Enumerating Vertices of 0/1-Polyhedra associated with 0/1-Totally Unimodular Matrices

Khaled Elbassioni
Masdar Institute, Khalifa University of Science and Technology, Abu Dhabi 54224, UAE
khaled.elbassioni@ku.ac.ae

Kazuhisa Makino
Research Institute for Mathematical Sciences (RIMS) Kyoto University, Kyoto 606-8502, Japan
makino@kurims.kyoto-u.ac.jp

Abstract

We give an incremental polynomial time algorithm for enumerating the vertices of any polyhedron $P = P(A, \underline{1}) = \{x \in \mathbb{R}^n \mid Ax \geq \underline{1},\ x \geq \underline{0}\}$, when A is a totally unimodular matrix. Our algorithm is based on decomposing the hypergraph transversal problem for unimodular hypergraphs using Seymour's decomposition of totally unimodular matrices, and may be of independent interest.

2012 ACM Subject Classification Mathematics of computing → Combinatorial algorithms, Mathematics of computing → Hypergraphs

Keywords and phrases Totally unimodular matrices, Vertices of polyhedra, Vertex enumeration, Hypergraph transversals, Hypergraph decomposition, Output polynomial-time algorithm

Digital Object Identifier 10.4230/LIPIcs.SWAT.2018.18

Acknowledgements We thank Endre Boros and Vladimir Gurvich for helpful discussions.

1 Introduction

1.1 The vertex enumeration problem

The well-known Minkowski-Weyl theorem states that any convex polyhedron $\mathcal{P} \subseteq \mathbb{R}^n$ can be represented as the Minkowski sum of the convex hull of the set $\mathcal{V}(\mathcal{P})$ of its extreme points and the conic hull of the set $\mathcal{D}(\mathcal{P})$ of its extreme directions (see e.g. [29]). Given a polyhedron $\mathcal{P}$ by its linear description as the intersection of finitely many halfspaces, obtaining the set $\mathcal{V}(\mathcal{P}) \cup \mathcal{D}(\mathcal{P})$, required by the other representation, is a well-known problem, called *Vertex Enumeration* (VE) (see,. e.g.,[14, 10]), which have been extensively studied in the literature in different (but polynomially equivalent) forms, e.g., , the *facet enumeration* problem [10] or the *polytope-polyhedron problem* [25]. Clearly, the size of the extreme set $\mathcal{V}(\mathcal{P}) \cup \mathcal{D}(\mathcal{P})$ can be (and typically is) exponential in the dimension n and the number of linear inequalities m, and thus when considering the computational complexity of the vertex enumeration problem, one is usually interested in *output-sensitive* algorithms [30], i.e., those whose running time depends not only on n and m, but also on $|\mathcal{V}(\mathcal{P}) \cup \mathcal{D}(\mathcal{P})|$. Alternatively, we may consider the following, polynomially equivalent, decision variant of the problem:

Dec($\mathcal{L}; \mathcal{X} \subseteq \mathcal{C}(\mathcal{P})$): Given a polyhedron $\mathcal{P}$, represented by a system of linear inequalities $\mathcal{L}$, and a subset $\mathcal{X} \subseteq \mathcal{C}(\mathcal{P})$, is $\mathcal{X} = \mathcal{C}(\mathcal{P})$?

In this description, $\mathcal{C}(\mathcal{P})$ could be either $\mathcal{V}(\mathcal{P})$, $\mathcal{D}(\mathcal{P})$, or $\mathcal{V}(P) \cup \mathcal{D}(P)$. The problem of enumerating the elements of $\mathcal{C}(\mathcal{P})$ is said to be solvable in *incremental polynomial* time if problem $\mathrm{Dec}(\mathcal{L}; \mathcal{X} \subseteq \mathcal{C}(\mathcal{P}))$ can be solved in time polynomial in the size of the description of

16th Scandinavian Symposium and Workshops on Algorithm Theory (SWAT 2018).
Editor: David Eppstein; Article No. 18; pp. 18:1–18:14
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

$\mathcal{L}$ and $\mathcal{X}$.[1] It is well-known that if the decision problem is NP-hard, then no *output* (or *total*) *polynomial-time* algorithm can generate the elements of $\mathcal{C}(\mathcal{P})$ unless P=NP (see e.g. [8]).

Vertex enumeration is an outstanding open problem in computational geometry and polyhedral combinatorics (see, e.g., [15, 25, 27]), and has numerous applications. For example, understanding the structure of the vertices helps in designing approximation algorithms for combinatorial optimization problems [33]; finding all vertices can be used for computing Nash equilibria for bimatrix games [5]. Numerous algorithmic ideas for vertex or facet enumeration have been introduced in the literature, see, e.g., [1, 2, 3, 10, 11, 15, 12, 14, 30, 28, 4].

The main result in [21] established that problem $\text{Dec}(\mathcal{L}; \mathcal{X} \subseteq \mathcal{V}(\mathcal{P}))$ is NP-hard for unbounded polyhedra, more precisely, when $|\mathcal{D}(\mathcal{P})|$ is exponentially large in the input size. This negative result holds, even when restricted to 0/1-*polyhedra* [9], that is, when $\mathcal{V}(\mathcal{P}) \subseteq \{0,1\}^n$, and comes in contrast with the fact that the VE problem for 0/1-polytopes (i.e., bounded polyhedra) is known to be solvable with *polynomial delay* (that is, the vertices are generated such that the delay between any successive outputs is polynomial *only* in the *input* size) and *polynomial space* (that is, the total space used for enumerating all the vertices is polynomial in the *input* size).

1.2 VE for 0/1-Polyhedra Associated with 0/1-Totally Unimodular Matrices

Let $A \in \{0,1\}^{m \times n}$ be an $m \times n$ 0/1-matrix such that the polyhedron

$$\mathcal{P}(A, \underline{1}) = \{x \in \mathbb{R}^n \mid Ax \geq \underline{1},\ x \geq \underline{0}\} \tag{1}$$

has only integral vertices, where $\underline{1}$ (resp., $\underline{0}$) denotes the vector of all ones (resp., zeros) of appropriate dimension. Then $\mathcal{P}(A, \underline{1})$ has only n extreme directions (namely the n unit vectors in $\mathbb{R}^n$), while the vertices of P are in one-to-one correspondence with the minimal transversals of the hypergraph $\mathcal{H}[A] \subseteq 2^{[n]}$, whose characteristic vectors of hyperedges are the rows of A. One of the most important examples is when the matrix A is *totally unimodular*: in this case, the polyhedron $\mathcal{P}(A, \underline{1})$ has integral vertices, and VE is equivalent to finding all minimal transversals[2] of a *unimodular hypergraph* $\mathcal{H}[A]$. Consequently, it follows from the well-known result in [19] that all vetrices of such polyhedra can be enumerated in *quasi-polynomial* time, and hence the VE problem in this case is unlikely to be NP-hard. Polynomial time algorithms for special cases of this problem are known; for example, enumerating minimal vertex/edge covers for a bipartite graphs [16, 26], enumerating minimal hitting sets/set covers of interval hypergraphs [8], and enumerating minimal path covers/cut conjunctions in directed trees [8]. However, the complexity of the VE problem for (1) remains open, even for the totally unimodular matrices A. In this paper, we settle the complexity of the VE problem in the latter case.

► **Theorem 1.** *Let $A \in \{0,1\}^{m \times n}$ be a totally unimodular matrix. Then the vertices of $\mathcal{P}(A, \underline{1})$ can be enumerated in incremental polynomial time.*

---

[1] Note that if the answer to the decision problem is "NO" then a new element in $\mathcal{C}(\mathcal{P}) \setminus \mathcal{X}$ can be found by a polynomial number of calls to the decision problem.

[2] Note that, it is not possible to reduce the problem of enumerating the vertices of $\mathcal{P}(A, \underline{1})$ to that of enumerating the vertices of the 0/1 polytope $\mathcal{P}' = \{x \in \mathbb{R}^n \mid Ax \geq \underline{1},\ \underline{0} \leq x \leq \underline{1}\}$, as $\mathcal{P}'$ can have exponentially more vertcies than those of $\mathcal{P}$ (namely, the vertices of $\mathcal{P}'$ are the (not necessarily minimal) transversals of $\mathcal{H}[A]$).

A celebrated result of Seymour [31] shows that any totally unimodular matrix (with 0, ± 1-entries) arises from (essentially) the so-called *network matrices*, by a small set of simple operations. Similar results for 0/1-totally unimodular matrices are derived in [32, Chapter 11], with the main building blocks replaced by 0/1-network matrices. On the other hand, it has been shown in [8] that for any polyhedron $\mathcal{P}(A, \underline{1})$, with a 0/1-network matrix A, the VE problem can be solved in incremental polynomial time. To prove Theorem 1, we show that the above mentioned decomposition of totally unimodular matrices yields a corresponding decomposition for the hypergraph transversal problem, that can be leveraged into a polynomial time algorithm for the problem. One of the natural ways to use such decomposition is to recursively partition the input polyhedron into two smaller polyhedra and then combine the outputs from the two subproblems. While such approach works for the simple cases of the decomposition (so-called 1- and 2-sum decompositions), it does not work for the more complicated case (so-called 3-sum decomposition). The main reason is that the number of vertices in either of the two subproblems may be exponentially larger than that in the original problem. To overcome this difficulty, we need to use the decomposition in a more sophisticated way, utilizing structural properties of the unimodular hypergraph $\mathcal{H}[A]$. On technical hurdle which arises is that the total input/output size of the resulting subproblems might exceed the input/output size of the original problem, which may eventually lead to an exponential blow-up in the overall running time of the algorithm in terms of the input *and* output sizes. To deal with this issue, we introduce a *volume* measure as the product of the input and output sizes, and show in each case of our decomposition that the total measure of the subproblems is smaller than the measure of the original problem.

2 Notation and Preliminaries

2.1 Hypergraphs and Transversals

Let V be a finite set. A hypergraph $\mathcal{H} \subseteq 2^V$ is a family of subsets of V. A hypergraph is called *Sperner* (*simple* or a *clutter*), if it has the property that no hyperedge contains another. For a hypergraph $\mathcal{H} \subseteq 2^V$, we denote by $\mathrm{Tr}(\mathcal{H})$ the family of *minimal transversals* of $\mathcal{H}$, i.e., (inclusion-wise) minimal subsets of V which have a nonempty intersection with each hyperedge of $\mathcal{H}$; $\mathrm{Tr}(\mathcal{H})$ is also called the *dual* of $\mathcal{H}$. Note that for the purpose of enumerating all minimal transversals of a hypergraph $\mathcal{H}$, it is enough to consider the Sperner hypergraph consisting of the minimal hyperedges of $\mathcal{H}$. We say that the hypergraph $\mathcal{H}$ is *trivial* if $\mathcal{H} = \emptyset$ or $\mathcal{H} = \{\emptyset\}$, and is *irredundant* if every $v \in V$ belongs to some $H \in \mathcal{H}$. As usual, we assume $\mathrm{Tr}(\{\emptyset\}) = \emptyset$ and $\mathrm{Tr}(\emptyset) = \{\emptyset\}$.

Given two hypergraphs $\mathcal{H}_1$ and $\mathcal{H}_2$ with vertex set V, denote by

$$\mathcal{H}_1 \wedge \mathcal{H}_2 = \mathrm{Min}\{H_1 \cup H_2 \mid H_1 \in \mathcal{H}_1 \text{ and } H_2 \in \mathcal{H}_2\},$$
$$\mathcal{H}_1 \vee \mathcal{H}_2 = \mathrm{Min}(\mathcal{H}_1 \cup \mathcal{H}_2),$$

the *conjunction* and *disjunction* of $\mathcal{H}_1$ and $\mathcal{H}_2$ respectively, where for hypergraph $\mathcal{H}$, $\mathrm{Min}(\mathcal{H})$ denotes the family of (inclusion-wise) minimal sets in $\mathcal{H}$. We denote by $\mathcal{H}_1 \dot\cup \mathcal{H}_2$ the *disjoint* union of $\mathcal{H}_1$ and $\mathcal{H}_2$. For two hypergraphs $\mathcal{H}_1 \subseteq 2^{V_1}$ and $\mathcal{H}_2 \subseteq 2^{V_2}$, we denote by $\mathcal{H}_1 \dot\wedge \mathcal{H}_2$ the conjunction of $\mathcal{H}_1$ and $\mathcal{H}_2$ when V_1 and V_2 are *disjoint*. By definition, $|\mathcal{H}_1 \dot\cup \mathcal{H}_2| = |\mathcal{H}_1| + |\mathcal{H}_2|$ and $|\mathcal{H}_1 \dot\wedge \mathcal{H}_2| = |\mathcal{H}_1| \cdot |\mathcal{H}_2|$.

For a hypergraph $\mathcal{H} \subseteq 2^V$ and a set $S \subseteq V$, we denote by $\mathcal{H}_S = \{H \in \mathcal{H} \mid H \subseteq S\}$ and $\mathcal{H}^S = \mathrm{Min}\{H \cap S \mid H \in \mathcal{H}\}$ the subhypergraph of $\mathcal{H}$ *induced by* S, and the *projection* of $\mathcal{H}$ on S, respectively. For $W, S \subseteq V$, we write $\mathcal{H}(W, S) = \{H \in \mathcal{H} \mid H \cap W = S\}$. Two vertices of $\mathcal{H}$ are said to be *identical* if they belong to exactly the same hyperedges, i.e., the corresponding columns in the hyperedge-vertex incidence matrix are identical.

The following propositions are straightforward (see e.g. [6, 17, 23]).

▶ **Proposition 2.** *Given a hypergraph $\mathcal{H} \subseteq 2^V$ and a set $S \subseteq V$, the following statements hold:*

(i) $\mathrm{Tr}(\mathrm{Tr}(\mathcal{H})) = \mathrm{Min}(\mathcal{H})$,

(ii) $\mathrm{Tr}(\mathcal{H}_S) = \mathrm{Tr}(\mathcal{H})^S$ *(and hence,* $\mathrm{Tr}(\mathcal{H}^S) = \mathrm{Tr}(\mathcal{H})_S$*) and*

(iii) $|\mathrm{Tr}(\mathcal{H}_S)| \leq |\mathrm{Tr}(\mathcal{H})|$.

▶ **Proposition 3.** *Given hypergraphs $\mathcal{H}_1, \ldots, \mathcal{H}_k \subseteq 2^V$,* $\mathrm{Tr}\left(\bigvee_{i=1}^r \mathcal{H}_i\right) = \bigwedge_{i=1}^r \mathrm{Tr}(\mathcal{H}_i)$.

As a corollary of Proposition 3 we have the following.

▶ **Proposition 4.** *Let $\mathcal{H} \subseteq 2^V$ be a hypergraph and $S_1, \ldots, S_r \subseteq V$ be subsets such that for every hyperhedge $H \in \mathcal{H}$ there exists an $i \in \{1, \ldots, r\}$ with $H \subseteq S_i$. Then* $\mathrm{Tr}(\mathcal{H}) = \bigwedge_{i=1}^r \mathrm{Tr}(\mathcal{H}_{S_i})$.

Throughout the paper, we use the notation: $n = n(\mathcal{H}) = |V|$, $m = m(\mathcal{H}) = |\mathcal{H}|$ and $k = k(\mathcal{H}) = |\mathrm{Tr}(\mathcal{H})|$.

2.2 Polyhedra

A convex polyhedron $P \subseteq \mathbb{R}^n$ is the intersection of finitely many halfspaces, determined by the *facets* of the polyhedron. A *vertex* or an *extreme point* of P is a point $v \in \mathbb{R}^n$ which cannot be represented as a convex combination of two other points of P, i.e., there exists no $\lambda \in (0, 1)$ and $v_1, v_2 \in P$ such that $v = \lambda v_1 + (1 - \lambda) v_2$. A *(recession) direction* of P is a vector $d \in \mathbb{R}^n$ such that $x_0 + \mu d \in P$ whenever $x_0 \in P$ and $\mu \geq 0$. An *extreme direction* of P is a direction d that cannot be written as a conic combination of two other directions, i.e., there exist no positive real numbers $\mu_1, \mu_2 \in \mathbb{R}_+$ and directions d_1, d_2 of P such that $d = \mu_1 d_1 + \mu_2 d_2$. Denote respectively by $\mathcal{V}(P)$ and $\mathcal{D}(P)$ the sets of extreme points and extreme directions of polyhedron P. A bounded polyhedron, i.e., one for which $\mathcal{D}(P) = \emptyset$ is called a *polytope*.

2.3 Totally Unimodular Matrices

A matrix $A \in \{0, 1\}^{m \times n}$ is *totally unimodular* if every square subdeterminant of it has value in $\{-1, 0, 1\}$. We denote by $\mathcal{U}^{m \times n}$ the set of $m \times n$ 0/1-totally unimodular matrices. For a matrix $A \in \{0, 1\}^{m \times n}$ we denote by $\mathcal{H}[A] \subseteq 2^{[n]}$ the hypergraph whose characteristic vectors of hyperedges are the rows of A. A hypergraph $\mathcal{H}$ is said to be *unimodular* [6] if $\mathcal{H} = \mathcal{H}[A]$ for a totally unimodular matrix A. Note by definition that if $\mathcal{H} \subseteq 2^V$ is unimodular then for any set $S \subseteq V$ and any subhypergraph $\mathcal{H}' \subseteq \mathcal{H}$, the hypergraph $(\mathcal{H}')^S$ is unimodular. A 0/1 matrix is said to be *ideal* (see, e.g., [13]) if the polyhedron $P = P(A, \underline{1})$ has only integral vertices. It is well-known that every totally unimodular matrix $A \in \{0, 1\}^{m \times n}$ is ideal. Furthermore, the following correspondence holds.

▶ **Proposition 5** ([24])**.** *Let A be an $m \times n$ ideal matrix. Then the vertices of the polyhedron $\mathcal{P}(A, \underline{1})$ are in one-to-one correspondence with the minimal transversals of the hypergraph $\mathcal{H}[A]$.*

2.4 0/1-Network matrices

A matrix $A \in \{0,1\}^{m \times n}$ is said to be a *network* matrix if there exists a directed tree[3] T such that the rows of A one-to-one correspond to the arcs in T, and each column of A is the characteristic vector of a directed path in T. Checking if a given matrix A is a network matrix and finding the corresponding tree representation can be done in polynomial time (see e.g., [29]). We call a hypergraph $\mathcal{H}$ a *network hypergraph* if $\mathcal{H} = \mathcal{H}[A]$ for some network matrix A or its transpose. It is known that network hypergraphs can be dualized in incremental polynomial time and polynomial space:

▶ **Theorem 6** ([8]). *Let $A \in \{0,1\}^{m \times n}$ be a network matrix. Then*
- **(i)** *all the vertices of $P(A, \underline{1})$ can be enumerated in incremental polynomial time using polynomial space;*
- **(ii)** *all the vertices of $P(A^T, \underline{1})$ can be enumerated in incremental polynomial time using polynomial space.*

2.5 Decomposition of 0/1-totally unimodular matrices

Seymour [31] gave a decomposition theorem that allows one to decompose (in polynomial time) any 0/1-totally unimodular matrix by repeatedly applying certain operations (called *i-sums*, for $i = 1, 2, 3$) until simple building blocks are obtained; the building blocks consist of 0/1-network matrices, their transposes and a specific 5×5 0/1-matrix. For our purposes this theorem can be stated as follows.

▶ **Theorem 7** (Decomposition of unimodular hyeprgraphs). *Let $\mathcal{H} \subseteq 2^V$ be a unimodular (nontrivial) irredundant Sperner hypergraph. Then $\mathcal{H}$ is a network hypergraph, (isomorphic to) the hypergraph $\mathcal{H}_0 = \{\{1,4,5\},\{1,2,5\},\{2,3,5\},\{3,4,5\}\}$, has two identical vertices, a hyperedge consisting of a singleton, or a vertex with degree 1, or there exists a nontrivial partition $V_1 \dot\cup V_2 = V$ such that $\mathcal{H}$ can be decomposed as follows:*

- 1*-sum decomposition:*
  - **(i)** $\mathcal{H}_{V_1} \neq \emptyset$, $\mathcal{H}_{V_2} \neq \emptyset$;
  - **(ii)** *for all $H \in \mathcal{H}$: either $H \subseteq V_1$ or $H \subseteq V_2$;*
- 2*-sum decomposition: there exists a nonempty set $S \subseteq V_1$ such that*
  - **(i)** $\mathcal{H}_{V_1} \neq \emptyset$, $\mathcal{H}(V_1, S) \neq \emptyset$, $\mathcal{H}(V_1, S)^{V_2} \neq \{\emptyset\}$;
  - **(ii)** *for all $H \in \mathcal{H}$ with $H \cap V_1 \neq \emptyset$ and $H \cap V_2 \neq \emptyset$: $H \cap V_1 = S$;*
- 3*-sum decomposition – case 1: there exist two nonempty sets $S_1 \subseteq V_1$ and $S_2 \subseteq V_2$, such that*
  - **(i)** $\mathcal{H}(V_1, S_1) \neq \emptyset$, $\mathcal{H}(V_1, S_1)^{V_2} \neq \{\emptyset\}$, $\mathcal{H}(V_2, S_2) \neq \emptyset$, $\mathcal{H}(V_2, S_2)^{V_1} \neq \{\emptyset\}$;
  - **(ii)** $|V_1| + |\mathcal{H}_{V_1} \cup \mathcal{H}(V_2, S_2)| \geq 4$, $|V_2| + |\mathcal{H}_{V_2} \cup \mathcal{H}(V_1, S_1)| \geq 4$;
  - **(iii)** *for all $H \in \mathcal{H}$ with $H \cap V_1 \neq \emptyset$ and $H \cap V_2 \neq \emptyset$: either $H \cap V_1 = S_1$ or $H \cap V_2 = S_2$;*
- 3*-sum decomposition – case 2: there exist three nonempty disjoint sets $S_0, S_1, S_2 \subseteq V_1$, such that*
  - **(i)** $\mathcal{H}_{V_1} \neq \emptyset$, $\mathcal{H}(V_1, S_0 \cup S_1) \neq \emptyset$, $\mathcal{H}(V_1, S_0 \cup S_1)^{V_2} \neq \{\emptyset\}$, $\mathcal{H}(V_1, S_0 \cup S_2) \neq \emptyset$, $\mathcal{H}(V_1, S_0 \cup S_2)^{V_2} \neq \{\emptyset\}$;
  - **(ii)** *for all $H \in \mathcal{H}$ with $H \cap V_1 \neq \emptyset$ and $H \cap V_2 \neq \emptyset$: either $H \cap V_1 = S_0 \cup S_1$, or $H \cap V_1 = S_0 \cup S_2$;*

[3] We say that a directed graph G is a *directed tree* if the underlying graph of G (i.e., the undirected graph obtained from G by ignoring orientation of arcs) is a tree.

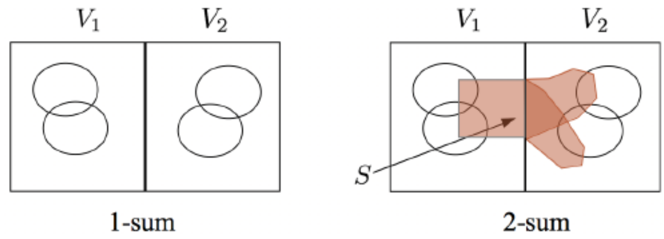

Figure 1 Decomposing a unimodular hypergraph: 1 and 2-sums.

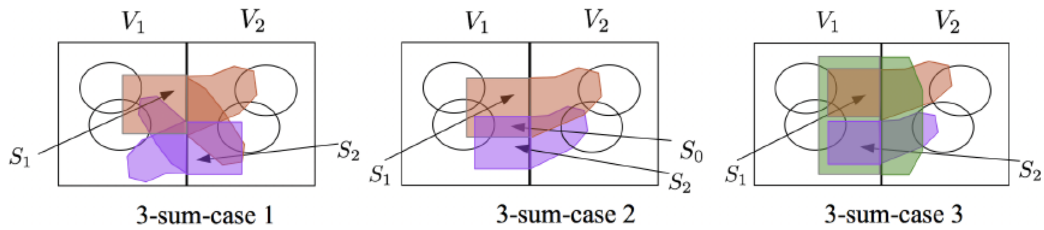

Figure 2 Decomposing a unimodular hypergraph: 3-sum.

- *3-sum decomposition – case 3: there exist two nonempty disjoint sets $S_1, S_2 \subseteq V_1$, such that*
  (i) *$\mathcal{H}_{V_1} \neq \emptyset$ and at least two of the following three conditions hold: (1) $\mathcal{H}(V_1, S_1) \neq \emptyset$ and $\mathcal{H}(V_1, S_1)^{V_2} \neq \{\emptyset\}$, (2) $\mathcal{H}(V_1, S_2) \neq \emptyset$, $\mathcal{H}(V_1, S_2)^{V_2} \neq \{\emptyset\}$, (3) $\mathcal{H}(V_1, S_1 \cup S_2) \neq \emptyset$, $\mathcal{H}(V_1, S_1 \cup S_2)^{V_2} \neq \{\emptyset\}$;*
  (ii) *$|V_1| + |\mathcal{H}_{V_1}| \geq 4$, $|V_2| + |\mathcal{H}_{V_2} \cup \mathcal{H}(V_1, S_1) \cup \mathcal{H}(V_1, S_2) \cup \mathcal{H}(V_1, S_1 \cup S_2)| \geq 4$;*
  (iii) *for all $H \in \mathcal{H}$ with $H \cap V_1 \neq \emptyset$ and $H \cap V_2 \neq \emptyset$: either $H \cap V_1 = S_1$, $H \cap V_1 = S_2$, or $H \cap V_1 = S_1 \cup S_2$.*

Discovering if $\mathcal{H}$ is a network hypergraph, or isomorphic to $\mathcal{H}_0$, and if not finding a decomposition as above can be done in polynomial time.

A schematic illustration of these decomposition rules is given in Figures 1 and 2.

▶ Remark. We note that the boundary condition (ii) in the 3-sum–case 1 is essential, since without insisting on this condition, any hypergraph can be decomposed according to the 3-sum–case 1 rule (take any $v \in V$ and $H \in \mathcal{H}$ such that $v \in H$ and let $V_1 = S_1 = \{v\}$, $V_2 = V \setminus \{v\}$ and $S_2 = H \setminus \{v\}$). Similarly, our analysis in the 3-sum–case 3 uses condition (ii). However, a similar condition is not needed for all other cases.

3 Decomposition of the hypergraph transversal problem

In the following, we show how to decompose the hypergraph transversal problem for a unimodular hypergraph $\mathcal{H}$, given the decomposition of $\mathcal{H}$ in Theorem 7. Such a decomposition yields naturally a recursive algorithm: each non-leaf node of the recursion tree is responsible for computing the dual of a unimodular hypergraph, while leaves involve the computation of the dual of a network hypergraph or the hypergraph $\mathcal{H}_0$. To ensure that the overall running time is polynomial, we need to bound the number of nodes of the recursion tree and the local computation time at each node, which consists of the time required for computing the decomposition and the time for combining the outputs from the recursive calls into the final output at the node. We will measure the "volume" of each subproblem to compute $\mathrm{Tr}(\mathcal{H})$

by $\mu(\mathcal{H}) = nmk = n(\mathcal{H})m(\mathcal{H})k(\mathcal{H})$. We let $T(\mu)$ be the number of nodes of the recursion subtree rooted at a node of volume μ, and let $L_1(\mu)$ and $L_2(\mu)$ be respectively the local computation time for the decomposition and combining the outputs at a node of volume μ. We stop the recursion when either $m = m(\mathcal{H})$, $n = n(\mathcal{H})$ or $k = k(\mathcal{H})$ drops below some constant C, in which case the hypergraph transversal problem can be solved in $\text{poly}(n, m)$ time using a simple procedures, such as *Berge Multiplication* [6, Chapter 2] for $n(\mathcal{H}) \leq C$ or $m(\mathcal{H}) \leq C$, and the methods in [7, 20] for $k(\mathcal{H}) \leq C$ which also show that the condition $k(\mathcal{H}) \leq C$ can be checked in $\text{poly}(n, m)$ time.

We will show by induction (on $\mu \geq 1$) that $T(\mu) \leq \mu$. We also show that $L_2(\mu) = O(\mu^c)$ for some constant $c \geq 1$. Since $L_1(\mu) = \text{poly}(n, m)$ [29, Chapter 20], it would follow then that the total time to compute $\text{Tr}(\mathcal{H})$ is at most $O(\mu^{1+c}) + \text{poly}(\mu)$, which is polynomial in n, m, and k. This would give a *total* polynomial-time algorithm for computing $\text{Tr}(\mathcal{H})$ which can be converted into an *incremental* polynomial-time algorithm by standard methods [22, 8]. Thus, we shall assume in the sequel that n, m, k are larger than any desired constant C.

Without loss of generality we assume that the input hypergraph is Sperner and irredundant, and this assumption is maintained for all hypergraphs arising as inputs to the recursive subproblems. We may also assume that $\mathcal{H}$ has neither singleton hyperedge nor vertex of degree 1 (i.e., contained in exactly one hyperedge). Indeed, if $\mathcal{H}$ contains a singleton hyperedge $H = \{v\}$, then by the Sperner property, no other hyperedge of $\mathcal{H}$ contains v. In this case, and also in the case when $\mathcal{H}$ has a vertex v contained exactly in one hyperedge $H \in \mathcal{H}$, $\text{Tr}(\mathcal{H})$ can be computed as follows:

$$\text{Tr}(\mathcal{H}) = \text{Tr}((\mathcal{H} \setminus \{H\}) \,\dot\cup\, \{H\}) = \text{Tr}((\mathcal{H} \setminus \{H\}) \wedge \text{Tr}(\{H\}), \tag{2}$$

where $\text{Tr}(\{H\} = \{\{w\} \mid w \in H\}$. By Proposition 2 (iii), $|\text{Tr}((\mathcal{H} \setminus \{H\})| \leq k(\mathcal{H})$ and thus, $\mu(\mathcal{H}') \leq (n-1)(m-1)k \leq \mu(\mathcal{H}) - 1$, where $\mathcal{H}' := \mathcal{H} \setminus \{H\}$ is the the subhypergraph induced by $V \setminus \{v\}$. Thus, we get by induction that $T(\mu(\mathcal{H})) \leq 1 + T(\mu(\mathcal{H}')) \leq \mu(\mathcal{H})$. Moreover, by (2), $\text{Tr}(\mathcal{H})$ can be computed from $\text{Tr}((\mathcal{H} \setminus \{H\})$ in $\text{poly}(n, m, k)$ time.

Finally, we may also assume that $\mathcal{H}$ does not have two identical vertices. Indeed, if it has two such vertices v, v' then we can reduce the problem by calling the algorithm on the hypergraph $\mathcal{H}' = \{H \setminus \{v'\} \mid H \in \mathcal{H}\}$ instead of $\mathcal{H}$. Then the dual of $\mathcal{H}$ can be obtained as follows:

$$\text{Tr}(\mathcal{H}) = \text{Tr}(\mathcal{H}') \dot\cup \{(T \setminus \{v\} \cup \{v'\}) \mid T \in \text{Tr}(\mathcal{H}'),\ v \in T\}. \tag{3}$$

Note that (3) implies that $k(\mathcal{H}') \leq k(\mathcal{H})$ and hence $\mu(\mathcal{H}') \leq (n-1)mk \leq \mu(\mathcal{H}) - 1$. Thus, in this case, we get the recurrence $T(\mu) \leq 1 + T(\mu - 1)$, which gives by induction on $\mu \geq 1$ that $T(\mu) \leq 1 + (\mu - 1) \leq \mu$. Moreover, by (3), $\text{Tr}(\mathcal{H})$ can be computed from $\text{Tr}(\mathcal{H}')$ in $\text{poly}(n, m, k)$ time.

The 1-sum decomposition case is straightforward and is omitted. As a warm-up, we present the 2-sum case in Section 3.1, and then move to the more complicated 3-sum decomposition. In fact, by Theorem 7 the latter can be divided into 3 subcases: we present 3-sum-case 1 in Section 3.2. Then for each of the 3-sum cases 2 and 3, we can identify 4 subcases. Due to space limitations, we present only the simplest cases (3-sum case 2-I and 3-sum case 2-II) in Section 3.3 and refer the reader to [18] for the remaining cases.

We will use the following simple facts in our analysis of the running time of the algorithm.

▶ **Fact 8.** *Let α, β, N, M be positive integers such that $\alpha \leq N/2$ and $\beta \leq M/2$. Consider the maximization problem:*

$$\begin{aligned} z^* = \max \quad & x_1 y_1 + x_2 y_2 \\ \text{s.t.} \quad & x_1 + x_2 \leq N, \\ & y_1 + y_2 \leq M, \\ & x_1, x_2 \geq \alpha, \\ & y_1, y_2 \geq \beta, \\ & x_1, x_2, y_1, y_2 \in \mathbb{Z}. \end{aligned}$$

Then $z^* = \alpha\beta + (N - \alpha)(M - \beta)$.

Proof. Let $(x_1^*, x_2^*, y_1^*, y_2^*)$ be an optimal solution. Clearly, $x_2^* = N - x_1^*$ and $y_2^* = M - y_1^*$. Without loss of generality assume that $x_1^* \geq \frac{N}{2}$. If $y_1^* < M - \beta$, then $(x_1^*, N - x_1^*, y_1^* + 1, M - y_1^* - 1)$ is also an optimal solution since

$$\begin{aligned} x_1^*(y_1^* + 1) + (N - x_1^*)(M - (y_1^* + 1)) &= x_1^* y_1^* + (N - x_1^*)(M - y_1^*) + 2x_1^* - N \\ &\geq x_1^* y_1^* + (N - x_1^*)(M - y_1^*). \end{aligned}$$

Thus we conclude in this case that $(x_1^*, N - x_1^*, M - \beta, \beta)$ is also an optimal solution. A symmetric argument shows that $(N - \alpha, \alpha, M - \beta, \beta)$ is an optimal solution of the maximization problem. ◀

▶ **Fact 9.** *Let x_i, y_i, for $i = 1, \ldots, h$, and M be positive integers such that $\sum_{i=1}^h x_i y_i \leq M$. Then $\sum_{i=1}^h (x_i + y_i) \leq M + h$.*

Proof. For $i = 1, \ldots, h$, let $\alpha_i = x_i y_i$. Note that the function $f(x_i) = x_i + \frac{\alpha}{x_i}$ is convex in $x_i > 0$, and hence $\max\{x_i + y_i \mid x_i y_i = \alpha_i,\ x_i \geq 1, y_i \geq 1\}$ is achieved at the boundary $(x_i, y_i) = (1, \alpha_i)$ or $(x_i, y_i) = (\alpha_i, 1)$. The claim follows by summing the inequality $x_i + y_i \leq \alpha_i + 1$ over $i = 1, \ldots, h$. ◀

3.1 2-sum decomposition

Given a nontrivial partition $V_1 \dot\cup V_2 = V$ and a nonempty set $S \subseteq V_1$ such that for all $H \in \mathcal{H}$ with $H \cap V_1 \neq \emptyset$ and $H \cap V_2 \neq \emptyset$: $H \cap V_1 = S$, we have the following decomposition of the dual hypergraph by Proposition 3:

$$\operatorname{Tr}(\mathcal{H}) = \operatorname{Tr}(\mathcal{H}_{V_1} \dot\cup \mathcal{H}_{V_2 \cup S}) = \operatorname{Tr}(\mathcal{H}_{V_1}) \wedge \operatorname{Tr}(\mathcal{H}_{V_2 \cup S}), \tag{4}$$

as $\mathcal{H} = \mathcal{H}_{V_1} \dot\cup \mathcal{H}_{V_2 \cup S}$ (note that $\mathcal{H}_{V_1} \cap \mathcal{H}_{V_2 \cup S} = \emptyset$ since $\mathcal{H}$ is Sperner and that both $\mathcal{H}_{V_1}$ and $\mathcal{H}_{V_2 \cup S}$ are unimodular). Thus in this case we get the recurrence:

$$T(\mu) \leq 1 + T(\mu_1) + T(\mu_2), \tag{5}$$

where $\mu = \mu(\mathcal{H})$, $\mu_1 = \mu(\mathcal{H}_{V_1})$, and $\mu_2 = \mu(\mathcal{H}_{V_2 \cup S})$. Let $n_1 = n(\mathcal{H}_{V_1}) = |V_1|$, $m_1 = |\mathcal{H}_{V_1}|$, $k_1 = |\operatorname{Tr}(\mathcal{H}_{V_1})|$, $n_2 = n(\mathcal{H}_{V_2 \cup S}) = |V_2| + |S|$, $m_2 = |\mathcal{H}_{V_2 \cup S}|$, and $k_2 = |\operatorname{Tr}(\mathcal{H}_{V_2 \cup S})|$. Note that $n_1, n_2, m_1, m_2 \geq 1$ by the assumptions of the 2-sum case (Theorem 7) and hence $\mu_1, \mu_2 \geq 1$. Then

$$\mu_1 + \mu_2 = n_1 m_1 k_1 + n_2 m_2 k_2 \leq (n-1)(m_1 + m_2)k = (n-1)mk \leq \mu - 1, \tag{6}$$

where $k_1 \leq k$ and $k_2 \leq k$ by Proposition 2 (iii). It follows by induction from (5) that

$$T(\mu) \leq 1 + \mu_1 + \mu_2 \leq 1 + (\mu - 1) = \mu. \tag{7}$$

Note that $\mathrm{Tr}(\mathcal{H})$ can be computed from $\mathrm{Tr}(\mathcal{H}_{V_1})$ and $\mathrm{Tr}(\mathcal{H}_{V_2 \cup S})$ using (4) in time $L_2(\mu) = \mathrm{poly}(n, m, k)$.

3.2 3-sum decomposition - case 1

Assume we are given a nontrivial partition $V_1 \dot\cup V_2 = V$ and two nonempty sets $S_1 \subseteq V_1$ and $S_2 \subseteq V_2$, such that for all $H \in \mathcal{H}$ with $H \cap V_1 \neq \emptyset$ and $H \cap V_2 \neq \emptyset$: either $H \cap V_1 = S_1$ or $H \cap V_2 = S_2$. Let $n_1 = |V_1|$, $n_2 = |V_2|$, $m_1 = |\mathcal{H}_{V_1} \cup \mathcal{H}(V_2, S_2)|$ and $m_2 = |\mathcal{H}_{V_2} \cup \mathcal{H}(V_1, S_1)|$. It is also assumed in this case that $n_1, n_2, m_1, m_2 \geq 1$, $n_1 + n_2 = n$, $m_1 + m_2 = m$, $n_1 + m_1 \geq 4$ and $n_2 + m_2 \geq 4$.

We consider two cases:

Case I: there is no hyperedge $H \in \mathcal{H}$ such that $H \subseteq S_1 \cup S_2$. Note that this, together with assumption (i) of the 3-sum–case 1 in Theorem 7, implies that $S_1 \subset V_1$ and $S_2 \subset V_2$. In this case, we have the following decomposition of the dual hypergraph:

$$\mathrm{Tr}(\mathcal{H}) = \mathrm{Tr}(\mathcal{H}_{V_1 \cup S_2} \dot\cup \mathcal{H}_{V_2 \cup S_1}) = \mathrm{Tr}(\mathcal{H}_{V_1 \cup S_2}) \wedge \mathrm{Tr}(\mathcal{H}_{V_2 \cup S_1}), \tag{8}$$

(Note by assumption that $\mathcal{H}_{V_1 \cup S_2} \cap \mathcal{H}_{V_2 \cup S_1} = \emptyset$.) Thus in this case we get the recurrence:

$$T(\mu) \leq 1 + T(\mu_1) + T(\mu_2), \tag{9}$$

where $\mu = \mu(\mathcal{H})$, $\mu_1 = \mu(\mathcal{H}_{V_1 \cup S_2})$, and $\mu_2 = \mu(\mathcal{H}_{V_2 \cup S_1})$. Let $n_1' = n(\mathcal{H}_{V_1 \cup S_2}) = |V_1| + |S_2|$, $m_1 = |\mathcal{H}_{V_1 \cup S_2}|$, $k_1 = |\mathrm{Tr}(\mathcal{H}_{V_1 \cup S_2})|$, $n_2' = n(\mathcal{H}_{V_2 \cup S_1}) = |V_2| + |S_1|$, $m_2 = |\mathcal{H}_{V_2 \cup S_1}|$, and $k_2 = |\mathrm{Tr}(\mathcal{H}_{V_2 \cup S_1})|$. Then

$$\mu_1 + \mu_2 = n_1' m_1 k_1 + n_2' m_2 k_2 \leq (n-1)(m_1 + m_2)k \leq \mu - 1, \tag{10}$$

where $k_1 \leq k$ and $k_2 \leq k$ by Proposition 2 (iii). It follows by induction from (9) that

$$T(\mu) \leq 1 + \mu_1 + \mu_2 \leq \mu. \tag{11}$$

Note that $\mathrm{Tr}(\mathcal{H})$ can be computed from $\mathrm{Tr}(\mathcal{H}_{V_1 \cup S_2})$ and $\mathrm{Tr}(\mathcal{H}_{V_2 \cup S_1})$ using (8) in time $L_2(\mu) = \mathrm{poly}(n, m, k)$.

Case II: there is a hyperedge $H_0 \in \mathcal{H}$ such that $H_0 \subseteq S_1 \cup S_2$. Note that $H_0 \cap S_1 \neq \emptyset$ and $H_0 \cap S_2 \neq \emptyset$ since otherwise by the simplicity of $\mathcal{H}$ we are in the 2-sum case. Without loss of generality, assume that $H_0 \cap V_1 = S_1$ and $H_0 \cap V_2 \subseteq S_2$. We assume that $\mathcal{H}(V_1, S_1)$ and $\mathcal{H}(V_2, S_2)$ are not empty; otherwise, we are in the 1-sum or 2-sum case. Given these assumptions, we use the following decomposition of the dual hypergraph:

$$\mathrm{Tr}(\mathcal{H}) = \mathrm{Tr}(\mathcal{H}_1 \dot\cup \mathcal{H}_2) = \mathrm{Tr}(\mathcal{H}_1) \wedge \mathrm{Tr}(\mathcal{H}_2), \tag{12}$$

where $\mathcal{H}_1 = \mathcal{H}_{V_1} \cup \mathcal{H}(V_2, S_2) \cup \{H_0\}$ and $\mathcal{H}_2 = \mathcal{H}_{V_2} \dot\cup \mathcal{H}(V_1, S_1)$. Note that H_0 is the the only hyperedge that belongs to both $\mathcal{H}_1$ and $\mathcal{H}_2$. Note also that neither $\mathcal{H}_1$ nor $\mathcal{H}_2$ may be an induced subhypergraph of $\mathcal{H}$ (i.e., the form $\mathcal{H}_S$ for some $S \subseteq V$) since there are hyperedges $H \subseteq S_1 \cup S_2$ that may not be included in $\mathcal{H}_1$ and $\mathcal{H}_2$. Hence, Proposition 2 cannot be used to bound the sizes of $\mathrm{Tr}(\mathcal{H}_1)$ and $\mathrm{Tr}(\mathcal{H}_2)$. Nevertheless, due to the special structure of the decomposition in this case, we can use the bounds given in Lemma 11 below instead. Let $\bar{\mathcal{H}}_1 \subseteq 2^{V_1 \cup \{v_2\}}$ (resp., $\bar{\mathcal{H}}_2 \subseteq 2^{V_2 \cup \{v_1\}}$) be the hypergraph obtained from $\mathcal{H}_1$ (resp., $\mathcal{H}_2$) by replacing S_2 (resp., S_1) by a *new* single vertex v_2 (resp., v_1), that is,

$$\bar{\mathcal{H}}_1 = \mathcal{H}_{V_1} \cup \bar{\mathcal{H}}(V_2, S_2) \cup \{\bar{H}_0\}, \qquad \bar{\mathcal{H}}_2 = \mathcal{H}_{V_2} \cup \bar{\mathcal{H}}(V_1, S_1),$$

where $\bar{\mathcal{H}}(V_2, S_2) = \{(H \setminus S_2) \cup \{v_2\} \mid H \in \mathcal{H}(V_2, S_2)\}$, $\bar{H}_0 = (H_0 \setminus S_2) \cup \{v_2\}$, and $\bar{\mathcal{H}}(V_1, S_1) = \{(H \setminus S_1) \cup \{v_1\} \mid H \in \mathcal{H}(V_1, S_1)\}$.

▶ **Lemma 10.** *If $\mathcal{H}$ is unimodular, then both $\bar{\mathcal{H}}_1$ and $\bar{\mathcal{H}}_2$ are unimodular.*

Proof. Let v be an arbitrary vertex in $H_0 \cap S_2$. Then the (hyperedge-vertex) incidence matrix of the hypergraph $\bar{\mathcal{H}}_1$ is a submatrix of that of $\mathcal{H}$, with rows restricted to $\mathcal{H}_{V_1} \cup \mathcal{H}(V_2, S_2) \cup \{H_0\}$, and columns restricted to $V_1 \cup \{v\}$. This shows that this submatrix is totally unimodular. A similar argument shows that $\bar{\mathcal{H}}_2$ is also unimodular. ◀

▶ **Lemma 11.** $|\operatorname{Tr}(\bar{\mathcal{H}}_1)| \leq |\operatorname{Tr}(\mathcal{H})|$ *and* $|\operatorname{Tr}(\bar{\mathcal{H}}_2)| \leq |\operatorname{Tr}(\mathcal{H})|$.

Proof. We prove the claim that $|\operatorname{Tr}(\bar{\mathcal{H}}_1)| \leq |\operatorname{Tr}(\mathcal{H})|$; the other claim can be proved similarly.

It is enough to show that for every minimal transversal $T \in \operatorname{Tr}(\bar{\mathcal{H}}_1)$, there is a minimal transversal $T' \in \operatorname{Tr}(\mathcal{H})$ such that for any distinct $T_1, T_2 \in \operatorname{Tr}(\bar{\mathcal{H}}_1)$, T_1' and T_2' are distinct.

Let $\mathcal{T}_1 = \{T \in \operatorname{Tr}(\bar{\mathcal{H}}_1) : v_2 \notin T\}$ and $\mathcal{T}_2 = \operatorname{Tr}(\bar{\mathcal{H}}_1) \setminus \mathcal{T}_1$. Consider first $T \in \mathcal{T}_1$. By assumption $T \cap S_1 \neq \emptyset$ since T has a nonempty intersection with $\bar{H}_0$. It follows that the only hyperedges of $\mathcal{H}$ having empty intersection with T are those in $\mathcal{H}_{V_2}$. Note that none of these hyperedges are contained in S_2 since $\mathcal{H}$ is Sperner. This implies that $\mathcal{H}_{V_2}^{V_2 \setminus S_2} \neq \{\emptyset\}$ and therefore $\operatorname{Tr}(\mathcal{H}_{V_2}^{V_2 \setminus S_2}) \neq \emptyset$. Let T'' be an arbitrary minimal transversal in $\operatorname{Tr}(\mathcal{H}_{V_2}^{V_2 \setminus S_2})$. Then it is easy to see that $T' = T \cup T''$ is in $\operatorname{Tr}(\mathcal{H})$.

Consider now $T \in \mathcal{T}_2$. By the minimality of T, there is a hyperedge $H \in \bar{\mathcal{H}}(V_2, S_2) \cup \{\bar{H}_0\}$ such that $H \cap T = \{v_2\}$. Furthermore, for every $v \in T \setminus \{v_2\}$, there is an $H \in \mathcal{H}_{V_1}$ such that $T \cap H = \{v\}$. Let $\mathcal{H}(T) = \{H \in \mathcal{H} \mid H \cap T \setminus \{v_2\} = \emptyset\}$ and note that $\mathcal{H}(T)^{V_2}$ is nontrivial. Pick $T'' \in \operatorname{Tr}(\mathcal{H}(T)^{V_2})$ arbitrarily. Then it is easy to see that $T' = (T \setminus \{v_2\}) \cup T''$ is in $\operatorname{Tr}(\mathcal{H})$.

Finally, note that for any distinct $T_1, T_2 \in \mathcal{T}_1$ (resp., $T_1, T_2 \in \mathcal{T}_2$), the constructed minimal transversals $T_1', T_2' \in T(\mathcal{H})$ are distinct. Moreover, for $T_1 \in \mathcal{T}_1$ and $T_2 \in \mathcal{T}_2$, T_1' and T_2' are distinct because $T_1' \cap S_2 = \emptyset$ while $T_2' \cap S_2 \neq \emptyset$. ◀

To compute (12), we find $\operatorname{Tr}(\bar{\mathcal{H}}_1)$ and $\operatorname{Tr}(\bar{\mathcal{H}}_2)$ recursively. Then $\operatorname{Tr}(\mathcal{H}_1)$ and $\operatorname{Tr}(\mathcal{H}_2)$ are given by the following claim.

▶ **Lemma 12.** *Let $\mathcal{T}_1 = \{T \in \operatorname{Tr}(\bar{\mathcal{H}}_1) \mid v_2 \notin T\}$, $\mathcal{T}_2 = \{T \in \operatorname{Tr}(\bar{\mathcal{H}}_1) \mid v_2 \in T,\ S_1 \cap T = \emptyset\}$, $\mathcal{T}_3 = \operatorname{Tr}(\bar{\mathcal{H}}_1) \setminus (\mathcal{T}_1 \cup \mathcal{T}_2)$, $\mathcal{T}_1' = \{T \in \operatorname{Tr}(\bar{\mathcal{H}}_2) \mid v_1 \notin T\}$ and $\mathcal{T}_2' = \operatorname{Tr}(\bar{\mathcal{H}}_2) \setminus \mathcal{T}_1'$. Then*

$$\begin{aligned} \operatorname{Tr}(\mathcal{H}_1) = \mathcal{T}_1 \ \dot{\cup}\ &\{(T \setminus \{v_2\}) \cup \{v\} \mid v \in H_0 \cap S_2 \text{ and } T \in \mathcal{T}_2\} \\ &\dot{\cup}\ \{(T \setminus \{v_2\}) \cup \{v\} \mid v \in S_2 \text{ and } T \in \mathcal{T}_3\}, \end{aligned} \tag{13}$$

$$\operatorname{Tr}(\mathcal{H}_2) = \mathcal{T}_1' \ \dot{\cup}\ \{(T \setminus \{v_1\}) \cup \{v\} \mid v \in S_1 \text{ and } T \in \mathcal{T}_2'\}. \tag{14}$$

Proof. Let us prove (13), since the proof of (14) is similar. Suppose $T \in \operatorname{Tr}(\mathcal{H}_1)$. If $T \cap S_2 = \emptyset$ then (it is easy to see that) $T \in \mathcal{T}_1$. If $T \cap S_2 \neq \emptyset$ then by minimality of T, $|T \cap S_2| = 1$; let $T \cap S_2 = \{v\}$. If $T \cap S_1 = \emptyset$ then necessarily $v \in H_0$, in which case $(T \setminus \{v\}) \cup \{v_2\} \in \mathcal{T}_2$; otherwise v can be any element in S_2, and hence, $(T \setminus \{v\}) \cup \{v_2\} \in \mathcal{T}_3$. On the other direction, if $T \in \mathcal{T}_1$ then clearly $T \in \operatorname{Tr}(\mathcal{H}_1)$; if $T \in \mathcal{T}_2$ then $T \cap \bar{H}_0 = \{v_2\}$

which implies that $(T \setminus \{v_2\}) \cup \{v\} \in \mathrm{Tr}(\mathcal{H}_1)$ for every $v \in H_0 \cap S_2$; finally, if $T \in \mathcal{T}_3$ then there is a hyperedge $H \in \bar{\mathcal{H}}(V_2, S_2)$ such that $H \cap T = \{v_2\}$, which implies in turn that $(T \setminus \{v_2\}) \cup \{v\} \in \mathrm{Tr}(\mathcal{H}_1)$. ◀

Note that $\mathrm{Tr}(\mathcal{H})$ can be computed from $\mathrm{Tr}(\bar{\mathcal{H}}_1)$ and $\mathrm{Tr}(\bar{\mathcal{H}}_2)$ using (12) and Lemma 12 in time $L_2(\mu) = \mathrm{poly}(n, m, k)$. Now, we bound $T(\mu)$.

Let $n'_1 = n(\bar{\mathcal{H}}_1) = n_1+1$, $m'_1 = |\bar{\mathcal{H}}_1| \in \{m_1, m_1+1\}$, $k_1 = |\,\mathrm{Tr}(\bar{\mathcal{H}}_1)|$, $n'_2 = n(\bar{\mathcal{H}}_2) = n_2+1$, $m_2 = |\bar{\mathcal{H}}_2|$, and $k_2 = |\,\mathrm{Tr}(\bar{\mathcal{H}}_2)|$. By the decomposition, $n'_1+n'_2 = n+2$ and $m'_1+m_2 = m+1$, and by Lemma 11, $k_1 \le k$ and $k_2 \le k$. Note that $n'_1, n'_2 \ge 2$, $m'_1, m_2 \ge 1$, $n_1 + m_1 \ge 4$, and $n'_2 m_2 \ge 4$, by the assumptions of the 3-sum case in Theorem 7.

We consider 3 subcases.

Case II-I: $2 \le n'_1 \le 3$. Then a simple procedure will be used to compute $\mathrm{Tr}(\bar{\mathcal{H}}_1)$, and hence we need only to recurse on $\bar{\mathcal{H}}_2$, giving the simpler recurrence: $T(\mu) \le 2 + T(\mu_2)$. Note that $m_2 \le m-2$ since $n_1 \le 2$ implies $m_1 \ge 2$ and hence $m_2 = m - m_1 \le m - 2$. Since $\mu_2 = n'_2 m_2 k_2 \le n(m-2)k \le \mu - 2$, we get by induction that

$$T(\mu) \le 2 + \mu_2 \le \mu. \tag{15}$$

Case II-II: $n'_2 = 2$. Then a simple procedure will be used to compute $\mathrm{Tr}(\bar{\mathcal{H}}_2)$, and hence we need only to recurse on $\bar{\mathcal{H}}_1$, giving the simpler recurrence: $T(\mu) \le 2 + T(\mu_1)$. As above, $m_1 \le m-3$ implying that $\mu_1 = n'_1 m'_1 k_2 \le n(m-2)k \le \mu - 2$, and giving by induction again that $T(\mu) \le \mu$.

Case II-III: $n'_1 \ge 4$ and $n'_2 \ge 3$. We first note that $m'_1, m_2 \ge 2$. Indeed, if $m'_1 = 1$ (resp., $m_2 = 1$), then $\mathcal{H}_{V_1} = \emptyset$ and $\mathcal{H}(V_2, S_2) = \{H_0\}$ (resp., $\mathcal{H}_{V_2} = \emptyset$ and $\mathcal{H}(V_1, S_1) = \{H_0\}$). Since we assume that $\mathcal{H}$ does not have identical vertices, we must have $n_1 = 1$ (resp., $n_2 = 1$). In either case we get a contradiction to the boundary assumtpions (ii) of the 3-sum–case 1 in Theorem 7. Lemmas 10 and 12 imply that, in this case, we get the recurrence:

$$T(\mu) \le 1 + T(\mu_1) + T(\mu_2), \tag{16}$$

where $\mu = \mu(\mathcal{H})$, $\mu_1 = \mu(\bar{\mathcal{H}}_1)$, and $\mu_2 = \mu(\bar{\mathcal{H}}_2)$.

Then by Fact 8, applied with $x_1 = n'_1$, $y_1 = m'_1$, $x_2 = n'_2$, $y_2 = m_2$, $N = n+2$, $M = m+1$, $\alpha = 3$ and $\beta = 2$, we get (as $n \ge 5$ and $m \ge 3$)

$$\begin{aligned}\mu_1 + \mu_2 = n'_1 m'_1 k_1 + n'_2 m_2 k_2 &\le (n'_1 m'_1 + n'_2 m_2)k \le ((n-1)(m-1)+6)k \\ &= nmk - (n+m-7)k \le \mu - 1.\end{aligned} \tag{17}$$

It follows by induction from (16) that

$$T(\mu) \le 1 + \mu_1 + \mu_2 \le \mu. \tag{18}$$

3.3 3-sum decomposition - case 2

Let $\mathcal{H}_1 = \mathcal{H}_{V_1}$ and $\mathcal{H}_2 = \mathcal{H}_{V_2}$. By Theorem 7, we have three nonempty disjoint sets S_0, S_1, S_2 in V_2, and the following two families are nonempty:

$$\mathcal{F}_1 = \{H \in \mathcal{H} \mid H \cap V_1 = S_0 \cup S_2, H \cap V_2 \ne \emptyset\}, \tag{19}$$

$$\mathcal{F}_2 = \{H \in \mathcal{H} \mid H \cap V_1 = S_0 \cup S_1, H \cap V_2 \ne \emptyset\}. \tag{20}$$

Note that $V_1, V_2 \neq \emptyset$, $\mathcal{H}_1 \neq \emptyset$, and $\mathcal{H}$ can be partitioned in the following way.

$$\mathcal{H} = \mathcal{H}_1 \dot{\cup} \mathcal{H}_2 \dot{\cup} \mathcal{F}_1 \dot{\cup} \mathcal{F}_2, \tag{21}$$

where $\dot{\cup}$ denotes the disjoint union. For $i = 0, 1, 2$, let

$$\mathcal{T}_i = \{T \in \mathrm{Tr}(\mathcal{H}_1) \mid T \cap S_i \neq \emptyset, T \cap S_j = \emptyset \ (j \neq i)\}, \tag{22}$$

and let

$$\mathcal{T} = \mathrm{Tr}(\mathcal{H}_1) \setminus (\mathcal{T}_0 \cup \mathcal{T}_1 \cup \mathcal{T}_2). \tag{23}$$

By definition, we have

$$\mathrm{Tr}(\mathcal{H}_1) = \mathcal{T} \dot{\cup} \mathcal{T}_0 \dot{\cup} \mathcal{T}_1 \dot{\cup} \mathcal{T}_2. \tag{24}$$

Let

$$\mathcal{P} = \mathcal{H}_{S_0 \cup S_1 \cup S_2} \ (= \{H \in \mathcal{H} \mid H \subseteq S_0 \cup S_1 \cup S_2\}). \tag{25}$$

We separatly consider the following 4 cases.

Case I: $\mathcal{P} = \emptyset$.
Case II: $\mathcal{P} = \{S_0 \cup S_1 \cup S_2\}$.
Case III: $\mathcal{P} \neq \emptyset, \{S_0 \cup S_1 \cup S_2\}$ and $\mathcal{T} \neq \emptyset$.
Case IV: $\mathcal{P} \neq \emptyset, \{S_0 \cup S_1 \cup S_2\}$ and $\mathcal{T} = \emptyset$.

Case I

$\mathcal{H}$ can be partitioned into $\mathcal{H}_1$ and $\mathcal{H}_{S_0 \cup S_1 \cup S_2 \cup V_2}$, i.e., $\mathcal{H} = \mathcal{H}_1 \dot{\cup} \mathcal{H}_{S_0 \cup S_1 \cup S_2 \cup V_2}$ and $\mathcal{H}_1$, $\mathcal{H}_{S_0 \cup S_1 \cup S_2 \cup V_2} \neq \emptyset$. Since $\mathrm{Tr}(\mathcal{H}) = \mathrm{Tr}(\mathcal{H}_1) \wedge \mathrm{Tr}(\mathcal{H}_{S_0 \cup S_1 \cup S_2 \cup V_2})$, we obtain $\mathrm{Tr}(\mathcal{H})$ by computing $\mathrm{Tr}(\mathcal{H}_1)$ and $\mathrm{Tr}(\mathcal{H}_{S_0 \cup S_1 \cup S_2 \cup V_2})$. Let $n_1 = |V_1|$, $m_1 = |\mathcal{H}_1|$, $k_1 = |\mathrm{Tr}(\mathcal{H}_1)|$, $n_2' = |S_0 \cup S_1 \cup S_2 \cup V_2|$, $m_2' = |\mathcal{H}_{S_0 \cup S_1 \cup S_2 \cup V_2}|$, $k_2 = |\mathrm{Tr}(\mathcal{H}_{S_0 \cup S_1 \cup S_2 \cup V_2})|$. Similar to the 2-sum decomposition case, we can show that $T(\mu) \leq \mu$ and the computation of $\mathrm{Tr}(\mathcal{H})$ can be done in time $L_2(\mu) = \mathrm{poly}(n, m, k)$.

Case II

We consider two cases: (1) $|\mathcal{H}_1| \geq 2$ and (2) $|\mathcal{H}_1| = 1$.

(1) $|\mathcal{H}_1| \geq 2$. Let $\mathcal{G}$ be a hypergraph obtained from $\mathcal{H}_{S_0 \cup S_1 \cup S_2 \cup V_2}$ by replacing S_0, S_1, and S_2 by new vertices v_0, v_1 and v_2, respectively. For any hyperedge $H \in \mathcal{H}_{S_0 \cup S_1 \cup S_2 \cup V_2}$, $H \cap S_i \neq \emptyset$ implies that $S_i \subseteq H$. Thus $\mathcal{G}$ is well-defined. Note that $\mathrm{Tr}(\mathcal{H}_{S_0 \cup S_1 \cup S_2 \cup V_2})$ can be obtained from $\mathrm{Tr}(\mathcal{G})$ in polynomial time by replacing v_i with any element in S_i. Since $\mathcal{H} = \mathcal{H}_1 \cup \mathcal{H}_{S_0 \cup S_1 \cup S_2 \cup V_2}$, we have $\mathrm{Tr}(\mathcal{H}) = \mathrm{Tr}(\mathcal{H}_1) \wedge \mathrm{Tr}(\mathcal{H}_{S_0 \cup S_1 \cup S_2 \cup V_2})$. We thus decompose $\mathcal{H}$ into $\mathcal{H}_1$ and $\mathcal{G}$. Namely we compute $\mathrm{Tr}(\mathcal{H})$ from $\mathrm{Tr}(\mathcal{H}_1)$ and $\mathrm{Tr}(\mathcal{G})$. Since $|\mathrm{Tr}(\mathcal{H}_1)|, |\mathrm{Tr}(\mathcal{G})| \leq |\mathrm{Tr}(\mathcal{H})| (= k)$, this can be done in time $L_2(\mu) = \mathrm{poly}(n, m, k)$.

Let us next show that $T(\mu) \leq \mu$. Let $n_1 = |V_1|$, $m_1 = |\mathcal{H}_1|$, $k_1 = |\mathrm{Tr}(\mathcal{H}_1)|$, $n_2' = |V_2| + 3$, $m_2' = |\mathcal{G}|$, and $k_2 = |\mathrm{Tr}(\mathcal{G})|$. Note that $\mathcal{H} = \mathcal{H}_1 \cup \mathcal{H}_{S_0 \cup S_1 \cup S_2 \cup V_2}$, $\mathcal{H}_1 \cap \mathcal{H}_{S_0 \cup S_1 \cup S_2 \cup V_2} = \{S_0 \cup S_1 \cup S_2\}$. This, together with definition and the discussion above, implies that

$$1 \leq n_1, n_2' \leq n, \ n_1 + n_2' = n + 3, \ 2 \leq m_1, m_2' \leq m - 1, \ m_1 + m_2' = m + 1, \ k_1, k_2 \leq k. \tag{26}$$

Thus we have

$$n_1 m_1 k_1 + n_2' m_2' k_2 \le (n_1 m_1 + n_2' m_2')k \le (n(m-1)+6)k \le nmk - 1 \tag{27}$$

where Fact 8 is used for the second inequality, and the third ineuqality is obtained by assuming that n is at least 7. It follows from (27) that $T(\mu) \le \mu$. We recall that $\mathrm{Tr}(\mathcal{H})$ is directly computed from $\mathcal{H}$ if at least one of n, m, and k is bounded by some constant C. Thus in this case we have $T(\mu) = 1$, which also satisfies $T(\mu) \le \mu$.

(2) $|\mathcal{H}_1| = 1$. In this case, we have $H_1 = \{S_0 \cup S_1 \cup S_2\}$. Therefore, the following lemma is satisfied.

▶ **Lemma 13.** *Let $\mathcal{H}$ be a hypergraph that satisfies (21) and $\mathcal{H}_1 = \{S_0 \cup S_1 \cup S_2\}$. Then we have* $\mathrm{Tr}(\mathcal{H}) = \{\{v\} \mid v \in S_0\} \dot\wedge \mathrm{Tr}(\mathcal{H}_2) \ \dot\cup \ \mathrm{Tr}(H^{V \setminus S_0})$.

Proof. From the definition, it is not difficult to see that $\mathrm{Tr}(\mathcal{H}) \supseteq \{\{v\} \mid v \in S_0\} \dot\wedge \mathrm{Tr}(\mathcal{H}_2) \ \dot\cup \ \mathrm{Tr}(H^{V \setminus S_0})$.

For the converse inclusion, let $T \in \mathrm{Tr}(\mathcal{H})$. If $T \cap S_0 = \emptyset$, then T is contained in $\mathrm{Tr}(H^{V \setminus S_0})$. Assume next that $T \cap S_0 \ne \emptyset$. For any $i = 0, 1, 2$ and any hyperedge $H \in \mathcal{H}$, $H \cap S_i \ne \emptyset$ inplies that $S_0 \subseteq H$. This means that $|T \cap S_0| = 1$ and $T \cap S_i = \emptyset$ for $i = 1, 2$. Moreover, we have $T \cap V_2 \in \mathrm{Tr}(\mathcal{H}_2)$, which completes the converse inclusion. ◀

Note that $\mathcal{H}_2, \mathcal{H}^{V \setminus S_0} \ne \{\emptyset\}$, and hence $\mathrm{Tr}(\mathcal{H}_2), \mathrm{Tr}(H^{V \setminus S_0}) \ne \emptyset$. Based on Lemma 13, we decompose $\mathcal{H}$ into $\mathcal{H}_2$ and $\mathcal{H}^{V \setminus S_0}$. Namely, we compute $\mathrm{Tr}(\mathcal{H})$ from $\mathrm{Tr}(\mathcal{H}_2)$ and $\mathrm{Tr}(H^{V \setminus S_0})$ in time $L_2(\mu) = \mathrm{poly}(n, m, k)$.

Let $n_1' = |V_2|$, $m_1' = |\mathcal{H}_2|$, $k_1' = |\mathrm{Tr}(\mathcal{H}_2)|$, $n_2' = n - |S_0|$, $m_2' = |\mathcal{H}^{V \setminus S_0}|$, and $k_2' = |\mathrm{Tr}(\mathcal{H}^{V \setminus S_0})|$. Then we have $n_1', n_2' \le n - 1$, $m_1', m_2' \le m$, and $k_1', k_2' \le k - 1$ and $k_1' + k_2' \le k$. Thus we have $n_1' m_1' k_1' + n_2' m_2' k_2' \le (n-1)(m-1)k \le nmk - 1$, where the last inequality is obtained from $n \ge 3$. This implies that $T(\mu) \le \mu$.

References

1 S. D. Abdullahi. *Vertex Enumeration and Counting for Certain Classes of Polyhedra*. PhD thesis, Computing (Computer Algorithms) Leeds University U.K., 2003.

2 D. Avis, B. Bremner, and R. Seidel. How good are convex hull algorithms. *Computational Geometry: Theory and Applications*, 7:265–302, 1997.

3 D. Avis and K. Fukuda. A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra. *Discrete and Computational Geometry*, 8(3):295–313, 1992.

4 D. Avis and K. Fukuda. Reverse search for enumeration. *Discrete Applied Mathematics*, 65(1-3):21–46, 1996.

5 D. Avis, G. D. Rosenberg, R. Savani, and B. von Stengel. Enumeration of Nash equilibria for two-player games. *Economic Theory*, 42(1):9–37, 2010.

6 C. Berge. *Hypergraphs*. Elsevier-North Holand, Amsterdam, 1989.

7 J. C. Bioch and T. Ibaraki. Complexity of identification and dualization of positive boolean functions. *Information and Computation*, 123(1):50–63, 1995.

8 E. Boros, K. Elbassioni, V. Gurvich, and K. Makino. Generating vertices of polyhedra and related problems of monotone generation. In D. Avis, D. Bremner, and A. Deza, editors, *Proceedings of the Centre de Recherches Mathématiques at the Université de Montréal, special issue on Polyhedral Computation*, volume 49, 2009.

9 E. Boros, K. Elbassioni, V. Gurvich, and H. R. Tiwary. The negative cycles polyhedron and hardness of checking some polyhedral properties. *Annals OR*, 188(1):63–76, 2011.

10 D. Bremner, K. Fukuda, and A. Marzetta. Primal-dual methods for vertex and facet enumeration. *Discrete and Computational Geometry*, 20:333–357, 1998.

11 M. R. Bussieck and M. E. Lübbecke. The vertex set of a 0/1 polytope is strongly $\mathcal{P}$-enumerable. *Computational Geometry: Theory and Applications*, 11(2):103–109, 1998.

12 V. Chvátal. *Linear Programming.* Freeman, San Francisco, CA, 1983.

13 G. Cornuéjols. *Combinatorial Optimization: Packing and Covering.* CBMS-NSF Regional Conference Series in Applied Mathematics. Society for Industrial and Applied Mathematics, 2001.

14 M. E. Dyer. The complexity of vertex enumeration methods. *Mathematics of Operations Research*, 8:381–402, 1983.

15 M. E. Dyer and L. G. Proll. An algorithms for determining all extreme points of a convex polytope. *Mathematical Programming*, 12:81–96, 1977.

16 T. Eiter and G. Gottlob. Identifying the minimal transversals of a hypergraph and related problems. *SIAM Journal on Computing*, 24(6):1278–1304, 1995.

17 T. Eiter, G. Gottlob, and K. Makino. New results on monotone dualization and generating hypergraph transversals. *SIAM Journal on Computing*, 32(2):514–537, 2003.

18 K. Elbassioni and K. Makino. Enumerating vertices of 0/1-polyhedra associated with 0/1-totally unimodular matrices. *CoRR*, abs/1707.03914, 2017. `arXiv:1707.03914`.

19 M. L. Fredman and L. Khachiyan. On the complexity of dualization of monotone disjunctive normal forms. *Journal of Algorithms*, 21:618–628, 1996.

20 V. Gurvich and L. Khachiyan. On generating the irredundant conjunctive and disjunctive normal forms of monotone Boolean functions. *Discrete Applied Mathematics*, 96-97(1):363–373, 1999.

21 L. Khachiyan, E. Boros, K. Borys, K. Elbassioni, and V. Gurvich. Generating all vertices of a polyhedron is hard. *Discrete & Computational Geometry*, 39(1-3):174–190, 2008.

22 L. Khachiyan, E. Boros, K. Elbassioni, V. Gurvich, and K. Makino. On the complexity of some enumeration problems for matroids. *SIAM Journal on Discrete Mathematics*, 19(4):966–984, 2005.

23 E. Lawler, J. K. Lenstra, and A. H. G. Rinnooy Kan. Generating all maximal independent sets: NP-hardness and polynomial-time algorithms. *SIAM Journal on Computing*, 9:558–565, 1980.

24 A. Lehman. On the width-length inequality, mimeographic notes. *Mathematical Programming*, 17:403–417, 1979.

25 L. Lovász. Combinatorial optimization: some problems and trends. DIMACS Technical Report 92-53, Rutgers University, 1992.

26 N. Mishra and L. Pitt. Generating all maximal independent sets of bounded-degree hypergraphs. In *COLT '97: Proceedings of the 10th annual conference on Computational learning theory*, pages 211–217, 1997.

27 M. E. Pfetsch. *The Maximum Feasible Subsystem Problem and Vertex-Facet Incidences of Polyhedra.* Dissertation, TU Berlin, 2002.

28 J.S. Provan. Efficient enumeration of the vertices of polyhedra associated with network **LP**'s. *Mathematical Programming*, 63(1):47–64, 1994.

29 A. Schrijver. *Theory of Linear and Integer Programming.* Wiley, New York, 1986.

30 R. Seidel. *Output-size sensitive algorithms for constructive problems in computational geometry.* Computer science, Cornell University, Ithaka, NY, 1986.

31 P. D. Seymour. Decomposition of regular matroids. *Journal of Combinatorial Theory, Series B*, 28(3):305–359, 1980.

32 K. Truemper. *Matroid Decomposition.* Academic Press, 1992.

33 V. V. Vazirani. *Approximation Algorithms.* Springer-Verlag New York, Inc., New York, NY, USA, 2001.