Scarf's Algorithm on Arborescence Hypergraphs

Karthekeyan Chandrasekaran[*], Yuri Faenza[†], Chengyue He[‡], and Jay Sethuraman[§]

Abstract

Scarf's algorithm—a pivoting procedure that finds a dominating extreme point in a down-monotone polytope—can be used to show the existence of a fractional stable matching in hypergraphs. The problem of finding a fractional stable matching in a hypergraph, however, is PPAD-complete. In this work, we study the behavior of Scarf's algorithm on arborescence hypergraphs, the family of hypergraphs in which hyperedges correspond to the paths of an arborescence. For arborescence hypergraphs, we prove that Scarf's algorithm can be implemented to find an integral stable matching in polynomial time. En route to our result, we uncover novel structural properties of bases and pivots for the more general family of network hypergraphs. Our work provides the first proof of polynomial-time convergence of Scarf's algorithm on hypergraphic stable matching problems, giving hope to the possibility of polynomial-time convergence of Scarf's algorithm for other families of polytopes.

1 Introduction

Scarf [21] proved the existence of a core allocation in a large class of cooperative games with non-transferable utility. Key ingredients in this proof include a lemma—*Scarf's lemma*—that asserts the existence of a dominating extreme point in certain polytopes, and a pivoting procedure—*Scarf's algorithm*—to find one. Scarf's results have profoundly influenced subsequent research in combinatorics, theoretical computer science, and game theory: Scarf's lemma has been used to show the existence of fair allocations such as cores and fractional cores [5, 21], strong fractional kernels [2], fractional stable solutions in hypergraphs [1], and stable paths [13]. Yet, it is not known how to efficiently construct these desirable allocations/solutions: It is widely believed that Scarf's algorithm is unlikely to be efficient for arbitrary applications. In this work, we investigate the possibility of efficient convergence of Scarf's algorithm for finding a stable matching in hypergraphs.

The stable matching problem in bipartite graphs (also known as the stable marriage problem) is a classic and well-studied problem. It is well-known that a stable matching in a bipartite graph with preferences always exists [12] and can be found in polynomial time via multiple algorithms, including Gale and Shapley's deferred acceptance algorithm [12], linear programming [3, 19, 20, 23] and other combinatorial algorithms [4, 8]. Recently, it was shown that Scarf's algorithm can be implemented to converge in polynomial time for the stable marriage problem [9]. This result motivated us to address the possibility of efficient convergence of Scarf's algorithm for hypergraph stable matching problems.

In contrast to graphs, the problem of finding a stable matching in hypergraphs is significantly more challenging. Even in the special case of tripartite 3-regular hypergraphs, it is NP-complete to

[*]University of Illinois, Urbana-Champaign, karthe@illinois.edu
[†]Columbia University, yuri.faenza@columbia.edu
[‡]Columbia University, ch3480@columbia.edu
[§]Columbia University, jay@ieor.columbia.edu

find a stable matching [17] (this is also known as the *stable family problem* proposed by Knuth [16]). Despite the hardness results of finding (integral) stable matchings in hypergraphs, Aharoni and Fleiner [1] used Scarf's lemma to show that a fractional stable matching exists in every hypergraph. However, this general implication from Scarf's result comes at a computational price: the problem of computing a fractional stable matching on hypergraphs is PPAD-complete [15]. The latter problem remains PPAD-complete even in extremely restrictive cases, for example, even if the hypergraph has node degree [14] or hyperedge size [7] upper bounded by 3.

In this paper, we define a new class of hypergraphs called *arborescence* hypergraphs (in which the hyperedges are paths in an arborescence), and design a polynomial-time algorithm to find an integer stable matching. We accomplish this through a pivoting rule that implements Scarf's algorithm and terminates in a polynomial number of iterations. The node-hyperedge incidence matrix of an arborescence hypergraph is totally unimodular and consequently, every extreme point of the associated fractional matching polytope is integral. Thus, every dominating extreme point is also integral. This observation and Aharoni and Fleiner's results imply that the extreme point found by Scarf's algorithm for arborescence hypergraphs corresponds to an *integral* stable matching. Our pivoting rule can be repurposed to apply to the more general family of *network hypergraphs*[1] but we are not able to prove polynomial convergence for this more general family.

Our Contributions. A hypergraph $H = (V, E)$ is specified by a vertex set V and a hyperedge set E, where every hyperedge $e \in E$ is a subset of V. A *hypergraphic preference system* is given by a pair $(H, \succ)$, where $H = (V, E)$ is a hypergraph and $\succ := \{\succ_i : i \in V\}$ is the preference profile, $\succ_i$ being a strict order over $\delta(i) = \{e \in E : i \in e\}$ for each $i \in V$. For $e, e' \in \delta(i)$, we write $e \succeq_i e'$ if either $e \succ_i e'$ or $e = e'$. In addition, we assume that for every $i \in V$, the *singleton hyperedge* $e_i = \{i\} \in \delta(i)$ and for every $e' \in \delta(i)$, $e' \succeq_i e_i$[2].

A *stable matching* for a hypergraphic preference system is a vector $x \in \{0,1\}^E$ so that for every $e \in E$, there exists a vertex $i \in e$ such that

$$\sum_{e' \in \delta(i), e' \succeq_i e} x_{e'} = 1. \tag{1}$$

Equation (1) with $e = e_i$ imposes that x is the characteristic vector of a matching. Moreover, for every hyperedge e, there exists a vertex $i \in e$ and a matching edge e' so that $e' \succeq_i e$. Correspondingly, if a fractional vector $x \in [0,1]^E$ still satisfies (1), then we call such a fractional vector x a *fractional stable matching*.

An *arborescence* is a directed graph that contains a vertex r (called the *root*) such that every vertex $v \neq r$ has a unique directed path from r. A hypergraph $H = (V, E)$ is an *arborescence hypergraph* if there exists an arborescence $\mathcal{T} = (U, \mathcal{A}_0)$ such that $V = \mathcal{A}_0$ and each hyperedge $e \in E$ is a subset of arcs in $\mathcal{A}_0$ that forms a directed path. We note that there exists a polynomial-time algorithm to verify whether a given hypergraph is an arborescence hypergraph and moreover, find an associated arborescence $\mathcal{T}$ [22].

As our main result, we show that Scarf's algorithm can be implemented to run in polynomial time for every arborescence hypergraphic preference system.

Theorem 1. *Let $(H = (V, E), \succ)$ be a hypergraphic preference system where H is an arborescence hypergraph. There exists a pivoting rule such that Scarf's algorithm terminates in at most $|V|$ iterations and outputs a stable matching on $(H = (V, E), \succ)$ in time $O(|V||E|)$.*

[1] The node-hyperedge incidence matrix of such a hypergraph is totally unimodular and consequently there is always an integer stable matching.

[2] This assumption corresponds, in the bipartite setting, to the usual hypothesis that an agent prefers to be matched rather than being unmatched.

A few remarks on this result are in order. To the best of our knowledge, this is the first result showing polynomiality of Scarf's algorithm beyond the case of stable marriage [9]. We note that the pivoting rule used to show polynomiality of Scarf's algorithm in stable marriage [9] was heavily inspired by Gale and Shapley's classical deferred acceptance mechanism, which is a purely combinatorial algorithm. In contrast, the pivoting rule developed in this work does not seem to have a combinatorial counterpart and is substantively different from the one from [9]. In Section 4, we provide evidence suggesting that classical approaches to stable marriage problems, such as a natural linear program with an exact description of the convex hull of all stable matchings [23], do not extend to even a subclass of arborescence hypergraphic preference systems.

Secondly, while most of the known results related to polynomiality of stable matching on hypergraphic preference system usually restricts either the degree of nodes [14] or the size of hyperedges [7], we do not assume any condition on them or on the preference lists of agents. Thirdly, previous work by [7] shows that there is a polynomial time algorithm for finding a stable matching on arborescence hypergraphic systems. They reduce the problem of finding a stable matching on such instances to finding a *kernel* in the clique-acyclic superorientation of a chordal graph, which can be solved in polynomial time [18]. We remark that the algorithm from [18] is different from Scarf's algorithm and does not seem to generalize to network hypergraphs, while, in our investigation, we uncover novel properties of bases and pivots in network hypergraphs, which may be of independent interest and could lead to polynomial-time convergence proofs for this more general class of hypergraphs.

We recall standard concepts on Scarf's algorithm, implication of Scarf's lemma to the stable matching problem in hypergraphs, and formally define arborescence hypergraphs in Section 2. We prove Theorem 1 in Section 3. In Section 4, we show that a natural relaxation of the convex hull of stable matchings in an arborescence hypergraph is fractional.

2 Preliminaries

We recall some standard concepts in this section as preliminaries. In Section 2.1, we present Scarf's lemma. In Section 2.2, we explain its implications to the stable matching problem in hypergraphs. In Section 2.3, we give details about Scarf's algorithm—in particular we describe cardinal pivot and ordinal pivot operations and how they are combined to show the existence of a dominating basis. In Section 2.4, we define network matrices and network hypergraphs. In Section 2.5, we define arborescence hypergraphs, the special case of interest to this work.

Throughout the paper, we use the following standard notations: For $n \in \mathbb{Z}_+$, we let $[n] := \{1, 2, \dots, n\}$. For sets S, S' with $|S \setminus S'| = 1$, we abuse notation and let $S \setminus S'$ be the unique element $e \in S \setminus S'$. For a matrix $A \in \mathbb{R}^{n \times m}$, $i \in [n], j \in [m]$, we let $a_{i,j}$ be the entry in the i-th row and j-th column of A. We denote the vector corresponding to j-th column by A_j. For $B \subseteq [m]$, we denote the submatrix of A corresponding to the columns in B as A_B.

2.1 Scarf's Lemma

Scarf's lemma deals with the standard form of down-monotone polytopes, as defined below.

Definition 2 (Standard form, cardinal basis, extreme point)**.** *Let $n \leq m$ be positive integers. Let $A \in \mathbb{R}^{n \times m}_{\geq 0}$ be a nonnegative matrix. Matrix A is in* standard form *if A has the form $A = (I_n | A')$ where I_n is the $n \times n$ identity matrix. Let $b \in \mathbb{R}^n_+$ be strictly positive. Consider the polytope*

$$P = \{x \in \mathbb{R}^m_{\geq 0} : Ax = b\}. \tag{2}$$

A set $B \subseteq [m]$ is a cardinal basis *for (A, b) if $|B| = n$ and the submatrix A_B of A indexed by the columns in B has full row rank. In addition, let B be a cardinal basis and $x = (x_B, 0) \in \mathbb{R}^m$ be such that $\bar{x} = x_B$ is the unique solution to the linear system $A_B \bar{x} = b$. Then, if $x \in P$, we call B a* feasible cardinal basis*, and the solution x is an* extreme point *of P corresponding to B.*

In addition to the polytope description, Scarf's lemma also takes as input a matrix C. Since no entry appears twice in any row of C, this matrix can be interpreted as a row-wise ordering of columns.

Definition 3 (Ordinal matrix, ordinal basis, utility vector)**.** *Let $n \leq m$ be positive integers. Let $C = (c_{i,j}) \in \mathbb{R}^{n \times m}$ be a matrix. Matrix C is an* ordinal matrix *if (i) for every $i \in [n]$ and $j \neq k \in [m]$, $c_{i,j} \neq c_{i,k}$, and (ii) for every distinct $i, j \in [n]$, $k \in [m] \setminus [n]$, $c_{i,i} < c_{i,k} < c_{i,j}$. Let O be a set of columns of C. For every row $i \in [n]$, define the utility of the row (w.r.t. O) as*

$$u_i^O := \min_{j \in O} c_{i,j}. \tag{3}$$

The set O is called an ordinal basis *of C if $|O| = n$ and for every column $j \in [m]$, there is at least one row $i \in [n]$ such that $u_i^O \geq c_{i,j}$. The associated vector $u^O \in \mathbb{R}^n$ is called the* utility vector *of the ordinal basis O.*

For two real vectors x, y in the same dimension, we succinctly write $x > y$ if $x_i > y_i$ for every entry i, and write $x \not> y$ if $x > y$ fails. We remark that, for every set O of n columns, we have that $C_j \geq u^O$ for every $j \in O$ by definition. Hence, a set O of n columns is an ordinal basis iff for every column $j \in [m]$, the column vector C_j satisfies $C_j \not> u^O$.

Definition 4 (Dominating basis)**.** *A feasible cardinal basis for (A, b) that is also an ordinal basis of C is called a* dominating basis *for (A, b, C). Given a dominating basis B for (A, b, C), the extreme point x of $P = \{x \in \mathbb{R}^m_{\geq 0} : Ax = b\}$ corresponding to B is called a* dominating extreme point *for (A, b, C).*

Scarf's lemma, presented next, shows the existence of a dominating extreme point under mild conditions on (A, b, C).

Theorem 5 (Scarf's Lemma)**.** *Let $A \in \mathbb{R}^{n \times m}_{\geq 0}$ be a matrix in standard form and $b \in \mathbb{R}^n_+$ such that the $P = \{x \in \mathbb{R}^m_{\geq 0} : Ax = b\}$ is bounded. Let $C \in \mathbb{R}^{n \times m}$ be an ordinal matrix. Then, there exists a dominating extreme point for (A, b, C).*

2.2 Implication of Scarf's Lemma for Fractional Hypergraph Stable Matching

Aharoni and Fleiner [1] used Scarf's Lemma to establish the existence of a fractional stable matching in hypergraphs. In this section, we briefly explain their approach via a special case.

Definition 6. *Let $(H = (V, E), \succ)$ be a hypergraphic preference system with $e_i := \{i\}$ for $i \in [n]$ being the singleton hyperedges. We say that matrices A, C are* block-partitioned *if they are constructed as follows (see Example 8 for an illustration):*

1. *For each $i \in V$, we define the i-th block $S_i := \{e \in E \setminus \{e_i\} : i = \max_{j \in e} j\}$.*

2. *Let A be the $V \times E$ incidence matrix of H whose columns are ordered as follows: The columns corresponding to the singleton hyperedges are the first n columns of A. Next, we group the remaining hyperedges into n blocks $S_1, \ldots, S_n$. The i-th block consists of S_i and is ordered in decreasing order of preference from left to right according to $\succ_i$. The i-th block appears before the $i+1$-th block for every $i \in [n-1]$.*

3. *The ordinal matrix C has the same number of rows and columns as A, and each column index $j \in [m]$ corresponds to the same hyperedge e_j in both A and C. We now describe the entries in C. For each $i \in [n], j \in [m]$ with $a_{i,j} = 1$, we set $c_{i,j} = |\delta(i)| - \ell$ if e_j is the ℓ-th best hyperedge with respect to $\succ_i$. The remaining entries in row i are assigned integers that are no less than $|\delta(i)|$ and in decreasing order from left to right.*

Let $(H = (V, E), \succ)$ be a hypergraphic preference system. Let $S_1, \ldots, S_n$ be the blocks and A, C be the block-partitioned matrices associated with this hypergraphic preference system. We observe that S_1 is empty since the only hyperedge e such that $\max_{j \in e} j = 1$ is $e = \{1\}$, which is a singleton. Aharoni and Fleiner [1] showed that every dominating extreme point for (A, b, C) is a fractional stable matching of the hypergraphic preference system $(H = (V, E), \succ)$.

Theorem 7 ([1])**.** *Let $(H = (V, E), \succ)$ be a hypergraphic preference system. Let A and C be the block-partitioned matrices associated with this hypergraphic preference system and let $b \in 1^V$ be the all ones vector. Then, the matrix A is in standard form, the vector b is a positive vector, the polytope $P = \{x \in \mathbb{R}^E_{\geq 0} : Ax = b\}$ is bounded, and the matrix C is ordinal. Moreover, every dominating extreme point for (A, b, C) is a fractional stable matching of $(H, \succ)$.*

We note that not every fractional stable matching is a dominating extreme point for (A, b, C) (see [9]). By Theorem 5 and Theorem 7, every hypergraphic preference system admits a fractional stable matching.

Example 8. *For $V = \{1, 2, 3, 4\}$, consider the hypergraph in Figure 1 and the following preference list:*

$$\begin{aligned}
&1 : \{1,3\} \succ_1 \{1,3,4\} \succ_1 \{1\},\\
&2 : \{2,3\} \succ_2 \{2\},\\
&3 : \{2,3\} \succ_3 \{1,3\} \succ_3 \{3,4\} \succ_3 \{1,3,4\} \succ_3 \{3\},\\
&4 : \{1,3,4\} \succ_4 \{3,4\} \succ_4 \{4\}.
\end{aligned}$$

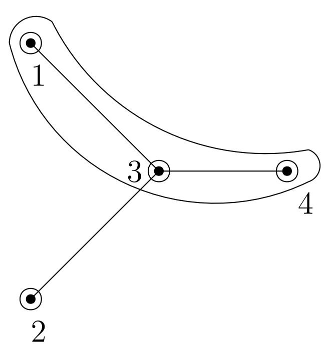

Figure 1: An example of a hypergraph. The circles around the nodes are singletons. Line segments represent edges. Other hyperedges (only $\{1, 3, 4\}$ in this example) are indicated by splinegons.

The blocks are $S_1 = \emptyset, S_2 = \emptyset, S_3 = \{\{2,3\}, \{1,3\}\}, S_4 = \{\{1,3,4\}, \{3,4\}\}$. The block-

partitioned incidence matrix and ordinal matrix are as follows:

$$A = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccccccc} \{1\} & \{2\} & \{3\} & \{4\} & \{2,3\} & \{1,3\} & \{1,3,4\} & \{3,4\} \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{pmatrix} \end{array},$$

$$C = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccccccc} \{1\} & \{2\} & \{3\} & \{4\} & \{2,3\} & \{1,3\} & \{1,3,4\} & \{3,4\} \end{array} \\ \begin{pmatrix} 0 & 7 & 6 & 5 & 4 & 2 & 1 & 3 \\ 7 & 0 & 6 & 5 & 1 & 4 & 3 & 2 \\ 7 & 6 & 0 & 5 & 4 & 3 & 1 & 2 \\ 7 & 6 & 5 & 0 & 4 & 3 & 2 & 1 \end{pmatrix} \end{array}.$$

2.3 Scarf's Algorithm

We review Scarf's algorithm in this section. It involves the iterative repetion of two operations, namely cardinal pivot and ordinal pivot. We describe these two operations in Sections 2.3.1 and 2.3.2 and then show how they are combined in Scarf's algorithm in Section 2.3.3.

2.3.1 Cardinal Pivot

We define the cardinal pivot operation in this section.

Definition 9 (Cardinal pivot)**.** *Let $A \in \mathbb{R}^{n\times m}$ be a matrix in standard form and $b \in \mathbb{R}^n$ be a vector. Let $B = \{j_1, \ldots, j_n\}$ be a feasible cardinal basis and let $j_t \in [m] \setminus B$. A* cardinal pivot *from B with* entering column *j_t returns a feasible cardinal basis B' that is defined as follows: Let x be the extreme point of P corresponding to B. Let A_B, A_{j_t} be the submatrices of A indicated by B, j_t, respectively. Let $y = A_B^{-1} A_{j_t} \in \mathbb{R}^B$. A column $j \in B$ is a* leaving candidate *if $y_j > 0$ and $j \in \arg\min_{j\in B: y_j>0}\{x_j/y_j\}$. We define $B' := B \cup \{j_t\} \setminus \{j_\ell\}$, where j_ℓ is a leaving candidate. The column $j_\ell = B - B'$ is known as the leaving column.*

The following is a standard result in linear programming, which implies that the cardinal pivot is well-defined.

Lemma 10 ([21])**.** *If $P = \{x \in \mathbb{R}^m_{\geq 0} : Ax = b\}$ is bounded, then for every feasible cardinal basis B and every $j_t \in [m] \setminus B$, there exists at least one leaving candidate. Also, if j_ℓ is a leaving candidate, then $B' = B \cup \{j_t\} \setminus \{j_\ell\}$ is a feasible cardinal basis.*

We note that a cardinal pivot could have multiple leaving candidates. A pivoting rule determines a unique leaving candidate.

Definition 11 (Pivoting rule, Degeneracy)**.** *Let $A \in \mathbb{R}^{n\times m}$ be a matrix in standard form and $b \in \mathbb{R}^n$ be a vector. A* pivoting rule *is a criterion that determines a unique leaving candidate for every tuple (B, j_t), where B is a feasible cardinal basis and $j_t \in [m]\setminus B$. A cardinal pivot $B \to B'$ is* degenerate *if the corresponding extreme points x and x' are such that $x = x'$. Otherwise, it is* non-degenerate.

2.3.2 Ordinal Pivot

We define the ordinal pivot operation in this section. In order to define the ordinal pivot operation, we need the following definition.

Definition 12 (Disliked relation)**.** *Let C be an ordinal matrix, O be an ordinal basis, and let $i \in [n]$. A column $j \in O$ is i-*disliked *w.r.t. O if $u_i^O = c_{ij}$. If O is clear from context, then we say that j is i-disliked (and omit "w.r.t. O").*

The "disliked" relation gives a bijection between the row set $[n]$ and the column set O as shown in the following lemma.

Lemma 13. *Let C be an ordinal matrix and O be an ordinal basis. Then, for every $j \in O$, there is a unique $i \in [n]$ such that column j is i-disliked. Also, for every distinct $j, j' \in O$, if j is i-disliked and j' is i'-disliked, then $i \neq i'$.*

Proof. By definition, since O is an ordinal basis, we have $C_j \not> u^O$ for every $j \in [m]$. Suppose that there exists $j \in O$ such that no row dislikes j. Then for every row $i \in [n]$, $u_i^O \neq c_{i,j}$ implies $u_i^O < c_{i,j}$, which results in $u^O < C_j$, a contradiction. Therefore, every $j \in O$ is disliked by at least one row. Since $|O| = n$, we deduce that every $j \in O$ is disliked by exactly one row $i \in [n]$.

Moreover, by definition of u^O and the fact that every row vector in C has distinct entries, no row can simultaneously dislike two columns. Therefore, the second part of the lemma holds. □

We now have the ingredients needed to define the ordinal pivot. Pseudocode is provided in Algorithm 1.

Definition 14 (Ordinal pivot)**.** *Let C be an ordinal matrix, O be an ordinal basis, and $j_\ell \in O$. An* ordinal pivot *from O with leaving column j_ℓ returns $O' = (O \setminus \{j_\ell\}) \cup \{j^*\}$ where j^* is defined as follows: Let i_ℓ be the unique row such that j_ℓ is i_ℓ-disliked w.r.t. O (exists and unique by Lemma 13). Let j_r be the column such that $u_{i_\ell}^{O-j_\ell} = c_{i_\ell,j_r}$ (recall that $u^{O-j_\ell} \in \mathbb{R}^n$ is the utility vector w.r.t. $O \setminus \{j_\ell\}$). Let i_r be the unique row in $[n]$ such that j_r is i_r-disliked w.r.t. O (exists and unique by Lemma 13). Define*

$$K := \{k \in [m] \setminus O : c_{i,k} > u_i^{O-j_\ell}, \text{ for all } i \neq i_r\}, \tag{4}$$
$$j^* := \arg\max_{k \in K} c_{i_r,k}.$$

We call i_r as the reference row *and j_r as the* reference column.

Algorithm 1 Ordinal Pivot

Let O be the current ordinal basis, associated with utility vector u^O. Suppose that $j_\ell \in O$ is the leaving column.
$i_\ell \leftarrow$ the unique row $i \in [n]$ that makes $u_i^O \leq c_{i,j_\ell}$ equal.
$j_r \leftarrow \arg\min_{j \in O - j_\ell}\{c_{i_\ell,j}\}$. ▷ Reference column.
$i_r \leftarrow$ the unique row $i \in [n]$ that makes $u_i^O \leq c_{i,j_r}$ equal.
$K \leftarrow \{k \in [m] \setminus O : c_{i,k} > u_i^{O-j_\ell}, \text{ for all } i \neq i_r\}$.
$j^* \leftarrow \arg\max_{k \in K} c_{i_r,k}$. ▷ Entering column.
$O \leftarrow O \cup \{j^*\} \setminus \{j_\ell\}$.

In contrast to the cardinal pivot, the ordinal pivot reverses the order of entering and leaving. Moreover, the ordinal pivot does not need a pivoting rule since the operation is unique. Scarf showed the following result which implies that the ordinal pivot is well-defined.

Lemma 15 ([21])**.** *Let C be an ordinal matrix, O be an ordinal basis, and $j_\ell \in O$. Let i_ℓ, i_r, j_r, and j^* be as defined in Definition 14. Then, we have the following:*

1. *Column $j = i_r \in K$, and thus $K \neq \emptyset$.*

2. *The set $O' = (O \setminus \{j_\ell\}) \cup \{j^*\}$ is an ordinal basis.*

3. *Let $j \in [m] \setminus O$. If $O'' = (O \setminus \{j_\ell\}) \cup \{j\}$ is an ordinal basis, then $j = j^*$. In other words, with a fixed leaving column j_ℓ, there exists a unique ordinal basis O'' with $|O \cap O''| = n - 1$ and $j_\ell \notin O'$.*

We recall that every ordinal basis is associated with an utility vector. Scarf showed certain helpful properties about the utility vectors: the row i_ℓ that dislikes the leaving column j_ℓ in the ordinal basis O will increase its utility, while the reference row i_r that dislikes the reference column j_r will decrease its utility. The utilities of the other rows stay the same. We state his result below since it will be useful in designing a potential function for poly-time convergence of Scarf's algorithm in certain settings.

Lemma 16 ([21])**.** *Let C be an ordinal matrix, O be an ordinal basis, and $j_\ell \in O$. Let i_ℓ, i_r, j_r, j^*, and O' be as defined in Definition 14. Then,*

1. $u^{O'}_{i_\ell} = c_{i_\ell, j_r} > c_{i_\ell, j_\ell} = u^{O}_{i_\ell}$,

2. $u^{O'}_{i_r} = c_{i_r, j^*} < c_{i_r, j_r} = u^{O}_{i_r}$, *and*

3. $u^{O'}_{i} = u^{O}_{i}$ *for every $i \in [n] \setminus \{i_\ell, i_r\}$.*

2.3.3 Initialization, Iteration, and Termination

Let $A \in \mathbb{R}^{n \times m}$ be a matrix in standard form, $b \in \mathbb{R}^m_+$ be a positive vector such that $P = \{x \in \mathbb{R}^m_{\geq 0} : Ax = b\}$ is bounded, and C be an ordinal matrix. Fix a cardinal pivoting rule. Scarf's algorithm initializes with a basis pair (B_0, O_0), where $B_0 = \{1, 2, \ldots, n\}$ and $O_0 = \{2, 3, \ldots, n, n+1\}$. The algorithm is iterative and maintains a pair (B, O) where B is a feasible cardinal basis, O is an ordinal basis and $|B \cap O| \geq n - 1$ (it can be verified that the initialized pair (B_0, O_0) satisfies these conditions). It proceeds alternatively with a cardinal pivot using the cardinal pivoting rule and an ordinal pivot. In particular, it goes through a sequence

$$(B_0, O_0) \rightarrow (B_1, O_0) \rightarrow (B_1, O_1) \rightarrow \cdots, \tag{5}$$

where for every $i \geq 0$, we have that $(B_i, O_i) \rightarrow (B_{i+1}, O_i)$ is a cardinal pivot from the feasible cardinal basis B_i with entering column $j_t = O_i - B_i$ using the cardinal pivoting rule, and $(B_{i+1}, O_i) \rightarrow (B_{i+1}, O_{i+1})$ is an ordinal pivot from the ordinal basis O_i with leaving column $j_\ell = B_{i+1} - O_i$.

Definition 17 (Scarf pair, iteration)**.** *A pair (B_i, O_i) in the sequence (5) with $B_i \neq O_i$ is called a* Scarf pair*. We denote a consecutive pair of cardinal pivot and ordinal pivot $(B_i, O_i) \rightarrow (B_{i+1}, O_i) \rightarrow (B_{i+1}, O_{i+1})$ as an* iteration *of Scarf's algorithm.*

We notice that $|B_0 \cap O_0| = n - 1$. The algorithm will maintain the invariant $|B_i \cap O_i| \geq n - 1$ and $|B_{i+1} \cap O_i| \geq n - 1$. The sequence terminates when $|B \cap O| = n$. If the sequence terminates, then the algorithm returns the extreme point x corresponding to the feasible cardinal basis B. Therefore, an iteration of Scarf's algorithm begins with a Scarf pair and returns either another Scarf pair or a dominating basis. Scarf showed the following result:

Theorem 18. *[21] Let $A \in \mathbb{R}^{n\times m}$ be a matrix in standard form, $b \in \mathbb{R}^m_+$ be a positive vector such that $P = \{x \in \mathbb{R}^m_{\geq 0} : Ax = b\}$ is bounded, and C be an ordinal matrix. There exists a cardinal pivoting rule such that Scarf's algorithm executed with that cardinal pivoting rule for (A, b, C) terminates within finite number of iterations. Moreover, the extreme point x associated with the terminating feasible cardinal basis B is a dominating extreme point.*

Theorem 18 implies Scarf's lemma (Theorem 5). The algorithm also provides a procedure to find a fractional stable matching for a given hypergraphic preference stystem (via Theorem 7). However, (similar to the simplex algorithm) the convergence may not be efficient for arbitrary (A, b, C) satisfying the hypothesis of Theorem 18.

2.4 Network Matrix and Network Hypergraph

Network hypergraphs are a special family of hypergraphs for which a hypergraphic preference system admits a stable matching (not just a fractional stable matching). In this section, we introduce network hypergraphs and certain subfamilies of network hypergraphs that are of interest to this work. We first recall certain terminology. A directed graph $\mathcal{D} = (U, \mathcal{A})$, is specified by a finite set U of *vertices* and a set $\mathcal{A} = \{(u, v) : u, v \in U\}$ of *arcs*.

Definition 19 ($\mathcal{D}$-Path, forward/backward arcs, and $\mathcal{D}$-directed path)**.** *Let $\mathcal{D} = (U, \mathcal{A})$ be a directed graph. Let $F \subseteq \mathcal{A}$, and $P = (a_1, a_2, \ldots, a_p)$ be an ordered set such that $F = \{a_1, a_2, \ldots, a_p\}$.*

1. *We say F (or P) is a $\mathcal{D}$-*path *if P forms an undirected path. When the underlying digraph is clear from context, we omit $\mathcal{D}$ and say F (or P) is a path.*
2. *An arc $a_i \in F$ is* forward *on P if following the order P we visit the tail of a_i before visiting its head, otherwise a_i is* backward *on P.*
3. *We say F (or P) is a $\mathcal{D}$-*directed path *if every arc in P is a forward arc on P.*

Definition 20 (Path concatenation)**.** *Let $P = (a_1, \ldots, a_p)$ and $P' = (a'_1, \ldots, a'_{p'})$ be two paths. The starting vertex (resp. ending vertex) of P is the unique vertex of $a_1 \backslash a_2$ (resp. $a_p \backslash a_{p-1}$). If the ending vertex of P is the same as the starting vertex of P', we define $P \oplus P' = (a_1, \ldots, a_p, a'_1, \ldots, a'_{p'})$ as the* concatenation *of the two paths.*

The concatenation of P and P' may not be a path since P and P' may share common vertices.

Definition 21 (Directed tree)**.** *A directed graph $\mathcal{T}$ is a* directed tree *if the underlying undirected graph (obtained by dropping the orientation of all arcs) is connected and acyclic.*

Definition 22 (Network Matrix)**.** *Let $\mathcal{D} = (U, \mathcal{A})$ be a directed graph and $\mathcal{T} = (U, \mathcal{A}_0)$ be a directed tree on vertex set U. The network matrix corresponding to $(\mathcal{D}, \mathcal{T})$ is the matrix $M \in \{0, \pm 1\}^{\mathcal{A}_0 \times \mathcal{A}}$ where for every $a = (u, v) \in \mathcal{A}$ and $a' \in \mathcal{A}_0$, we have*

$$M_{a',a} = \begin{cases} 1 & \text{if } a' \text{ is a forward arc on the unique } \mathcal{T}\text{-path from } u \text{ to } v; \\ -1 & \text{if } a' \text{ is a backward arc on the unique } \mathcal{T}\text{-path from } u \text{ to } v; \\ 0 & \text{if } a' \text{ does not belong to the unique } \mathcal{T}\text{-path from } u \text{ to } v. \end{cases} \tag{6}$$

A matrix M is a network matrix *if there exists a directed graph $\mathcal{D} = (U, \mathcal{A})$ and a directed tree $\mathcal{T} = (U, \mathcal{A}_0)$ such that the network matrix corresponding to $(\mathcal{D}, \mathcal{T})$ is M. We say $\mathcal{T}$ is the* principal tree *corresponding to M.*

We note that a network matrix M corresponding to $(\mathcal{D} = (U, \mathcal{A}), \mathcal{T} = (U, \mathcal{A}_0))$ is in standard form if and only if $\mathcal{A}_0 \subseteq \mathcal{A}$ and the first n columns of M correspond to $\mathcal{A}_0$. Network matrices are totally unimodular (see, e.g. [22]). A consequence of this fact is the following:

Theorem 23. *Let $M \in \{0, \pm 1\}^{n \times m}$ be a network matrix. Then, all extreme points of the polyhedron $\{x \in \mathbb{R}^m_{\geq 0} : Mx = b\}$ with $b \in \mathbb{Z}^n$ are integer vectors.*

We will focus on $\{0,1\}$-valued network matrices. A $\{0,1\}$-valued matrix M can naturally be represented by a hypergraph whose node-hyperedge incidence matrix is M. If M is additionally a network matrix, then we call the associated hypergraph as a network hypergraph. We define them formally below. See Figure 2 for an illustration.

Definition 24 (Network Hypergraph)**.** *A hypergraph $H = (V, E)$ is a* network hypergraph *if the node-hyperedge incidence matrix A of H is a network matrix. We say $(\mathcal{D}, \mathcal{T})$ is the* underlying network *of H if the node-hyperedge incidence matrix A of H is the network matrix corresponding to $(\mathcal{D}, \mathcal{T})$.*

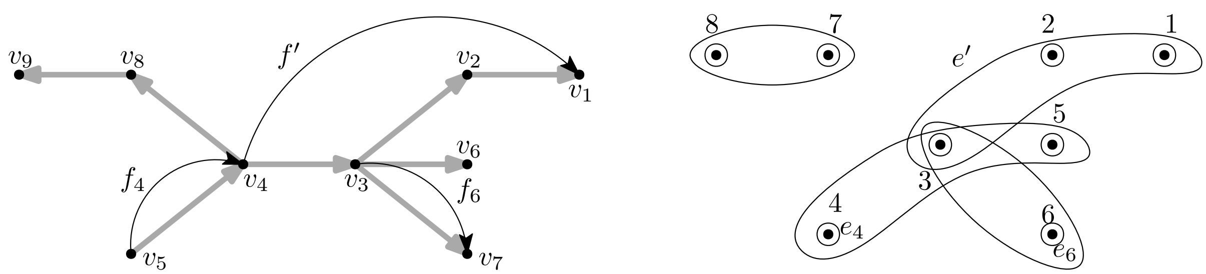

Figure 2: An example of a network hypergraph. On the left we present the principal tree using grey arcs. On the right we have a hypergraph with $V = [8]$ and 12 hyperedges (8 of them are singletons). We illustrate the hyperedges $e_4 = \{4\}$, $e_6 = 6$ and $e' = \{1, 2, 3\}$ on the right by the paths f_4, f_6 and f' using black arcs on the left, respectively. In the principal tree we present, the unique source is v_5. The sinks are v_1, v_6, v_7, v_9. There are two branching vertices, namely v_3 and v_4.

We note that there exists a polynomial time algorithm to verify whether a given matrix is a network matrix and if so, then find the principal tree [22]. Therefore, we can also use the same algorithm to verify whether a given hypergraph is a network hypergraph (equivalent to verifying whether a given $\{0,1\}$-valued matrix is a network matrix). By Theorems 5, 7 and 23, we have the following corollary:

Corollary 25. *Let $H = (V, E)$ be a network hypergraph. For every preference set $\succ$, the hypergraphic preference system $(H = (V, E), \succ)$ admits a stable matching.*

2.5 Arborescence Hypergraph, Interval Hypergraph

We focus primarily on the network hypergraphic preference system where the principal tree $\mathcal{T}$ of the underlying network $(\mathcal{D}, \mathcal{T})$ is an *arborescence*.

Definition 26 (Arborescence)**.** *A directed tree $\mathcal{T} = (U, \mathcal{A}_0)$ is an* arborescence *if there is a distinguished vertex $r \in U$ called* root *such that for every $v \in U$, the unique $\mathcal{T}$-path from r to v is a $\mathcal{T}$-directed path.*

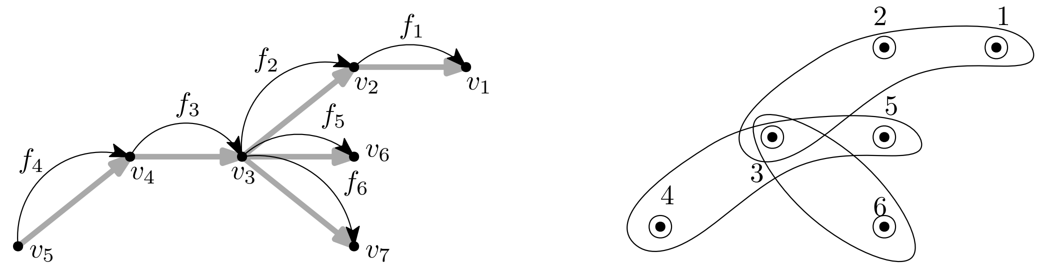

Figure 3: An example of an arborescence and an arborescence hypergraph. On the left we present an arborescence $\mathcal{T}$ with the root v_5. On the right we have a hypergraph H with principal tree $\mathcal{T}$ with $V = [6]$ and 9 hyperedges (6 of them are singletons). We present the arcs in the arborescence which are corresponding to the singletons in the hypergraph.

Definition 27 (Arborescence Hypergraph)**.** *A network hypergraph $H = (V, E)$ with underlying network $(\mathcal{D} = (U, \mathcal{A}), \mathcal{T} = (U, \mathcal{A}_0))$ is an* arborescence hypergraph *if the principal tree $\mathcal{T} = (U, \mathcal{A}_0)$ is an arborescence.*

Definition 28 (Interval Hypergraph)**.** *A hypergraph $H = (V, E)$ is an* interval hypergraph *if $V = [n]$ and every $e \in E$ is such that $e = \{i, i+1, \ldots, j\}$ for some $i, j \in [n]$ with $i \leq j$.*

An interval hypergraph is an arborescence hypergraph: Indeed, given an interval hypergraph $H = (V, E)$ with $V = [n]$, let $\mathcal{T} = (U, \mathcal{A}_0)$ be a directed graph where $U = [n+1]$ and $\mathcal{A}_0 = \{(i, i+1) : i \in [n]\}$. Let $\mathcal{D} = (U, \mathcal{A})$ be such that $\mathcal{A} = \{(i, j+1) : i \leq j, [i, j] \in E\}$. Then H is a network hypergraph with underlying network $(\mathcal{D}, \mathcal{T})$ where $\mathcal{T}$ is an arborescence.

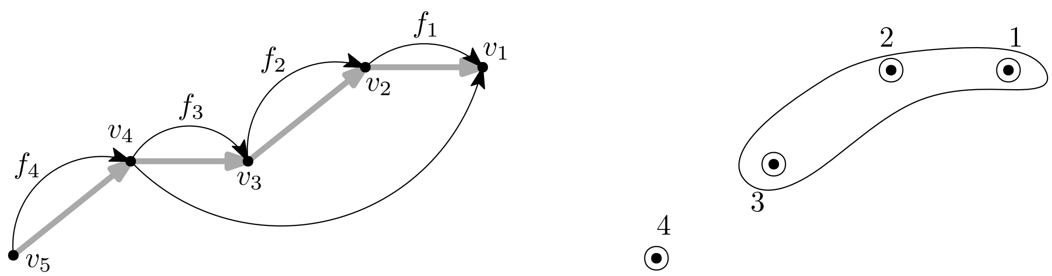

Figure 4: An example of an interval hypergraph. On the left we present the principal tree. On the right we have an interval hypergraph with $V = [4]$. The hyperedge $\{1, 2, 3\}$ can be seen as an interval $[1, 3]$, and correspond to the black arc/grey path from v_4 to v_1 on the principal tree.

Definition 29 (source, sink, branching vertex, leaf)**.** *Let $\mathcal{T} = (U, \mathcal{A}_0)$ be a directed tree on U and let $v \in U$. We denote $\delta^+_{\mathcal{T}}(v) = \{(v, u) : u \in U, (u, v) \in \mathcal{A}_0\}$ and $\delta^-_{\mathcal{T}}(v) = \{(w, v) : w \in U, (w, v) \in \mathcal{A}_0\}$. Let $d^+_{\mathcal{T}}(v) := |\delta^+_{\mathcal{T}}(v)|$ and $d^-_{\mathcal{T}}(v) := |\delta^-_{\mathcal{T}}(v)|$, and we call them out-degree, in-degree of v in $\mathcal{T}$, respectively. If $\delta^+_{\mathcal{T}}(v) = 0$ (resp., $\delta^-_{\mathcal{T}}(v) = 0$), then we call v a* sink *(resp.,* source*). If $\delta^+_{\mathcal{T}}(v) = 2$, then we say that v is a* branching vertex*. An arc $e \in \mathcal{A}_0$ is called a* leaf *if its tail vertex is a sink.*

See Figure 2 for an illustration of source, sink, branching vertex, and leaf. It can be verified that an arborescence is a directed tree with a unique source. A natural related question is the following: is it possible to verify whether a given network hypergraph is an arborescence hypergraph in polynomial-time? We leave this as an open question. For the purposes of this paper, we assume that the input arborescence hypergraph is given by its underlying network where the principal tree is an arborescence.

3 Polynomiality of Scarf's Algorithm on Arborescence Hypergraphic Preference System

According to Corollary 25, every hypergraphic preference system over an arborescence hypergraph admits a stable matching. The key contribution of this paper is showing that a stable matching in an arborescence hypergraphic preference system can be found in polynomial time via Scarf's algorithm.

Theorem 1. *Let $(H = (V, E), \succ)$ be a hypergraphic preference system where H is an arborescence hypergraph. There exists a pivoting rule such that Scarf's algorithm terminates in at most $|V|$ iterations and outputs a stable matching on $(H = (V, E), \succ)$ in time $O(|V||E|)$.*

To show Theorem 1, we establish certain properties of Scarf's algorithm applied to a hypergraphic preference system. From now on, let $(H = (V, E), \succ)$ be the hypergraphic preference system of interest. Let A and C be the block-partitioned matrices associated with this hypergraphic preference system and let $b \in \{1\}^V$ be the all-ones vector. In Section 3.1, we discuss some general properties of Scarf pairs arising in the execution of Scarf's algorithm on hypergraphic preference system; in particular, we define the notion of a *separator* associated with each Scarf pair which will act as a potential function to bound the number of iterations. In Section 3.2, we focus on the case where H is a network hypergraph. For this family, we give a combinatorial interpretation of the cardinal pivot. In Section 3.3.1 and Section 3.3.2, we focus on the case where H is an arborescence hypergraph. We design a cardinal pivoting rule. In Section 3.3.3, we prove Theorem 1 based on the properties established in Sections 3.1–3.3.2.

3.1 Ordinal Pivots for Hypergraphic Preference System

Let $(H = (V, E), \succ)$ be the hypergraphic preference system of interest. Let A and C be the block-partitioned matrices associated with this hypergraphic preference system and let $b \in \{1\}^V$ be the all-ones vector.

3.1.1 Controlling node

We first make an assumption about (A, b, C). When implementing Scarf's algorithm, the index $i = 1$ can be seen as a controlling row/column in Scarf's algorithm (see [21]). Without loss of generality, we can let the first row and column be artificial, which eliminates the asymmetry among original rows/agents. Formally, we have the following:

Lemma 30. *Let (A, b, C) be as above. Construct a new tuple (A', b', C') where*

$$A' = \begin{pmatrix} 1 & 0^T \\ 0 & A \end{pmatrix}, b' = \begin{pmatrix} 1 \\ b \end{pmatrix}, C' = \begin{pmatrix} 0 & \xi^T \\ \chi & C \end{pmatrix}, \tag{7}$$

where $\chi = (M, \dots, M)^T$ such that $M > \max_{i,j} c_{ij}$, and $\xi = (m, m-1, \dots, 2, 1)^T$. We index the new row/column in A', C' by 0-th row/column. Then B is a dominating basis for (A, b, C) if and only if $B' = B \cup \{0\}$ is a dominating basis for (A', b', C').

Proof. For an $n \times n$ submatrix A_B of A, we have that

$$A_{B'} = \begin{array}{cc} & \begin{array}{cccc} 0 & \cdots & \cdots & \cdots \end{array} \\ \begin{array}{c} 0 \\ \vdots \\ \vdots \\ \vdots \end{array} & \left(\begin{array}{c:ccc} 1 & 0 & \cdots & 0 \\ \hdashline 0 & & & \\ \vdots & & A_B & \\ 0 & & & \end{array}\right) \end{array}$$

is an $(n+1) \times (n+1)$ submatrix of A', with $\det A_{B'} \neq 0$ if and only if $\det A_B \neq 0$. Also, for $x \in \mathbb{R}^n_{\geq 0}$, $x' = (1, x^T)^T$ yields that $A_B x = b$ if and only if $A_{B'} x' = b'$. Thus equivalently, B' is an (A', b')-feasible basis if and only if B is an (A, b)-feasible basis.

Similarly, for an $n \times n$ submatrix C_O of C we define the following $(n+1) \times (n+1)$ submatrix of C' as

$$C'_O = \begin{array}{cc} & \begin{array}{cccc} 0 & \cdots & \cdots & \cdots \end{array} \\ \begin{array}{c} 0 \\ \vdots \\ \vdots \\ \vdots \end{array} & \left(\begin{array}{c:ccc} 0 & \times & \cdots & \times \\ \hdashline M & & & \\ \vdots & & C_O & \\ M & & & \end{array}\right) \end{array}.$$

By definition, if O is an ordinal basis of C, then the utility vector u^O cannot have $u^O < C_j$ for any column $j \in [n+m]$. Thus, the utility vector $u^{O'} = (0, (u^O)^T)^T$ generated by O' cannot have $u^{O'} < C'_j$ for any column $j \in \{0, 1, \ldots, n+m\}$. Thus, O' is an ordinal basis for C'. Conversely, if O' is an ordinal basis of C', then again by definition, utility vector $u^{O'}$ has the form $u^{O'} = (0, (u^O)^T)^T$, where $u^O_i = \min_{\ell \in O} c_{i,\ell}$ for $i \in [n]$, since by assumption we have $c_{i,\ell} < M$. By the dominating property $u^{O'} < C_j$ is impossible for every column $j \in \{0, 1, \ldots, n+m\}$. Since $u^{O'}_0 = 0 < c_{0j}$, we must have $u^O_i = u^{O'}_i \geq c_{i,j}$ for some $i \in [n]$, which implies the dominating property for O. Hence, B is a dominating basis for (A, b, C) if and only if B' is a dominating basis for (A', b', C'). □

Another nice property of this modified instance is that the terminating pivot of Scarf's algorithm is always an ordinal pivot.

Lemma 31. *Let (A', b', C') be as in Lemma 30, let (B_0, O_0) be the initial Scarf pair and consider the sequence of pivot (5). Then, for every $i \geq 0$, $B_{i+1} \neq O_i$. Therefore, Scarf's algorithm terminates when $B_i = O_i$ for some $i \in \mathbb{Z}_+$, which happens only after an ordinal pivot.*

Proof. Let (B, O) be a Scarf pair. Then we have $|B \cap O| = n$ (notice that $|B| = |O| = n+1$ since we have 0-th row) and $0 = B - O$. Consider the cardinal pivot $(B, O) \to (B', O)$, we claim that $0 \in B'$ and thus $B' \neq O$. In fact, if $0 \notin B'$, then the 0-th row of the submatrix $A_{B'}$ has all 0's, which implies $A_{B'}$ is not full-rank and thus B' is not a cardinal basis, a contradiction. Therefore, since $0 \in B'$ and $0 \notin O$, we have $B' \neq O$. Thus, the algorithm does not terminate and perform the next step, which is an ordinal pivot $(B', O) \to (B', O')$. Since a cardinal pivot can not make the algorithm halt, the last pivot is an ordinal pivot and thus the last ordinal pivot gives $B_i = O_i$ for some $i \in \mathbb{Z}_+$. □

In light of Lemma 31, it follows that Scarf's algorithm with (A', b', C') will terminate at the end of an iteration (i.e., the algorithm cannot terminate after just the cardinal pivot of an iteration).

We remark that the property that the algorithm terminates after an ordinal pivot may not be true in general.

We will henceforth assume that our matrices A, C have such a 0-th row and column as shown in (7).

3.1.2 Separator

When applied to the hypergraphic preference system, Lemma 30 implies that without loss of generality, we can add a node 0 to the instance, with the only hyperedge that contains 0 being $e_0 = \{0\}$. After a stable matching for the new hypergraphic preference system is obtained, removing $\{0\}$ gives a stable matching in the original instance. This modification to the instance, however, changes the node that "controls" Scarf's algorithm. In the modified instance, we have $V = \{0, 1, \ldots, n\}$ and $E = \{e_0, e_1, \ldots, e_m\}$ where the hyperedge indices are ordered according to the block-partition of A, C, that is, e_j corresponds to the j-th column of A (as well as C). In particular, for $i \in V$, we use $e_i = \{i\}$ to denote a singleton hyperedge.

Let (B, O) be a Scarf pair. Suppose the 0-disliked column w.r.t. O is $j^{\rightarrow}$. We recall that by Lemma 30, the 0-th row of C is $(0, m, m-1, \ldots, 2, 1)^T$, which is decreasing except for the 0-th column. Since (B, O) is a Scarf pair, $0 \notin O$. Thus, the 0-disliked column $j^{\rightarrow}$, by definition, is the rightmost column in O (i.e. the column with the largest index). This is the reason for denoting it as $j^{\rightarrow}$.

Definition 32. *Let (B, O) be a Scarf pair. Suppose the* 0*-disliked column w.r.t. O is $j^{\rightarrow}$. If $j^{\rightarrow}$ belongs to the i-th block S_i, then we say that i is the* separator *of (B, O).*

Since the blocks form a partition of the non-singleton hyperedges, it follows that the separator is unique. We show that the separator satisfies the following properties:

Lemma 33. *Let (B, O) be a Scarf pair and suppose i is the separator of (B, O). Let e_i be the column which corresponds to the singleton hyperedge $\{i\}$. Then, e_i belongs to O and e_i is i-disliked.*

Proof. Suppose i is the separator. Consider the 0-disliked column $j^{\rightarrow}$ in O. We have $u_0^O = c_{0,j^{\rightarrow}}$ by definition. By Lemma 13, the i-disliked column cannot be $j^{\rightarrow}$, which gives $u_i^O < c_{i,j^{\rightarrow}}$. Therefore, there exists some $j \in O$ such that $c_{i,j} < c_{i,j^{\rightarrow}}$.

Let $j \in O$ be such that $c_{i,j} < c_{i,j^{\rightarrow}}$. Recall the permutation of entries in C: j is either in block i or $j = i$ (in which case $c_{i,j} = c_{i,i} = 0$). If the former happens, since the entries are decreasing inside the i-th block, we must have $j > j^{\rightarrow}$, which is a contradiction. Therefore, $j = i$ and thus $i \in O$. Since $c_{i,i} = 0$, we have $u_i^O = 0$ and column i (singleton e_i) is i-disliked. □

Lemma 34. *Let $(B, O) \rightarrow (B', O) \rightarrow (B', O')$ be an iteration in Scarf's algorithm. Suppose j_ℓ is the leaving column of the ordinal pivot $O \rightarrow O'$. Let j_r and j^* be the reference column and the entering column of the ordinal pivot $O \rightarrow O'$ respectively. If j_ℓ is i-disliked w.r.t. O and i is the separator of (B, O), then we have the following:*

(i) $j_r = j^{\rightarrow}$ is the rightmost column in O.

(ii) j_r is 0*-disliked w.r.t. O and j^* is* 0*-disliked w.r.t. O'.*

(iii) The separator of (B', O') is i' for some $i' > i$.

Proof. Given the hypothesis that i is the separator of (B, O), by Lemma 33 we know $j_\ell = i$. Also, we have argued in Lemma 33 that if $j \in O$ such that $c_{i,j} < c_{i,j^{\rightarrow}}$, then $j = i$. Therefore, $c_{i,j^{\rightarrow}}$ is the second least entry of row i among O. By definition of the reference column, we have $j_r = j^{\rightarrow}$. This shows (i).

By Definition 32, $j^{\rightarrow}$ is 0-disliked w.r.t. O, so is j_r. By Lemma 16 with $i_r = 0$, the ordinal pivot $O \to O'$ will introduce a new 0-disliked column j^* w.r.t. O'. This shows (ii).

By Lemma 15, when choosing the new entering column j^*, we need to satisfy that $c_{i',j^*} > u_{i'}^{O-j_\ell}$ for $i' \neq 0$. Thus in particular $c_{i,j^*} > u_i^{O-j_\ell} = u_i^{O-i} = c_{i,j_r}$ (the last equality holds since $c_{i,j_r} = c_{i,j^{\rightarrow}}$ is the second least entry of row i among O). Thus, by the permutation rule of row i, either $j^* < j^{\rightarrow}$, or $j^* > j^{\rightarrow}$ and j^* is not in block i. Suppose $j^* < j^{\rightarrow}$ happens. If $j^* = 0$, then we have $B' = O'$ and the algorithm terminates. If $0 < j^* < j^{\rightarrow}$, then we have $c_{0,j^*} > c_{0,j^{\rightarrow}} = c_{0,j_r}$, which violates Lemma 16, a contradiction. Thus, if (B', O') is a new Scarf pair, we can claim that $j^* > j^{\rightarrow}$ and j^* is not in block i. Suppose that j^* is in block i', then by the fact $j^* > j^{\rightarrow}$, we have $i' > i$. By definition 32, since j^* is 0-disliked w.r.t. O', i' is the new separator. This shows (iii). $\square$

3.2 Feasible Cardinal Bases and Cardinal Pivots for Network Matrices

In this section, we present properties about bases and cardinal pivots of network matrices. We start with the following property about network matrices of directed trees.

Lemma 35. *Suppose that $\mathcal{T}_1 = (U, \mathcal{A}_1)$ and $\mathcal{T}_2 = (U, \mathcal{A}_2)$ are two directed trees. Let A_1 be the $\mathcal{A}_1 \times \mathcal{A}_2$ network matrix corresponding to $(\mathcal{D}_1 = (U, \mathcal{A}_2), \mathcal{T}_1)$, and A_2 be the $\mathcal{A}_2 \times \mathcal{A}_1$ network matrix corresponding to $(\mathcal{D}_2 = (U, \mathcal{A}_1), \mathcal{T}_2)$. Then,*

$$A_1 \cdot A_2 = I_n. \tag{8}$$

Proof. Without loss of generality, we assume $\mathcal{A}_1 \cap \mathcal{A}_2 = \emptyset$. Otherwise, we can remove the common arcs from each set, split them into connected components, and then apply the result for each connected component, which corresponds to the same block of matrix A_1, A_2. Putting the submatrices together gives the desired result.

Let $\mathcal{D}_{1,2} = (U, \mathcal{A}_1 \cup \mathcal{A}_2)$ be the directed graph whose arcs are all and only the arcs from $\mathcal{T}_1$ and $\mathcal{T}_2$, denote by A the *incidence matrix* of $\mathcal{D}_{1,2}$, i.e., $A \in \{0, \pm 1\}^{U \times (\mathcal{A}_1 \cup \mathcal{A}_2)}$ such that

$$A(v,a) = \begin{cases} 1 & \text{if } v \text{ is the head of } a; \\ -1 & \text{if } v \text{ is the tail of } a; \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

Denote by $A = (X_1 | X_2)$, where the columns of X_1, X_2 correspond to the arcs in $\mathcal{A}_1, \mathcal{A}_2$, respectively. One can see that X_1, X_2 is the incidence matrix of $\mathcal{T}_1, \mathcal{T}_2$, respectively. We claim the following:

Claim 36. $X_1 \cdot A_1 = X_2$ *and* $X_2 \cdot A_2 = X_1$.

Proof of Claim 36. Indeed, to show the first equation, consider a column y^a in matrix X_2 and suppose it corresponds to the arc $a = (u,v) \in \mathcal{A}_2$. Then, y^a has only two nonzeros, i.e. $y^a_u = 1$ and $y^a_v = -1$. Consider the linear system $X_1 \cdot z = y^a$. Here, the solution z represents a feasible unit flow sent from u (as a supply) to v (as a demand) in the directed graph $\mathcal{T}_1 = (U, \mathcal{A}_1)$ since X_1 is the incidence matrix of $\mathcal{T}_1$. Since $\mathcal{T}_1$ is a tree on U, the only way to send a flow from u to v is to find the unique $\mathcal{T}_1$-path from u to v, and assign 1 unit flow on the forward arcs and -1 unit flow on the backward arcs. Thus, there is a unique solution z to $X_1 \cdot z = y^a$ such that $z^{a'} \neq 0$ if and only if a' appears in the $\mathcal{T}_1$-path from u to v, and is 1 (resp. -1) if a' is forward (resp. backward). Compare with the definition of (6), we have that the feasible solution z is exactly the column vector of A_1 that corresponds to a. This argument applies to every $a \in \mathcal{A}_2$, hence $X_1 \cdot A_1 = X_2$. Similarly it holds that $X_2 \cdot A_2 = X_1$. ■

We notice that both X_1 and X_2 have rank $|U|-1$, as they are the incidence matrix of a directed tree. Let X_1' and X_2' be obtained from X_1 and X_2 by deleting the same row from both, respectively. Then X_1' and X_2' are both invertible matrices with

$$X_1' \cdot A_1 = X_2' \text{ and } X_2' \cdot A_2 = X_1'.$$

Thus, we have $A_1 \cdot A_2 = (X_1')^{-1} X_2' (X_2')^{-1} X_1' = I_n$. □

From Lemma 35, we deduce that every cardinal basis for a network matrix corresponds to a directed tree on U.

Theorem 37 ([22])**.** *Let $\mathcal{D} = (U, \mathcal{A})$ be a directed graph with $|U| = n$, $|\mathcal{A}| = m$ and $\mathcal{T} = (U, \mathcal{A}_0)$ be a directed tree. Let A be the $\mathcal{A} \times \mathcal{A}_0$ network matrix corresponding to $(\mathcal{D}, \mathcal{T})$. Consider a subset $B \subset [m]$ of n columns. Denote the corresponding arc set by $\mathcal{A}_B$. Then the following statements are equivalent:*

(i) B is a cardinal basis.

(ii) $\mathcal{T}_B = (U, \mathcal{A}_B)$ is a directed tree.

Proof. If B is a basis, then $|B| = n$. If $(U, \mathcal{A}_B)$ is not a tree, then there exists a cycle in $\mathcal{A}_B$. Suppose that the arcs $a_{j_1} \to \cdots \to a_{j_k}$ corresponding to the columns $j_1, \ldots, j_k \in B$ create a cycle. Then, let $A_{j_1}, \ldots, A_{j_k}$ be the corresponding column vectors of A. Begin with the tail of a_{j_1}, walk along the cycle through the arcs $a_{j_1} \to \cdots \to a_{j_k}$, define $\lambda_i = 1$ if a_{j_i} goes forward and $\lambda_i = -1$ if a_{j_i} goes backward. By definition of network matrix, $\lambda_1 A_{j_1} + \cdots + \lambda_k A_{j_k} = 0$. Therefore, $A_{j_1}, \ldots, A_{j_k}$ are linearly dependent, a contradiction. Thus $(U, \mathcal{A}_B)$ is a directed tree.

Conversely, if $\mathcal{T}_B = (U, \mathcal{A}_B)$ is a directed tree, the submatrix A_B is a network matrix corresponding to $(\mathcal{T}, \mathcal{T}_B)$. By Lemma 35, A_B is invertible, thus B is a basis. □

Next, we will give a combinatorial interpretation of the cardinal pivot for a network matrix. We need the following lemma.

Lemma 38. *Let $\mathcal{D} = (U, \mathcal{A})$ be a directed graph with $|U| = n$, $|\mathcal{A}| = m$ and $\mathcal{T} = (U, \mathcal{A}_0)$ be a directed tree. Let A be the $\mathcal{A} \times \mathcal{A}_0$ network matrix corresponding to $(\mathcal{D}, \mathcal{T})$, $b = 1^V$, and let $P := \{x \in \mathbb{R}^{\mathcal{A}_0}_{\geq 0} : Ax = b\}$. Let B be a cardinal basis of A and suppose $j_t \notin B$. Let $a_{j_t} = (v, v')$ be the arc corresponding to the column j_t. Let $P_B(v, v')$ be the $\mathcal{T}_B$-path from v to v' where $\mathcal{T}_B = (U, \mathcal{A}_B)$. Define $y := A_B^{-1} \cdot A_{j_t}$. Then, we have*

$$y_a = \begin{cases} 1 & \text{if } a \in \mathcal{A}_B \text{ is a forward arc on path } P_B(v, v'), \\ -1 & \text{if } a \in \mathcal{A}_B \text{ is a backward arc on path } P_B(v, v'), \text{ and} \\ 0 & \text{if } a \text{ does not belong to the path } P_B(v, v'). \end{cases} \tag{10}$$

Proof. For simplicity, we abuse the notation by denoting $A_B^{-1} = (\beta_{ij})_{i,j \in [n]}$ and $A_{j_t} = (\gamma_j)_{j \in [n]}$. Also, to distinguish the arcs in $\mathcal{A}_0$ and $\mathcal{A}_B$ we let f_i, a_j be the representative arcs in $\mathcal{A}_0, \mathcal{A}_B$, respectively.

Since $\mathcal{T} = (U, \mathcal{A}_0)$ is the principal tree and B is a basis, by Lemma 35, the network matrix A_2 defined by $\mathcal{D}_2 = (U, \mathcal{A}_0, \mathcal{T}_B)$ is exactly A_B^{-1}. Therefore we have $y = A_2 \cdot A_{j_t}$.

First, suppose that the arc $a_{j_t} = (f_{i_1}, f_{i_2}, \ldots, f_{i_q})$ is composed by the unique $\mathcal{T}$-path $P_0(v, v') = (f_{i_1}, f_{i_2}, \ldots, f_{i_q})$ from v to v'. We abuse the notation by writing $f_k \in P_0(v, v')$ if $f_k \in \{f_{i_1}, f_{i_2}, \ldots, f_{i_q}\}$.

Notice that $\gamma_k \neq 0$ if and only if $f_k \in P_0(v, v')$ by (6). Thus, we have

$$y_i = \sum_{f_k \in P_0(v,v')} \beta_{ik}\gamma_k = \sum_{f_k \in P_0(v,v'), \gamma_k = 1} \beta_{ik} - \sum_{f_k \in P_0(v,v'), \gamma_k = -1} \beta_{ik}$$
$$= \sum_{f_k \in P_0(v,v'),\ f_k \text{ forward}} \beta_{ik} - \sum_{f_k \in P_0(v,v'),\ f_k \text{ backward}} \beta_{ik}. \tag{11}$$

For every $s \in [q]$, we define P_B^s as the unique $\mathcal{T}_B$-path generated by f_{i_s}, i.e. one can define P_B^s as the sequence $P_B^s = (a_{k_1}, a_{k_2}, \dots, a_{k_\ell})$, where P_B^s starts from the tail of f_{i_s} and ends at the head of f_{i_s}. By definition, since $A_2 = A_B^{-1}$ is a network matrix corresponding to $(\mathcal{T}, \mathcal{T}_B)$, β_{ik} indicates the direction of the arc $a_i \in \mathcal{A}_B$ on P_B^s. Therefore, $\beta_{ik} = 1$ (resp. -1) if there exists $s \in [q]$ such that $k = i_s$, and a_i is a forward (resp. backward) arc on P_B^s. From (11), we obtain

$$\begin{aligned} y_i = &\sum_{s \in [q],\ f_{i_s} \text{ forward}} \left(\mathbf{1}(a_i \in P_B^s, a_i \text{ forward}) - \mathbf{1}(a_i \in P_B^s, a_i \text{ backward})\right) \\ &- \sum_{s \in [q],\ f_{i_s} \text{ backward}} \left(\mathbf{1}(a_i \in P_B^s, a_i \text{ forward}) - \mathbf{1}(a_i \in P_B^s, a_i \text{ backward})\right). \end{aligned} \tag{12}$$

Let $\bar{P}_B^s = P_B^s$ if f_{i_s} is forward on a_{j_t}, and $\bar{P}_B^s$ is the reverse (reversing the order of the arcs in the sequence, for example, if $P_B^s = (a_{k_1}, a_{k_2}, \dots, a_{k_\ell})$, then $\bar{P}_B^s = (a_{k_\ell}, \dots, a_{k_2}, a_{k_1})$) of P_B^s if f_{i_s} is backward on a_{j_t}. Then (12) becomes

$$y_i = \sum_{s \in [q]} (\mathbf{1}(a_i \in \bar{P}_B^s, a_i \text{ forward}) - \mathbf{1}(a_i \in \bar{P}_B^s, a_i \text{ backward})). \tag{13}$$

Now consider a walk $W = \bar{P}_B^1 \oplus \cdots \oplus \bar{P}_B^q$ (see Definition 20). Then according to (13), y_i counts exactly the difference between the number of times a_i appears forward on W and that of times a_i appears backward on W. Notice that our goal is to relate y with the path $P_0(v, v')$, where such path $P_0(v, v')$ is a reduced version of W without redundant cycles. Notice that $P_0(v, v')$ can be obtained by deleting from W the arcs that appear multiple times, and when deleting them, we cancel the forward and backward arcs pair by pair, which results in that the difference between the number of occurrences maintain the same. At the end, each arc a_i either goes forward or backward, thus we have exactly the meaning shown in (10). □

As a consequence of Lemma 38, we show that the leaving column of a cardinal pivot should necessarily be a forward arc in the unique path from the tail of the entering arc to the head of the entering arc.

Corollary 39. *Let $\mathcal{D} = (U, \mathcal{A})$ be a directed graph with $|U| = n$, $|\mathcal{A}| = m$ and $\mathcal{T} = (U, \mathcal{A}_0)$ be a directed tree. Let A be the $\mathcal{A} \times \mathcal{A}_0$ network matrix corresponding to $(\mathcal{D}, \mathcal{T})$, $b = 1^V$, and $P := \{x \in \mathbb{R}_{\geq 0}^{\mathcal{A}_0} : Ax = b\}$. Let B be a feasible cardinal basis and let $\mathcal{T}_B = (U, \mathcal{A}_B)$. Consider the cardinal pivot from B with entering column j_t. Let $a_{j_t} = (v, v')$ be the arc corresponding to column j_t. Let $P_B(v, v')$ be the unique $\mathcal{T}_B$-path from v to v' (unique by Theorem 37). Then, j_ℓ is a leaving candidate of the cardinal pivot only if the arc a_{j_ℓ} is a forward arc on $P_B(v, v')$.*

Proof. By Lemma 38, $y_a > 0$ if and only if a is a forward arc on $P_B(v, v')$. Thus, by Definition 9, if j_ℓ is a leaving candidate, it has to satisfy $y_{a_{j_\ell}} > 0$, which implies that a_{j_ℓ} is an arc that is forward on $P_B(v, v')$. □

In order to give a combinatorial interpretation of non-degenerate cardinal pivot for a network matrix, we will use the notion of augmenting and descending paths.

Definition 40 (Augmenting and Descending Paths)**.** *Let $\mathcal{D} = (U, \mathcal{A})$, $x \in \mathbb{R}^{\mathcal{A}}$, and $P = (a_{j_1}, \dots, a_{j_p})$ be a $\mathcal{D}$-path.*

1. *The path P is x-*augmenting *if every forward arc a_{fwd} on P has $x_{a_{fwd}} = 1$ and every backward arc a_{bwd} on P has $x_{a_{bwd}} = 0$.*

2. *The path P is x-*descending *if every forward arc a_{fwd} on P has $x_{a_{fwd}} = 0$ and every backward arc a_{bwd} on P has $x_{a_{bwd}} = 1$.*

We show that a non-degenerate cardinal pivot is characterized by an x-augmenting path.

Theorem 41. *Let $\mathcal{D} = (U, \mathcal{A})$ be a directed graph with $|U| = n$, $|\mathcal{A}| = m$ and $\mathcal{T} = (U, \mathcal{A}_0)$ be a directed tree. Let A be the $\mathcal{A} \times \mathcal{A}_0$ network matrix corresponding to $(\mathcal{D}, \mathcal{T})$, $b = 1^V$, and $P := \{x \in \mathbb{R}^{\mathcal{A}_0}_{\geq 0} : Ax = b\}$. Let B be a feasible cardinal basis and let $\mathcal{T}_B = (U, \mathcal{A}_B)$. Consider a cardinal pivot from B to B' with j_t as the entering column where $a_{j_t} = (v, v') \in \mathcal{A}$. Let x, x' be the extreme points corresponding to B, B', respectively. Let $P_B(v, v')$ be the unique $\mathcal{T}_B$-path from v to v' (unique by Theorem 37). The following statements are equivalent:*

(i) The cardinal pivot is non-degenerate.

(ii) $x_{j_t} = 0$ and $x'_{j_t} = 1$.

(iii) $P_B(v, v')$ is x-augmenting.

(iv) $P_B(v, v')$ is x'-descending.

Proof. If the cardinal pivot is non-degenerate, then we have that $x \neq x'$. We claim that $x'_{j_t} \neq 0$. Indeed, by the equation $A_B x_B + A_{j_t} x_{j_t} = A_B x'_B + A_{j_t} x'_{j_t} = b$, if $x'_{j_t} = 0$, then since A_B has full rank, we must have $x_B = x'_B$, which results in $x = x'$, a contradiction. Since x' is feasible, we have $x'_{j_t} > 0$. By the fact that A is totally unimodular, we have $x'_{j_t} = 1$. Thus, (i) implies (ii). (ii) obviously implies (i) since by definition $x \neq x'$, thus the cardinal pivot is non-degenerate. Therefore, we obtain (i)$\Leftrightarrow$(ii).

By definition of pivoting (c.f. Section 2.3.1), we have $A_B x_B = 1$ and $A_B x_B + A_{j_t} x_{j_t} = 1$. Thus we have $A_B x'_B + A_{j_t} x'_{j_t} = 1$, which gives $x'_B = x_B - A_B^{-1} A_{j_t} x'_{j_t} = x_B - x'_{j_t} y$ where y is given by (10).

Now, assume (ii) holds. Observe that $x'_{j_t} = 1$ and $x'_B \geq 0$ means for all $a \in \mathcal{A}_B$, $y_a = 1$ implies $x_a = 1$. By Lemma 38, $y_a = 1$ if and only if a is a forward arc on $P_B(v, v')$, thus every forward arc a_{fwd} has $x_{a_{fwd}} = 1$ and $x'_{a_{fwd}} = 0$. On the other hand, consider any backward arc a_{bwd} on $P(v, v')$, if $x_{a_{bwd}} = 1$, then $x'_{a_{bwd}} = 1 - 1 \times (-1) = 2$, a contradiction with feasibility of x'. Therefore, $x_{a_{bwd}} = 0$ and $x'_{a_{bwd}} = 1$. Thus, we obtain (ii)$\Rightarrow$(iii) and (ii)$\Rightarrow$(iv).

On the other hand, if (iii) holds, then by Lemma 38 and $A_B x_B + A_{j_t} x_{j_t} = 1$, we obtain (ii). If (iv) holds, then again by Lemma 38 and $A_B x'_B + A_{j_t} x'_{j_t} = 1$, we can deduce (ii). Therefore, we obtain (iii)$\Rightarrow$(ii) and (iv)$\Rightarrow$(ii). $\square$

3.3 Polynomiality of Scarf's algorithm

In this section, we describe a pivoting rule and establish polynomiality of Scarf's algorithm using this pivoting rule for every arborescence hypergraphic preference system.

3.3.1 Depth-first Order

In order to guarantee some nice properties on the cardinal bases (or equivalently, directed trees) visited by the algorithm, we permute the vertices and arcs of the principal tree according to a depth-first order. This order is a common topological order on trees. Details can be found in [6].

Definition 42. *Let $\mathcal{T} = (U, \mathcal{A}_0)$ be an arborescence with root r. We define a partial order $\geq_{\mathcal{T}}$ over U such that $v \geq_{\mathcal{T}} v'$ if the $\mathcal{T}$-directed path from r to v' passes through v. We say that $\mathcal{T}$ is* depth-first *if*

1. *$U = \{v_1, \ldots, v_{n+1}\}$, and for $i, j \in [n+1]$, $v_i \geq_{\mathcal{T}} v_j$ implies $i \geq j$. In particular, v_{n+1} is the unique root of $\mathcal{T}$.*
2. *$\mathcal{A}_0 = \{f_1, \ldots, f_n\}$, and for every $i \in [n]$, the head of f_i is v_i.*

For example, the arborescence in Figure 5 is depth-first. For every given arborescence $\mathcal{T} = (U, \mathcal{A}_0)$ with $|U| = n + 1$, a permutation of U and $\mathcal{A}_0$ such that $\mathcal{T}$ is depth-first ordered can be found in time $O(n)$.

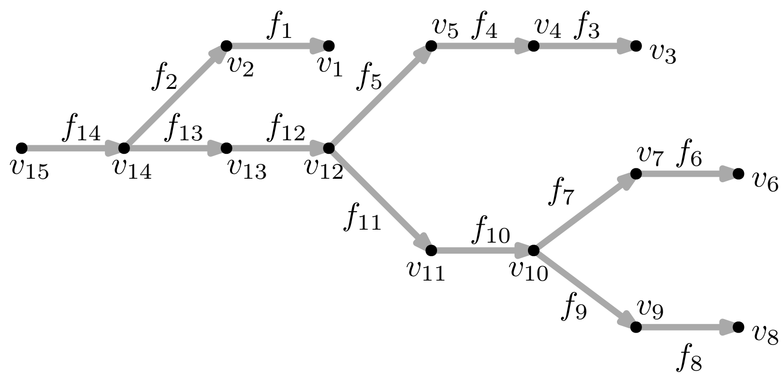

Figure 5: A depth-first arborescence $\mathcal{T} = (U, \mathcal{A}_0)$. $U = \{v_1, \ldots, v_{15}\}$ and $r = v_{15}$ is the root. $\mathcal{A}_0 = \{f_1, \ldots, f_{14}\}$. Notice that for every $i \in [14]$, the head of f_i is v_i, yet the tail of f_i may not be v_{i+1} (for example, $f_2 = (v_{14}, v_2)$).

3.3.2 The Pivoting Rule

We have seen the interpretation of cardinal pivot for network matrices in Section 3.2. When applied to the incidence matrix of an arborescence hypergraph corresponding to $(\mathcal{D} = (U, \mathcal{A}), \mathcal{T} = (U, \mathcal{A}_0))$, a variable x_a represents a specific arc $a \in \mathcal{A}$. From this perspective, by saying that the entering arc (resp. leaving arc) is a_{j_t} (resp. a_{j_ℓ}), we mean the entering variable/column (resp. leaving variable/column) is x_{j_t}/j_t (resp. x_{j_ℓ}/j_ℓ).

Let B be a feasible cardinal basis and consider a cardinal pivot from B. Suppose that we have a_{j_t} as an entering arc and $a_{j_t} = (v, v')$. Then, there is a $\mathcal{T}_B$-path from v to v', denoted by $P_B(v, v')$. By Corollary 39, the leaving arc a_{j_ℓ} has to be a forward arc on $P_B(v, v')$. We will use the following *first forward arc leaving (FFL) rule* as the pivoting rule.

Definition 43 (First forward arc leaving rule)**.** *Let B be a feasible cardinal basis. Let $a_{j_t} = (v, v')$ be the entering arc and $P_B(v, v') = (\bar{a}_1, \cdots, \bar{a}_p)$ be the unique $\mathcal{T}_B$-path. If there exists a forward arc $\bar{a}_k$ in $P_B(v, v')$ with $x_{\bar{a}_k} = 0$, choose the one with smallest subscript k as the leaving arc. If all forward arcs $\bar{a}_{fwd}$ in $P_B(v, v')$ have $x_{\bar{a}_{fwd}} = 1$, then let k be the smallest index such that $\bar{a}_k$ is forward on $P_B(v, v')$, and let $\bar{a}_k$ be the leaving arc.*

We first justify that FFL rule is valid, meaning that for every entering variable $j_t \notin B$, (i) the leaving arc defined above exists and is unique, and (ii) suppose that j_ℓ is the leaving variable defined by the FFL rule, then j_ℓ is indeed a leaving candidate (c.f. Definition 11).

Lemma 44. *The FFL rule is a valid pivoting rule.*

Proof. We begin by observing that for every $a = (u, u') \in \mathcal{A}$, we have that $u \geq_{\mathcal{T}} u'$. Let $a = (u, u') \in \mathcal{A}$. By definition, there exists a $\mathcal{T}$-directed path $P_0(u, u')$ from u to u'. Let r be the root of the arborescence $\mathcal{T}$, then the $\mathcal{T}$-path $P_0(r, u)$ from r to u has only forward arcs. Therefore, $P_0(r, u) \oplus P_0(u, u')$ also has only forward arcs, which implies that $P_0(r, u) \oplus P_0(u, u')$ is a $\mathcal{T}$-directed path from r to u'. Thus, $u \geq_{\mathcal{T}} u'$.

Let $a_{j_t} = (v, v') \in \mathcal{A}$ be the entering arc. In order to show validity of the pivoting rule, we first show that there exists at least one forward arc in $P_B(v, v')$. By the observation in the previous paragraph, we have that $v \geq_{\mathcal{T}} v'$. For the sake of contradiction, suppose that $P_B(v, v') = (\bar{a}_1, \ldots, \bar{a}_p)$ only contains backward arcs. Let $\bar{a}_i = (\bar{v}_i, \bar{u}_i)$ for every $i \in [p]$. Since $\bar{a}_i \in \mathcal{A}$, we have that $\bar{v}_i \geq_{\mathcal{T}} \bar{u}_i$ for every $i \in [p]$. Thus, we have $v' = \bar{v}_p \geq_{\mathcal{T}} \bar{u}_p = \bar{v}_{p-1} \geq_{\mathcal{T}} \cdots \geq_{\mathcal{T}} \bar{u}_1 = v$. This contradicts with $v \geq_{\mathcal{T}} v'$. Consequently, there exists at least one forward arc in $P_B(v, v')$. By the definition of the FFL rule, the choice of $\bar{a}_k$ is unique.

Now, in order to show validity of the pivoting rule, it suffices to show that if $a_{j_\ell} = \bar{a}_k$, then j_ℓ is a leaving candidate. In order to show this, we need to show that $y_\ell > 0$ and $\ell \in \arg\min\{x_j/y_j : j \in B, y_j > 0\}$. By Lemma 38, we have $y_\ell > 0$ (and indeed $y_a > 0$ if and only if a is a forward arc on $P_B(v, v')$). Thus, j_ℓ is always the minimizer as defined above, since either $x_{j_\ell} = 0$, or $x_{j_\ell} = 1$ and every $j \in B$ such that $y_j > 0$ satisfies $x_j/y_j = 1$ according to the FFL rule. □

We rewrite the cardinal pivot operation with FFL rule in Algorithm 2.

Algorithm 2 Cardinal Pivot with FFL Rule

Let B be the current feasible cardinal basis, associated with basic feasible solution x. Suppose that $j_t \notin B$ is the entering column.
Find $a_{j_t} = (v, v')$ and the $\mathcal{T}_B$-path $P_B(v, v') = (\bar{a}_1, \ldots, \bar{a}_p)$
$I \leftarrow \{i \in [p], \bar{a}_i \text{ is forward and } x_{\bar{a}_i} = 0\}$
$J \leftarrow \{i \in [p], \bar{a}_i \text{ is forward}\}$
if $I \neq \emptyset$ **then**
    $k \leftarrow \min\{i : i \in I\}$
else
    $k \leftarrow \min\{i : i \in J\}$
end if
Let the leaving column j_ℓ be such that $a_{j_\ell} = \bar{a}_k$.
$B \leftarrow B \cup \{j_t\} \setminus \{j_\ell\}$

3.3.3 Proof of Theorem 1

From now on, we assume the following: the arborescence hypergraph $H = (V, E)$ with principal tree $\mathcal{T} = (U, \mathcal{A}_0)$ where $U = \{v_1, \ldots, v_{n+1}\}$ and $\mathcal{A}_0 = \{f_1, \ldots, f_n\}$ are depth-first (c.f. Definition 42). The matrices A, C as the input of Scarf's algorithm are block-partitioned (c.f. Definition 6) with additional 0-th row and column as in Lemma 30. Every cardinal pivot follows the FFL rule (c.f. Definition 43). We will analyze Scarf's algorithm and show Theorem 1 under these assumptions.

Let (B_0, O_0) be the initial Scarf pair of the algorithm where $B_0 = \{0, 1, \ldots, n\}$ and $O_0 = \{1, 2, \ldots, n, n+1\}$. The algorithm executes alternatively a cardinal pivot with the FFL rule and an ordinal pivot. Then, we obtain a unique sequence (5) and each (B_k, O_k) in that sequence is a Scarf pair. We will maintain $|B_k \cap O_k| \geq n$, and if $B_k \neq O_k$, then $0 = B_k - O_k$. We now define a notion of a well-structured basis and will subsequently show that all cardinal bases visited by the algorithm are well-structured. This structural property of the basis will be helpful in bounding the number of iterations of the algorithm.

Definition 45. *Let B be a feasible cardinal basis and $\mathcal{T}_B = (U, \mathcal{A}_B)$ be the directed tree corresponding to B. Let x be the extreme point associated with B. Let $i \in [n]$. We say that B is an i-*nice basis *if the following properties hold:*

(i) Let $v \in U \setminus \{v_{n+1}\}$, and $P_B(v_{n+1}, v)$ be the $\mathcal{T}_B$-path from v_{n+1} to v. Then, $P_B(v_{n+1}, v)$ is x-augmenting.

(ii) $\{f_i, f_{i+1}, \ldots, f_n\} \subset \mathcal{A}_B$.

(iii) Let $f \in \{f_i, f_{i+1}, \ldots, f_n\}$. Let $\mathcal{T}^{-f} = (U, \mathcal{A}_0 - \{f\})$ be the subgraph of $\mathcal{T}$ with exactly two connected components (in the undirected sense). Denote by $U = R \cup W$ the partition of vertices of the two components such that $r \in R$ and $r \notin W$. If an arc $a \in \mathcal{A}_B$ has its end vertices in R and W, then $a = f$. In other words, the removal of each arc $f \in \{f_i, f_{i+1}, \ldots, f_n\}$ from $\mathcal{T}_B$ partitions U into the same sets as the removal of the same arc from $\mathcal{T}$.

See Figure 6 for an example of an i-nice basis. The following lemma is the main result of the section. It will allow us to bound the number of iterations in the algorithm.

Lemma 46. *Let (B, O) be a Scarf pair and we consider the iteration (a cardinal pivot and an ordinal pivot) $(B, O) \to (B', O) \to (B', O')$. Suppose $j_t = O - B$ is the entering column of the cardinal pivot, $j_\ell = B' - B$ is the leaving column of the cardinal pivot, and $j^* = O' - O$ is the entering column of the ordinal pivot. Let $i \in [n]$ be the separator of (B, O). Suppose B is an i-nice basis and $j^* \neq 0$. Then, we have the following:*

1. *Let $a_{j_t} = (v_j, v_k)$. Then, $f_i = (v_j, v_i)$ (recall that f_i is the unique arc entering v_i in $\mathcal{T}$). In other words, a_{j_t} and f_i share the same tail. In addition, f_i is the first arc in $P_0(v_j, v_k)$, where $P_0(v_j, v_k)$ be the $\mathcal{T}$-path from v_j to v_k.*
2. *$a_{j_\ell} = f_i$.*
3. *The separator of (B', O') is i' for some $i' > i$.*
4. *B' is an i'-nice basis.*

Proof. We use the notation in Definition 45.

1. By Definition 32, let $j^{\rightarrow}$ be the 0-th disliked column w.r.t. O. Then, $j^{\rightarrow}$ is in S_i, namely the i-th block of C. By the block structure, the hyperedge $e_{j^{\rightarrow}} \in E$ has $i \in e \subseteq \{1, \ldots, i\}$. Suppose $a_{j^{\rightarrow}} = (w, w')$. Then, the $\mathcal{T}$-directed path $P_0(w, w')$ from w to w' contains f_i and only contains arcs among $\{f_1, \ldots, f_i\}$. Since $\mathcal{T}$ is depth-first ordered, $P_0(w, w')$ visits a sequence of arcs with subscripts in a descending order, thus the first arc is f_i. Therefore, $w = v_j$ where $(v_j, v_i) = f_i$. Also, $w' \neq v_i$ since $e_{j^{\rightarrow}} \neq e_i$ (notice that $j^{\rightarrow}$ is in the i-th block of C, but i is among the first $n+1$ columns of C). We claim that $j^{\rightarrow} = O - B$ and thus $j_t = j^{\rightarrow}$. In fact, denote

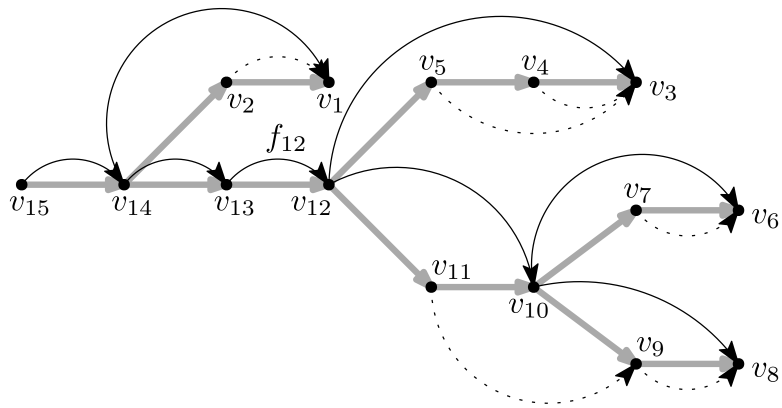

Figure 6: An example of of a basis B that is a 12-nice basis. The grey arcs are the arcs in $\mathcal{T} = (U, \mathcal{A}_0)$ while the black arcs are the arcs in $\mathcal{T}_B = (U, \mathcal{A}_B)$. We note that $\mathcal{T}$ is the depth-first arborescence from Figure 5 and $\mathcal{T}_B$ is associated with a feasible cardinal basis B. The solid circular arcs are associated to variables among B with x-value 1 and dotted circular arcs to variables among B with x-value 0.

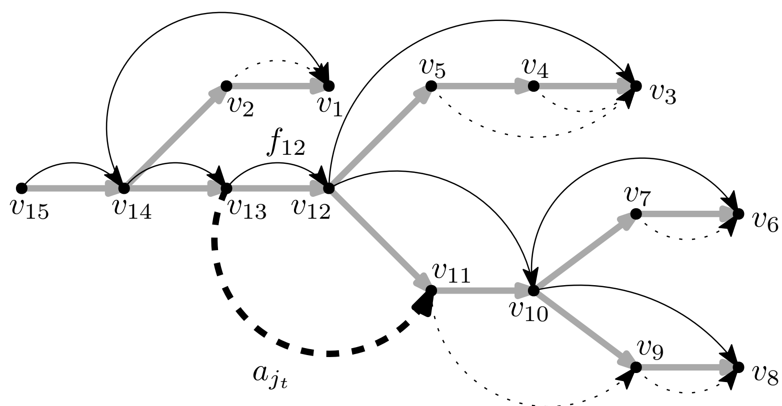

Figure 7: An example of cardinal pivot from the basis B given in Figure 6 with entering arc $a_{j_t} = (v_{13}, v_{11})$ present by a dashed circular arc. This graph is $(U, \mathcal{A}_B \cup \{a_{j_t}\})$. The $\mathcal{T}_B$-path from v_{13} to v_{11} is $P_B(v_{13}, v_{11}) = ((v_{13}, v_{12}), (v_{12}, v_{10}), (v_{10}, v_8), (v_9, v_8), (v_{11}, v_9))$, where the first three arcs are forward and the last two arcs are backward.

by $U = R_i \cup W_i$ the partition of vertices of the two components in $\mathcal{T}^{-f_i} = (U, \mathcal{A}_0 - \{f_i\})$ as defined in Definition 45(iii). Then, we have $w = v_j \in R_i$ and $w' \in W_i$ since $P_0(w, w')$ contains f_i. Therefore, $a_{j^\rightarrow}$ has its end vertices in R_i and W_i. If $j^\rightarrow \in B$, then $a_{j^\rightarrow} = f_i$ by the i-nice property (iii) of B, which contradicts with $e_{j^\rightarrow} \neq e_i$. Hence, $j^\rightarrow = j_t$ since $j^\rightarrow \in O$ and $j^\rightarrow \notin B$. Notice that $a_{j^\rightarrow}$ satisfies the the properties mentioned as conclusions of the lemma, and hence, a_{j_t} also satisfies these properties.

2. Let $P_B(v_j, v_k)$ be the $\mathcal{T}_B$-path from v_j to v_k (notice that $a_{j_t} = (v_j, v_k)$). We first claim that

$$P_B(v_j, v_k) = f_i \oplus P_B(v_i, v_k), \text{ and} \tag{14}$$

$$P_B(v_{n+1}, v_k) = P_B(v_{n+1}, v_j) \oplus P_B(v_j, v_k). \tag{15}$$

In fact, as claimed before, since $v_j \in R_i$ and $v_k \in W_i$, by i-nice property (iii) of B, $P_B(v_j, v_k)$ contains f_i. Also, if f_i is not the first arc in $P_B(v_j, v_k)$, we visit v_j twice through $P_B(v_j, v_k)$, a

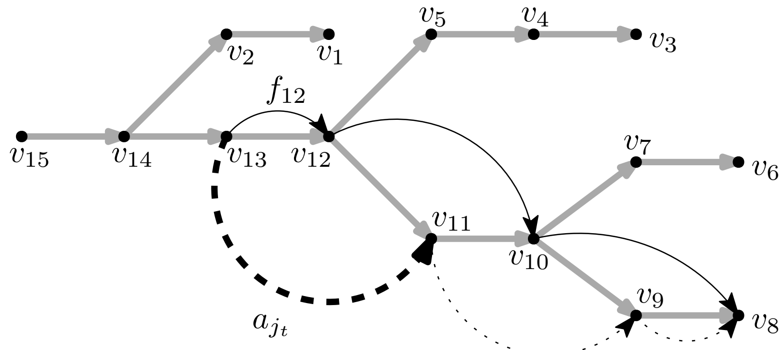

Figure 8: The cycle F created by $a_{j_t} = (v_{13}, v_{11})$ and $P_B(v_{13}, v_{11})$ from Figure 7. We note that f_{12} is the first arc in $P_B(v_{13}, v_{11})$. By the FFL rule, the arc f_{12} will leave $\mathcal{A}_B$.

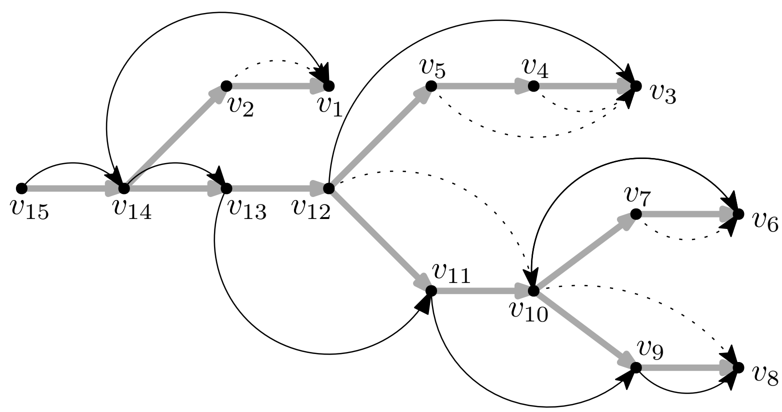

Figure 9: The feasible cardinal basis B' after the cardinal pivot $B \to B'$. We note that the leaving arc $a_{j_\ell} = f_{12}$ is removed from the basis.

contradiction. Therefore, equation (14) holds. Again, since $v_{n+1} \in R_i$ and $v_k \in W_i$, by i-nice property (iii) of B we deduce that the path $P_B(v_{n+1}, v_k)$ passes through $f_i = (v_j, v_i)$. Thus, $P_B(v_{n+1}, v_k) = P_B(v_{n+1}, v_j) \oplus f_i \oplus P_B(v_i, v_k) = P_B(v_{n+1}, v_j) \oplus P_B(v_j, v_k)$, which gives (15). Now, by i-nice property (i) of B, since $P_B(v_{n+1}, v_k)$ is x-augmenting, the subpath $P_B(v_j, v_k)$ is also x-augmenting. By (14), we know that f_i is the first arc on the x-augmenting path $P_B(v_j, v_k)$. By the FFL rule, f_i is the leaving arc. Therefore, $a_{j_\ell} = f_i$.

3. By the conclusion in part 2 of the lemma, the leaving column j_ℓ is i, which corresponds to the singleton e_i. Notice that $i \in B$ and $0 \in B - O$, we have $i \in O$. Thus, $0 \le u_i^O \le c_{i,i} = 0$, which means $u_i^O = c_{i,i}$ and j_ℓ is i-disliked w.r.t. O. By Lemma 34, the separator of (B', O') is i' with some $i' > i$.

4. We verify the i'-nice properties (i)-(iii) of B' in order. By definition, we have $B' = B \cup \{j_t\} \setminus \{j_\ell\}$.

   (i) Let $v \in U \setminus \{v_{n+1}\}$. Let $P_B(v_{n+1}, v)$ and $P_{B'}(v_{n+1}, v)$ be the $\mathcal{T}_B$-path and $\mathcal{T}_{B'}$-path from v_{n+1} to v, respectively. Let $F \subset \mathcal{A}$ be the arcs of the cycle formed by $P_B(v_j, v_k)$ and a_{j_t}, i.e. $a \in F$ iff either $a = a_{j_t}$ or a is in $P_B(v_j, v_k)$. We claim that:

   Claim 47. *If v_p is incident to F, i.e. v_p is an endpoint of some arc in F, then either $v_p = v_j$ or $v_p \in W_i$. Also, $v_j \geq_{\mathcal{T}} v_p$.*

Proof of Claim 47. If v_p is incident to F, then v_p is also visited by $P_B(v_j, v_k)$. By (14) and the i-nice property (iii) of B, $P_B(v_j, v_k)$ traverses from R_i to W_i through f_i, and never go back to R_i as the unique bridge f_i between R_i and W_i can be only used once. Thus, either $v_p = v_j$ or $v_p \in W_i$. If the latter happens, notice that f_i is also the bridge between R_i and W_i on $\mathcal{T}$, the $\mathcal{T}$-directed path $P_0(v_{n+1}, v_p)$ visits v_j, which implies $v_j \geq_{\mathcal{T}} v_p$. ■

By the i-nice property (i) of B and (15), $P_B(v_j, v_i)$ is x-augmenting and by (14), it takes f_i as a forward arc. Thus, $x_{f_i} = 1$. Since $a_{j_\ell} = f_i$ and $f_i \notin \mathcal{A}_{B'}$, we have $x'_{f_i} = 0$. Therefore, by Theorem 41, the cardinal pivot $B \to B'$ is non-degenerate and $P_B(v_j, v_k)$ is x-augmenting and x'-descending. Also, by the equation $x'_B - x_B = A_B^{-1} A_{j_t}(x_{j_t} - x'_{j_t}) = y(-1)$, we have $x'_B = x_B - y$. By Lemma 38, for $s \in [m]$, $x_s \neq x'_s$ if and only if $a_s \in F$ (notice that for the non-basic column $s \notin B$, $x_s = x'_s = 0$).

Consider the directed graph $\mathcal{T}_{B'} = (U, \mathcal{A}_B \cup \{a_{j_t}\} \setminus \{a_{j_\ell}\})$, and let $P_{B'}(v_{n+1}, v)$ be a $\mathcal{T}_{B'}$-path from v_{n+1} to v. If $v \in R_i$, then $P_{B'}(v_{n+1}, v) = P_B(v_{n+1}, v)$, and they do not contain the arcs in F. Indeed, let $v \in R_i$ and by the i-nice property (iii) of B, if $P_B(v_{n+1}, v)$ visits W_i, then it takes f_i forwardly. However, since the destination $v \in R_i$, $P_B(v_{n+1}, v)$ needs to travel back through f_i backwardly, which implies that $P_B(v_{n+1}, v)$ takes f_i at least twice, which cannot be a path. Therefore, $P_B(v_{n+1}, v)$ never visits W_i and f_i is not part of the path. Thus, $P_B(v_{n+1}, v)$ is an $\mathcal{T}_{B'}$-path from v_{n+1} to v. It is unique since $\mathcal{T}_{B'}$ is a tree. Thus, $P_{B'}(v_{n+1}, v) = P_B(v_{n+1}, v)$. By Claim 47, $P_B(v_{n+1}, v)$ does not intersect with F, otherwise it visits W_i, a contradiction. Thus, $P_{B'}(v_{n+1}, v)$ also does not intersect with F. Notice that $P_B(v_{n+1}, v)$ is x-augmenting, thus $P_{B'}(v_{n+1}, v)$ is x-augmenting, which is also x'-augmenting since $x'_s = x_s$ for $a_s \notin F$. If $v \in W_i$, then by i-nice property (iii) of B, we have $P_B(v_{n+1}, v) = P_B(v_{n+1}, v_j) \oplus f_i \oplus P_B(v_i, v)$. Since $f_i \in F$, $P_B(v_{n+1}, v)$ visits the arc(s) in F and begins with f_i, as $P_B(v_{n+1}, v_j)$ is fully contained in R_i as we argued above (if we let $v = v_j$). Let v_q be the last vertex incident to F that $P_B(v_{n+1}, v)$ visits, then we have

$$P_B(v_{n+1}, v) = P_B(v_{n+1}, v_j) \oplus f_i \oplus P_B(v_j, v_q) \oplus P_B(v_q, v),$$

$$P_{B'}(v_{n+1}, v) = P_{B'}(v_{n+1}, v_j) \oplus (v_j, v_k) \oplus P_{B'}(v_k, v_q) \oplus P_{B'}(v_q, v),$$

where $a_{j_t} = (v_j, v_k) \in \mathcal{A}_{B'}$. $P_{B'}(v_{n+1}, v_j)$ is x'-augmenting since $P_B(v_{n+1}, v_j)$ is x-augmenting, they are the same path and $x_s = x'_s$ for every a_s in such path. $(v_j, v_k) = a_{j_t}$ is a path with single forward arc such that $x'_{j_t} = 1$, thus is x'-augmenting. $P_{B'}(v_k, v_q)$ is x'-augmenting, since $P_B(v_k, v_q)$ is x'-descending, and every forward (resp. backward) arc on $P_{B'}(v_k, v_q)$ is a backward (resp. forward) arc on $P_B(v_k, v_q)$. $P_{B'}(v_q, v)$ is also x'-augmenting, since the same path $P_B(v_q, v)$ is x-augmenting and $x_s = x'_s$ for every a_s in such path. Therefore, $P_{B'}(v_{n+1}, v)$ is x'-augmenting.

(ii) Notice that $\{f_i, \dots, f_n\} \subset \mathcal{A}_B$ by the i-nice property of B, then $\{f_{i+1}, \dots, f_n\} \subset \mathcal{A}_{B'}$ since $\mathcal{A}_B \setminus \{f_i\} \subset \mathcal{A}_{B'}$ by $a_{j_\ell} = f_i$. Therefore, $\{f_{i'}, f_{i'+1}, \dots, f_n\} \subset \mathcal{A}_{B'}$ as $i' \geq i + 1$.

(iii) Let $f_q \in \{f_{i'}, f_{i'+1}, \dots, f_n\}$. Let $U = R, \cup W$ be the partition of vertices defined in Definition 45(iii). Now, if $a \in \mathcal{A}_{B'}$ crosses between R and W, we claim that $a = f_q$. Indeed, if we also have $a \in \mathcal{A}_B$, then by the i-nice property of B, we know that $a = f_q$ immediately. Otherwise, if $a \notin \mathcal{A}_B$, then $a = B' - B = a_{j_t}$. By $a_{j_t} = (v_j, v_k)$ and the assumption that a corsses between R and W, we have that $v_j \in R$ and $v_k \in W$. Notice that v_j is the tail of $f_i = (v_j, v_i)$. If it holds $v_i \in W$, then $f_i \in \mathcal{A}_B$ also crosses between R and W, which implies $f_i = f_q$, a contradiction with $f_q \in \{f_{i'}, f_{i'+1}, \dots, f_n\}$. Therefore, we have $v_i \in R$. Consider the $\mathcal{T}$-path $P_0(v_j, v_k)$ from v_j to v_k. By the conclusion in part 1 of the lemma, we know that $f_i = (v_j, v_i)$ is the first arc in $P_0(v_j, v_k)$. By the fact that $v_i \in R$ and $v_k \in W$, f_q is part of $P_0(v_k, v_k)$, and appears after f_i. Notice that since $a_{j_t} = (v_j, v_k) \in \mathcal{A}_B$, $P_0(v_j, v_k)$ is a $\mathcal{T}$-directed path, we traverse along $P_0(v_j, v_k)$ through a sequence of vertices with decreasing order with respect to $\geq_{\mathcal{T}}$, which implies that the head of f_i is greater than the head of f_q. Since $\mathcal{T}$ is depth-first, we have $v_i \geq_{\mathcal{T}} v_q$, and thus $i \geq q$, a contradiction with $q \geq i' > i$.

Therefore, B' is an i'-nice basis.

□

Lemma 48. *Scarf's algorithm with FFL rule terminates in at most n iterations.*

Proof. We first verify that the initial bases (B_0, O_0) satisfies the condition of Lemma 46, i.e. if the column $n+1 = O_0 - B_0$ is in i-th block, then B_0 is an i-nice basis.

Indeed, we have $\mathcal{A}_{B_0} = \mathcal{A}_0 = \{f_1, \ldots, f_n\}$ and $x_{f_j} = 1$ for every $j \in [n]$. One can verify that for every $k \in [n]$, B_0 is an k-nice basis, thus in particular B_0 is an i-nice basis.

By Lemma 46, every iteration moves the separator from i to i' with $i' > i$. Inductively applying this argument shows that the index of the separator strictly increases in each iteration. Thus, we have an upper bound of n on the number of iterations. □

We now show that Scarf's algorithm will terminate in polynomial time by using Lemma 48. Remember that each iteration of Scarf's algorithm contains two pivots, cardinal pivot and ordinal pivot, where each single pivot can be achieved in polynomial time [21]. Futhermore, we show the running time is indeed $O(nm)$ stated in Theorem 1.

Proof of Theorem 1. To embed the input into Scarf's algorithm with tuple (A, b, C), we can construct the matrices A, C in time $O(m)$.

The cardinal pivot with the implementation present in Algorithm 2 involves finding a path between two vertices in a given tree, and checking whether such path is x-augmenting or not. Since the length of such path is bounded by n, the running time is $O(n)$, and thus by Lemma 48, the total running time on cardinal pivots is $O(n^2)$.

The ordinal pivot shown in Algorithm 1 needs to find set K where the "candidate" entering column belongs to, which needs at most mn times of comparisons. However, according to Lemma 34 and Lemma 46, for every iteration we change the separator of the Scarf pair, and such ordinal pivots are generally easier. In fact, we can rewrite ordinal pivot (Algorithm 1) in our problem as Algorithm 3. Notice that, as j^* is the maximizer of $c_{0,k}$ for $k \in K$, which is equivalent to say j^* is the smallest index in K, since we have $c_{0,0} > c_{0,1} > \cdots > c_{0,m}$. Therefore, we do not need to find the whole set K, instead we start from the block S_{i+1} and from left to right search if there exists a column $j > j_r$ such that for all $\bar{\imath} \neq 0$, $c_{\bar{\imath},j} > u_{\bar{\imath}}^{O-i}$. Thus, in an iteration when we move the separator from i to i', we only search the columns between the block S_{i+1} and $S_{i'}$, which means at most m columns are searched among all iterations. Since for each searching round we need to compare the n entries between two columns, the total running time for all ordinal pivots is $O(nm)$.

Therefore, the total running time of Scarf's algorithm with our implementation is bounded by $O(nm)$. □

4 Non-integrality of the Fractional Stable Matching Polytope

A classical approach to stable matching problem in graphs is to describe the stable matching polytope [23], that is, find an polyhedral description of the convex hull of all characteristic vectors of stable matchings. This approach has been successful for the classical marriage model [20, 24],

Algorithm 3 Ordinal Pivot with Separator Change

Let (O, i) be the current ordinal basis with separator i, associated with utility vector u^O.
$j_\ell \leftarrow i$ ▷ Leaving column is set to correspond to the arc f_i (Lemma 46(ii))
$i_\ell \leftarrow i$ ▷ The arc f_i is i-disliked w.r.t. O
$j_r \leftarrow \max O$ ▷ Lemma 34(i)
$i_r \leftarrow 0$ ▷ Lemma 34(ii)
if $\{j > j_r : c_{\bar{\imath},j} > u_{\bar{\imath}}^{O-i}, \forall \bar{\imath} \neq 0\} = \emptyset$ **then**
  $O' = O \cup \{0\} \setminus \{j_\ell\}$ is a dominating basis for (A, b, C).
else
  $j^* \leftarrow \min\{j > j_r : c_{\bar{\imath},j} > u_{\bar{\imath}}^{O-i}, \forall \bar{\imath} \neq 0\}$ ▷ Decreasing order $c_{0,0} > c_{0,1} > \cdots > c_{0,m}$
  $O \leftarrow O \cup \{j^*\} \setminus \{j_\ell\}$
  Let j^* belong to the block $S_{i'}$ in matrix C.
  $i \leftarrow i'$
end if

as well as some of its generalizations [10, 11]. We construct a hypergraphic preference system $I = (H = (V, E), \succ)$ where H is an interval hypergraph, on which the above classical approach does not succeed. This result provides evidence that the stable matching problem in arborescence hypergraphic preference system is technically challenging compared to the classical stable matching problem in bipartite graphs.

We first define

$$P(I) = conv\left(\left\{x \in \{0,1\}^E : x \text{ is a stable matching}\right\}\right), \tag{16}$$

and

$$Q(I) = \left\{ x \in \mathbb{R}^E \;\middle|\; \begin{array}{ll} x(\delta(i)) \leq 1, & \forall i \in V, \\ x(e^{\succeq}) \geq 1, & \forall e \in E, \\ 0 \leq x_e \leq 1, & \forall e \in E. \end{array} \right\}, \tag{17}$$

where $\delta(i) = \{e \in E : i \in e\}$ and $e^{\succeq} = \{e' \in E : \exists i \in e, e' \succ_i e\} \cup \{e\}$. We call $P(I)$ the *stable matching polytope* of I and $Q(I)$ the *fractional stable matching polytope* of I. When H is a bipartite graph, it is known that $P(I) = Q(I)$ (see, e.g., [23]). In contrast, we show that this equality fails to hold for hypergraph preference systems where the hypergraph is an interval hypergraph.

Theorem 49. *There is a hypergraphic preference system $I = (H = (V, E), \succ)$ where H is an interval hypergraph, such that the fractional stable matching polytope $Q(I)$ is not integral, thus $P(I) \neq Q(I)$.*

We give the following example that shows Theorem 49.

Example 50. *Let $I = (H = (V, E), \succ)$ where $V = \{1, 2, \ldots, 9\}$, $E = \{e_1, e_2, e_3, f_1, f_2, f_3, g_1, g_2, g_3\}$*

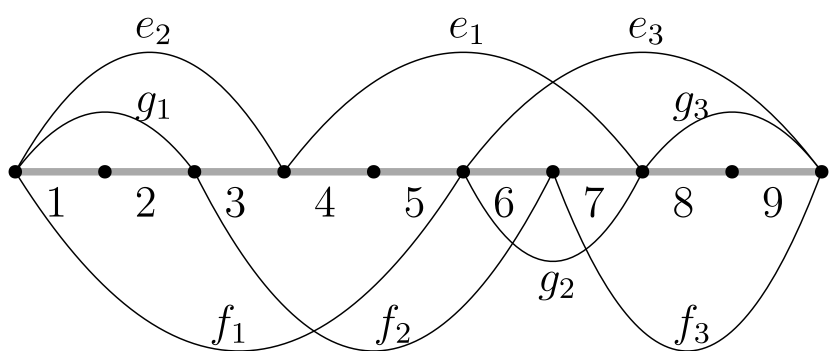

Figure 10: The underlying hypergraph in Example 50. A node belongs to an edge iff the latter covers the former in this figure.

are shown in Figure 10. The preference list is as follows:

$$
\begin{aligned}
&1 : e_2 \succ_1 g_1 \succ_1 f_1,\\
&2 : f_1 \succ_2 e_2 \succ_2 g_1,\\
&3 : f_2 \succ_3 f_1 \succ_3 e_2,\\
&4 : e_1 \succ_4 f_2 \succ_4 f_1,\\
&5 : f_1 \succ_5 e_1 \succ_5 f_2,\\
&6 : e_1 \succ_6 f_2 \succ_6 e_3 \succ_6 g_2,\\
&7 : f_3 \succ_7 e_1 \succ_7 e_3 \succ_7 g_2,\\
&8 : e_3 \succ_8 g_3 \succ_8 f_3,\\
&9 : f_3 \succ_9 e_3 \succ_9 g_3.
\end{aligned}
$$

Consider the solution x where $x(e_1) = x(e_2) = x(e_3) = 0$, $x(f_1) = x(f_2) = x(f_3) = x(g_1) = x(g_2) = x(g_3) = \frac{1}{2}$. We show that x is an extreme point of $Q(I)$.

First, observe that $\{g_1, f_2, f_3\}$ and $\{f_1, g_2, g_3\}$ are two partitions of the interval, thus every $i \in I$ yields $x(\delta(i)) = \frac{1}{2} + \frac{1}{2} = 1$. One can check that the stability constraints are satisfied:

$$
\begin{aligned}
&x(e_1^{\succeq}) = x(e_1) + x(f_1) + x(f_3) = 1,\\
&x(e_2^{\succeq}) = x(e_2) + x(f_1) + x(f_2) = 1,\\
&x(e_3^{\succeq}) = x(e_3) + x(e_1) + x(f_2) + x(f_3) = 1,\\
&x(f_1^{\succeq}) = x(f_1) + x(e_2) + x(g_1) + x(f_2) + x(e_1) = \frac{3}{2} > 1,\\
&x(f_2^{\succeq}) = x(f_2) + x(e_1) + x(f_1) + x(e_1) = 1,\\
&x(f_3^{\succeq}) = x(f_3) + x(e_3) + x(g_3) = 1,\\
&x(g_1^{\succeq}) = x(g_1) + x(e_2) + x(f_1) + x(e_2) = 1,\\
&x(g_2^{\succeq}) = x(g_2) + x(e_1) + x(f_2) + x(e_3) + x(f_3) = \frac{3}{2} > 1,\\
&x(g_3^{\succeq}) = x(g_3) + x(e_3) + x(f_3) = 1.
\end{aligned}
$$

Now, pick the tight constraints corresponding to $e_1^{\succeq}, e_2^{\succeq}, e_3^{\succeq}$, $\delta(1), \delta(6), \delta(9)$ and restrict them to the positive iariables $x(f_i), x(g_i)$ with $i = 1, 2, 3$, we find a linear system

$$
\begin{array}{rccc}
 & \begin{array}{ccccccccc} f_1 & f_2 & f_3 & g_1 & g_2 & g_3 & e_1 & e_2 & e_3 \end{array} & & RHS \\
\begin{array}{r} e_1^{\succ} \\ e_2^{\succ} \\ e_3^{\succ} \\ \delta(1) \\ \delta(6) \\ \delta(9) \\ x(e_1) \\ x(e_2) \\ x(e_3) \end{array} &
\left(\begin{array}{ccccccccc}
1 & & 1 & & & & 1 & & \\
1 & 1 & & & & & & 1 & \\
 & 1 & 1 & & & & 1 & & 1 \\
1 & & & 1 & & & & 1 & \\
 & 1 & & & 1 & & 1 & & 1 \\
 & & 1 & & & 1 & & & 1 \\
\hdashline
 & & & & & & 1 & & \\
 & & & & & & & 1 & \\
 & & & & & & & & 1
\end{array}\right) & x = &
\left(\begin{array}{c} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{array}\right),
\end{array}
$$

where the left-hand-side coefficient matrix has full rank. In fact, it is sufficient to check the upper-left 6×6 *matrix has full rank. Thus, we find* 9 *linearly independent tight constraints that uniquely determine* x*.*

Therefore, x *is indeed an extreme point of* $Q(I)$*, however it is not integral.*

5 Conclusions and Future Work

We showed that Scarf's algorithm converges in polynomial time and returns an integral stable matching on arborescence hypergraphic preference systems. Our result is the first proof of polynomial-time convergence of Scarf's algorithm on hypergraphic stable matching problems. We note that some of our results hold for hypergraphs that are more general than arborescence hypergraphs. We mention three directions for future research: Firstly, it would be interesting to generalize our approach to show polynomial-time convergence of Scarf's algorithm for more general classes of hypergraphs, such as network hypergraphs. Secondly, it would be insightful to find an interpretation as a purely combinatorial algorithm of our implementation of Scarf's algorithm for arborescence hypergraphs, if any such interpretation exists. Thirdly, is it possible to verify whether a given network hypergraph is an arborescence hypergraph in polynomial-time? More generally, is it possible to construct a principal tree associated with a given network hypergraph with the least number of sources? We leave the above as open questions.

Acknowledgments. Yuri Faenza and Chengyue He are supported by the NSF Grant 2046146 and by a Meta Research Award. Karthekeyan Chandrasekaran is supported in part by the NSF grant CCF-2402667. Part of this work was completed while the first three authors were attending the semester program on *Discrete Optimization: Mathematics, Algorithms, and Computation* at ICERM, Brown University, USA.

References

[1] R. Aharoni and T. Fleiner. On a lemma of Scarf. *Journal of Combinatorial Theory, Series B*, 87(1):72–80, 2003.

[2] R. Aharoni and R. Holzman. Fractional kernels in digraphs. *J. Combinatorial Theory*, 73(1):1–6, 1998.

[3] M. Baïou and M. Balinski. The stable admissions polytope. *Mathematical programming*, 87:427–439, 2000.

[4] M. Baïou and M. Balinski. The stable allocation (or ordinal transportation) problem. *Mathematics of Operations Research*, 27(3):485–503, 2002.

[5] P. Biró and T. Fleiner. Fractional solutions for capacitated ntu-games, with applications to stable matchings. *Discrete Optimization*, 22:241–254, 2016.

[6] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to algorithms*. MIT press, 2022.

[7] G. Csáji. On the complexity of stable hypergraph matching, stable multicommodity flow and related problems. *Theoretical Computer Science*, 931:1–16, 2022.

[8] P. Dworczak. Deferred acceptance with compensation chains. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pages 65–66, 2016.

[9] Y. Faenza, C. He, and J. Sethuraman. Scarf's algorithm and stable marriages. *arXiv preprint arXiv:2303.00791*, 2023.

[10] Y. Faenza and X. Zhang. Affinely representable lattices, stable matchings, and choice functions. *Mathematical Programming*, 197(2):721–760, 2023.

[11] T. Fleiner. On the stable b-matching polytope. *Mathematical Social Sciences*, 46(2):149–158, 2003.

[12] D. Gale and L. S. Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962.

[13] P. E. Haxell and G. T. Wilfong. A fractional model of the border gateway protocol (bgp). In *Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 193–199. Citeseer, 2008.

[14] T. Ishizuka and N. Kamiyama. On the complexity of stable fractional hypergraph matching. In *29th International Symposium on Algorithms and Computation (ISAAC 2018)*. Schloss-Dagstuhl-Leibniz Zentrum für Informatik, 2018.

[15] S. Kintali, L. J. Poplawski, R. Rajaraman, R. Sundaram, and S.-H. Teng. Reducibility among fractional stability problems. *SIAM Journal on Computing*, 42(6):2063–2113, 2013.

[16] D. E. Knuth. Marriages stables. *Technical report*, 1976.

[17] C. Ng and D. S. Hirschberg. Three-dimensional stabl matching problems. *SIAM Journal on Discrete Mathematics*, 4(2):245–252, 1991.

[18] A. Pass-Lanneau, A. Igarashi, and F. Meunier. Perfect graphs with polynomially computable kernels. *Discrete Applied Mathematics*, 272:69–74, 2020.

[19] A. E. Roth, U. G. Rothblum, and J. H. Vande Vate. Stable matchings, optimal assignments, and linear programming. *Mathematics of operations research*, 18(4):803–828, 1993.

[20] U. G. Rothblum. Characterization of stable matchings as extreme points of a polytope. *Mathematical Programming*, 54:57–67, 1992.

[21] H. E. Scarf. The core of an n person game. *Econometrica: Journal of the Econometric Society*, pages 50–69, 1967.

[22] A. Schrijver. *Theory of linear and integer programming.* John Wiley & Sons, 1998.

[23] C.-P. Teo and J. Sethuraman. The geometry of fractional stable matchings and its applications. *Mathematics of Operations Research*, 23(4):874–891, 1998.

[24] J. H. V. Vate. Linear programming brings marital bliss. *Operations Research Letters*, 8(3):147–153, 1989.