Decomposing Probability Marginals Beyond Affine Requirements*

Jannik Matuschke

KU Leuven

Abstract. Consider the triplet $(E, \mathcal{P}, \pi)$, where E is a finite ground set, $\mathcal{P} \subseteq 2^E$ is a collection of subsets of E and $\pi : \mathcal{P} \to [0, 1]$ is a *requirement function*. Given a vector of *marginals* $\rho \in [0, 1]^E$, our goal is to find a distribution for a random subset $S \subseteq E$ such that $\mathbf{Pr}[e \in S] = \rho_e$ for all $e \in E$ and $\mathbf{Pr}[P \cap S \neq \emptyset] \geq \pi_P$ for all $P \in \mathcal{P}$, or to determine that no such distribution exists.

Generalizing results of Dahan, Amin, and Jaillet [6], we devise a generic decomposition algorithm that solves the above problem when provided with a suitable sequence of *admissible support candidates (ASCs)*. We show how to construct such ASCs for numerous settings, including *supermodular requirements*, Hoffman-Schwartz-type *lattice polyhedra* [14], and *abstract networks* where π fulfils a conservation law. The resulting algorithm can be carried out efficiently when $\mathcal{P}$ and π can be accessed via appropriate oracles. For any system allowing the construction of ASCs, our results imply a simple polyhedral description of the set of marginal vectors for which the decomposition problem is feasible. Finally, we characterize *balanced hypergraphs* as the systems $(E, \mathcal{P})$ that allow the *perfect decomposition* of any marginal vector $\rho \in [0, 1]^E$, i.e., where we can always find a distribution reaching the highest attainable probability $\mathbf{Pr}[P \cap S \neq \emptyset] = \min\left\{\sum_{e \in P} \rho_e, 1\right\}$ for all $P \in \mathcal{P}$.

1 Introduction

Given a set system $(E, \mathcal{P})$ on a finite ground set E with $\mathcal{P} \subseteq 2^E$ and a *requirement function* $\pi : \mathcal{P} \to (-\infty, 1]$, consider the polytope

$$Z_\pi := \left\{ z \in [0,1]^{2^E} \;:\; \textstyle\sum_{S \subseteq E} z_S = 1 \text{ and } \sum_{S : S \cap P \neq \emptyset} z_S \geq \pi_P \;\forall\, P \in \mathcal{P} \right\},$$

which corresponds to the set of all probability distributions over 2^E such that the corresponding random subset $S \subseteq E$ hits each $P \in \mathcal{P}$ with probability at least its requirement value π_P.[1] We are interested in describing the projection of Z_π to the corresponding marginal probabilities on E, i.e.,

$$Y_\pi := \left\{ \rho \in [0,1]^E \;:\; \exists z \in Z_\pi \text{ with } \rho_e = \textstyle\sum_{S \subseteq E : e \in S} z_S \;\forall\, e \in E \right\}.$$

* Proofs of results marked with (♣) can be found in the the full version [24].

[1] Note that we can assume $\pi_P \in [0, 1]$ without loss of generality in the definition of Z_π, but we allow negative values for notational convenience in later parts of the paper.

For $\rho \in Y_\pi$, we call any $z \in Z_\pi$ with $\rho_e = \sum_{S \subseteq E : e \in S} z_S$ for all $e \in E$ a *feasible decomposition of ρ for $(E, \mathcal{P}, \pi)$*. Note that every $\rho \in Y_\pi$ fulfils

$$\textstyle\sum_{e \in P} \rho_e \geq \pi_P \qquad \forall\, P \in \mathcal{P} \tag{$\star$}$$

because $\sum_{S : S \cap P \neq \emptyset} z_S \leq \sum_{e \in P} \rho_e$ for any feasible decomposition z of ρ. Hence

$$Y_\pi \ \subseteq\ Y^\star := \left\{\rho \in [0,1]^E \ :\ \rho \text{ fulfils } (\star)\right\}.$$

We say that $(E, \mathcal{P}, \pi)$ is $(\star)$*-sufficient* if $Y_\pi = Y^\star$. Our goal is to identify classes of such $(\star)$-sufficient systems, along with corresponding decomposition algorithms that, given $\rho \in Y^\star$, find a feasible decomposition of ρ. Using such decomposition algorithms, we can reduce optimization problems over Z_π whose objectives and other constraints can be expressed via the marginals to optimization problems over $Y^\star$, yielding an exponential reduction in dimension.

1.1 Motivation

Optimization problems over Z_π and polytopes with a similar structure arise, e.g., in the context of *security games*. In such a game, a defender selects a random subset $S \subseteq E$ of resources to inspect while an attacker selects a strategy $P \in \mathcal{P}$, balancing their utility from the attack against the risk of detection (which occurs if $P \cap S \neq \emptyset$). Indeed, the decomposition setting described above originates from the work of Dahan, Amin, and Jaillet [6], who used it to describe the set of mixed Nash equilibria for such a security game using a compact LP formulation when the underlying system is $(\star)$-sufficient.

Two further application areas of marginal decomposition are *randomization in robust or online optimization*, which is often used to overcome pessimistic worst-case scenarios [17–19, 27], and *social choice and mechanism design*, where randomization is frequently used to satisfy otherwise irreconcilable axiomatic requirements [3] and where decomposition results in various flavors are applied, e.g., to define auctions via interim allocations [2, 12], to improve load-balancing in school choice [7], and to turn approximation algorithms into truthful mechanisms [21, 22]. In [24, Appendix A], we discuss several applications from these three areas, including different security games, a robust randomized coverage problem, and committee election with diversity constraints. There we also show how the structures for which we establish $(\star)$-sufficiency here arise naturally in these applications and imply efficient algorithms for these settings.

1.2 Previous Results

As mentioned above, Dahan et al. [6] introduced the decomposition problem described above to characterize mixed Nash equilibria of a network security game played on $(E, \mathcal{P})$. They observed that such equilibria can be described by a compact LP formulation if $(E, \mathcal{P}, \pi)$ is $(\star)$-sufficient for all requirements π of the *affine* form

$$\pi_P = 1 - \textstyle\sum_{e \in P} \mu_e \quad \forall\, P \in \mathcal{P} \tag{A}$$

for some $\mu \in [0,1]^E$. They showed that this is indeed the case when E is the set of edges of a directed acyclic graph (DAG) and $\mathcal{P}$ the set of s-t-paths in this DAG and provide a polynomial-time (in $|E|$) algorithm for computing feasible decompositions in this case. Matuschke [25] extended this result by providing an efficient decomposition algorithm for *abstract networks*, a generalization of the system of s-t-paths in a (not necessarily acyclic) digraph; see Section 3 for a definition. He also showed that a system $(E, \mathcal{P}, \pi)$ is ($\star$)-sufficient for all affine requirement functions π if and only if the system has the *weak max-flow/min-cut property*, i.e., the polyhedron $\{y \in \mathbb{R}^E_+ \,:\, \sum_{e \in P} y_e \geq 1 \;\forall\, P \in \mathcal{P}\}$ is integral.

While the affine setting (A) is well-understood, little is known for the case of more general requirement functions. A notable exception is the *conservation law* studied by Dahan et al. [6], again for the case of directed acyclic graphs:

$$\pi_P + \pi_Q = \pi_{P \times_e Q} + \pi_{Q \times_e P} \quad \forall\, P, Q \in \mathcal{P}, e \in P \cap Q, \tag{C}$$

where $P \times_e Q$ for two paths $P, Q \in \mathcal{P}$ containing a common edge $e \in P \cap Q$ denotes the path consisting of the prefix of P up to e and the suffix of Q starting with e. Dahan et al. [6] established ($\star$)-sufficiency for requirements fulfilling (C) in DAGs by providing another combinatorial decomposition algorithm. It was later observed in [25] and independently in a different context in [4] that (C) for DAGs is in fact equivalent to (A). However, this equivalence no longer holds for the natural generalization of (C) to arbitrary digraphs.

1.3 Contribution and Structure of this Paper

In this article, we present an algorithmic framework for computing feasible decompositions of marginal vectors fulfilling ($\star$) for a wide range of set systems and requirement functions, going beyond the affine setting (A). Our algorithm, described in Section 2, iteratively adds a so-called *admissible support candidate (ASC)* to the constructed decomposition. The definition of ASCs is based on a transitive dominance relation on $\mathcal{P}$, which has the property that a decomposition of $\rho \in Y^\star$ is feasible for $(E, \mathcal{P}, \pi)$ if and only if it is feasible for the restriction of the system to non-dominated sets.

Our algorithmic framework can be seen as a generalization of Dahan et al.'s [6] Algorithm 1 for requirements fulfilling (C) in DAGs. An important novelty which allows us to establish ($\star$)-sufficiency for significantly more general settings is the use of the dominance relation and the definition of ASCs, which are more flexible than the properties implicitly used in [6]. A detailed comparison of the two algorithms can be found in [24, Appendix B.1].

To establish correctness of our algorithm for a certain class of systems, which also implies ($\star$)-sufficiency for those systems, it suffices to show the existence of an ASC in each iteration of the algorithm. We assume that the set E is of small cardinality and given explicitly, while $\mathcal{P}$ might be large (possibly exponential in $|E|$) and is accessed by an appropriate oracle. To establish polynomial runtime of our algorithm in $|E|$, it suffices to show that the following two tasks can be carried out in polynomial time in $|E|$:

(i) In each iteration, construct an ASC.
(ii) Given $\rho \in [0,1]^E$, either assert $\rho \in Y^\star$ or find a maximum violated inequality of ($\star$), i.e., $P \in \mathcal{P}$ maximizing $\pi_P - \sum_{e \in P} \rho_e > 0$.

We prove the existence and computability of admissible sets for a variety of settings, which we describe in the following.

Supermodular Requirements. A basic example for which our algorithm implies ($\star$)-sufficiency is the case where $\mathcal{P} = 2^E$ and π is a *supermodular* function, i.e., $\pi_{P \cap Q} + \pi_{P \cup Q} \geq \pi_P + \pi_Q$ for all $P, Q \in \mathcal{P}$. In Section 2.3, we show the existence of ASCs for this setting and observe that both (i) and (ii) can be solved when π is given by a *value oracle* that given $P \in \mathcal{P}$ returns π_P.

Abstract Networks under Weak Conservation of Requirements. We prove ($\star$)-sufficiency for the case that $(E, \mathcal{P})$ is an abstract network and π fulfils a relaxed version of the conservation law (C) introduced by Hoffman [15]. Such systems generalize systems of s-t-paths in digraphs, capturing some of their essential properties that suffice to obtain results such as Ford and Fulkerson's [9] max-flow/min-cut theorem or Dijkstra's [8] shortest-path algorithm; see Section 3 for a formal definition and an in-depth discussion. In particular, our results generalize the results of Dahan et al. [6] for DAGs under (C) to arbitrary digraphs.

Lattice Polyhedra. We also study the case where $\mathcal{P} \subseteq 2^E$ is a *lattice*, i.e., a partially ordered set in which each pair of incomparable elements have a unique maximum common lower bound, called *meet* and a unique minimum common upper bound, called *join*, and where π is supermodular with respect to these meet and join operations. Hoffman and Schwartz [14] showed that under two additional assumptions on the lattice, called *submodularity* and *consecutivity*, the system defined by ($\star$) and $\rho \geq 0$ is totally dual integral (the corresponding polyhedron, which is the dominant of $Y^\star$, is called *lattice polyhedron*). These polyhedra generalize (contra-)polymatroids and describe, e.g., r-cuts in a digraph [10] or paths in s-t-planar graphs [26]. When π is monotone with respect to the partial order on $\mathcal{P}$, a two-phase (primal-dual) greedy algorithm introduced by Kornblum [20] and later generalized by Frank [10] can be used to efficiently optimize linear functions over lattice polyhedra using an oracle that returns maxima of sublattices. We show the existence and computability of admissible sets under the same assumptions by carefully exploiting the structure of extreme points implicit in the analysis of the Kornblum-Frank algorithm; see Section 4 for complete formal definitions and an in-depth discussion of these results.

Perfect Decompositions and Balanced Hypergraphs. We call a set system $(E, \mathcal{P})$ *decomposition-friendly* if it is ($\star$)-sufficient for all requirement funtions π. Note that $(E, \mathcal{P})$ is decomposition-friendly if and only if every $\rho \in [0,1]^E$ has a feasible decomposition for $(E, \mathcal{P}, \pi^\rho)$, where $\pi^\rho_P := \min\left\{\sum_{e \in P} \rho_e, 1\right\}$ for $P \in \mathcal{P}$. We call such a decomposition *perfect*, as it simultaneously reaches the maximum intersection probability attainable under ρ for each $P \in \mathcal{P}$. In Section 5 we show that $(E, \mathcal{P})$ is decomposition-friendly if and only if it is a *balanced hypergraph*, a set system characterized by the absence of certain odd-length induced cycles.

1.4 Notation and Preliminaries

For $m \in \mathbb{N}$, we use the notation $[m]$ to denote the set $\{1, \dots, m\}$. Moreover, we use the notation $\mathbb{1}_A$ to indicate whether expression A is true ($\mathbb{1}_A = 1$) or false ($\mathbb{1}_A = 0$). We will further make use of the following observation.

Lemma 1 ([25, Lemma 3]). *There is an algorithm that given $\rho \in [0,1]^E$ and $z \in Z_\pi$ with $\sum_{S:e\in S} z_S \le \rho_e$ for all $e \in E$, computes a feasible decomposition of ρ in time polynomial in $|E|$ and $|\{S \subseteq E \,:\, z_S > 0\}|$.*

2 Decomposition Algorithm

We describe a generic algorithm that is able to compute feasible decompositions of marginals for a wide range of systems. The algorithm makes use of a dominance relation defined in Section 2.1. We describe the algorithm in Section 2.2 and state the conditions under which it is guaranteed to produce a feasible decomposition. In Section 2.3, we provide a simple yet relevant example where these conditions are met. Finally, we prove correctness of the algorithm in Section 2.4.

2.1 The Relation $\sqsubseteq_{\pi,\rho}$ and Admissible Support Candidates

For $P, Q \in \mathcal{P}$ we write $P \sqsubseteq_{\pi,\rho} Q$ if either $P = Q$, or if $\pi_P \le \pi_Q - \sum_{e\in Q\setminus P} \rho_e$ and $\pi_P < \pi_Q$. We say that P is *non-dominated* with respect to π and ρ in $\mathcal{P}' \subseteq \mathcal{P}$ if $P \in \mathcal{P}'$ and there exists no $Q \in \mathcal{P}' \setminus \{P\}$ with $P \sqsubseteq_{\pi,\rho} Q$.

Lemma 2 (♣). *The relation $\sqsubseteq_{\pi,\rho}$ is a partial order. In particular, for any $\mathcal{P}' \subseteq \mathcal{P}$, there exists at least one P' that is non-dominated in $\mathcal{P}'$.*

As we will see in the analysis below, it suffices to ensure $\sum_{S:S\cap P} z_S \ge \pi_P$ for non-dominated $P \in \mathcal{P}$ to construct a feasible decomposition. This motivates the following definition. A set $S \subseteq E$ is an *admissible support candidate (ASC) for π and ρ* if the following three conditions are fulfilled:

(S1) $S \subseteq E_\rho := \{e \in E \,:\, \rho_e > 0\}$.
(S2) $|S \cap P| \le 1$ for all $P \in \mathcal{P}^{=}_{\pi,\rho} := \left\{Q \in \mathcal{P} \,:\, \sum_{e\in Q} \rho_e = \pi_Q\right\}$.
(S3) $|P \cap S| \ge 1$ for all non-dominated (w.r.t. π and ρ) P in $\{Q \in \mathcal{P} \,:\, \pi_Q > 0\}$.

We now present an algorithm, that when provided with a sequence of ASCs computes a feasible decomposition for $\rho \in Y^\star$.

2.2 The Algorithm

The algorithm constructs a decomposition by iteratively selecting an ASC S for a requirement function $\bar{\pi}$ and a marginal vector $\bar{\rho}$, which can be thought of as residuals of the original requirements and marginals, respectively, with $\bar{\pi} = \pi$ and $\bar{\rho} = \rho$ initially. It shifts a probability mass of

$$\varepsilon_{\bar{\pi},\bar{\rho}}(S) := \min\left\{\min_{e\in S} \bar{\rho}_e,\ \max_{P\in\mathcal{P}} \bar{\pi}_P,\ \delta_{\bar{\pi},\bar{\rho}}(S)\right\}$$

to S, where $\delta_{\bar{\pi},\bar{\rho}}(S) := \inf_{P \in \mathcal{P} : |P \cap S| > 1} \frac{\bar{\pi}_P - \sum_{e \in P} \bar{\rho}_e}{1 - |P \cap S|}$. Intuitively, $\varepsilon_{\bar{\pi},\bar{\rho}}(S)$ corresponds to the maximum amount of probability mass that can be shifted to the set S without losing feasibility of the remaining marginals for the remaining requirements. The residual marginals $\bar{\rho}$ are reduced by $\varepsilon_{\bar{\pi},\bar{\rho}}(S)$ for all $e \in S$, and so are the requirements of all $P \in \mathcal{P}$ (including those P with $P \cap S = \emptyset$).

Algorithm 1: Generic Decomposition Algorithm

```
Initialize π̄ := π, ρ̄ := ρ.
Initialize z_∅ = 1 and z_S := 0 for all S ⊆ E with S ≠ ∅.
while max_{P∈𝒫} π̄_P > 0 do
    Let S be an ASC for π̄ and ρ̄.
    Let ε := ε_{π̄,ρ̄}(S).
    Set z_S := z_S + ε and z_∅ := z_∅ − ε.
    Set ρ̄_e := ρ̄_e − ε for all e ∈ S.
    Set π̄_P := π̄_P − ε for all P ∈ 𝒫.
Apply Lemma 1 to z to obtain a feasible decomposition z′ of ρ.
return z′
```

Our main result establishes that the algorithm returns a feasible decomposition after a polynomial number of iterations, if an ASC for $\bar{\pi}$ and $\bar{\rho}$ exists in every iteration. To show that a certain system is ($\star$)-sufficient, it thus suffices to establish the existence of the required ASCs.

Theorem 3. *Let $(E, \mathcal{P})$ be a set system and $\pi : \mathcal{P} \to (-\infty, 1]$. Let $\rho \in Y^\star$. If there exists an ASC for $\bar{\pi}$ and $\bar{\rho}$ in every iteration of Algorithm 1, then the algorithm terminates after $\mathcal{O}(|E|^2)$ iterations and returns a feasible decomposition of ρ for $(E, \mathcal{P}, \pi)$.*

Note that Theorem 3 implies that Algorithm 1 can be implemented to run in time $\mathcal{O}(\mathcal{T}|E|^2)$, when provided with an oracle that computes the required ASCs along with the corresponding values of $\varepsilon_{\bar{\pi},\bar{\rho}}(S)$ in time $\mathcal{T}$.[2] Before we prove Theorem 3, we first provide an example to illustrate its application.

2.3 Basic Example: Supermodular Requirements

Consider the case that $\mathcal{P} = 2^E$ and π is supermodular, i.e., for all $P, Q \in \mathcal{P}$ it holds that $\pi_{P \cap Q} + \pi_{P \cup Q} \geq \pi_P + \pi_Q$. Note that if π is supermodular, then $\bar{\pi}$ is supermodular throughout Algorithm 1, as subtracting a constant does not affect supermodularity. Moreover, we show in Section 2.4 that $\bar{\rho}$ fulfils ($\star$) for $\bar{\pi}$ throughout the algorithm. To apply Algorithm 1, it thus suffices to show existence of an ASC when $\rho \in Y^\star$ and π is supermodular. To obtain the ASC, we define $Q := \bigcup_{P \in \mathcal{P}^{=}_{\pi,\rho}} P$ and distinguish two cases: If $Q \cap E_\rho = \emptyset$, we let $S' := E_\rho$. Otherwise, we let $S' := (E_\rho \setminus Q) \cup \{e_Q\}$ for an arbitrary $e_Q \in Q \cap E_\rho$.

[2] In particular, note that $\varepsilon_{\bar{\pi},\bar{\rho}}(S)$ can be computed using at most $|S|$ iterations of the discrete Newton algorithm if we can solve problem (ii) from Section 1.3, i.e., the maximum violated inequality problem for $Y^\star$.

Lemma 4. *If $\mathcal{P} = 2^E$, π is supermodular, and $\rho \in Y^\star$, then S' is an ASC.*

Proof. Note that S' fulfils (S1) by construction and it fulfils (S2) because $P \subseteq Q$ and hence $P \cap S' \subseteq \{e_Q\}$ for all $P \in \mathcal{P}^{=}_{\pi,\rho}$. To see that S' fulfils (S3), assume by contradiction that $P \cap S' = \emptyset$ for some non-dominated $P \in \mathcal{P}$. Note that $P \cap E_\rho \subseteq Q \setminus \{e_Q\}$. Because $Q \in \mathcal{P}^{=}_{\pi,\rho}$ by standard uncrossing arguments, we obtain $\pi_P \leq \sum_{e \in P} \rho_e = \sum_{e \in Q \cap P} \rho_e = \pi_Q - \sum_{e \in Q \setminus P} \rho_e$ and thus $P \sqsubseteq_{\pi,\rho} Q$ (note that $e_Q \in Q \setminus P$ and hence $\pi_P < \pi_Q$), a contradiction. □

We remark that both the described ASC and maximum violated inequalities of $Y^\star$ can be found in polynomial time using submodular function minimization [29] when π is given by a value oracle, that given P returns π_P.

2.4 Analysis (Proof of Theorem 3)

Throughout this section we assume that $(E, \mathcal{P}, \pi)$ and ρ fulfil the conditions of the Theorem 3. In particular, $\rho \in Y^\star$ and in each iteration of the algorithm there exists an ASC. We show that under these conditions the while loop terminates after $\mathcal{O}(|E|^2)$ iterations (Lemma 6) and that after termination of the loop, $z \in Z_\pi$ (Lemma 8) and $\sum_{S : e \in S} z_S \leq \rho_e$ for all $e \in E$ (Lemma 5(a) for $k = \ell$). This implies that Lemma 1 can indeed be applied to z in the algorithm to obtain a feasible decomposition of ρ, thus proving Theorem 3.

We introduce the following notation. Let $S^{(i)}$ and $\varepsilon^{(i)}$ denote the set S and the value of ε chosen in the ith iteration of the while loop in the algorithm. Let further $\pi^{(i)}$ and $\rho^{(i)}$ denote the values of $\bar{\pi}$ and $\bar{\rho}$ at the beginning of the ith iteration (in particular, $\pi^{(1)} = \pi$ and $\rho^{(1)} = \rho$). Let $K \subseteq \mathbb{N}$ denote the set of iterations of the while loop. If the algorithm terminates, $K = \{1, \ldots, \ell\}$, where $\ell \in \mathbb{N}$ denotes the number of iterations. In that case, let $\rho^{(\ell+1)}$ and $\pi^{(\ell+1)}$ denote the state of $\bar{\rho}$ and $\bar{\pi}$ after termination.

Using this notation, we can establish the following three invariants, which follow directly from the construction of $\rho^{(i)}$ and $\varepsilon^{(i)}$ in the algorithm and the defining properties of the ASC $S^{(i)}$.

Lemma 5 (♣). *For all $k \in K$, the following statements hold true:*
- *(a)* $\rho_e^{(k+1)} = \rho_e - \sum_{i=1}^{k} \mathbb{1}_{e \in S^{(i)}} \cdot \varepsilon^{(i)} \geq 0$ *for all* $e \in E$,
- *(b)* $\sum_{e \in P} \rho_e^{(k+1)} \geq \pi_P^{(k+1)} = \pi_P - \sum_{i=1}^{k} \varepsilon^{(i)}$ *for all* $P \in \mathcal{P}$, *and*
- *(c)* $S^{(k)} \neq \emptyset$ *and* $\varepsilon^{(k)} > 0$.

The next lemma shows that the while loop indeed terminates after $\mathcal{O}(|E|^2)$ iterations. Its proof follows from the fact that in every non-final iteration $k \in K$, there is an element $e \in S^{(k)}$ for which the value of $\bar{\rho}_e$ drops to 0, or there are two elements $e, e' \in S^{(k)}$ such that $e, e' \in P$ for some $P \in \mathcal{P}^{=}_{\pi^{(k+1)},\rho^{(k+1)}}$. It can be shown that the same pair e, e' cannot appear in two distinct iterations of the latter type, from which we obtain the following bound.

Lemma 6 (♣). *The while loop in Algorithm 1 takes at most $\binom{|E|}{2} + |E|$ iterations, i.e., $K = \{1, \ldots, \ell\}$ with $\ell \leq \binom{|E|}{2} + |E|$.*

The termination criterion of the while loop implies the following lemma.

Lemma 7 (♣). *It holds that* $\sum_{i=1}^{\ell} \varepsilon^{(i)} = \max_{P \in \mathcal{P}} \pi_P$.

Finally, we can use the properties of the ASCs $S^{(k)}$ to show that $z \in Z_\pi$.

Lemma 8. *After termination of the while loop, it holds that* $z \in Z_\pi$.

Proof. Note that $z_S = \sum_{i=1}^{\ell} \mathbb{1}_{S=S^{(i)}} \cdot \varepsilon^{(i)} \geq 0$ for $S \subseteq E$ with $S \neq \emptyset$ and that $z_\emptyset = 1 - \sum_{i=1}^{\ell} \varepsilon^{(i)} \geq 0$, where the nonnegativity follows from Lemma 5(c) and Lemma 7 with $\max_{P \in \mathcal{P}} \pi_P \leq 1$, respectively. This also implies $\sum_{S \subseteq E} z_S = 1$.

We will prove that $\sum_{i=k}^{\ell} \mathbb{1}_{P \cap S^{(i)} \neq \emptyset} \cdot \varepsilon^{(i)} \geq \pi_P^{(k)}$ for all $k \in [\ell + 1]$ and $P \in \mathcal{P}$, which, for $k = 1$, implies $\sum_{S : P \cap S \neq \emptyset} z_S = \sum_{i=1}^{\ell} \mathbb{1}_{P \cap S^{(i)} \neq \emptyset} \cdot \varepsilon^{(i)} \geq \pi_P^{(1)} = \pi_P$ and hence $z \in Z_\pi$. We prove the above statement by induction on k, starting from $k = \ell + 1$ and going down to $k = 1$. For the base case $k = \ell + 1$, observe that the left-hand side is 0 and $\pi_P^{\ell+1} \leq 0$ by termination criterion of the while loop.

For the induction step, let $k \in [\ell]$, assuming that the statement is already established for $k + 1$ and let $P \in \mathcal{P}$. We distinguish two cases.

- Case $P \cap S^{(k)} \neq \emptyset$: We can apply the induction hypothesis to obtain $\sum_{i=k}^{\ell} \mathbb{1}_{P \cap S^{(i)} \neq \emptyset} \cdot \varepsilon^{(i)} = \varepsilon^{(k)} + \sum_{i=k+1}^{\ell} \mathbb{1}_{P \cap S^{(i)} \neq \emptyset} \cdot \varepsilon^{(i)} \geq \varepsilon^{(k)} + \pi_P^{(k+1)} = \pi_P^{(k)}$.
- Case $P \cap S^{(k)} = \emptyset$: If $\pi_P^{(k)} \leq 0$ then the desired statement follows from $\varepsilon^{(i)} > 0$ for all $i \in [\ell]$ by Lemma 5(c). Thus, we can assume $\pi_P^{(k)} > 0$. By property (S3), there is $Q \in \mathcal{P}$ with $P \sqsubseteq_{\pi^{(k)}, \rho^{(k)}} Q$ and $Q \cap S^{(k)} \neq \emptyset$. Hence we can apply the induction step proven in the first case to Q, yielding $\sum_{i=k}^{\ell} \mathbb{1}_{Q \cap S^{(i)} \neq \emptyset} \cdot \varepsilon^{(i)} \geq \pi_Q^{(k)}$. From this, we conclude that

$$\begin{aligned} \textstyle\sum_{i=k}^{\ell} \mathbb{1}_{P \cap S^{(i)} \neq \emptyset} \cdot \varepsilon^{(i)} &\geq \textstyle\sum_{i=k}^{\ell} \mathbb{1}_{P \cap Q \cap S^{(i)} \neq \emptyset} \cdot \varepsilon^{(i)} \\ &\geq \pi_Q^{(k)} - \textstyle\sum_{i=k}^{\ell} \mathbb{1}_{(Q \setminus P) \cap S^{(i)} \neq \emptyset} \cdot \varepsilon^{(i)} \\ &\geq \pi_Q^{(k)} - \textstyle\sum_{e \in Q \setminus P} \rho_e^{(k)} \;\geq\; \pi_P^{(k)}, \end{aligned}$$

  where the first and second inequality use $\varepsilon^{(i)} > 0$ by Lemma 5(c), the third inequality uses $\rho_e^{(k)} = \sum_{i=k}^{\ell} \mathbb{1}_{e \in S^{(i)}} \cdot \varepsilon^{(i)}$ by Lemma 5(a) and the final inequality uses $P \sqsubseteq_{\pi^{(k)}, \rho^{(k)}} Q$. □

3 Abstract Networks Under Weak Conservation Law

An *abstract network* is a tuple $(E, \mathcal{P}, \preceq, \times)$, where $(E, \mathcal{P})$ is a set system, $\preceq_P$ for each $P \in \mathcal{P}$ is a linear order of the elements in P, and $\times$ is an operator that takes $P, Q \in \mathcal{P}$ and $e \in P \cap Q$ as arguments and maps them to a member of $\mathcal{P}$, such that $P \times_e Q \in \mathcal{P}$ fulfils $P \times_e Q \subseteq \{p \in P \,:\, p \preceq_P e\} \cup \{q \in Q \,:\, e \preceq_Q q\}$ for all $P, Q \in \mathcal{P}$ and $e \in P \cap Q$. Note that the definition of abstract networks does not impose any requirements on the order $\preceq_{P \times_e Q}$. In particular, it does not need to be consistent with $\preceq_P$ and $\preceq_Q$.

Abstract networks were introduced by Hoffman [15] in an effort to encapsulate the essential properties of systems of paths in classic networks that enable the proof of Ford and Fulkerson's [9] max-flow/min-cut theorem. Indeed, the set of s-t-paths in a digraph constitutes a special case of an abstract network (however, see [16] for examples of abstract networks that do not arise in this way) and the elements of $\mathcal{P}$ are therefore also referred to as *abstract paths*. The *maximum weighted abstract flow (MWAF)* problem and the *minimum weighted abstract cut (MWAC)* problem correspond to the linear programs

$$\begin{array}{rl} \max & \sum_{P \in \mathcal{P}} \pi_P \; x_P \\ \text{s.t.} & \sum_{P : e \in P} x_P \leq u_e \; \forall e \in E \\ & x \geq 0 \end{array} \qquad \begin{array}{rl} \min & \sum_{e \in E} u_e \; y_e \\ \text{s.t.} & \sum_{e \in P} y_e \geq \pi_P \; \forall P \in \mathcal{P} \\ & y \geq 0 \end{array}$$

where $u \in \mathbb{R}^E_+$ is a capacity vector and π determines the reward for each unit of flow sent along the abstract path $P \in \mathcal{P}$.

Hoffman [15] proved that MWAC is totally dual integral, if the reward function π fulfils the following weak conservation law:

$$\pi_{P \times_e Q} + \pi_{Q \times_e P} \geq \pi_P + \pi_Q \quad \forall\, P, Q \in \mathcal{P}, e \in P \cap Q. \tag{C'}$$

McCormick [28] complemented this result by a combinatorial algorithm for solving MWAF when $\pi \equiv 1$. This was extended by Martens and McCormick [23] to a combinatorial algorithm for solving MWAF with arbitrary π fulfilling (C') when π is given a separation oracle for the constraints of MWAC.

A combinatorial algorithm for marginal decomposition in abstract networks under affine requirements (A) based on a generalization of Dijkstra's shortest-path algorithm is presented in [25]. Here, we prove ($\star$)-sufficiency for the more general setting (C') by showing that ASCs can be constructed in this setting.

Theorem 9. *Let $(E, \mathcal{P}, \preceq, \times)$ be an abstract network, let π fulfil (C'), and let $\rho \in Y^\star$. Then $S := \{e \in E_\rho \;:\;$ there is no $P \in \mathcal{P}^=_{\pi,\rho}$ and $p \in P \cap E_\rho$ with $p \prec_P e\}$ is an ASC for π and ρ.*

Proof. Note that S fulfils (S1) and (S2) by construction. It remains to show that S also fulfils (S3). For this, let $Q \in \mathcal{P}$ be non-dominated with $\pi_Q > 0$ and assume by contradiction $Q \cap S = \emptyset$. We use the notation $(P, e) := \{p \in P \;:\; p \prec_P e\}$ and $[e, P] := \{p \in P \;:\; e \preceq_P p\}$ for $P \in \mathcal{P}$ and $e \in P$.

Note that $Q \cap E_\rho \neq \emptyset$ because $\sum_{e \in Q} \rho_e \geq \pi_Q > 0$. Let $q := \min_{\preceq_Q} Q \cap E_\rho$. Observe that $q \notin S$ by our assumption, and hence, by construction of S, there must be $r \in E_\rho$ and $R \in \mathcal{P}^=_{\pi,\rho}$ such that $r \prec_R q$.

Let $Q' := R \times_q Q$ and $R' := Q \times_q R$. Note that $R' \cap E_\rho \subseteq [q, R]$ because $(Q, q) \cap E_\rho = \emptyset$ by choice of q as $\prec_q$-minimal element in $Q \cap E_\rho$. Using ($\star$), we obtain $\sum_{e \in [q,R]} \rho_e \geq \sum_{e \in R'} \rho_e \geq \pi_{R'}$. We conclude that

$$\pi_{Q'} + \sum_{e \in [q,R]} \rho_e \;\geq\; \pi_{Q'} + \pi_{R'} \;\geq\; \pi_Q + \pi_R \;=\; \pi_Q + \sum_{e \in R} \rho_e,$$

where the second inequality follows from (C') and the final identity is due to the fact that $\pi_R = \sum_{e \in R} \rho_e$ because $R \in \mathcal{P}_{\pi,\rho}$. Subtracting $\sum_{e \in [q,R]} \rho_e$ on both sides yields $\pi_{Q'} \geq \pi_Q + \sum_{e \in (R,q)} \rho_e$.

Using $Q' \backslash Q \subseteq (R, q)$ by construction of Q', we obtain $\pi_{Q'} \geq \pi_Q + \sum_{e \in Q' \backslash Q} \rho_e$. Note further that $\pi_{Q'} > \pi_Q$ because $r \in (R, q) \cap E_\rho$ and hence $\sum_{e \in (R,q)} \rho_e > 0$. We conclude that $Q \sqsubseteq_{\pi,\rho} Q'$, a contradiction to Q being non-dominated. $\square$

We remark that the corresponding ASCs and hence feasible decompositions can be computed in polynomial time in $|E|$ if the abstract network is given via an oracle that solve the maximum violated inequality problem for $Y^\star$ and returns π_P and $\prec_P$ for the corresponding $P \in \mathcal{P}$.

4 Lattice Polyhedra

We now consider the case where $\mathcal{P}$ is equipped with a partial order $\preceq$ so that $(\mathcal{P}, \preceq)$ is a *lattice*, i.e., the following two properties are fulfilled for all $P, Q \in \mathcal{P}$:

- The set $\{R \in \mathcal{P} \,:\, R \preceq P,\, R \preceq Q\}$ has a unique maximum w.r.t. $\preceq$, denoted by $P \wedge Q$ and called the *meet of P and Q*.
- The set $\{R \in \mathcal{P} \,:\, R \succeq P,\, R \succeq Q\}$ has a unique minimum w.r.t. $\preceq$, denoted by $P \vee Q$ and called the *join of P and Q*.

We will further assume that $\mathcal{P}$ fulfils the following two additional properties:

$$\mathbb{1}_{e \in P \vee Q} + \mathbb{1}_{e \in P \wedge Q} \leq \mathbb{1}_{e \in P} + \mathbb{1}_{e \in Q} \quad \forall\, P, Q \in \mathcal{P},\ e \in E \tag{SM}$$
$$P \cap R \subseteq Q \quad \forall\, P, Q, R \in \mathcal{P} \text{ with } P \prec Q \prec R \tag{CS}$$

which are known as *submodularity* and *consecutivity*, respectively.

Furthermore, we assume that the requirement function π is *supermodular* w.r.t. the lattice $(\mathcal{P}, \preceq)$, i.e.,

$$\pi_{P \vee Q} + \pi_{P \wedge Q} \geq \pi_P + \pi_Q \quad \forall\, P, Q \in \mathcal{P}$$

and *monotone* w.r.t. $\preceq$, i.e., $\pi_P \leq \pi_Q$ for all $P, Q \in \mathcal{P}$ with $P \preceq Q$.

Hoffman and Schwartz [14] showed that under these assumptions (even when foregoing monotonicity of π) the system defining the polyhedron

$$Y^+ := \left\{\rho \in \mathbb{R}_+^E \,:\, \textstyle\sum_{e \in P} \rho_e \geq \pi_P \ \forall\, P \in \mathcal{P}\right\},$$

which they call *lattice polyhedron*, is totally dual integral. For the case that π is monotone, Kornblum [20] devised a two-phase (primal-dual) greedy algorithm for optimizing linear functions over Y^+, which was extended by Frank [10] to the more general notions of sub- and supermodularity (still requiring montonicity). The algorithm runs in strongly polynomial time when provided with *lattice oracle* that, given $U \subseteq E$ returns the maximum member (w.r.t. $\preceq$) of the sublattice $\mathcal{P}[U] := \{P \in \mathcal{P} \,:\, P \subseteq U\}$ along with the value of π_P. We prove the following decomposition result under the same assumptions as in [10, 20].

Theorem 10 (♣). *Let $(\mathcal{P}, \preceq)$ be a submodular, consecutive lattice and let π be monotone and supermodular with respect to $\preceq$. Then $(E, \mathcal{P}, \pi)$ is ($\star$)-sufficient. Moreover, there is an algorithm that, given $\rho \in [0,1]^E$, finds in polynomial time in $|E|$ and $\mathcal{T}$, a feasible decomposition of ρ or asserts that $\rho \notin Y^\star$, where $\mathcal{T}$ is the time for a call to a lattice oracle for $(\mathcal{P}, \preceq)$ and π.*

Our strategy for proving Theorem 10 is the following: If $\rho \in Y^\star$, we can express it as a convex combination of extreme points (and possibly rays) of Y^+. We can use the structure of these extreme points, implied by the optimality of the two-phase greedy algorithm, to construct ASCs and hence, via Algorithm 1, a feasible decomposition of each extreme point. These can then be recomposed to a feasible decomposition for ρ. The two-phase greedy algorithm allows us to carry out these steps efficiently as it implies a separation oracle for $Y^\star$.

In the remainder of this section, we show how to construct an ASC for the case that $\rho \in Y^\star$ is an extreme point of Y^+. We start by describing the properties of extreme points implied by the correctness of the two-phase greedy algorithm.

Theorem 11 ([10]). *Let $\rho \in Y^+$. Then ρ is an extreme point of Y^+ if and only if there exists $e_1, \dots, e_m \in E$ and $P_1, \dots, P_m \in \mathcal{P}$ with the following properties:*

(G1) $e_i \in P_i$ *for all* $i \in [m]$,
(G2) $P_i = \max_{\succeq} \mathcal{P}[E \setminus \{e_1, \dots, e_{i-1}\}]$ *for all* $i \in [m]$,
(G3) $\pi_{P_i} > 0$ *for all* $i \in [m]$ *and* $\pi_Q \leq 0$ *for all* $Q \in \mathcal{P}[E \setminus \{e_1, \dots, e_m\}]$,
(G4) ρ *is the unique solution to the linear system*

$$\begin{aligned} \textstyle\sum_{e \in P_i} \rho_e &= \pi_{P_i} && \forall\, i \in [m], \\ \rho_e &= 0 && \forall\, e \in E \setminus \{e_1, \dots, e_m\}. \end{aligned}$$

We call such $e_1, \dots, e_m \in E$ and $P_1, \dots, P_m \in \mathcal{P}$ fulfilling these properties a *greedy support for ρ*. Indeed, note that (G4) implies $E_\rho \subseteq \{e_1, \dots, e_m\}$. Properties (G1)-(G4) also imply that greedy supports have a special interval structure, enabling the following algorithmic and structural result.

Lemma 12 (♣). *Given a greedy support $e_1, \dots, e_m$ and $P_1, \dots, P_m$ of an extreme point ρ of Y^+ one can compute in time $\mathcal{O}(m)$ a set S fulfilling*

$$S \subseteq E_\rho \text{ and } |S \cap P_i| = 1 \text{ for all } i \in [m]. \tag{1}$$

The corresponding algorithm iterates through $e_1, \dots, e_m$ in reverse order and adds element e_i to S if it does not result in $|S \cap P_i| > 1$. We now show that S as constructed above is indeed an ASC.

Theorem 13 (♣). *If S fulfils (1) for the greedy suppport of an extreme point ρ of Y^+, then S is an ASC for π and ρ.*

Proof (sketch). Note that S fulfils (S1) as $S \subseteq E_\rho$ by (1). Next, we show that S also fulfils (S2). Assume by contradiction that there is $Q \in \mathcal{P}^{=}_{\pi,\rho}$ with $|Q \cap S| > 1$. Without loss of generality, we can assume Q to be $\succeq$-maximal with this property. We distinguish three cases.

- Case 1: $Q \preceq P_m$. Note that $e_j \notin Q$ for all $j \in [m]$ with $j < m$, as otherwise $Q \prec P_m \prec P_j$ would imply $e_j \in P_m$ by (CS), contradicting (G2), which requires $P_m \subseteq E \setminus \{e_1, \dots, e_{m-1}\}$. Therefore $Q \cap S \subseteq \{e_m\}$, from which we conclude $|Q \cap S| \leq 1$.

- Case 2: There is $i \in [m]$ with $P_i \succeq Q \succ P_{i+1}$. It can be shown that (G2) and (CS) imply $P_i \cap E_\rho \subseteq Q \cap E_\rho$ in this case. Moreover, $P_i, Q \in \mathcal{P}^{=}_{\pi,\rho}$ and monotonicty imply $\sum_{e \in P_i} \rho_e = \pi_{P_i} \geq \pi_Q = \sum_{e \in Q} \rho_e$. We conclude that in fact $P_i \cap E_\rho = Q \cap E_\rho$. Thus $P_i \cap S = Q \cap S$ and $|Q \cap S| \leq 1$ by (1).
- Case 3: There is $i \in [m]$ such that $Q \sim P_i$ (i.e., Q and P_i are incomparable w.r.t. $\preceq$). Let $i \in [m]$ be maximal with that property and define $Q_+ := Q \vee P_i$ and $Q_- := Q \wedge P_i$. Using standard uncrossing arguments we can show that $P_i, Q \in \mathcal{P}^{=}_{\pi,\rho}$ implies $Q_+, Q_- \in \mathcal{P}^{=}_{\pi,\rho}$ and $\mathbb{1}_{Q_+ \cap S} + \mathbb{1}_{Q_- \cap S} = \mathbb{1}_{P_i \cap S} + \mathbb{1}_{Q \cap S}$. Note that $|Q_+ \cap S| \leq 1$ by maximality of $Q \in \mathcal{P}^{=}_{\pi,\rho}$ with $|Q \cap S| > 1$. We will show that $|Q_- \cap S| \leq 1$, which, using the above and $|P_i \cap S| = 1$ by (1), implies $|Q \cap S| \leq |Q_+ \cap S| + |Q_- \cap S| - |P_i \cap S| \leq 1$, a contradiction.
  It remains to show $|Q_- \cap S| \leq 1$, for which we distinguish two subcases. First, if $i = m$, then $Q_- \cap S \subseteq \{e_m\}$ as shown in case 1 above. Second, if $i < m$, then maximality of i with $P_i \sim Q$ implies $Q \succ P_{i+1}$ and therefore $P_i \succ Q_- = P_i \wedge Q \succeq P_{i+1}$. Thus either $Q_- = P_{i+1}$ and hence $|Q_- \cap S| = 1$ by (1) or $P_i \succ Q_- \succ P_{i+1}$, in which case $|Q_- \cap S| \leq 1$ by case 2 above.

The proof that S fulfils (S3) follows similar lines but requires the use of some additional consequences of (G1)-(G4). □

5 Perfect Decompositions and Balanced Hypergraphs

In this section, we study decomposition-friendly systems, where every $\rho \in [0,1]^E$ has a perfect decomposition that attains requirements $\pi^\rho_P := \min\{\sum_{e \in P} \rho_e, 1\}$ for all $P \in \mathcal{P}$. We show that such systems are characterized by absence of certain substructures that hinder perfect decomposition.

A *special cycle* of $(E, \mathcal{P})$ consists of ordered subsets $C = \{e_1, \dots, e_k\} \subseteq E$ and $\mathcal{C} = \{P_1, \dots, P_k\} \subseteq \mathcal{P}$ such that $P_i \cap C = \{e_i, e_{i+1}\}$ for $i \in [k]$, where we define $e_{k+1} = e_1$. The *length* of such a special cycle $(C, \mathcal{C})$ is $|C| = k = |\mathcal{C}|$. A *balanced hypergraph* is a system $(E, \mathcal{P})$ that does not have any special cycles of odd length at least 3. Balanced hypergraphs were introduced by Berge [1] and have been studied extensively, see, e.g., the survey by Conforti et al. [5]. Our main result in this section is the following:

Theorem 14 (♣). *A set system $(E, \mathcal{P})$ is decomposition-friendly if and only if it is a balanced hypergraph. If $(E, \mathcal{P})$ is a balanced hypergraph, a perfect decomposition of $\rho \in [0,1]^E$ can be computed in polynomial time in $|E|$.*[3]

Proof (sketch). To see that every decomposition-friendly system needs to be a balanced hypergraph, consider any odd-length special cycle $(C, \mathcal{C})$ and observe that the marginals defined by $\rho_e = \frac{1}{2}$ for $e \in C$ and $\rho_e = 0$ for $e \in E \setminus C$ do not have a perfect decomposition. The existence of perfect decompositions in balanced hypergraphs can be established by a reduction to the case $\pi^\rho \equiv 1$, for which a perfect decomposition can be obtained using an integrality result of Fulkerson et al. [11]. □

[3] Note that $|\mathcal{P}|$ is bounded by $\mathcal{O}(|E|^2)$ for any balanced hypergraph [13]. Thus, the stated running time holds even when $\mathcal{P}$ is given explicitly.

References

[1] Berge, C.: The rank of a family of sets and some applications to graph theory. In: Recent progress in combinatorics, pp. 49–57, Academic Press New York (1969)

[2] Border, K.C.: Implementation of reduced form auctions: A geometric approach. Econometrica **59**, 1175–1187 (1991)

[3] Brandl, F., Brandt, F., Seedig, H.G.: Consistent probabilistic social choice. Econometrica **84**, 1839–1880 (2016)

[4] Çela, E., Klinz, B., Lendl, S., Woeginger, G.J., Wulf, L.: A linear time algorithm for linearizing quadratic and higher-order shortest path problems. In: Integer Programming and Combinatorial Optimization, Lecture Notes in Computer Science, vol. 13904, pp. 466–479, Springer (2023)

[5] Conforti, M., Cornuéjols, G., Vušković, K.: Balanced matrices. Discrete Mathematics **306**, 2411–2437 (2006)

[6] Dahan, M., Amin, S., Jaillet, P.: Probability distributions on partially ordered sets and network interdiction games. Mathematics of Operations Research **47**, 458–484 (2022)

[7] Demeulemeester, T., Goossens, D., Hermans, B., Leus, R.: A pessimist's approach to one-sided matching. European Journal of Operational Research **305**, 1087–1099 (2023)

[8] Dijkstra, E.W.: A note on two problems in connexion with graphs. Numerische Mathematik **269**, 271 (1959)

[9] Ford, L.R., Fulkerson, D.R.: Maximal flow through a network. Canadian Journal of Mathematics **8**, 399–404 (1956)

[10] Frank, A.: Increasing the rooted-connectivity of a digraph by one. Mathematical Programming **84**, 565–576 (1999)

[11] Fulkerson, D.R., Hoffman, A.J., Oppenheim, R.: On balanced matrices. Pivoting and Extension: In honor of AW Tucker pp. 120–132 (1974)

[12] Gopalan, P., Nisan, N., Roughgarden, T.: Public projects, boolean functions, and the borders of Border's theorem. ACM Transactions on Economics and Computation (TEAC) **6**, 1–21 (2018)

[13] Heller, I.: On linear systems with integral valued solutions. Pacific Journal of Mathamatics **7**, 1351–1364 (1957)

[14] Hoffman, A., Schwartz, D.: On lattice polyhedra. In: Proceedings of the Fifth Hungarian Combinatorial Colloquium, North-Holland (1978)

[15] Hoffman, A.J.: A generalization of max flow—min cut. Mathematical Programming **6**, 352–359 (1974)

[16] Kappmeier, J.P.W., Matuschke, J., Peis, B.: Abstract flows over time: A first step towards solving dynamic packing problems. Theoretical Computer Science **544**, 74–83 (2014)

[17] Kawase, Y., Sumita, H.: Randomized strategies for robust combinatorial optimization. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 7876–7883 (2019)

[18] Kawase, Y., Sumita, H., Fukunaga, T.: Submodular maximization with uncertain knapsack capacity. SIAM Journal on Discrete Mathematics **33**, 1121–1145 (2019)

[19] Kobayashi, Y., Takazawa, K.: Randomized strategies for cardinality robustness in the knapsack problem. Theoretical Computer Science **699**, 53–62 (2017)

[20] Kornblum, D.: Greedy algorithms for some optimization problems on a lattice polyhedron. Ph.D. thesis, City University of New York (1978)

[21] Kraft, D., Fadaei, S., Bichler, M.: Fast convex decomposition for truthful social welfare approximation. In: Web and Internet Economics: 10th International Conference, WINE 2014, Beijing, China, December 14-17, 2014. Proceedings 10, pp. 120–132, Springer (2014)

[22] Lavi, R., Swamy, C.: Truthful and near-optimal mechanism design via linear programming. Journal of the ACM (JACM) **58**, 1–24 (2011)

[23] Martens, M., McCormick, S.T.: A polynomial algorithm for weighted abstract flow. In: Integer Programming and Combinatorial Optimization, Lecture Notes in Computer Science, vol. 5035, pp. 97–111, Springer (2008)

[24] Matuschke, J.: Decomposing probability marginals beyond affine requirements. Tech. rep., arXiv:2311.03346 (2023)

[25] Matuschke, J.: Decomposition of probability marginals for security games in max-flow/min-cut systems. Tech. rep., arXiv:2211.04922 (A prelimary version appeared under the title "Decomposition of Probability Marginals for Security Games in Abstract Networks" at IPCO 2023.) (2023)

[26] Matuschke, J., Peis, B.: Lattices and maximum flow algorithms in planar graphs. In: Graph-Theoretic Concepts in Computer Science (WG 2010), Lecture Notes in Computer Science, vol. 6410, pp. 324–335, Springer (2010)

[27] Matuschke, J., Skutella, M., Soto, J.A.: Robust randomized matchings. Mathematics of Operations Research **43**, 675–692 (2018)

[28] McCormick, S.T.: A polynomial algorithm for abstract maximum flow. In: Proceedings of the 7th annual ACM-SIAM Symposium on Discrete Algorithms, pp. 490–497 (1996)

[29] Schrijver, A.: Combinatorial optimization: polyhedra and efficiency. Springer (2003)