Decomposition of Probability Marginals for Security Games in Abstract Networks

Jannik Matuschke
KU Leuven

Abstract. Given a set system \((E, \mathcal{P})\), let \(\pi \in [0, 1]^\mathcal{P}\) be a vector of requirement values on the sets and let \(\rho \in [0, 1]^E\) be a vector of probability marginals with \(\sum_{e \in P} \rho_e \geq \pi_P\) for all \(P \in \mathcal{P}\). We study the question under which conditions the marginals \(\rho\) can be decomposed into a probability distribution on the subsets of \(E\) such that the resulting random set intersects each \(P \in \mathcal{P}\) with probability at least \(\pi(P)\).

Extending a result by Dahan, Amin, and Jaillet [4] motivated by a network security game in directed acyclic graphs, we show that such a distribution exists if \(\mathcal{P}\) is an abstract network and the requirements are of the form \(\pi_P = 1 - \sum_{e \in P} \mu_e\) for some \(\mu \in [0, 1]^E\). Our proof yields an explicit description of a feasible distribution that can be computed efficiently. As a consequence, equilibria for the security game studied in [4] can be efficiently computed even when the underlying digraph contains cycles.

As a subroutine of our algorithm, we provide a combinatorial algorithm for computing shortest paths in abstract networks, answering an open question by McCormick [14]. We further show that a conservation law proposed in [4] for requirement functions in partially ordered sets can be reduced to the setting of affine requirements described above.

1 Introduction

Consider a set system \((E, \mathcal{P})\), where \(E\) is a finite ground set and \(\mathcal{P} \subseteq 2^E\) is a collection of subsets of \(E\). Given probability marginals \(\rho \in [0, 1]^E\) and requirements \(\pi \in [0, 1]^\mathcal{P}\), we are interested in finding a probability distribution on the power set \(2^E\) of \(E\) that is consistent with these marginals and that ensures that each set in \(P \in \mathcal{P}\) is hit with probability at least \(\pi_P\). In other words, we are looking for a solution \(x\) to the system

\[
\begin{align*}
\sum_{S \subseteq E, e \in S} x_S &= \rho_e \quad \forall e \in E, \quad (1) \\
\sum_{S \subseteq E : S \cap P \neq \emptyset} x_S &\geq \pi_P \quad \forall P \in \mathcal{P}, \quad (2) \\
\sum_{S \subseteq E} x_S &= 1, \quad (3) \\
x_S &\geq 0 \quad \forall S \subseteq E. \quad (4)
\end{align*}
\]

Throughout this paper, we will call a distribution \(x\) fulfilling (1) to (4) a feasible decomposition of \(\rho\) for \((E, \mathcal{P})\) and \(\pi\), and we will say that the marginals \(\rho\) are feasible for \((E, \mathcal{P})\) and \(\pi\) if such a feasible decomposition exists.
A necessary condition for the existence of a feasible decomposition is that the marginals suffice to cover each set of the system individually, i.e.,
\[ \sum_{e \in P} \rho_e \geq \pi_P \quad \forall P \in \mathcal{P}. \] (⋆)

We are particularly interested in identifying classes of systems and requirement functions for which (⋆) is not only a necessary but also a sufficient condition. For such systems, it is possible to describe the set of distributions on \( 2^E \) fulfilling (2) via the polytope of feasible marginals defined by (⋆), which is of exponentially lower dimension.

1.1 Motivation

A natural application for feasible decompositions in the setting described above lies in network security games; see, e.g., [1, 2, 4, 9, 17, 18] for various examples and applications of network security games. In fact, such a game was also the motivation of Dahan, Amin, and Jaillet [4], who originally introduced the decomposition setting described above. We will discuss their game in detail in Section 5. Here, we describe a simpler yet relevant problem as an illustrative example.

Consider the following game played on a set system \((E, \mathcal{P})\), where each element \( e \in E \) is equipped with a usage cost \( c_e \geq 0 \) and an inspection cost \( d_e \geq 0 \). A defender \( D \) determines a random subset \( S \) of elements from \( E \) to inspect at cost \( \sum_{e \in S} d_e \) (e.g., a set of links of a network at which passing traffic is monitored). She anticipates that an attacker \( A \) is planning to carry out an illegal action, where \( A \) chooses a set in \( P \in \mathcal{P} \) (e.g., a route in the network along which he smuggles contraband), for which he will receive utility \( U_1 - \sum_{e \in P} c_e \) for some constant \( U_1 > 0 \). However, if \( P \) intersects with the random set \( S \) of elements inspected by \( D \), then \( A \) is discovered while carrying out his illegal action, reducing his utility by a penalty \( U_2 \geq U_1 \). The attacker also has the option to not carry out any attack, resulting in utility 0. Thus, \( A \) will refrain from using \( P \in \mathcal{P} \) if the probability that \( S \cap P \neq \emptyset \) exceeds \( \pi_P := (U_1 - \sum_{e \in P} c_e) / U_2 \).

A natural goal for \( D \) is to discourage \( A \) from attempting any attack at all, while keeping the incurred inspection cost as small as possible. Note that the randomized strategies that achieve this goal correspond exactly to vectors \( x \) that minimize \( \sum_{S \subseteq E} \sum_{S \in e} d_e x_S \) subject to constraints (2) to (4). Unfortunately, the corresponding LP has both an exponential number of variables and an exponential number of constraints in the size of the ground set \( E \).

However, assume that we can establish the following three properties for our set system: (i) condition (⋆) is sufficient for the feasibility of marginals, (ii) we can efficiently compute the corresponding feasible decompositions, and (iii) given \( \gamma \in \mathbb{R}_+^E \), we can efficiently solve \( \min_{P \in \mathcal{P}} \sum_{e \in P} \gamma_e \). Then (i) allows us to formulate \( D \)’s problem in terms of the marginals, i.e., \( \min_{\rho \in [0,1]^E} \sum_{e \in E} d_e \rho_e \) subject to constraints (⋆), (iii) allows us to separate the linear constraints (⋆) and obtain optimal marginals \( \rho \), and (ii) allows us to turn these marginals into a distribution corresponding to an optimal inspection strategy for the defender \( D \). In this paper, we will establish all three conditions for a generic type of set systems called abstract networks.
1.2 Abstract Networks

An *abstract network* consists of a set system \((E, \mathcal{P})\) where each set \(P \in \mathcal{P}\) (also referred to as an (abstract) path) is equipped with an internal linear order \(\preceq_P\) of its elements, such that the following property is fulfilled:

\[
\forall P, Q \in \mathcal{P}, e \in P \cap Q : \exists R \in \mathcal{P} : R \subseteq \{ p \in P : p \preceq_P e \} \cup \{ q \in Q : e \preceq_Q q \}.
\]

Given \(P, Q \in \mathcal{P}\) and \(e \in P \cap Q\), we use the notation \(P \times_e Q\) to denote an arbitrary but fixed feasible choice for such an \(R \in \mathcal{P}\).

Interesting special cases of abstract networks include \(\mathcal{P}\) being the set of maximal chains in a partially ordered set \((E, \preceq)\) (here, \(\preceq_P\) is simply the restriction of \(\preceq\) to \(\mathcal{P}\)) and \(\mathcal{P}\) being the set of simple \(s\)-\(t\)-paths in a digraph \(D = (V, A)\) (here, \(E = V \cup A\) and each path is identified with the sequence of its nodes and arcs). We remark that in both cases, the order \(\preceq_P \times_e Q\) is consistent with \(\preceq_P\) and \(\preceq_Q\), which is not a general requirement for abstract networks; see, e.g., [11] for examples of abstract networks where this is not the case.

Abstract networks were introduced by Hoffman [8] to illustrate the generality of Ford and Fulkerson’s [7] max-flow/min-cut theorem. McCormick [14] provided a combinatorial algorithm for computing maximum flows in abstract networks using a membership oracle that, given \(F \subseteq E\), returns \(P \in \mathcal{P}\) with \(P \subseteq F\) together with the corresponding order \(\preceq_P\), or certifies that no such \(P\) exists. Martens and McCormick [13] later extended this result by giving a combinatorial algorithms for a weighted version of the problem, using a stronger oracle. Applications of abstract networks include, e.g., line planning for public transit systems [12] and route assignment in evacuation planning [10, 15].

1.3 Previous Results

Dahan et al. [4] studied the case where \(\mathcal{P}\) is the set of maximal chains of a partially ordered set (poset), or, equivalently, the set of \(s\)-\(t\)-paths in a directed acyclic graph (DAG). They showed that \((\ast)\) is sufficient for the existence of a feasible distribution when the requirements fulfil the following conservation law:

\[
\pi(P) + \pi(Q) = \pi(P \times_e Q) + \pi(Q \times_e P) \quad \forall P, Q \in \mathcal{P}, e \in P \cap Q.
\]

Although their result is algorithmic, the corresponding algorithm requires explicitly enumerating all maximal chains and hence does not run in polynomial time in the size of \(E\). However, Dahan et al. [4] provide a polynomial-time algorithm for the case of affine requirements, in which there exists a vector \(\mu \in [0, 1]^E\) such that the requirements are of the form

\[
\pi(P) = 1 - \sum_{e \in P} \mu(e) \quad \forall P \in \mathcal{P}.
\]

1 Given an abstract network \((E, \mathcal{P})\) with capacities \(u \in \mathbb{R}_E^+\), a flow is a vector \(f \in \mathbb{R}_E^+\) fulfilling capacity constraints \(\sum_{P \subseteq \mathcal{P}, e \in P} f_P \leq u_e\) for all \(e \in E\). The maximum abstract flow problem asks for a flow of maximum value \(\sum_{P \subseteq \mathcal{P}} f_P\). Hoffman [8] showed that the corresponding dual linear program is totally dual integral (even in a more general weighted setting), thus generalizing the max-flow/min-cut theorem.
As a consequence of this latter result, the authors were able to characterize Nash equilibria for their network security game (which is a flow-interdiction game played on s-t-paths in a digraph) by means of a compact arc-flow LP formulation and compute such equilibria in polynomial time, under the condition that the underlying digraph is acyclic. Indeed, this positive result is particularly surprising, as similar—and seemingly simpler—flow-interdiction games had previously been shown to be NP-hard, even on DAGs [6].

1.4 Our Results

We extend the results of Dahan et al. [4] for posets/DAGs in multiple directions:

1. For the affine requirements case (A), we show that (*) is a sufficient condition for the feasibility of marginals when (E, P) is an abstract network, by providing an explicit description of a feasible decomposition for this case, based on a natural generalization of shortest-path distances to abstract networks (see Section 2). The described solutions have the property that the sets in their support can be represented by an interval matrix. A special case of this result is the case where P is the set of s-t-paths in a digraph (which is not necessarily acyclic). In this case, a feasible decomposition can be computed efficiently by a single run of a standard shortest-path algorithm.

2. We also provide an algorithm for efficiently computing the corresponding feasible decompositions for the general case of an arbitrary abstract network given by a membership oracle (see Section 3). This algorithm makes use of the following result as a subroutine.

3. We provide a combinatorial strongly polynomial algorithm for computing shortest paths in abstract networks when P is given by a membership oracle (see Section 4). Beyond its relevance for the present work, this also partially answers a question by McCormick [14], who conjectured that such an algorithm might enable a strongly polynomial algorithm for computing maximum flows in abstract networks.

4. As a consequence of our results, Nash equilibria for the network security game studied by Dahan et al. [4] can be described by a compact polyhedron and computed efficiently even when the game is played on an abstract network, including the case of a digraph with cycles (see Section 5).

5. We further show that the conservation law (C) proposed in [4] for maximal chains in posets can be reduced to the affine requirements case (A) (see Section 6). We provide a polynomial-time algorithm for computing the corresponding weights µ when the requirements π are given by a value oracle. As a consequence, the corresponding feasible decompositions can be computed efficiently in this case as well.

6. Finally, we discuss other types of set systems (see Section 7). We observe that (*) is not sufficient for the feasibility of the marginals when P consists of the bases of a matroid, perfect matchings of a bipartite graph, or paths in a multicommodity network. We further show that deciding whether a given set of marginals is feasible is NP-hard in general, even when P is given by an explicit list of small sets and the requirements are all equal to 1.
1.5 Notation

Before we discuss our results in detail, we introduce some useful notation concerning abstract networks. Let \((E, \mathcal{P})\) be an abstract network. For \(P \in \mathcal{P}\) and \(e \in P\), we use the following notation to denote prefixes of \(P\) ending at \(e\) and suffixes of \(P\) starting at \(e\), respectively:

\[
[P, e] := \{p \in P : p \preceq_P e\} \quad [e, P] := \{p \in P : e \preceq_P p\}
\]

\[
(P, e) := \{p \in P : p \prec_P e\} \quad (e, P) := \{p \in P : e \prec_P p\}
\]

For any path \(P \in \mathcal{P}\), we further denote the maximal and minimal element of with respect to \(\preceq_P\) by \(s_P\) and \(t_P\), respectively.

Throughout the paper, proofs of results marked with (♦) can be found in the appendix.

2 Feasible Decompositions in Abstract Networks

In this section we prove the following theorem, providing an explicit description of feasible decompositions of marginals in abstract networks assuming that requirements are of the form (A) and fulfil the necessary condition (★). The construction, described in the following theorem, is based on a natural generalization of shortest-path distances in abstract networks.

**Theorem 1.** Let \((E, \mathcal{P})\) be an abstract network and let \(\rho, \mu \in [0, 1]^E\) fulfilling condition (★), i.e., \(\sum_{e \in P} \rho_e \geq \pi_P := 1 - \sum_{e \in P} \mu_e\) for all \(P \in \mathcal{P}\). Define

\[
\alpha_e := \min\{\sum_{f \in (Q, e)} \mu_f + \rho_f : Q \in \mathcal{P}, e \in Q\} \cup \{1 - \rho_e\}
\]

for \(e \in E\). For \(\tau \sim U[0, 1]\) drawn uniformly at random from \([0, 1]\), let

\[
S_\tau := \{e \in E : \alpha_e \leq \tau < \alpha_e + \rho_e\}.
\]

Then \(x\) defined by \(x_S := \Pr[S_\tau = S]\) for \(S \subseteq E\) is a feasible decomposition of \(\rho\) for \((E, \mathcal{P})\) and \(\pi\).

Before we prove Theorem 1, let us first discuss some of its implications.

**Interval Structure and Explicit Computation of \(x\).** Given the vector \(\alpha\), the non-zero entries of \(x\) can be easily determined in polynomial time. Indeed, note that the set \(A := \{\alpha_e, \alpha_e + \rho_e : e \in E\}\) induces a partition of \([0, 1]\) into at most \(2|E| + 1\) intervals, with \(S_\tau' = S_\tau''\) whenever \(\tau'\) and \(\tau''\) are in the same interval. Thus, there are at most \(2|E| + 1\) non-zero entries in \(x\), whose values can be determined by sorting \(A\), determining all corresponding intervals, computing \(S_\tau\) for one \(\tau\) in each of these intervals, and then, for each occurring set \(S\), setting \(x_S\) to the total length of all intervals in which this set is attained.
**Special Case: Directed Graphs.** Consider the case where \( \mathcal{P} \) is the set of simple \( s-t \)-paths in a digraph \( D = (V,E) \) and \( E = V \cup A \). For a set \( V \in V \), let \( \mathcal{P}_{sv} \) denote the set of simple \( s-v \)-paths in \( D \). If we are given explicit access to \( D \) (rather than accessing \( \mathcal{P} \) via a membership oracle), we can compute feasible decompositions as follows. Without loss of generality, we can assume that for any \( v \in V \) and \( Q \in \mathcal{P}_{sv} \), there is \( Q' \in \mathcal{P}_{sv} \) with \( Q \subseteq Q' \).\(^2\) Then \( \alpha_v = \min_{Q \in \mathcal{P}_{sv}} \sum_{f \in Q \setminus \{v\}} \mu_f + \rho_f \) for \( v \in V \) and \( \alpha_a = \min_{Q \in \mathcal{P}_{sv}} \sum_{f \in Q} \mu_f + \rho_f \) for \( a = (v,w) \in A \). Hence, the vector \( \alpha \) corresponds to shortest-path distances in \( D \) with respect to \( \rho + \mu \) (with costs on both arcs and nodes). Both \( \alpha \) and the corresponding feasible decomposition of \( \rho \) can be computed by a single run of Dijkstra’s \([5]\) algorithm in \( D \).

Computing feasible decompositions in the general case of arbitrary abstract networks is more involved. We show how this can be achieved in Section 3.

**Proof of Theorem 1.** We show that \( x \) as constructed in Theorem 1 is a feasible decomposition. Note that \( x \) fulfills (3) and (4) by construction. Note further that \( x \) fulfills (1) because \( \sum_{S \in \mathcal{P} \setminus \{e\}} x_S = \Pr[e \in S] = \Pr[\alpha_e \leq \tau < \alpha_e + \rho_e] = \rho_e \) for all \( e \in E \), where the second identity follows from \( 0 \leq \alpha_e \leq 1 - \rho_e \). It remains to prove that \( x \) fulfills (2). The following lemma will be helpful in this endeavour.

**Lemma 2.** Given \((E, \mathcal{P}), \rho, \mu, \) and \( \alpha \) as described in Theorem 1, the following two conditions are fulfilled for every \( P \in \mathcal{P} \):

1. \( \alpha_{t_P} + \mu_{t_P} + \rho_{t_P} \geq 1 \) and
2. for every \( e \in P \setminus \{t_P\} \) there is \( e' \in (e, P) \) with \( \alpha_{e'} \leq \alpha_e + \mu_e + \rho_e \).

**Proof.** We first show statement 1. By contradiction assume \( \alpha_{t_P} + \mu_{t_P} + \rho_{t_P} < 1 \). Let \( Q \in \mathcal{P} \) with \( t_P \in Q \) and \( \sum_{f \in (Q,t_P)} \mu_f + \rho_f = \alpha_{t_P} \) and let \( R := Q \times_{t_P} P \). Note that \( R \subseteq [Q,t_P] \) and hence \( \sum_{e \in R} \mu_e + \rho_e \leq \alpha_{t_P} + \mu_{t_P} + \rho_{t_P} < 1 \), implying \( \sum_{e \in R} \rho_e < 1 - \sum_{e \in R} \mu_e \), a contradiction to (\(*\)\).

We now turn to statement 2. If \( \alpha_e \geq 1 - \mu_e - \rho_e \), then the statement follows with \( e' = t_P \) because \( \alpha_{t_P} \leq 1 \leq \alpha_e + \mu_e + \rho_e \). Thus assume \( \alpha_e < 1 - \mu_e - \rho_e \) and let \( Q \in \mathcal{P} \) with \( \alpha_e = \sum_{f \in (Q,e)} \mu_f + \rho_f \). Let \( R := Q \times_e P \). By (\(*\)\) we observe that \( \sum_{f \in R} \mu_f + \rho_f \geq 1 > \alpha_e + \mu_e + \rho_e \), which implies \( R \setminus [Q,e] \neq \emptyset \) because \( \mu, \rho \geq 0 \). Thus, let \( e' \in R \setminus [Q,e] \) be minimal with respect to \( \prec_R \). Observe that \( R \setminus [Q,e] \subseteq (e, P) \) and hence \( e' \in (e, P) \). The statement then follows from

\[ \alpha_{e'} \leq \sum_{f \in (R,e')} \mu_f + \rho_f \leq \sum_{f \in [Q,e]} \mu_f + \rho_f = \alpha_e + \mu_e + \rho_e, \]

where the second inequality is due to the fact that \( (R,e') \subseteq [Q,e] \) by choice of \( e' \) and the fact that \( \mu, \rho \geq 0 \). \( \blacksquare \)

With the help of Lemma 2, we can prove that \( x \) fulfills (2) as follows. Let \( P \in \mathcal{P} \). For \( e \in P \) define

\[ \phi(e) := \Pr[S \cap [P,e] \neq \emptyset] = \tau \leq \alpha_e + \rho_e] + \sum \mu_f. \]

\(^2\) This can be ensured by introducing arcs \( (v,t) \) with \( \mu_{(v,t)} = 1 \) and \( \rho_{(v,t)} = 0 \) for all \( v \in V \setminus \{t\} \). Note that this does not change the set of feasible decompositions of \( \rho \).
Let $F := \{e \in P : \phi(e) \geq \alpha_e + \mu_e + \rho_e\}$. We will show that $t_P \in F$. Note that this suffices to prove (2), because the definition of $F$ together with statement 1 of Lemma 2 imply $\phi(t_P) \geq \alpha_{t_P} + \mu_{t_P} + \rho_{t_P} \geq 1$, which in turn yields

$$\sum_{S \subseteq E : S \cap P \neq \emptyset} x_S = \Pr[S \cap P \neq \emptyset] \geq \phi(t_P) - \sum_{f \in P} \mu_f \geq 1 - \sum_{f \in P} \mu_f = \pi_P.$$  

We proceed to show $t_P \in F$. By contradiction assume this is not the case. Note that $F \neq \emptyset$ because $\alpha_{s_P} = 0$ and $\phi(s_P) = \Pr[s_P \in S] + \mu_{s_P} = \rho_{s_P} + \mu_{s_P}$. Thus let $e \in F$ be maximal with respect to $\prec_P$. Because $e \neq t_P$, we can invoke statement 2 of Lemma 2 and obtain $e' \in (e, P)$ with

$$\alpha_{e'} \leq \alpha_e + \mu_e + \rho_e. \quad (5)$$

We will show that $e' \in F$, contradicting our choice of $e$. Note that the definition of $\phi$ and the fact that $e' \not\succ_P e$ imply

$$\phi(e') \geq \phi(e) + \Pr[e' \in S_{e'} \land \tau > \alpha_e + \rho_e] + \mu_{e'}$$

$$\geq \alpha_e + \mu_e + \rho_e + \Pr[e' \in S_{e'} \land \tau > \alpha_e + \rho_e] + \mu_{e'}, \quad (6)$$

where the second inequality follows from $e \in F$. Moreover, observe that $e' \in S_{e'}$ if and only if $\alpha_{e'} \leq \tau < \alpha_{e'} + \rho_{e'}$ and hence

$$\Pr[e' \in S_{e'} \land \tau > \alpha_e + \rho_e] = \alpha_{e'} + \rho_{e'} - \max\{\alpha_{e'}, \alpha_e + \rho_e\} \geq \alpha_{e'} + \rho_{e'} - (\alpha_e + \mu_e + \rho_e),$$

where the inequality follows from (5). Combining this bound with (6) yields $\phi(e') \geq \alpha_{e'} + \mu_{e'} + \rho_{e'}$ and hence $e' \in F$, contradicting our choice of $e$ and completing the proof of Theorem 1. \hfill \Box

### 3 Computing Feasible Decompositions

Complementing our existence result from the previous section, we now discuss how to compute corresponding feasible decompositions. We will assume that the ground set $E$ is given explicitly, while the set of abstract paths $P$ is given by a membership oracle that, given $F \subseteq E$, either returns $P \in P$ with $P \subseteq F$ and the corresponding order $\preceq_P$, or confirms that no $P \in P$ with $P \subseteq F$ exists.

By our arguments in Section 2, it suffices to compute the values of $\alpha_e$ for all $e \in E$. Unfortunately, a complication arises in that even finding a path containing a certain element $e \in E$ is NP-hard.\footnote{Note that even for the special case where $P$ corresponds to the set of simple $s$-$t$-paths in a digraph, finding $P \in P$ containing a certain arc is equivalent to the 2-disjoint path problem. Simply side-stepping this issue by introducing additional elements as done in the second remark after Theorem 1 is not possible here, because we are restricted to accessing $P$ only via the membership oracle.} However, as we show below, it is possible to identify a subset $U \subseteq E$ for which we can compute the values of $\alpha$, while the elements in $E \setminus U$ turn out to be redundant w.r.t. the feasibility of the marginals. From this, we obtain the following theorem.
Theorem 3. There is an algorithm that, given an abstract network \((E, \mathcal{P})\) via a membership oracle and \(\rho, \mu \in [0, 1]^E\) such that \(\sum_{e \in P} \rho_e \geq \pi_P := 1 - \sum_{e \in P} \mu_e\) for all \(P \in \mathcal{P}\), computes a feasible decomposition \(\rho\) for \((E, \mathcal{P})\) and \(\pi\) in time \(O(|E|^3 \cdot T_P)\), where \(T_P\) denotes the time for a call to the membership oracle of \(\mathcal{P}\).

The Algorithm Theorem 3 is established via Algorithm 1, which computes values \(\bar{\alpha}_e\) for elements \(e\) in a subset \(U \subseteq E\) as follows. Starting from \(U = \emptyset\), the algorithm iteratively computes a path \(P\) minimizing \(\sum_{f \in P \setminus U} \mu_f + \rho_f\) and adds the first element \(e\) of \(P \setminus U\) to \(U\), determining \(\bar{\alpha}_e\) based on the length of \((P, e)\).

Algorithm 1: Computing a feasible decomposition

```plaintext
Initialize \(U := \emptyset\).

while \(\min_{P \in \mathcal{P}} \sum_{f \in P \setminus U} \mu_f + \rho_f < 1\) do
  Let \(P \in \text{argmin}_{P \in \mathcal{P}} \sum_{f \in P \setminus U} \mu_f + \rho_f\).
  Let \(e := \min_{\rho \in P \setminus U} |P|\).
  Set \(U := U \cup \{e\}\) and \(\bar{\alpha}_e := \min \left\{ \sum_{f \in (P, e)} \mu_f + \rho_f, 1 - \rho_e \right\}\).

return \(\bar{\alpha}, U\)
```

Analysis First note that in every iteration of the while loop, the set \(P \setminus U\) is nonempty because \(\sum_{f \in P} \mu_f + \rho_f \geq 1\) by the assumption on the input in Theorem 3. Hence the algorithm is well-defined and terminates after at most \(|E|\) iterations. We further remark that finding \(P \in \mathcal{P}\) minimizing \(\sum_{e \in P \setminus U} \rho_f + \mu_f\) can be done in time \(O(|E|^2 T_P)\) using the Algorithm 2 described in Section 4.

The following lemma then suffices to complete the proof of Theorem 3.

Lemma 4 (\(\dagger\)). Let \(\bar{\alpha}, U\) be the output of Algorithm 1 and define \(\bar{\rho}_e := \rho_e\) and \(\bar{\mu}_e := \mu_e\) for \(e \in U\) and \(\bar{\rho}_e := 0\) and \(\bar{\mu}_e := 0\) for \(e \in E \setminus U\). Then

1. \(\sum_{e \in P} \bar{\rho}_e \geq \bar{\pi}_P := 1 - \sum_{e \in P} \bar{\mu}_e\) for all \(P \in \mathcal{P}\) and
2. \(\bar{\alpha}_e = \min \left\{ \sum_{f \in (Q, e)} \bar{\rho}_f + \bar{\mu}_f : Q \in \mathcal{P}, e \notin Q \right\} \cup \{1 - \bar{\rho}_e\}\) for all \(e \in U\).

Indeed, observe that Lemma 4 together with Theorem 1 implies that \(\bar{\alpha}\) induces a feasible decomposition \(\bar{x}\) of \(\bar{\rho}\) for \((E, \mathcal{P})\) and \(\bar{\pi}\). Because \(\bar{\rho}_e \leq \rho_e\) for all \(e \in E\) and \(\bar{\pi}_P \geq \pi_P\) for all \(P \in \mathcal{P}\), this decomposition can be extended to a feasible decomposition of \(\rho\) for \((E, \mathcal{P})\) and \(\pi\) by arbitrarily incorporating the elements from \(E \setminus U\). This completes the proof of Theorem 3.

4 Computing Shortest Paths in Abstract Networks

In this section, we consider the following natural generalization of the classic shortest \(s\text{-}t\)-path problem in digraphs: Given an abstract network \((E, \mathcal{P})\) and a cost vector \(\gamma \in \mathbb{R}^E\), find a path \(P \in \mathcal{P}\) minimizing \(\sum_{e \in P} \gamma_e\). We provide a combinatorial, strongly polynomial algorithm for this problem, accessing \(\mathcal{P}\).
Decomposition of Probability Marginals in Abstract Networks

only via a membership oracle. In fact, the question for such an algorithm was already raised by McCormick [14], who conjectured that it can be used to turn (an adaptation of) his combinatorial, but only weakly polynomial algorithm for the maximum abstract flow problem into a strongly polynomial one.

Theorem 5. There is an algorithm that, given an abstract network \((E, \mathcal{P})\) via a membership oracle and a cost vector \(\gamma \in \mathbb{R}_+^E\) computes a path \(P \in \mathcal{P}\) minimizing \(\sum_{e \in P} \gamma_e\) in time \(O(|E|^2 \cdot T_P)\), where \(T_P\) denotes the time for a call to the membership oracle of \(\mathcal{P}\).

The Algorithm For notational convenience, we assume that there is \(s, t \in E\) with \(s_P = s\) and \(t_P = t\) for all \(P \in \mathcal{P}\). Note that this assumption is without loss of generality, as it can be ensured by adding dummy elements \(s\) and \(t\) to \(E\) and including them at the start and end of each path, respectively.

The algorithm is formally described as Algorithm 2. It can be seen as a natural extension of Dijkstra’s [5] algorithm in that it maintains for each element \(e \in E\) a (possibly infinite) label \(\psi_e\) indicating the length of the shortest segment \([Q_e, e]\) for some \(Q_e\) found so far, and in that its outer loop iteratively chooses an element with currently smallest label for processing. However, updating these labels is more involved, as an abstract network does not provide local concepts such as “the set of arcs leaving a node”. In its inner loop, the algorithm therefore carefully tries to extend the segment \(Q_e\) for the currently processed element \(e\) to find new shortest segments \(Q_{e'}\) for other elements \(e'\).

Algorithm 2: Computing a shortest path in an abstract network

| Initialize \(T := \emptyset\), \(\psi_s := \gamma_s\), and \(\psi_e := \infty\) for all \(e \in E \setminus \{s\}\). |
| Let \(Q_s \in \mathcal{P}\). |
| while \(\psi_t > \min_{f \in E \setminus T} \psi_f\) do |
| Let \(e := \text{argmin}_{f \in E \setminus T} \psi_f\). |
| Let \(F := (E \setminus T) \cup [Q_e, e]\). |
| while there is \(P \in \mathcal{P}\) with \(P \subseteq F\) do |
| Let \(e' := \min_{P \setminus P} P \setminus [Q_e, e]\). |
| Set \(F := F \setminus \{e'\}\). |
| if \(\sum_{f \in [P, e']} \gamma_f < \psi_{e'}\) then |
| Set \(\psi_{e'} := \sum_{f \in [P, e']} \gamma_f\) and \(Q_{e'} := P\). |
| Set \(T := T \cup \{e\}\). |
| return \(Q_t\) |

Analysis The proof of the correctness of Algorithm 2 crucially relies on the following lemma, which essentially certifies that the algorithm does not overlook any shorter path segments when processing an element.

Lemma 6 (♦). Algorithm 2 maintains the following invariant: For all \(P \in \mathcal{P}\), there is \(e \in P\) with \([e, P] \cap T = \emptyset\) and \(\psi_e \leq \sum_{f \in [P, e]} \gamma_f\).
Proof of Theorem 5. When Algorithm 2 terminates, ψ_t ≤ ψ_f for all f ∈ E \ T by the termination criterion of the outer while loop. Let P ∈ ℙ. By Lemma 6 there is an element e ∈ P \ T with ψ_e ≤ \sum f \in [P_e] \gamma_f. Note that this implies \sum f \in Q_t \gamma_f = ψ_t ≤ ψ_e = \sum f \in [P_e] \gamma_f ≤ \sum f \in P \gamma_f, where the last inequality uses the fact that \gamma_f ≥ 0 for all f ∈ E. We conclude that the path Q_t returned by the algorithm is indeed a shortest path.

To see that the algorithm terminates in polynomial time, observe that the outer while loop stops after at most |E| − 1 iterations, as in each iteration an element from E \ \{t\} is added to T and the termination criterion is fulfilled if T = E \ \{t\}. Furthermore, each iteration of the inner while loop removes an element from F and hence after at most |F| ≤ |E| iterations no path P ⊆ F exists anymore, implying that the inner while loop terminates. □

5 Dahan et al.’s Network Security Game

Dahan et al. [4] studied the following network security game. The input is a set system (E, ℙ) with capacities u ∈ ℝ^E_+, transportation cost c ∈ ℝ^E_+ and interdiction costs d ∈ ℝ^E_+. There are two players: the routing entity R, whose strategy space is the set of flows F := \{f ∈ ℙ : \sum p \in ℙ \sum e \in p \ f_p ≤ u_e \ ∀ e \in E\}, and the interdictor I, who selects a subset of elements S ⊆ E to interdict, with the intuition that all flow on interdicted elements is disrupted. Given strategies f ∈ F and S ⊆ E, the payoffs for R and I, respectively, are given by

\begin{align*}
P_R(f, S) := & \sum p \in ℙ \sum e \in p \ f_p - \sum p \in ℙ \sum e \in p \ c_e f_p \quad \text{and} \\
P_I(f, S) := & \sum p \in ℙ \sum e \in p \ f_p - \sum e \in S \ d_e,
\end{align*}

respectively. That is, R’s payoff is the total amount of non-disrupted flow, reduced by the cost for sending flow f, while I’s payoff is the total amount of flow that is disrupted, reduced by the interdiction cost for the set S.

We are interested in finding (mixed) Nash equilibria (NE) for this game, i.e., random distributions σ_R and σ_I over the strategy spaces of I and R, respectively, such that no player can improve their expected payoff by unilateral deviation. However, the efficient computation of such equilibria is hampered by the fact that the strategy spaces of both players are of exponential size/dimension in the size of the ground set E. To overcome this issue, Dahan et al. [4] proposed to consider the following pair of primal and dual linear programs:

\begin{align*}
\text{[LP}_R\text{]} \quad & \max \sum_{p \in ℙ} \pi^f_p f_p & \text{[LP}_I\text{]} \quad & \min \sum_{e \in E} u_e \mu_e + d_e \rho_e \\
\text{s.t.} \quad & \sum_{p \in ℙ \cap e \in p} f_p ≤ u_e \quad \forall e \in E & \text{s.t.} \quad & \sum_{e \in ℙ} \mu_e + \rho_e ≥ \pi^f_p \quad \forall P \in ℙ \\
& \sum_{p \in ℙ \cap e \in p} f_p ≤ d_e \quad \forall e \in E & \mu ≥ 0 \\
& f ≥ 0 & \rho ≥ 0,
\end{align*}

where π^f_p := 1 - \sum a \in P c_a. Dahan et al. [4] showed the following result.
Theorem 7 (Dahan et al. [4]). Let \( f^* \) and \((\mu^*, \rho^*)\) be optimal solutions to \([LP_R]\) and \([LP_I]\), respectively. Let \( \sigma_I \) be a feasible decomposition of \( \rho^* \) for \((E, P)\) and \( \pi_P := \pi_P^* - \sum_{e \in P} \mu_e^* \) and let \( \sigma_R \) be a distribution over \( F \) with \( \sum_{f \in F} \sigma_R,f f_P = f^*_P \). Then \((\sigma_R, \sigma_I)\) is a Nash equilibrium.

In particular, note that any feasible solution to \([LP_I]\) defines marginals \( \rho \) that fulfil \( \text{(⋆)} \) for \( \pi_P := \pi_P^* - \sum_{e \in P} \mu_e \). Hence, if condition \( \text{(⋆)} \) is sufficient for feasibility of marginals in the set system \( P \) under affine requirements, any pair of optimal solutions to the LPs induces a Nash equilibrium. If we can moreover efficiently compute optimal solutions to the LPs and the corresponding feasible decompositions, we can efficiently find a Nash equilibrium.

Dahan et al. [4] showed that this is possible for the case that \( P \) is the set of \( s-t \)-paths in a DAG and hence NE for the game can be found efficiently in that setting. This positive result is made particularly interesting by the fact that equilibria are hard to compute for the variant of the game in which the interdictor is limited by a budget, even when interdiction costs are uniform and transportation costs are zero and the game is played on a DAG [6].

Our results in Sections 2 to 4 imply that all three conditions for the computability of NE are also met when \((E, P)\) is an abstract network (note that we can use Algorithm 2 to separate the constraints of \([LP_I]\)). Hence we can compute Nash equilibria for the above game when \((E, P)\) is an abstract network given by a membership oracle, in time polynomial in \(|E|\), including the case where the game is played on a digraphs with cycles.

We remark that Dahan et al. [4] also showed that, if there is at least one dual solution with a decomposition that assigns positive probability to the empty set, then all NE of the game are of the form described in Theorem 7. They showed that this condition is always fulfilled in the DAG case when all transportation costs are positive. Via a small adjustment to our construction in Section 2, the same result can be proven for the case of abstract networks (♦).

6 The Conservation Law for Partially Ordered Sets

As discussed in Section 1, Dahan et al. [4] established the sufficiency of \( \text{(⋆)} \) in partially ordered sets not only for the case of affine requirements \( \text{(A)} \) but also for the case where requirements fulfill the conservation law \( \text{(C)} \). However, they left it open whether it is possible to efficiently compute the corresponding decompositions in the latter case. In this section, we resolve this question by showing that the conservation law \( \text{(C)} \) for maximal chains in a poset can be reduced to the case of affine requirements \( \text{(A)} \) in the corresponding Hasse diagram,\(^4\) for which a feasible decomposition then can be computed efficiently.

Theorem 8 (♦). Let \( D = (V, A) \) be a directed acyclic graph, let \( s, t \in V \), and let \( P = 2^{V \cup A} \) be the set of \( s-t \)-paths in \( D \). Let \( \pi \in [0, 1]^P \) such that \( \text{(C)} \) is fulfilled.

\(^4\) The Hasse diagram of a poset \((E, \preceq)\) is a directed acyclic graph whose nodes are the elements of \( E \) and whose maximal paths correspond to the maximal chains of \( \preceq \). See Appendix D.2 for details on this transformation and why it is necessary.
Then there exists $\mu \in [0,1]^{V \cup A}$ such that $\pi_P = 1 - \sum_{e \in P} \mu_e$. Furthermore, $\mu$ can be computed in strongly polynomial time in $|V|$ and $|A|$ when $\pi$ is given by an oracle that, given $P \in \mathcal{P}$, returns $\pi_P$.

Proof (sketch). By Farkas’ lemma, the existence of $\mu$ is equivalent to showing that $\sum_{P \in \mathcal{P}} (1 - \pi_P) y_P \geq 0$ for every $y \in \mathbb{R}^P$ with $\sum_{P \in \mathcal{P}, e \in P} y_P \geq 0$ for all $e \in V \cup A$. This property can be established by iteratively applying (C) to transform $y$ into a nonnegative vector without changing $\sum_{P \in \mathcal{P}} (1 - \pi_P) y_P$. \qed

7 Other Set Systems

The results in this paper lead to the question whether sufficiency of (⋆) and computability of feasible decompositions can be established for other set systems, beyond abstract networks. We give negative answers for several natural candidates of such systems and point out interesting questions for future research.

Sufficiency of (⋆) (♦) There are simple counterexamples for the sufficiency of (⋆) in the following cases, even when assuming that $\pi \equiv 1$: when $\mathcal{P}$ is the set of bases of a matroid; when $\mathcal{P}$ is the set of perfect matchings in a bipartite graph; when $\mathcal{P}$ is the set of $s_i$-$t_i$-paths in a digraph with multiple terminal pairs $(s_i,t_i)$. An interesting question in this context is whether we can describe the systems for which (⋆) is sufficient by means of forbidden substructures.

Approximately Feasible Decompositions (♦) Given the non-existence result mentioned above, one may be interested in finding decompositions that satisfy the requirements at least approximately. We say a decomposition $x$ of marginals $\rho$ is $\beta$-approximately feasible, for $\beta \in [0,1]$, if it fulfils (1), (3), (4) and $\sum_{S \subseteq E: S \cap P \neq \emptyset} x_S \geq \beta \cdot \pi_P$ for all $P \in \mathcal{P}$. Indeed, if marginals $\rho$ fulfil (⋆) for requirements $\pi$, a $(1 - 1/e)$-approximately feasible decomposition always exists: Simply include each element $e \in E$ in the random set independently with probability $\rho_e$. While this bound is tight for general set systems (e.g., for the bases of the $k$-uniform matroid), an interesting question for future research is whether better guarantees may be achieved for some classes of systems.

Computing Feasible Decompositions and Optimization (♦) For a given instance, we may also be interested in finding a decomposition of the given marginals that is $\beta$-approximately feasible for the largest possible value of $\beta$. Note that this also includes the case of finding a feasible decomposition if it exists (resulting in $\beta = 1$). Unfortunately, this latter problem is NP-complete, even in quite restricted cases, as evidenced by the theorem below. However, note that this hardness result still leaves room for approximating the best possible $\beta$.

Theorem 9 (♦). The following decision problem is NP-complete: Given a set system $(E, \mathcal{P})$ with $|P| = 3$ for all $P \in \mathcal{P}$ and marginals $\rho \in [0,1]^E$, is there a feasible decomposition of $\rho$ for $(E, \mathcal{P})$ and requirement vector $\pi \equiv 1$?
References

[1] Dimitris Bertsimas, Ebrahim Nasrabadi, and James B. Orlin. On the power of randomization in network interdiction. *Operations Research Letters*, 44:114–120, 2016.
[2] José Correa, Tobias Harks, Vincent J.C. Kreuzen, and Jannik Matuschke. Fare evasion in transit networks. *Operations Research*, 65:165–183, 2017.
[3] Mathieu Dahan, Saurabh Amin, and Patrick Jaillet. Probability distributions on partially ordered sets and network security games. Technical report, arXiv:1811.08516v1, 2018.
[4] Mathieu Dahan, Saurabh Amin, and Patrick Jaillet. Probability distributions on partially ordered sets and network interdiction games. *Mathematics of Operations Research*, 47:458–484, 2022.
[5] Edsger W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 269:271, 1959.
[6] Yann Disser and Jannik Matuschke. The complexity of computing a robust flow. *Operations Research Letters*, 48:18–23, 2020.
[7] Lester R. Ford and Delbert R. Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 8:399–404, 1956.
[8] Alan J. Hoffman. A generalization of max flow—min cut. *Mathematical Programming*, 6:352–359, 1974.
[9] Tim Holzmann and J. Cole Smith. The shortest path interdiction problem with randomized interdiction strategies: Complexity and algorithms. *Operations Research*, 69:82–99, 2021.
[10] Jan-Philipp W. Kappmeier. *Generalizations of flows over time with applications in evacuation optimization*. PhD thesis, TU Berlin, 2015.
[11] Jan-Philipp W. Kappmeier, Jannik Matuschke, and Britta Peis. Abstract flows over time: A first step towards solving dynamic packing problems. *Theoretical Computer Science*, 544:74–83, 2014.
[12] Marika Karbstein. *Line planning and connectivity*. PhD thesis, TU Berlin, 2013.
[13] Maren Martens and S. Thomas McCormick. A polynomial algorithm for weighted abstract flow. In *Integer Programming and Combinatorial Optimization*, volume 5035 of Lecture Notes in Computer Science, pages 97–111. Springer, 2008.
[14] S. Thomas McCormick. A polynomial algorithm for abstract maximum flow. In *Proceedings of the 7th annual ACM-SIAM Symposium on Discrete Algorithms*, pages 490–497, 1996.
[15] Urmila Pyakurel, Durga Prasad Khanal, and Tanka Nath Dhamala. Abstract network flow with intermediate storage for evacuation planning. *European Journal of Operational Research*, 2022.
[16] Thomas J. Schaefer. The complexity of satisfiability problems. In *Proceedings of the tenth annual ACM Symposium on Theory of Computing*, pages 216–226, 1978.
[17] Dávid Szaszlér. Security games on matroids. *Mathematical Programming*, 161:347–364, 2017.
[18] Milind Tambe. *Security and game theory: Algorithms, deployed systems, lessons learned*. Cambridge University Press, 2011.

[19] Éva Tardos. A strongly polynomial algorithm to solve combinatorial linear programs. *Operations Research*, 34:250–256, 1986.
A Missing Details from Section 3

Lemma 4 (⋆). Let \( \bar{\alpha}, U \) be the output of Algorithm 1 and define \( \bar{\mu}_e := \mu_e \) and \( \bar{\mu}_e := 0 \) for \( e \in U \) and \( \bar{\mu}_e := 0 \) for \( e \in E \setminus U \). Then

1. \( \sum_{e \in P} \bar{\mu}_e \geq \pi_P := 1 - \sum_{e \in P} \bar{\mu}_e \) for all \( P \in \mathcal{P} \) and
2. \( \bar{\alpha}_e = \min \left\{ \sum_{f \in (Q,e)} \bar{\mu}_f + \bar{\rho}_f : Q \in \mathcal{P}, e \in Q \right\} \cup \{ 1 - \bar{\mu}_e \} \) for all \( e \in U \).

Proof. Let \( P^{(i)} \in \mathcal{P} \) and \( e^{(i)} \in E \) be the path and the element chosen in iteration \( i \) of the while loop. Let \( U^{(0)} := \emptyset \) and \( U^{(i)} := \{ e^{(1)}, \ldots, e^{(i)} \} \). Note that \( U^{(i-1)} \) is the state of \( U \) at the beginning of iteration \( i \) of the while loop. Note that (⋆) implies \( \sum_{f \in P^{(i)}} \mu_f + \mu_f \geq 1 > \sum_{f \in P^{(i)} \cap U^{(i-1)}} \mu_f + \rho_f \), where the final inequality follows from the termination criterion of the while loop. Thus \( P^{(i)} \setminus U^{(i-1)} \neq \emptyset \), implying that the second line of the while loop is well-defined and the algorithm adds a new element to \( U \) in each iteration of the while loop. We conclude that the algorithm terminates after \( k \) iterations for some \( k \leq |E| \).

The termination criterion implies \( \sum_{e \in P \cap U} \mu_e \geq 1 - \sum_{e \in P \cap U} \mu_e \geq \pi_P \) for all \( P \in \mathcal{P} \), proving statement 1 of the lemma. We next establish statement 2 of the lemma. To this end let \( e \in U \) and \( Q \in \mathcal{P} \) with \( e \in Q \). Let \( i \in \{ 1, \ldots, k \} \) be such that \( e = e^{(i)} \) and define \( R := Q \times_e P^{(i)} \). Note that

\[
\sum_{f \in P^{(i)} \cap U^{(i-1)}} \mu_f + \rho_f \leq \sum_{f \in R \cap U^{(i-1)}} \mu_f + \rho_f \leq \sum_{f \in (Q,e) \cap U^{(i-1)}} \mu_f + \rho_f + \sum_{f \in [e,P^{(i)} \cap U^{(i-1)}]} \mu_f + \rho_f,
\]

where the first inequality follows from the choice of \( P^{(i)} \) by the algorithm in iteration \( i \), and the second inequality follows from construction of \( R \). From this, we conclude

\[
\alpha_e = \sum_{f \in (P^{(i)}, e)} \mu_f + \rho_f = \sum_{f \in (P^{(i)}, e) \cap U^{(i-1)}} \mu_f + \rho_f \leq \sum_{f \in (Q,e) \cap U^{(i-1)}} \mu_f + \rho_f \leq \sum_{f \in (Q,e)} \mu_f + \rho_f,
\]

where the second identity follow from \( (P^{(i)}, e) \subseteq U^{(i-1)} \) by choice of \( e = e^{(i)} \) as first element on \( P^{(i)} \setminus U^{(i-1)} \). \( \square \)

B Missing Details from Section 4

Lemma 6 (⋆). Algorithm 2 maintains the following invariant: For all \( P \in \mathcal{P} \), there is \( e \in P \) with \( [e, P] \cap T = \emptyset \) and \( \psi_e \leq \sum_{f \in [P,e]} \gamma_f \).

Proof. The invariant is clearly fulfilled initially as \( T = \emptyset \) and \( \psi_s = \gamma_s = \sum_{e \in [P,s]} \gamma_e \) for all \( P \in \mathcal{P} \) (recall our assumption \( s = s_P \in P \) for all \( P \in \mathcal{P} \)). Now assume that the invariant holds at the beginning of an iteration of the outer while loop, i.e., for each \( P \in \mathcal{P} \) there is \( e_P \in P \) with \([e_P, P] \cap T = \emptyset \) and \( \psi(e_P) = \sum_{f \in Q_{e_P}} \gamma_f \leq \sum_{f \in [P,e_P]} \gamma_f \). Let \( e \in \arg\min_{f \in \emptyset \setminus T} \psi_e \) be the element selected at the beginning of this iteration and added to \( T \) at its end. Consider any path \( P \in \mathcal{P} \) and distinguish two cases:
- \( e \notin [e_p, P] \): In this case, \( e_p \) continues to fulfill the conditions of the invariant for \( P \), as \([e_p, P] \cap (T \cup \{e\}) = \emptyset \) and \( \psi_{e_p} = \sum_{f \in [Q_{e_p}, e_p]} \gamma_f \) either remains unchanged or is decreased during the course of the inner while loop.

- \( e \in [e_p, P] \): Let \( R := Q_e \times e \). Observe that \( R \subseteq E \setminus T \cup [Q_e, e] \) and hence there must be an iteration of the inner while loop in which some \( e' \in R \) is chosen for removal of \( F \) with \( e' \) being \( \prec_{p'} \)-minimal in \( P' \setminus [Q_e, e] \) for some \( P' \in \mathcal{P} \). In that iteration of the inner while loop, \( Q_{e'} \) is set to \( P' \) and \( \psi_{e'} \) is set to

\[
\psi_{e'} = \sum_{f \in [P', e']} \gamma_f \leq \sum_{f \in [Q_e, e]} \gamma_f + \gamma_{e'} = \psi_e + \gamma_{e'} \leq \psi_{e_p} + \gamma_{e'},
\]

where the fist inequality follows from \( [P', e'] \subseteq [Q_e, e] \cup \{e'\} \) by choice of \( e' \) as first element of \( P' \) not in \( [Q_e, e] \), and the second inequality follows from the choice of \( e \in \text{argmin}_{f \in E \setminus T} \psi_f \) and the fact that \( e_p \notin T \). Note that \( e' \in R \setminus [Q_e, e] \) implies \( e' \in (e, P) \) and \( [e', P] \cap (T \cup \{e\}) = \emptyset \), as \( (e, P) \subseteq [e_p, P] \). Note further that \( \psi_{e'} \leq \psi_{e_p} + \gamma_{e'} \leq \sum_{f \in [P', e']} \gamma_f \), from which we conclude that \( e' \) fulfills the statement of the invariant for \( P \). \( \square \)

C Missing Details from Section 5

Dahan et al. [4] showed that the linear programs \([LP_R]\) and \([LP_I]\) describe all NE for their security game (in the sense that all NE can be constructed from optimal solutions to the LPs as described in Theorem 7) if the following conditions are met:

- \((*)\) is a sufficient condition for the feasibility of marginals and
- there exists an optimal solution \( \mu^*, \rho^* \) to \([LP_I]\) such that \( \rho^* \) has a feasible decomposition \( x \) w.r.t. \( \pi_p := 1 - \sum_{e \in P} c_e + \mu^*_e \) with \( x_0 > 0 \).

They showed that both conditions are met if \((E, \mathcal{P})\) is the set of \( s-t \)-paths in a DAG and \( c_e > 0 \) for all \( e \in E \). Here, we argue that the same conditions are met if \((E, \mathcal{P})\) is an abstract network and \( c_P > 0 \) for all \( P \in \mathcal{P} \). The first condition is satisfied because of Theorem 1. For the second condition, assume \( c_P > 0 \). Let \( \mu^*, \rho^* \) be an optimal solution to \([LP_I]\). Because \( \pi^*_P < 1 \) for all \( P \in \mathcal{P} \), we can assume without loss of generality that \( \rho^*_e < 1 \) for every \( e \in E \).

Consider the abstract network \((E', \mathcal{P}')\) with \( E' := E \cup \{s\} \) and \( \mathcal{P}' := \{\{s\} \cup P : P \in \mathcal{P}\} \) with \( s \prec_{p'} e \) and \( e \prec_{p'} f \) for every \( P' = \{s\} \cup P \in \mathcal{P}' \) and \( e, f \in P \) with \( e \prec_{p'} f \). That is \((E', \mathcal{P}')\) is constructed from \((E, \mathcal{P})\) by attaching a global starting node \( s \) to every path of \( \mathcal{P} \).

Let \( E_s := \{e \in E : e = s_P \text{ for some } P \in \mathcal{P}\} \). Define \( \varepsilon := \min_{e \in E_s} c_e \) and let \( \mu'_e := \varepsilon, \mu'_e := c_e + \mu_e - \varepsilon \geq 0 \) for \( e \in E_s \), and \( \mu'_e := c_e + \mu^*_e \) for \( e \in E \setminus E_s \). Let \( \rho'_e := 0 \) and \( \rho'_e := \rho^*_e \) for \( e \in E \) and let \( \pi^*_{p'} := 1 - \sum_{e \in p'} \mu'_e \) for \( P' \in \mathcal{P}' \). Note that for \( P' \in \mathcal{P}' \) and \( P \in \mathcal{P} \) with \( P' = \{s\} \cup P \), by construction \( \sum_{e \in P} \rho^*_e = \sum_{e \in P} \rho^*_e \) and \( \pi^*_{p'} = \pi_p - (|E_s| - 1) \varepsilon \).

The following lemma then implies that the second condition is fulfilled for \( \rho^* \) in \((E, \mathcal{P})\).
Lemma 10. There is a feasible decomposition \( x \) of \( \rho' \) for \((E', \mathcal{P}')\) and \( \pi' \) with \( x_\emptyset > 0 \). Moreover, \( \sum_{S \subseteq E : S \cap P \neq \emptyset} x_S \geq \pi_P \) for all \( P \in \mathcal{P} \), i.e., \( x \) restricted to \( E \) is a feasible decomposition of \( \rho^* \) for \((E, \mathcal{P})\) and \( \pi \).

Proof. First observe that \( \sum_{e \in P} \rho'_e = \sum_{e \in P} \rho_e \geq \pi_P \geq \pi_P' \) for all \( P' \in \mathcal{P}' \) and \( P \in \mathcal{P} \) with \( P' = \{ s \} \cup P \). Hence \( \rho' \) fulfills (\( \star \)) for \((E', \mathcal{P}')\) and \( \pi' \). Thus, we can apply Theorem 1 to obtain a feasible decomposition \( x \) of \( \rho' \). Recall that where \( x_S = \Pr[S_\tau = S] \) where

\[
S_\tau = \{ e \in E : \alpha_e \leq \tau < \alpha_e + \rho_e \}
\]

for \( \tau \sim U[0, 1] \) and

\[
\alpha_e := \min \{ \sum_{f \in (Q, e)} \mu_f + \rho_f' : Q \in \mathcal{P}', e \in Q \} \cup \{ 1 - \rho_e' \}.
\]

Note that by our assumption \( \rho'_e < \rho_e < 1 \) for all \( e \in E \) and moreover, \( \sum_{f \in (Q, e)} \mu_f' + \rho_f' \geq \mu_e' = \epsilon > 0 \) for all \( Q \in \mathcal{P}' \) and \( e \in E \). This implies that \( \delta := \min_{e \in E} \alpha_e > 0 \). Because moreover \( \rho'_e = 0 \), we conclude that \( S_\tau = \emptyset \) for all \( \tau < \delta \) and hence \( x_\emptyset = \Pr[S_\tau = \emptyset] \geq \delta > 0 \).

It remains to show that \( x \) is also a feasible decomposition for \( \rho^* \). Because \( \rho'_e = 0 \) and \( \rho_e = \rho_e^* \) for all \( e \in E \), it is easy to see that \( x_\emptyset > 0 \) implies \( S \subseteq E \) and moreover, \( x \) fulfills (1), (3), and (4). Let \( P \in \mathcal{P} \) and \( P' = \{ s \} \cup P \in \mathcal{P}' \). Let \( s' \in P \cap E_s \) be \( \leq_P \)-maximal. Let \( Q \in \mathcal{P} \) with \( s' = s_Q \). Let \( R := Q \times_{s'} P \). Note that \( R \subseteq [s', P] \) and \( R \cap E_s = \{ s' \} \) by construction. Let \( R' := \{ s \} \cup R \in \mathcal{P}' \). We conclude that

\[
\sum_{S \subseteq E : S \cap P \neq \emptyset} x_S \geq \sum_{S \subseteq E : S \cap R \neq \emptyset} x_S \geq \pi_{P'} = \pi_R \geq \pi_P,
\]

implying that \( x \) also fulfills (2).

\[\square\]

D Missing Details from Section 6

D.1 Proof of Theorem 8

Theorem 8 (\( \blacklozenge \)). Let \( D = (V, A) \) be a directed acyclic graph, let \( s, t \in V \), and let \( \mathcal{P} \subseteq 2^{V \cup A} \) be the set of \( s \)-\( t \)-paths in \( D \). Let \( \pi \in [0, 1]^\mathcal{P} \) such that (C) is fulfilled. Then there exists \( \mu \in [0, 1]^{V \cup A} \) such that \( \pi_P = 1 - \sum_{e \in P} \mu_e \). Furthermore, \( \mu \) can be computed in strongly polynomial time in \(|V| \) and \(|A| \) when \( \pi \) is given by an oracle that, given \( P \in \mathcal{P} \), returns \( \pi_P \).

Proof. Consider the system

\[
\begin{align*}
\sum_{e \in P} \mu_e &= 1 - \pi_P & \forall P \in \mathcal{P}, \\
\mu_e &\geq 0 & \forall e \in E := V \cup A,
\end{align*}
\]

and note that any feasible solution \( \mu \) to this system satisfies the conditions established in the theorem (as \( \mu_e \leq 1 \) for all \( e \in E \) is implied by (7) and (8)).

The proof of the theorem is split in two parts. In the first part we establish the existence of \( \mu \), i.e., feasibility of the system (7) and (8). In the second part, we show how this \( \mu \) can be efficiently computed.
Existence of μ For any vector \( y \in \mathbb{R}^P \), define

\[
\bar{\pi}(y) := \sum_{P \in \mathcal{P}} (1 - \pi_P) y_P \quad \text{and} \quad y_e := \sum_{P \in \mathcal{P}, e \in P} y_P \text{ for } e \in E.
\]

By Farkas’ lemma, system (7) and (8) has a feasible solution if and only if

\[
\bar{\pi}(y) \geq 0 \text{ for all } y \in \mathbb{R}^P \text{ with } y_e \geq 0 \text{ for all } e \in E.
\]

To show that (9) is fulfilled, we will prove the following auxiliary lemma.

**Lemma 11.** Let \( z, z' \in \mathbb{R}^P_+ \) with \( z_e \geq z'_e \) for all \( e \in E \). Then \( \bar{\pi}(z) \geq \bar{\pi}(z') \).

Note that Lemma 11 immediately implies (9): Given \( y \in \mathbb{R}^P \) with \( y_e \geq 0 \) for all \( e \in E \), define \( z, z' \in \mathbb{R}^P_+ \) by \( z_P := \max\{y_P, 0\} \) and \( z'_P := \max\{-y_P, 0\} \). Note that \( z_e - z'_e = y_e \geq 0 \) for all \( e \in E \) and hence Lemma 11 implies \( \bar{\pi}(y) = \bar{\pi}(z) - \bar{\pi}(z') \geq 0 \), proving (9). It thus only remains to prove Lemma 11.

**Proof of Lemma 11.** Let \( z' \in \mathbb{R}^P_+ \). Let

\[
X_{z'} := \{ z \in \mathbb{R}^P_+ : z_e \geq z'_e \text{ for all } e \in E \}.
\]

We show that \( \min_{z \in X_{z'}} \bar{\pi}(z) \geq \bar{\pi}(z') \), which proves the lemma.

We show this by induction on the cardinality of

\[
supp(z') := \{ P \in \mathcal{P} : z'_P > 0 \}.
\]

To start the induction, note that if \( supp(z') = \emptyset \), then \( \bar{\pi}(z') = 0 \leq \bar{\pi}(z) \) for all \( z \in \mathbb{R}^P_+ \), because \( 1 - \pi_P \geq 0 \) for all \( P \in \mathcal{P} \).

To complete the induction fix an arbitrary \( P \in \mathcal{P} \) with \( z'_P > 0 \). For \( a \in P \cap A \) and \( z \in X_{z'} \), let

\[
Q(a) := \{ Q \in \mathcal{P} : a \in Q \text{ and } [Q, a] = [P, a] \} \quad \text{and} \quad q_z(a) := \sum_{Q \in Q(a)} z_P.
\]

Now choose \( z \) so that it maximizes \( \sum_{a \in P \cap A} q_z(a) \) among all \( z \in \text{argmin}_{z \in X_{z'}} \bar{\pi}(z) \).

We will show that \( z_P \geq z'_P \). By contradiction assume that this is not the case. Let \( a = (u, v) \in P \cap A \) be the last arc on \( P \) such that \( q_z(a) \geq z'_P \), and let \( a' = (v, w) \in P \cap A \) be the first arc on \( P \) such that \( q_z(a) < z'(P') \); note that such a and \( a' \) exist because the first arc of \( P \) fulfills \( q_z(a) \geq z'_P \) but the last arc of \( P \) does not, as \( z_P < z'_P \).

Let \( Q \in Q(a) \) with \( a' \notin Q \) and \( z_Q > 0 \); note that such a \( Q \) exists because \( q_z(a') < z'_P \leq q_z(a) \) by choice of \( a \) and \( a' \). Let \( R \in \mathcal{P} \setminus Q(a) \) with \( a' \in R \) and \( z_R > 0 \); note that such an \( R \) exists because \( z_a \geq z'_P > q_z(a) \).

Note that \( v \in Q \cap R \) and let \( \bar{Q} := Q \times_v R \) and \( \bar{R} := R \times_v Q \). Let \( \varepsilon := \min\{z_Q, z_R\} > 0 \) and define \( \bar{z} \) by

\[
\bar{z}(\bar{P}) := \begin{cases} z_P + \varepsilon & \text{if } \bar{P} \in \{Q, \bar{R}\}, \\ z_P - \varepsilon & \text{if } \bar{P} \in \{Q, R\}, \\ z_P & \text{otherwise.} \end{cases}
\]
Note that $\bar{\pi}(z) = \bar{\pi}(z)$ by choice of $\varepsilon$. Furthermore, $\bar{\pi}(z) = z_e$ for all $e \in E$, because the graph is acyclic and hence $Q = [\bar{Q}, v] \cup [v, \bar{R}]$ and $R = [R, v] \cup [v, Q]$. Finally, observe that $\bar{\pi}(z) = \bar{\pi}(z)$ by (C). These three observations imply that $\bar{\pi}$ is contained in $X_\varepsilon$ and it is a minimizer of $\bar{\pi}$. Moreover, $q_\varepsilon(\bar{a}) \geq q_\varepsilon(\bar{a})$ for all $\bar{a} \in P \cap A$ because $Q \in Q(\bar{a})$ implies $Q \in Q(\bar{a})$ and $R \in Q(\bar{a})$ implies $R \in Q(\bar{a})$, respectively. Furthermore, $Q \in Q(a')$ by construction. Because $Q, R \notin Q(a')$, this implies $q_\varepsilon(a') > q_\varepsilon(a')$. We conclude that $\sum_{a \in P \cap A} q_\varepsilon(a) > \sum_{a \in P \cap A} q_\varepsilon(a)$, contradicting our choice of $z$. We have thus shown that $z_P > z_P'$.

Now construct $\hat{z}, \hat{z}'$ by $\hat{z} := z_P - z_P' > 0$, $\hat{z}_P := 0$, and $\hat{z}_P := z_P$, $\hat{z}'_P := z'_P$, for $P \in \mathcal{P} \setminus \{P\}$. Note that $\hat{z}_e \geq \hat{z}'_e$ for all $e \in E$ and supp($\hat{z}'$) = supp($\hat{z}'$) \{P\} by construction. Hence

$$\bar{\pi}(z) - (1 - \pi(P))z_P = \bar{\pi}(z) \geq \bar{\pi}(z') = \bar{\pi}(z') - (1 - \pi(P))z'_P$$

by induction hypothesis. We conclude that $\bar{\pi}(z) \geq \bar{\pi}(z')$, which completes the induction and proves the lemma.

**Efficient computation of $\mu$** We now show how to compute such a solution $\mu$ in polynomial time when $\pi$ is given implicitly by an oracle that given $P$ returns $\pi_P$.

Note that to do this, it suffices to determine a basis of the rows of the system (7), as the set of solutions does not change when restricting (7) to this basis. The following lemma shows how to obtain such a basis efficiently. A solution fulfilling both (7) and (8) can then be determined in strongly polynomial time, e.g., using the algorithm of Tardos [19] for linear programs with binary constraint matrices.

**Lemma 12.** There is an algorithm that, given a directed acyclic graph $D = (V, A)$ and two nodes $s, t \in V$, computes in strongly polynomial time a set $\mathcal{P}$ of $s$-$t$-paths in $D$ such that the incidence vectors of the paths in $\mathcal{P}$ are a basis of the subspace of $\mathbb{R}^{V \cup A}$ spanned by the incidence vectors of all $s$-$t$-paths in $D$.

**Proof.** Let $\hat{V} \subseteq V$ be the set of nodes such that there is both an $s$-$v$-path and a $v$-$t$-path in $D$, and let $A := A[\hat{V}]$ be the set of arcs with both endpoints in $\hat{V}$. Note that the set of $s$-$t$-paths in $D$ is the same as the subgraph $\hat{D} = (\hat{V}, A)$.

Let $T \subseteq A$ be an arborescence rooted at $s$ and spanning $\hat{V}$ (such an arborescence exists by construction of $\hat{V}$). For $v \in \hat{V}$, let $T[v]$ denote the unique $s$-$v$-path in $T$. Let $B := A \setminus T$. For $b = (v, w) \in B$, let $P_b$ be an $s$-$t$-path resulting from the concatenation of $T[v] \cup b$ with an arbitrary $w$-$t$-path in $\hat{D}$ (note that such a concatenation indeed results in a simple path as the graph is acyclic). Define $P_0 := T[t]$ and $\mathcal{P} := \{P_0\} \cup \{P_b : b \in B\}$. We claim $\mathcal{P}$ is indeed the desired set of paths.

For any $P \in \mathcal{P}$, let $\chi_P \in \mathbb{R}^A$ denote the arc-incidence vector of $P$. Let

$$U := \text{span}\{\chi_P : P \in \mathcal{P}\}$$

and $C := \text{span}\{u \in \mathbb{R}^A : u \text{ is a circulation in } D\}$.

Note that $U \subseteq \{\chi_{P_0} + u : u \in C\}$ because $u_P := \chi_P - \chi_{P_0}$ is a circulation for any $P \in \mathcal{P}$. It is well-known that $\dim C = |A| - (|V| - 1) = |B|$. Hence, $\dim U \leq \dim U + 1 \leq |B| + 1$. Moreover, because the graph is acyclic, there is
an ordering \( b_1, \ldots, b_k \) of \( B \) such that \( P_b \cap \{ b_1, \ldots, b_{i-1} \} = \emptyset \) for \( i \in \{ 1, \ldots, k \} \). Because also \( P_b \cap B = \emptyset \), the incidence vectors of the paths in \( \mathcal{P} \) are linearly independent. As \( |\mathcal{P}| = |\mathcal{B}| + 1 = \dim U \), we conclude that \( \mathcal{P} \) is indeed a basis of \( U \). Finally, observe that extending the arc-incidence vectors by node incidences does not change the dimension of \( U \), as for every path \( P \) these entries are fully determined by the arcs \( P \).

This completes our proof of Theorem 8. □

D.2 From Poset to DAG (Hasse diagram)

In this section, we discuss the case where \( \mathcal{P} \) is the set of maximal chains of a partially ordered set \((E, \mathcal{P})\) and \( \pi \in [0, 1]^E \) fulfills the conservation law (C). Note that Theorem 8 requires \( \mathcal{P} \) to the the set of \( s\)-\( t \)-paths in a directed acyclic graph (where each path corresponds to the set of its arcs and nodes), which is a special case of maximal chains in a poset (in that each maximal chain in a DAG consists of an alternating sequence of nodes and arcs).

We first observe that Theorem 8 does not apply directly to the case where \( \mathcal{P} \) is the set of maximal chains in an arbitrary poset. Indeed, consider the poset \( E = \{ a_0, a_1, z_0, z_1 \} \) with \( a_i \prec z_j \) for all \( i, j \in \{ 0, 1 \} \) and requirements defined by \( \pi_{\{a_0, z_0\}} = 0 \) and \( \pi_{\{a_0, z_1\}} = \pi_{\{a_1, z_0\}} = \pi_{\{a_1, z_1\}} = 1. \) Note that \( \pi \) fulfills (C),\(^5\) but is not of the form \( 1 - \sum_{e \in P} \mu_e \) for any \( \mu \in [0, 1]^E \) because \( \pi_{\{a_0, z_1\}} = \pi_{\{a_1, z_0\}} = \pi_{\{a_1, z_1\}} = 1 \) implies \( \mu_{a_0} = \mu_{a_1} = \mu_{z_0} = \mu_{z_1} = 0, \) contradicting \( \pi_{\{a_0, z_0\}} = 0. \)

However, we can still make use of Theorem 8 when given an arbitrary poset, by using its Hasse diagram representation. The Hasse diagram of a poset \((E, \preceq)\) is the digraph \( D = (V, A) \) with node set \( V = E \) and arc set \( A = \{(e, f) : e, f \in E, e \prec f \text{ and there is no } f' \in E : e \prec f' \prec f \}. \) Note that \( D \) is acyclic. Furthermore, if we assume w.l.o.g. that \( \preceq \) has a unique minimal element \( s \in E \) and a unique maximal element \( t \in E \), then each \( s\)-\( t \)-path \( P \) in \( D \) corresponds to the maximal chain \( C = P \cap V \) of \( \preceq \) and vice versa. We can thus interpret \( \pi \) also as a requirement function on the paths of \( D \) and note that (C) is not affected by this transformation.

Applying Theorem 8 on \( D \) and \( \pi \) then lets us represent \( \pi \) by weights \( \mu \in [0, 1]^{V \cup A}. \)\(^6\) Given any marginals \( \rho \in [0, 1]^E \) in the poset fulfilling (\( \ast \)), we can thus find a feasible decomposition by applying Theorem 3 on the corresponding marginals in \( D \) (which are simply extending \( \rho \) by \( \rho_a = 0 \) for all \( a \in A \)). Note that the corresponding decomposition only uses subsets of \( V = E \), and hence is also a feasible decomposition for \( \rho \) in \( E \).

---

\(^5\) In fact, any requirement vector fulfills (C) for this poset.

\(^6\) Note that these weights \( \mu \) are also defined for the arcs of \( D \), hence giving \( |A| \) additional degrees of freedom that would not present when trying to describe \( \pi \) by weights on the elements of the poset alone. Our earlier example shows that these degrees of freedom are indeed necessary.
E Missing Details from Section 7 for Other Systems

The results from Sections 2 and 3 lead to the question whether (⋆) is a sufficient condition for the feasibility of marginals in other types of systems beyond abstract networks, and whether we can compute feasible decompositions in these systems.

E.1 Sufficiency of Condition (⋆)

Dahan et al. [4] showed that (⋆) is in general not a sufficient condition for the feasibility. In particular, they provide an example [4, Example 2] of a partially ordered set with a requirement vector \( \pi \) not fulfilling their conservation law (C) and marginals \( \rho \) such that (⋆) is fulfilled, but were no feasible decomposition of \( \rho \) for \( \pi \) exists.

Here we observe that (⋆) is also not sufficient for the feasibility of marginals, even when \( \pi \equiv 1 \), for many fundamental types of set systems. To verify these counterexamples, it will be useful to consider following LP:

\[
\text{[D]} \quad \max \sum_{e \in E} \rho_e y_e + \sum_{P \in \mathcal{P}} \pi_P z_P \\
\text{s.t.} \quad \sum_{e \in S} y_e + \sum_{P \in \mathcal{P} : P \cap S \neq \emptyset} z_P \leq 1 \quad \forall S \subseteq E \\
\quad \quad z \geq 0
\]

Recall that feasible decompositions of marginals \( \rho \) correspond exactly to feasible solutions \( x \) to the system (1) to (4). In particular, a feasible decomposition induces a solution of value 1 to the LP

\[
\text{[P]} \quad \min \sum_{S \subseteq E} x_S \quad \text{s.t. } x \text{ fulfils (1), (2), and (4)}.
\]

Note that [D] is the dual to [P] and hence a solution to [D] of value at least 1 implies that marginals \( \rho \) have no feasible decomposition for \((E, \mathcal{P})\) and \( \pi \).

Spanning Trees\(^7\) Consider the graph \( G = (V, E) \) given by

\[ V = \{v_1, v_2, v_3\} \] and \( E = \{e_1 = \{v_1, v_2\}, e_2 = \{v_2, v_3\}, e_3 = \{v_1, v_3\}\} \).

Let \( \mathcal{P} \subseteq 2^E \) be the set of spanning trees in \( G \), i.e.,

\[ \mathcal{P} = \{\{e_1, e_2\}, \{e_1, e_3\}, \{e_2, e_3\}\} . \]

Let \( \pi_P = 1 \) for every \( P \in \mathcal{P} \) and \( \rho_e = 0.5 \) for \( e \in E \). Note that \( \rho \) and \( \pi \) fulfil condition (⋆). However, \( y_e = -1 \) for \( e \in E \) and \( z_P = 1 \) for \( P \in \mathcal{P} \) defines a feasible solution to [D] of value 1.5.

Note that this example also establishes that (⋆) is not a sufficient condition when \( \mathcal{P} \) is the set of bases of a matroid and \( \pi \equiv 1 \).

\(^7\) The same example also appeared earlier in a preliminary version of Dahan et al.’s work to motivate the restriction to partially ordered sets [3, Section 2.2].
Perfect Matchings in a Bipartite Graph Consider the bipartite graph \( G = (V, E) \) depicted in Fig. 1 (a). Let \( \mathcal{P} \subseteq 2^E \) be the set of perfect matchings in \( G \). Let \( \pi_P = 1 \) for every \( P \in \mathcal{P} \) and let \( \rho_e = 0.5 \) for \( e = \{v_i, w_i\} \) for \( i \in \{1, 2, 3\} \) and \( \rho_e = 0 \) otherwise. Note that there are exactly three perfect matchings in \( G \) and that each of these matchings contains exactly two of the three edges \( \{v_1, w_1\}, \{v_2, w_2\}, \{v_3, w_3\} \), which are the only edge with positive marginals. In particular, this implies that \( \rho \) and \( \pi \) fulfil condition \((*)\). However,

\[
y_e = \begin{cases} 
-1 & \text{for } e \in E \text{ with } \rho_e > 0 \\
-3 & \text{for all } e \in E \text{ with } \rho_e = 0,
\end{cases}
\]

together with \( z_P = 1 \) for \( P \in \mathcal{P} \) defines a feasible solution to \([D]\) of value \(-1.5 + |\mathcal{P}| = 1.5\). Indeed, note that, because \( |\mathcal{P}| = 3 \), the constraints for \([D]\) for \( S \subseteq E \) are trivially fulfilled if \( S = \emptyset \), or if \( S \) contains an \( e \) with \( y_e = -3 \), or if \( S \) contains at least two elements \( e, f \) with \( y_e = y_f = -1 \). The only remaining case is that \( S = \{v_i, w_i\} \) for some \( i \in \{1, 2, 3\} \). But then \( S \) only intersects with two matchings, and again the constraint is fulfilled. Hence the marginals are not feasible.

\( s_1-t_1 \)-paths in a DAG Consider the digraph \( D = (V, A) \) depicted in Fig. 1 (b), with marginals \( \rho \) on \( E := V \cup A \) as depicted. Let \( \mathcal{P} \subseteq 2^E \) be the set of \( s_1-t_1 \)-paths and \( s_2-t_2 \)-paths in \( D \). Note that there are exactly three such paths. Each of these three paths contains exactly two elements with positive marginals and each element with positive marginals is contained in exactly two paths of \( \mathcal{P} \). Hence \( \rho \) and \( \pi \) fulfil condition \((*)\), and by the same argument as in the example for perfect matchings, the solution defined by

\[
y_e = \begin{cases} 
-1 & \text{for } e \in E \text{ with } \rho_e > 0 \\
-3 & \text{for all } e \in E \text{ with } \rho_e = 0,
\end{cases}
\]

together with \( z_P = 1 \) for \( P \in \mathcal{P} \) yields a feasible solution to \([D]\) of value \(-1.5 + |\mathcal{P}| = 1.5\). Hence the marginals are not feasible.
E.2 Approximately Feasible Marginals

Given the non-existence result mentioned above, one may be interested in finding decompositions that satisfy the requirements at least approximately. We say a decomposition $x$ of marginals $\rho$ is $\beta$-approximately feasible, for $\beta \in [0, 1]$, if it fulfills (1), (3), (4), and

$$\sum_{S \subseteq E: S \cap P \neq \emptyset} x_S \geq \beta \cdot \pi_P \quad \forall P \in \mathcal{P}.$$ 

Indeed, for $\beta = 1 - 1/e$ such a decomposition exists for any set of marginals fulfilling (⋆), without imposing any further conditions on $(E, \mathcal{P})$ or $\pi$.

Lemma 13. Let $(E, \mathcal{P})$ be a set system, $\pi \in [0, 1]^\mathcal{P}$, and $\rho \in [0, 1]^E$ such that condition (⋆) is fulfilled, i.e., $\sum_{e \in P} \rho_e \geq \pi_P$ for all $P \in \mathcal{P}$. Let $S_\rho \subseteq E$ be a random set including each $e \in E$ independently with probability $\rho_e$ and let $x_S = \Pr[S_\rho = S]$. Then $x$ is $(1 - 1/e)$-approximately feasible decomposition of $\rho$ for $(E, \mathcal{P})$ and $\pi$.

Proof. By construction, $x$ fulfills (1), (3), and (4). Let $P \in \mathcal{P}$. Using that $\sum_{e \in P} \rho_e \geq \pi_P$ and $\pi_P \in [0, 1]$, we observe that

$$\Pr[S_\rho \cap P \neq \emptyset] = 1 - \prod_{e \in P} \Pr[e \notin S_\rho] = 1 - \prod_{e \in P} (1 - \rho_e)$$

$$\geq 1 - \left(1 - \frac{\sum_{e \in P} \rho_e}{|P|}\right)^{|P|} \geq 1 - \left(1 - \frac{\pi_P}{|P|}\right)^{|P|}$$

$$\geq 1 - e^{-\pi_P} \geq (1 - e^{-1})\pi_P,$$

which proves the lemma. \qed

E.3 Computing Feasible Decompositions for a Given Instance

Finally, we consider the problem of deciding whether a given instance of marginals in a set system is feasible. We show that this problem is NP-hard, even when considering set systems $(E, \mathcal{P})$ where each set $P \in \mathcal{P}$ contains at most 3 elements of $E$ (note that $|\mathcal{P}| \leq |E|^3$ in such systems and hence the hardness holds even when we assume $\mathcal{P}$ to be given explicitly in the input rather than via an oracle).

Theorem 9 (♦). The following decision problem is NP-complete: Given a set system $(E, \mathcal{P})$ with $|P| = 3$ for all $P \in \mathcal{P}$ and marginals $\rho \in [0, 1]^E$, is there a feasible decomposition of $\rho$ for $(E, \mathcal{P})$ and requirement vector $\pi \equiv 1$?

Proof. To see that the problem is indeed in NP, note that a given instance is a yes instance if and only if the corresponding linear system (1) to (4) has a feasible solution. As the number of constraints (1) to (3) is polynomial in $|E|$ (because $|\mathcal{P}| \leq |E|^3$), extreme point solutions of (1) to (4) have at most a polynomial number of non-zeros. Such an extreme point solution is thus a polynomially sized and verifiable certificate for a yes instance.
We now show that the NP-hard via reduction from the NP-hard Not-all-equal 3-Satisfiability (NAE3SAT) problem [16]. In this problem, we are given a finite set of boolean variables \( Y \) and a finite set of clauses \( C \), with each clause consisting of the disjunction of three literals of the variables. We let \( L := \{ y, \neg y : y \in Y \} \) be the set of literals on \( Y \) and identify each clause \( C \in C \) with the set of its literals (e.g., \( C = \{ y_1, \neg y_3, y_5 \} \)). A truth assignment is a subset of \( A \subseteq L \) of the literals containing for each variable \( y \in Y \) either \( y \) or \( \neg y \). The task in NAE3SAT is to decide whether there is a truth assignment \( A \) such that \( |A \cap C| \in \{1, 2\} \) for each \( C \in C \), i.e., the truth assignment satisfies at least one, but not all literals of each clause.

For a truth assignment \( A \), let \( \bar{A} := Y \setminus A \) denote the corresponding complementary assignment. For a clause \( C \in C \), let

\[
\bar{C} := \{ \neg y : y \in C \cap Y \} \cup \{ y : \neg y \in C \}
\]
denote the corresponding complementary clause consisting of the negation of its literals.

**Observation 14.** For any truth assignment \( A \), the following statements are equivalent:

- \( |A \cap C| \in \{1, 2\} \) for every \( C \in C \).
- \( A \cap C \neq \emptyset \) and \( A \cap C \neq \emptyset \) for all \( C \in C \).
- \( A \cap C \neq \emptyset \) and \( A \cap C \neq \emptyset \) for all \( C \in C \).
- \( A \cap \bar{C} \neq \emptyset \) and \( A \cap \bar{C} \neq \emptyset \) for all \( C \in C \).

Given an instance of NAE3SAT, define the set system \( (E, P) \) with \( E := L \) and

\[
P := \{ C, \bar{C} : C \in C \} \cup \{ P_y : y \in Y \},
\]
where \( P_y := \{ y, \neg y \} \). Let \( \pi = 1 \) and \( \rho_e = 0.5 \) for all \( e \in E \). We show that there is a feasible decomposition of \( \rho \) for \( (E, P) \) and \( \pi \) if and only if there is a truth assignment \( A \) of the variables in \( Y \) with \( |A \cap C| \in \{1, 2\} \) for all \( C \in C \).

First assume there is a truth assignment \( A \) of the variables in \( Y \) with \( |A \cap C| \in \{1, 2\} \) for all \( C \in C \). Then define \( x \) by setting \( x_A = 0.5 \), \( x_{\bar{A}} = 0.5 \), and \( x_S = 0 \) for all \( S \in 2^E \setminus \{A, \bar{A}\} \). Note that \( x \) fulfils (3) and (C) by construction. Moreover, because \( A, \bar{A} \) is a partition of \( E = L \), note that \( x \) fulfils (1) and \( \sum_{S : S \cap \{y, \neg y\} \neq \emptyset} x_S = x_A + x_{\bar{A}} = 1 \) for all \( y \in Y \). Finally, because \( |A \cap C| \in \{1, 2\} \), note that \( A \cap C, A \cap \bar{C}, \bar{A} \cap C, \bar{A} \cap \bar{C} \) are all nonempty for every \( C \in C \) by Observation 14, and hence \( \sum_{S : S \cap C} x_S = x_A + x_{\bar{A}} = 1 \) and \( \sum_{S : S \cap \bar{C}} x_S = x_A + x_{\bar{A}} = 1 \) for all \( C \in C \). Thus \( x \) also fulfils (2) and we can conclude that \( x \) is a feasible decomposition of \( \rho \).

Now assume \( \rho \) has a feasible decomposition \( x \). Let \( A \subseteq E \) with \( x_S > 0 \). Note that

\[
1 = \pi_{P_y} \leq \sum_{S : S \cap P_y \neq \emptyset} x_S \leq \sum_{S : S \cap P_y \neq \emptyset} |S \cap P_y| x_S \leq \sum_{S \subseteq E} x_S \leq 1,
\]
for all \( y \in Y \), and hence \( |S \cap P_y| = 1 \) for all \( S \) with \( x_S > 0 \). Thus also \( |A \cap P_y| = 1 \) for all \( y \in Y \), i.e., \( A \) contains exactly one of the literals \( y, \neg y \) for every \( y \in Y \).
Y. Hence $A$ is a truth assignment. Similarly, note that $1 \leq \sum_{S \subseteq C \neq \emptyset} x_S \leq \sum_{S \subseteq E} x_S \leq 1$ and $1 \leq \sum_{S \subseteq \overline{C} \neq \emptyset} x_S \leq \sum_{S \subseteq E} x_S \leq 1$ for all $C \in \mathcal{C}$, and hence $x_A > 0$ implies $A \cap C \neq \emptyset \neq A \cap \overline{C}$. Hence, by Observation 14, $A$ is a truth assignment fulfilling $|A \cap C| \in \{1, 2\}$ for all $C \in \mathcal{C}$. \qed