that $\{v_2, v_5\}$ is a circuit, again contradicting independence. Hence M_A, M_B, and M_C must be pairwise distinct matroids containing v_3, which contradicts the assumption that $(S, \mathcal{I})$ is a 2-matchoid.

7.4   Parallel Extension, Restriction, and Contraction in Independence Systems

In matroid theory, the most important classes are those closed under basic operations, such as restriction, contraction, and parallel extension. Determining whether analogous closure properties hold for more general independence systems is also of particular interest. In this section, we investigate this problem and related questions. We first define the counterpart of these two operations for independence systems.

Definition 17 (Parallel extension). Let $\mathcal{F} = (S, \mathcal{I})$ be an independence system and $s \in S$ be a non-loop element. Set $S_{os} := (S - s) \cup \{s_1, s_2\}$, and define the *projection* of a set $X \subseteq S_{os}$ as $\pi_s(X) = X$ if $X \cap \{s_1, s_2\} = \emptyset$ and $\pi_s(X) = (X \setminus \{s_1, s_2\}) + s$ otherwise. Then, the *parallel extension* of $\mathcal{F}$ along e is the system $\mathcal{F}_{os} = (S_{oe}, \mathcal{I}_{os})$ with

$$\mathcal{I}_{os} = \{X \subseteq S_{os} : \pi_s(X) \in \mathcal{I}\}.$$

Definition 18 (Restriction). Let $\mathcal{F} = (S, \mathcal{I})$ be an independence system and $Y \subseteq S$. We define the *restriction* of $\mathcal{F}$ to Y as the system $\mathcal{F} = (S|_Y, \mathcal{I}|_Y)$ with

$$\mathcal{I}|_Y = \{X \subseteq Y : X \in \mathcal{I}\}.$$

The two operations above agree with the corresponding ones on matroids.

Definition 19 (Contraction). Let $\mathcal{F} = (S, \mathcal{I})$ be an independence system and $I \in \mathcal{I}$ be an independent set. We define the *contraction* of I in $\mathcal{F}$ as the system $\mathcal{F}/I = (S \setminus I, \mathcal{I}/I)$ with

$$\mathcal{I}/I = \{X \subseteq S \setminus I : S \cup I \in \mathcal{I}\}.$$

The contraction of an independent set coincides with the notion of contraction in matroids when we contract an independent set, but it is undefined when we try to contract a set that is not independent.

The next result shows that most of the classes behave well with respect to these operations.

Lemma 9. *If $\mathcal{F} = (S, \mathcal{I})$ is k-matroid intersection/k-matchoid/k-circuit bounded/k-growth/k-extendible, then so are its parallel extensions, restrictions and contractions.*

Proof. The claim for restrictions is direct. For parallel extensions and contractions, we prove the statement for the classes one by one. We first consider parallel extensions. Let $s \in S$ be an arbitrary element.

k-matchoids. If $\mathcal{F}$ is defined by matroids $(S_i, \mathcal{I}_i)$ for $i \in [N]$, then its parallel extension $\mathcal{F}_{os}$ is the k-matchoid defined by the matroids obtained by performing the same parallel extension on each of them.

k-circuit-bounded. Take an independent set $I \in \mathcal{I}_{os}$ and let $e \in S_{os} \setminus I$. If $I \cap \{s_1, s_2\} = \emptyset$ or $e \notin \{s_1, s_2\}$, then there is a one to one correspondence between the circuits of $\mathcal{F}_{os}$ in $I + e$ and of $\mathcal{F}$ in $\pi_s(I + e)$. If $I \cap \{s_1, s_2\} \neq \emptyset$ and $e \in \{s_1, s_2\}$, then $I + e$ contains the unique circuit $\{s_1, s_2\}$. Since $\mathcal{F}$ is k-circuit bounded, $I + e$ contains at most k such circuits in both cases.

k-growth. Take an arbitrary set $X \subseteq S_{os}$ and an independent set $I \in \mathcal{I}_{os}$. If $|(I \cup X) \cap \{s_1, s_2\}| \leq 1$ or $\{s_1, s_2\} \subseteq X$, then the k-growth axiom (kBG) simply follows from that for the projections in the original system. Otherwise, since $I \in \mathcal{I}_{os}$, we have $|(X \setminus I) \cap \{s_1, s_2\}| = |(I \setminus X) \cap \{s_1, s_2\}| = 1$; we may assume without loss of generality that $s_1 \in S \setminus I$ and $s_2 \in I \setminus X$. Then (kBG) for $\pi_s(X)$ and $\pi_s(Z)$ in the original system yields a set $Z \subseteq \pi_s(I) \setminus \pi_s(X)$ such that $P \cup (\pi_s(I) \setminus Z) \in \mathcal{I}$ for every basis P of $\pi_s(S)$, and $|Z| \leq k|(\pi_s(S) \setminus \pi_s(I)| = k|X \setminus I| - k$. Here, the last inequality follows from $s \in \pi_s(X) \cap \pi_s(Z)$. Then, choosing $Z' := Z + s_2$ satisfies $P' \cup (I \setminus Z') \in \mathcal{I}_{os}$ for every basis P' of X and $|Z'| = |Z| + 1 \leq k|X \setminus Z| - k + 1 \leq k|X \setminus Z|$.

k-extendible. Take any triple (A, B, e) with $A \subseteq B$, $B \in \mathcal{I}_{os}$, $e \in S_{os} \setminus B$, and $A + e \in \mathcal{I}_{os}$. If $B \cap \{s_1, s_2\} = \emptyset$ or $e \notin \{s_1, s_2\}$, then the k-extension property of $\mathcal{F}$ implies that there exists $Z \subseteq \pi_s(B) \setminus \pi_s(A)$ with $|Z| \leq k$ such that $(\pi_s(B) + \pi_s(e)) \setminus Z \in \mathcal{I}$. Let $Z' := Z$ if $s \notin Z$ and $Z' := (Z - s) \cup (B \cap \{s_1, s_2\})$ otherwise. Then Z' satisfies $(B + e) \setminus Z' \in \mathcal{I}_{os}$ and $|Z'| \leq k$. If $B \cap \{s_1, s_2\} \neq \emptyset$ and $e \in \{s_1, s_2\}$, then $B + e$ contains the unique circuit $\{s_1, s_2\}$, thus $Z = \{s_1, s_2\} - e$ is a proper choice.

Now we consider contractions. Let I be an independent set in $\mathcal{I}$ and $J \subseteq S \setminus I$ an independent set of $\mathcal{F}/I$, that is, $I \cup J \in \mathcal{I}$.

k-matchoids. If $\mathcal{F}$ is defined by matroids $(S_i, \mathcal{I}_i)$ for $i \in [N]$, then the contraction $\mathcal{F}/I$ is the k-matchoid defined by the matroids $(S_i \setminus I, \mathcal{I}_i/(I \cap S_i))$ for $i \in [N]$.

k-circuit-bounded. Let $e \in (S \setminus I) \setminus J$ and C be a circuit of $\mathcal{F}/I$ contained in $J + e$. Then $I \cup (J + e) \notin \mathcal{I}$, but $I \cup (J + e - f) \in \mathcal{I}$ for every $e, f \in C$. These together imply that $C = C' \setminus I$

for some circuit $C' \subseteq (I \cup J) + e$ of $\mathcal{F}$. Since $(I \cup J) + e$ contains at most k circuits of $\mathcal{F}$, we have that $J + e$ contains at most k circuits of $\mathcal{F}/I$.

k-growth. Take an arbitrary set $X \subseteq S \setminus I$ and consider the sets $Y := X \cup I$ and $F := J \cup I$. Note that $F \in \mathcal{I}$ by definition and, since M is a k-growth system, we have that there exists $Z \subseteq F \setminus Y$ such that $|Z| \leq k|Y \setminus F|$ and for any basis P of Y, we have $P \cup (F \setminus (Y \cup Z)) \in \mathcal{I}$. However, we have that $I \subseteq Y \cap F$, and thus $Y \setminus F = X \setminus J$, $F \setminus Y = J \setminus X$, and $F \setminus (Y \cup Z) = J \setminus (X \cup Z)$. Let Q be a basis of X, which implies that Q is an independent set of Y. Since $Q \cup (F \setminus (Y \cup Z)) \in \mathcal{I}$, we get that $Q \cup (J \setminus (X \cup Z)) \in \mathcal{J}$, and thus $\mathcal{F}/I$ is a k-growth system.

k-extendible. We show the k-extension property for a triple (A, B, e) with $A \subseteq B$, $B \in \mathcal{I}/I$, $e \in (S \setminus I) \setminus B$, and $A + e \in \mathcal{I}/I$. By the definition of contraction, we have $I \cup A \subseteq I \cup B \in \mathcal{I}$, and $I \cup (A+e) \in \mathcal{I}$. Since the original system is k-extendible, then there exists $Z \subseteq (B \cup I) \setminus (A \cup I) = B \setminus A$ with $|Z| \leq k$ such that $(I \cup (B + e)) \setminus Z \in \mathcal{I}$. Therefore $(B + e) \setminus Z \in \mathcal{I}/I$. $\qquad\square$

Interestingly, the case for k-systems is different.

Lemma 10. *Every parallel extension and restriction of a k-system is a k-system. However, for $k \geq 2$, the contraction of a k-system is not necessarily a k-system.*

Proof. Closure under parallel extension follows immediately, since duplicating an element into parallel copies does not affect the relative sizes of maximal independent sets in any subset, while closure under restriction is trivial by definition. For closure under contraction, consider the system with $S = \{x, a, b, c, d\}$ and bases $\{a, b, c, d\}$, $\{x, a\}$, $\{x, b\}$, $\{x, c\}$, and $\{x, d\}$. Since every set of size two is independent, for every $X \subseteq S$ with $|X| \leq 4$ and for every pair P, Q of bases of X, we have $|P| \geq |X|/2 \geq |Q|/2$. If $X = S$, then all bases of X have size either 2 or 4, so the system is a 2-system. However, after contracting a, both $P = \{b, c, d\}$ and $Q = \{x\}$ are bases, with $|P|/|Q| = 3$, and the resulting independence system is not a 2-system. $\qquad\square$

The following theorem reveals a key structural link between k-systems and k-extendible systems: every k-extendible system arises as a contraction of a k-system.

Theorem 13. *An independence system $\mathcal{F} = (S, \mathcal{I})$ is k-extendible precisely when all of its contractions are k-systems.*

Proof. If $\mathcal{F}$ is k-extendible, then by Lemma 9, all its contractions are k-extendible, and therefore also k-systems.

Now suppose that $\mathcal{F}$ is a k-system whose contractions are k-systems. We show that it is k-extendible. Let $A \subseteq B \in \mathcal{I}$ and $e \in S \setminus B$ with $A + e \in \mathcal{I}$. Since $A + e \subseteq B + e$, choose $Z \subseteq B \setminus A$ of minimum size such that $(B + e) \setminus Z \in \mathcal{I}$. If $B + e \in \mathcal{I}$, then $Z = \emptyset$ works and we are done; otherwise assume $B + e \notin \mathcal{I}$. Then $\{e\} \in \mathcal{I}/(B \setminus Z)$ and $Z \in \mathcal{I}/(B \setminus Z)$. We claim that both $\{e\}$ and Z are bases of $Z + e$ in the contracted system $\mathcal{F}/(B \setminus Z)$. Indeed, if Z were not a basis, then $Z + e$ would be independent in $\mathcal{I}/(B \setminus Z)$, implying $(Z + e) \cup (B \setminus Z) = B + e \in \mathcal{I}$, a contradiction. Similarly, if there existed $f \in Z$ such that $\{e, f\} \in \mathcal{I}/(B \setminus Z)$, then $(B \setminus Z) \cup \{e, f\} = (B + e) \setminus (Z - f) \in \mathcal{I}$, contradicting the minimality of Z as $Z - f$ would have smaller size. Therefore, both $\{e\}$ and Z are bases of $Z + e$ in $\mathcal{I}/(B \setminus Z)$. As $\mathcal{I}/(B \setminus Z)$ is a k-system, we must have $|Z| \leq k|\{e\}| = k$, and hence $(S, \mathcal{I})$ is k-extendible. $\qquad\square$

References

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

[2] A. Abels, L. Ladewig, K. Schewior, and M. Stinzendörfer. Knapsack secretary through boosting. In P. Chalermsook and B. Laekhanukit, editors, *WAOA*, pages 61–81, 2022.

[3] M. Adamczyk, A. Borodin, D. Ferraioli, B. D. Keijzer, and S. Leonardi. Sequential posted-price mechanisms with correlated valuations. *ACM Trans. Econ. Comput.*, 5(4), 12 2017.

[4] S. Alaei. Bayesian combinatorial auctions: Expanding single buyer mechanisms to many buyers. *SIAM Journal on Computing*, 43(2):930–972, 2014.

[5] S. Assadi, T. Kesselheim, and S. Singla. Improved truthful mechanisms for subadditive combinatorial auctions: Breaking the logarithmic barrier. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 653–661. SIAM, 2021.

[6] S. Assadi and S. Singla. Improved truthful mechanisms for combinatorial auctions with submodular bidders. *SIGecom Exch.*, 18(1):19–27, 2020.

[7] M. Babaioff, N. Immorlica, D. Kempe, and R. Kleinberg. A knapsack secretary problem with applications. In *APPROX/RANDOM*, page 16–28, 2007.

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

[9] K. Bérczi, V. Livanos, J. A. Soto, and V. Verdugo. Matroid secretary via labeling schemes. In *International Conference on Integer Programming and Combinatorial Optimization (IPCO)*, pages 128–141. Springer, 2025.

[10] H. Beyhaghi, N. Golrezaei, R. P. Leme, M. Pál, and B. Sivan. Improved revenue bounds for posted-price and second-price mechanisms. *Operations Research*, 69(6):1805–1822, 2021.

[11] L. Blumrosen and T. Holenstein. Posted prices vs. negotiations: An asymptotic analysis. In *ACM Conference on Electronic Commerce (EC)*, page 49. Association for Computing Machinery, 2008.

[12] A. Braun, M. Buttkus, and T. Kesselheim. Asymptotically optimal welfare of posted pricing for multiple items with MHR distributions. In P. Mutzel, R. Pagh, and G. Herman, editors, *Annual European Symposium on Algorithms (ESA)*, volume 204, pages 22:1–22:16, 2021.

[13] B. Brubach, N. Grammel, W. Ma, and A. Srinivasan. Improved guarantees for offline stochastic matching via new ordered contention resolution schemes. *Mathematics of Operations Research*, 2024.

[14] A. Bubna and A. Chiplunkar. Prophet inequality: Order selection beats random order. In *ACM Conference on Economics and Computation (EC)*, page 302–336. Association for Computing Machinery, 2023.

[15] G. Calinescu, C. Chekuri, M. Pal, and J. Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing*, 40(6):1740–1766, 2011.

[16] S. Chawla, J. D. Hartline, and R. Kleinberg. Algorithmic pricing via virtual valuations. In *ACM Conference on Electronic Commerce (EC)*, page 243–251. Association for Computing Machinery, 2007.

[17] S. Chawla, J. D. Hartline, D. L. Malec, and B. Sivan. Multi-parameter mechanism design and sequential posted pricing. In *ACM Symposium on Theory of Computing (STOC)*, page 311–320. Association for Computing Machinery, 2010.

[18] C. Chekuri and V. Livanos. On submodular prophet inequalities and correlation gap. *Theoretical Computer Science*, 1019:114814, 2024.

[19] J. Correa and A. Cristi. A constant factor prophet inequality for online combinatorial auctions. In *ACM Symposium on Theory of Computing (STOC)*, page 686–697. Association for Computing Machinery, 2023.

[20] S. Dobzinski. Breaking the logarithmic barrier for truthful combinatorial auctions with submodular bidders. *SIAM J. Comput.*, 50(3), 2021.

[21] S. Dobzinski and J. Vondrák. Impossibility results for truthful combinatorial auctions with submodular valuations. *J. ACM*, 63(1):5:1–5:19, 2016.

[22] S. Dughmi. The outer limits of contention resolution on matroids and connections to the secretary problem. In A. Czumaj, A. Dawar, and E. Merelli, editors, *International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 168, pages 42:1–42:18, 2020.

[23] S. Dughmi. Matroid secretary is equivalent to contention resolution. In M. Braverman, editor, *Innovations in Theoretical Computer Science Conference (ITCS)*, volume 215, pages 58:1–58:23, 2022.

[24] S. Dughmi. From contention resolution to matroid secretary and back. *SIAM J. Comput.*, 54(3):585–624, 2025.

[25] P. Dütting, M. Feldman, T. Kesselheim, and B. Lucier. Prophet inequalities made easy: Stochastic optimization by pricing nonstochastic inputs. *SIAM Journal on Computing*, 49(3):540–582, 2020.

[26] P. Dütting, T. Kesselheim, and B. Lucier. An $o(\log \log m)$ prophet inequality for subadditive combinatorial auctions. *SIAM Journal on Computing*, 53(6):239–275, 2024.

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

[28] T. Ezra, M. Feldman, N. Gravin, and Z. G. Tang. Online stochastic max-weight matching: Prophet inequality for vertex and edge arrival models. In *ACM Conference on Economics and Computation (EC)*, page 769–787, 2020.

[29] T. Ezra, M. Feldman, N. Gravin, and Z. G. Tang. General graphs are easier than bipartite graphs: Tight bounds for secretary matching. In *ACM Conference on Economics and Computation (EC)*, page 1148–1177, 2022.

[30] M. Feldman, N. Gravin, and B. Lucier. *Combinatorial Auctions via Posted Prices*, pages 123–135. SIAM, 2015.

[31] M. Feldman, J. Naor, R. Schwartz, and J. Ward. Improved approximations for k-exchange systems. In *European Symposium on Algorithms*, pages 784–798. Springer, 2011.

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

[33] M. Feldman, O. Svensson, and R. Zenklusen. Online contention resolution schemes with applications to bayesian selection problems. *SIAM J. Comput.*, 50(2):255–300, 2021.

[34] M. Feldman, O. Svensson, and R. Zenklusen. A framework for the secretary problem on the intersection of matroids. *SIAM Journal on Computing*, 51(3):766–819, 2022.

[35] T. S. Ferguson. Who Solved the Secretary Problem? *Statistical Science*, 4(3):282 – 289, 1989.

[36] M. T. Hajiaghayi, R. Kleinberg, and T. Sandholm. Automated online mechanism design and prophet inequalities. In *AAAI*, pages 58–65, 2007.

[37] P. Jaillet, J. A. Soto, and R. Zenklusen. Advances on matroid secretary problems: Free order model and laminar case. *arXiv preprint arXiv:1207.1333*, 2012.

[38] P. Jaillet, J. A. Soto, and R. Zenklusen. Advances on matroid secretary problems: Free order model and laminar case. In M. X. Goemans and J. R. Correa, editors, *Integer Programming and Combinatorial Optimization (IPCO)*, pages 254–265, 2013.

[39] T. Jenkyns. The efficacy of the "greedy" algorithm. In *Proc. 7th Southeastern Conf. on Combinatorics, Graph Theory and Computing*, pages 341–350, 1976.

[40] M. Kapralov, I. Post, and J. Vondrák. Online submodular welfare maximization: Greedy is optimal. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1216–1225. SIAM, 2013.

[41] R. M. Karp, U. V. Vazirani, and V. V. Vazirani. An optimal algorithm for online bipartite matching. In *ACM Symposium on Theory of Computing (STOC)*, page 352–358, 1990.

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

[43] T. Kesselheim, A. Tönnis, K. Radke, and B. Vöcking. Primal beats dual on online packing lps in the random-order model. In *ACM Symposium on Theory of Computing (STOC)*, page 303–312, 2014.

[44] R. Kleinberg. A multiple-choice secretary algorithm with applications to online auctions. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, page 630–631, 2005.

[45] R. Kleinberg and S. M. Weinberg. Matroid prophet inequalities and applications to multi-dimensional mechanism design. *Games and Economic Behavior*, 113:97–115, 2019.

[46] M. Klimm and M. Knaack. Generalized assignment and knapsack problems in the random-order model. In N. Megow and A. Basu, editors, *Integer Programming and Combinatorial Optimization*, pages 341–354, Cham, 2025. Springer Nature Switzerland.

[47] B. Korte and D. Hausmann. An analysis of the greedy heuristic for independence systems. In *Annals of Discrete Mathematics*, volume 2, pages 65–74. Elsevier, 1978.

[48] N. Korula and M. Pál. Algorithms for secretary problems on graphs and hypergraphs. In S. Albers, A. Marchetti-Spaccamela, Y. Matias, S. Nikoletseas, and W. Thomas, editors, *Automata, Languages and Programming*, pages 508–520, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[49] U. Krengel and L. Sucheston. Semiamarts and finite values. *Bulletin of the American Mathematical Society*, 83(4):745–747, 07 1977.

[50] U. Krengel and L. Sucheston. On semiamarts, amarts, and processes with finite value. *Probability on Banach spaces*, 4:197–266, 1978.

[51] O. Lachish. O(log log rank) competitive ratio for the matroid secretary problem. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 326–335, 2014.

[52] E. Lee and S. Singla. Optimal online contention resolution schemes via ex-ante prophet inequalities. In H. Bast, G. Herman, and Y. Azar, editors, *26th European Symposium on Algorithms, ESA 2018*, Leibniz International Proceedings in Informatics, LIPIcs, Germany, Aug. 2018. Schloss Dagstuhl- Leibniz-Zentrum fur Informatik GmbH, Dagstuhl Publishing.

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

[54] V. Livanos. Simple and optimal greedy online contention resolution schemes. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 9687–9699, 2022.

[55] W. Ma, C. MacRury, and P. Nuti. Online matching and contention resolution for edge arrivals with vanishing probabilities. In *ACM Conference on Economics and Computation (EC)*, page 159, New York, NY, USA, 2024. Association for Computing Machinery.

[56] C. MacRury, W. Ma, and N. Grammel. On (random-order) online contention resolution schemes for the matching polytope of (bipartite) graphs. *CoRR*, abs/2209.07520, 2022.

[57] J. Marinkovic, J. A. Soto, and V. Verdugo. Online combinatorial assignment in independence systems. *Mathematical Programming*, Mar 2025.

[58] A. Mehta, A. Saberi, U. Vazirani, and V. Vazirani. Adwords and generalized online matching. *Journal of the ACM (JACM)*, 54(5):22–es, 2007.

[59] J. Mestre. Greedy in approximation algorithms. In *European Symposium on Algorithms*, pages 528–539. Springer, 2006.

[60] D. Naori and D. Raz. Online Multidimensional Packing Problems in the Random-Order Model. In *International Symposium on Algorithms and Computation (ISAAC)*, volume 149, pages 10:1–10:15, 2019.

[61] P. Nuti and J. Vondrák. Towards an optimal contention resolution scheme for matchings. *Mathematical Programming*, 210(1):761–792, Mar 2025.

[62] J. Oxley. *Matroid Theory*. Oxford University Press, 02 2011.

[63] B. Peng and Z. Tang. Order selection prophet inequality: From threshold optimization to arrival time design. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 171–178, 2022.

[64] T. Pollner, M. Roghani, A. Saberi, and D. Wajc. Improved online contention resolution for matchings and applications to the gig economy. In D. M. Pennock, I. Segal, and S. Seuken, editors, *ACM Conference on Economics and Computation (EC)*, pages 321–322. ACM, 2022.

[65] A. Rubinstein. Beyond matroids: Secretary problem and prophet inequality with general constraints. In *Annual ACM Symposium on Theory of Computing (STOC)*, page 324–332, New York, NY, USA, 2016. Association for Computing Machinery.

[66] A. Rubinstein and S. Singla. Combinatorial prophet inequalities. In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, page 1671–1687, 2017.

[67] E. Samuel-Cahn. Comparison of threshold stop rules and maximum for independent nonnegative random variables. *The Annals of Probability*, 12(4):1213–1216, 1984.

[68] R. R. Saxena, S. Velusamy, and S. M. Weinberg. An improved lower bound for matroid intersection prophet inequalities. In Y. T. Kalai, editor, *Innovations in Theoretical Computer Science Conference, (ITCS)*, volume 251, pages 95:1–95:20.

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

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

[71] Q. Yan. Mechanism design via correlation gap. In D. Randall, editor, *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 710–719. SIAM, 2011.