Some Fixed-Parameter Tractable Classes of Hypergraph Duality and Related Problems

Khaled Elbassioni[1], Matthias Hagen[2,3], and Imran Rauf[1]

[1] Max-Planck-Institut für Informatik, D–66123 Saarbrücken
{elbassio,irauf}@mpi-inf.mpg.de
[2] Friedrich-Schiller-Universität Jena, Institut für Informatik, D–07737 Jena
hagen@cs.uni-jena.de
[3] University of Kassel, Research Group Programming Languages / Methodologies, Wilhelmshöher Allee 73, D–34121 Kassel, Germany
hagen@uni-kassel.de

Abstract. In this paper we present fixed-parameter algorithms for the problem DUAL—given two hypergraphs, decide if one is the transversal hypergraph of the other—and related problems. In the first part, we consider the number of edges of the hypergraphs, the maximum degree of a vertex, and a vertex complementary degree as our parameters.

In the second part, we use an Apriori approach to obtain FPT results for generating all maximal independent sets of a hypergraph, all minimal transversals of a hypergraph, and all maximal frequent sets where parameters bound the intersections or unions of edges.

1 Introduction

In many situations, one might be interested in finding all objects or configurations satisfying a certain *monotone property*. Consider, for instance, the problem of finding all (inclusion-wise) maximal/minimal collections of items that are frequently/infrequently bought together by customers in a supermarket. More precisely, let $\mathcal{D} \in \{0,1\}^{m \times n}$ be a binary matrix whose rows represent the subsets of items purchased by different customers in a supermarket. For a given integer $t \geq 0$, a subset of items is said to be *t-frequent* if at least t rows (transactions) of $\mathcal{D}$ contain it, and otherwise is said to be *t-infrequent*. Finding frequent itemsets is an essential problem in finding the so-called *association rules* in data mining applications [AIS93]. By monotonicity, it is enough to find the *border* which is defined by the minimal t-infrequent and maximal t-frequent sets. While it was shown in [BGKM02] that finding maximal frequent sets is an NP-hard problem, finding the minimal t-infrequent sets, as well as many other enumeration problems in different areas (see e.g. [BEGK03, EG95]), turn out to be polynomially equivalent with the *hypergraph transversal problem*, defined as follows.

Let V be a finite set and $\mathcal{H} \subseteq 2^V$ be a hypergraph on V. A *transversal* of $\mathcal{H}$ is a subset of V that intersects every hyperedge of $\mathcal{H}$. Let $\mathcal{H}^d \subseteq 2^V$ be the hypergraph of all inclusion-wise minimal transversals of $\mathcal{H}$, also called the *dual* of $\mathcal{H}$. For hypergraphs $\mathcal{F}$ and $\mathcal{G}$ on vertex set V, the hypergraph transversal

M. Grohe and R. Niedermeier (Eds.): IWPEC 2008, LNCS 5018, pp. 91–102, 2008.

problem, denoted $\textsc{Dual}(\mathcal{F}, \mathcal{G})$, asks to decide whether they are dual to each other, i. e., $\mathcal{F}^d = \mathcal{G}$. Equivalently, the problem is to check if two monotone Boolean functions $f, g : \{0,1\}^n \mapsto \{0,1\}$ are dual to each other, i. e., $f(x) = \bar{g}(\bar{x})$ for all $x \in \{0,1\}^n$.

Let $\mathcal{F}, \mathcal{G} \subseteq 2^V$ be Sperner hypergraphs (i.e. no hyperedge of which contains another), and let $(\mathcal{F}, \mathcal{G})$ be an instance of $\textsc{Dual}$. By definition of dual hypergraphs we may assume throughout that

$$F \cap G \neq \emptyset \text{ for all } F \in \mathcal{F} \text{ and } G \in \mathcal{G}. \tag{1}$$

A *witness* for the non-duality of the pair $(\mathcal{F}, \mathcal{G})$ satisfying (1) is a subset $X \subseteq V$, such that

$$X \cap F \neq \emptyset \text{ for all } F \in \mathcal{F}, \text{ and } X \not\supseteq G \text{ for all } G \in \mathcal{G}. \tag{2}$$

We shall say that the hypergraphs $\mathcal{F}$ and $\mathcal{G}$ satisfying (1) are dual if no such witness exists. Intuitively, a witness of non duality of $(\mathcal{F}, \mathcal{G})$ is a transversal of $\mathcal{F}$ (not necessarily minimal) that does not include any hyperedge of $\mathcal{G}$. Also, by definition, the pair $(\emptyset, \{\emptyset\})$ is dual. Note that the condition (2) is symmetric in $\mathcal{F}$ and $\mathcal{G}$: $X \subseteq V$ satisfies (2) for the pair $(\mathcal{F}, \mathcal{G})$ if and only if $\bar{X}$ satisfies (2) for $(\mathcal{G}, \mathcal{F})$.

In the following we present some fixed-parameter algorithms for this problem. Briefly, a parameterized problem with parameter k is *fixed-parameter tractable* if it can be solved by an algorithm running in time $O(f(k) \cdot poly(n))$, where f is a function depending on k only, n is the size of the input, and $poly(n)$ is any polynomial in n. The class FPT contains all fixed-parameter tractable problems. For more general surveys on parameterized complexity and fixed-parameter tractability we refer to the monographs of Downey and Fellows, and Niedermeier [DF99, Nie06].

A related problem $\textsc{Dualization}(\mathcal{F})$ is to generate $\mathcal{F}^d$ given $\mathcal{F}$. Given an algorithm for $\textsc{Dualization}$ we can decide if $\mathcal{F}$ and $\mathcal{G}$ are dual by generating the dual hypergraph of one explicitly and compare it with the other (actually, $\textsc{Dual}$ and $\textsc{Dualization}$ are even equivalent in the sense of solvability in appropriate terms of polynomial time [BI95]). Due to the fact that the size of $\mathcal{F}^d$ may be exponentially larger than the the size of $\mathcal{F}$, we consider *output-sensitive* fixed-parameter algorithms for $\textsc{Dualization}$, i. e., which are polynomial in both the input and output size $|\mathcal{F}| + |\mathcal{F}^d|$. In this sense, the time required to produce each new output is usually called the *delay* of the algorithm.

Both, $\textsc{Dual}$ and $\textsc{Dualization}$ have many applications in such different fields like artificial intelligence and logic [EG95, EG02], database theory [MR92], data mining and machine learning [GKM+03], computational biology [Dam06, Dam07], mobile communication systems [SS98], distributed systems [GB85], and graph theory [JPY88, LLK80]. The currently best known algorithms for $\textsc{Dual}$ run in quasi-polynomial time or use $\mathcal{O}(\log^2 n)$ nondeterministic bits [EGM03, FK96, KS03]. Thus, on the one hand, $\textsc{Dual}$ is probably not coNP-complete, but on the other hand a polynomial time algorithm is not yet known.

In this paper, we show that $\textsc{Dual}(\mathcal{F}, \mathcal{G})$ is fixed parameter-tractable with respect to the following parameters:

- the numbers of edges $m = |\mathcal{F}|$ and $m' = |\mathcal{G}|$ (cf. Section 2),
- the maximum degrees of vertices in $\mathcal{F}$ and $\mathcal{G}$, i. e., $p = \max_{v \in V} |\{F \in \mathcal{F} \ : \ v \in F\}|$, $p' = \max_{v \in V} |\{G \in \mathcal{G} \ : \ v \in G\}|$ (cf. Section 3),
- the maximum complementary degrees $q = \max_{v \in V} |\{F \in \mathcal{F} \ : \ v \notin F\}|$ and $q' = \max_{v \in V} |\{G \in \mathcal{G} \ : \ v \notin G\}|$ (cf. Section 4), and
- the maximum c such that $|F_1 \cup F_2 \cup \cdots \cup F_k| \geq n - c$, where $F_1, \ldots, F_k \in \mathcal{F}$ and k is a constant—and the symmetric parameter c' with respect to $\mathcal{G}$ (cf. Section 5.2).

We shall prove the bounds with respect to the parameters m, p, q, c; the other symmetric bounds follow by exchanging the roles of $\mathcal{F}$ and $\mathcal{G}$. Our results for the parameters m and q improve the respective results from [Hag07].

Other related FPT results were obtained by Damaschke who studied counting and generating minimal transversals of size up to k and showed both problems to be FPT if hyperedges have constantly bounded size [Dam06, Dam07].

In Section 5.3 we consider the related problem of finding all maximal frequent sets, and show that it is fixed parameter-tractable with respect to the maximum size of intersection of k rows of the database $\mathcal{D}$ for a constant k, thus generalizing the well-known Apriori algorithm, which is fixed-parameter with respect to the size of the largest transaction.

Let V be a finite set of size $|V| = n$. For a hypergraph $\mathcal{F} \subseteq 2^V$ and a subset $S \subseteq V$, we use the following notations: $\bar{S} = V \setminus S$, $\mathcal{F}_S = \{F \in \mathcal{F} \mid F \subseteq S\}$ and $\mathcal{F}^S = \text{minimal}\,(\{F \cap S \mid F \in \mathcal{F}\})$, where for a hypergraph $\mathcal{H}$, minimal$(\mathcal{H})$ denotes the Sperner hypergraph resulting from $\mathcal{H}$ by removing hyperedges that contain any other hyperedge of $\mathcal{H}$.

2 Number of Edges as Parameter

Let $(\mathcal{F}, \mathcal{G})$ be an instance of DUAL and let $m = |\mathcal{F}|$. We show that the problem is fixed-parameter tractable with parameter m and improve the running time of [Hag07].

Given a subset $S \subseteq V$ of vertices, [BGH98] gave a criterion to decide if S is a *sub-transversal* of $\mathcal{F}$, i. e., if there is a minimal transversal $T \in \mathcal{F}^d$ such that $T \supseteq S$. In general, testing if S is a sub-transversal is an NP-hard problem even if $\mathcal{F}$ is a graph (see [BEGK00]). However, if $|S|$ is bounded by a constant, then such a check can be done in polynomial time. This observation was used to solve the hypergraph transversal problem in polynomial time for hypergraphs of bounded edge size in [BEGK00], or more generally of bounded conformality [BEGK04]. To describe this criterion, we need a few more definitions. For a subset $S \subseteq V$, and a vertex $v \in S$, let $\mathcal{F}_v(S) = \{H \in \mathcal{F} \mid H \cap S = \{v\}\}$. A selection of $|S|$ hyperedges $\{H_v \in \mathcal{F}_v(S) \mid v \in S\}$ is called *covering* if there exists a hyperedge $H \in \mathcal{F}_{V \setminus S}$ such that $H \subseteq \bigcup_{v \in S} H_v$.

Proposition 2.1 (cf. [BGH98]). *A non-empty subset $S \subseteq V$ is a sub-transversal for $\mathcal{F} \subseteq 2^V$ if and only if there exists a non-covering selection $\{H_v \in \mathcal{F}_v(S) \mid v \in S\}$ for S.*

If the size of S is bounded we have the following.

Lemma 2.2. *Given a hypergraph $\mathcal{F} \subseteq 2^V$ of size $|\mathcal{F}| = m$ and a subset $S \subseteq V$, of size $|S| = s$, checking whether S is a sub-transversal of $\mathcal{F}$ can be done in time $\mathcal{O}(nm(m/s)^s)$.*

Proof. For every possible selection $\mathcal{F} = \{H_v \in \mathcal{F}_v(S) \mid v \in S\}$, we can check if $\mathcal{F}$ is non-covering in $\mathcal{O}(n|\mathcal{F}_{\bar{S}}|)$ time. Since the families $\mathcal{F}_v(S)$ are disjoint, we have $\sum_{v \in S} |\mathcal{F}_v(S)| \leq m$, and thus the arithmetic-geometric mean inequality gives for the total number of selections

$$\prod_{v \in S} |\mathcal{F}_v(S)| \leq \left(\frac{\sum_{v \in S} |\mathcal{F}_v(S)|}{s}\right)^s \leq \left(\frac{m}{s}\right)^s.$$

□

Procedure DUALIZE1($\mathcal{F}, S, V$)**:**
Input: A hypergraph $\mathcal{F} \subseteq 2^V$, and a subset $S \subseteq V$
Output: The set $\{T \in \mathcal{F}^d \ : \ T \supseteq S\}$

1. **if** S is not a sub-transversal for $\mathcal{F}$ **then return**
2. **if** $S \in \mathcal{F}^d$ **then** output S and **return**
3. Find $e \in V \setminus S$, such that $S \cup \{e\}$ is a sub-transversal for $\mathcal{F}$
4. DUALIZE1($\mathcal{F}, S \cup \{e\}, V$)
5. DUALIZE1($\mathcal{F}^{V \setminus \{e\}}, S, V \setminus \{e\}$)

Fig. 1. The backtracking method for finding minimal transversals

The algorithm is given in Figure 1, and is based on the standard backtracking technique for enumeration (see e.g. [RT75, Eit94]). The procedure is called initially with $S = \emptyset$. It is easy to verify that the algorithm outputs all elements of the dual hypergraph $\mathcal{F}^d$, without repetition, and in lexicographic ordering (assuming some order on the vertex set V). Since the algorithm essentially builds a backtracking tree whose leaves are the minimal transversals of $\mathcal{F}$, the time required to produce each new minimal transversal is bounded by the depth of the tree (at most $\min\{n, m\}$) times the maximum time required at each node. By Lemma 2.2, the latter time is at most $n \cdot \mathcal{O}(nm) \cdot \max\{(m/s)^s \ : \ 1 \leq s \leq m\} = \mathcal{O}(n^2 m \cdot e^{m/e})$.

Lemma 2.3. *Let $\mathcal{F} \subseteq 2^V$ be a hypergraph with $|\mathcal{F}| = m$ edges on $|V| = n$ vertices. Then all minimal transversals of $\mathcal{F}$ can be found with $\mathcal{O}(n^2 m^2 e^{m/e})$ delay.*

Theorem 2.4. *Let $\mathcal{F}, \mathcal{G} \subseteq 2^V$ be two hypergraphs with $|\mathcal{F}| = m, |\mathcal{G}| = m'$ and $|V| = n$. Then $\mathcal{F}^d = \mathcal{G}$ can be decided in time $\mathcal{O}(n^2 m^2 e^{(m/e)} \cdot m')$.*

Proof. We generate at most m' members of $\mathcal{F}^d$ by calling DUALIZE1 (if there are more then obviously $\mathcal{F}^d \neq \mathcal{G}$). Assuming that hyperedges are represented by bit vectors (defined by indicator functions), we can check whether $\mathcal{G}$ is identical to $\mathcal{F}^d$ by lexicographically ordering the hyperedges of both and simply comparing the two sorted lists. The time to sort and compare m' hyperedges each one of size at most $\log n$ can be bounded by $\mathcal{O}(m' \log m' \log n)$. $\square$

As a side remark we note an interesting implication of Lemma 2.3. For a hypergraph $\mathcal{F}$ with $|\mathcal{F}| \leq c \log n$ for a constant c, the algorithm DUALIZE1 finds all its minimal transversals with polynomial delay $\mathcal{O}(n^{c/e+2} \log^2 n)$ improving the previous best bound of $\mathcal{O}(n^{2c+6})$ by Makino [Mak03]. Similarly, if the number of minimal transversals is bounded by $\mathcal{O}(\log n)$, then DUALIZE1 can be used to find all these transversals in incremental polynomial time. Another implication which we will need in Section 5 is the following.

Corollary 2.5. *For a hypergraph $\mathcal{F} \subseteq 2^V$, we can generate the first k minimal transversals in time $\mathcal{O}(n^2 k^3 e^{(k/e)} \cdot m)$, where $n = |V|$ and $m = |\mathcal{F}|$.*

Proof. We keep a partial list $\mathcal{G}$ of minimal transversals, initially empty. If $|\mathcal{G}| < k$, we call DUALIZE1 on $\mathcal{G}$ to generate at most $m+1$ elements of $\mathcal{G}^d$. If it terminates with $\mathcal{G}^d = \mathcal{F}$, then all elements of $\mathcal{F}^d$ have been generated. Otherwise, $X \in \mathcal{G}^d \backslash \mathcal{F}$ is a witness for the non-duality of $(\mathcal{G}, \mathcal{F})$, and so by symmetry $\bar{X}$ contains a new minimal transversal of $\mathcal{F}$ that extends $\mathcal{G}$. $\square$

3 Maximum Degree as Parameter

Let p be the maximum degree of a vertex in hypergraph $\mathcal{F} \subseteq 2^V$, i. e., $p = \max_{v \in V} |\{F \in \mathcal{F} : v \in F\}|$. We show that $\textsc{Dual}(\mathcal{F}, \mathcal{G})$ is fixed-parameter tractable with parameter p (a result which follows by similar techniques, but with weaker bounds, from [EGM03]).

For a labeling of vertices $V = \{v_1, v_2, \ldots, v_n\}$, let $\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_n$ be a partition of hypergraph $\mathcal{F}$ defined as $\mathcal{F}_i = \{F \in \mathcal{F} \ : \ F \ni v_i, F \subseteq \{v_1, \ldots, v_i\}\}$. By definition the size of each set $\mathcal{F}_i$ in this partition is bounded by p. The algorithm is given in Figure 2 and essentially combines the technique of the previous section with the method of [LLK80] (see also [BEGK04]). We proceed inductively, for $i = 1, \ldots, n$, by finding $(\mathcal{F}_1 \cup \ldots \cup \mathcal{F}_{i-1})^d$. Then for each set X in this transversal hypergraph we extend it to a minimal transversal to $(\mathcal{F}_1 \cup \ldots \cup \mathcal{F}_i)^d$ by finding $(\{F \in \mathcal{F}_i \ : \ F \cap X = \emptyset\})^d$, each set of which is combined with X, possibly also deleting some elements from X, to obtain a minimal transversal to $\mathcal{F}_1 \cup \ldots \cup \mathcal{F}_i$.

For a hypergraph $\mathcal{H}$ and its transversal X (not necessarily minimal), let $\delta(X)$ denote a minimal transversal of $\mathcal{H}$ contained in X.

The following proposition states that with the partition $\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_n$, the size of intermediate hypergraphs in this incremental algorithm never gets too large.

Proposition 3.1 (cf. [EGM03, LLK80]). *(i) $\forall S \subseteq V \ : \ |(\mathcal{F}_S)^d| \leq |\mathcal{F}^d|$, (ii) $|(\mathcal{F}_1 \cup \ldots \cup \mathcal{F}_i)^d| \leq |\mathcal{F}^d|$, (iii) For every $X \in (\mathcal{F}_1 \cup \ldots \cup \mathcal{F}_{i-1})^d$,*

$$|(\{F \in \mathcal{F}_i \ : \ F \cap X = \emptyset\})^d| \leq |(\mathcal{F}_1 \cup \ldots \cup \mathcal{F}_i)^d|.$$

Proof. All three follow from the fact that $(\mathcal{F}_S)^d$ is a truncation of $\mathcal{F}^d$ on S, where $S = \{v_1, \ldots, v_i\}$ in (ii) and $S = \{v_1, \ldots, v_i\} \setminus X$ in (iii). $\square$

Procedure DUALIZE2($\mathcal{F}, V$)**:**
Input: A hypergraph $\mathcal{F} \subseteq 2^V$
Output: The set $\mathcal{F}^d$

1. $\mathcal{X}_0 = \{\emptyset\}$
2. **for** $i = 1, \ldots, n$ **do**
3. **for** each $X \in \mathcal{X}_{i-1}$ **do**
4. Let $\mathcal{A} = \{F \in \mathcal{F}_i \ : \ F \cap X = \emptyset\}$
5. Use DUALIZE1 to compute $\mathcal{A}^d$ if not already computed
6. $\mathcal{X}_i \leftarrow \{\delta(X \cup Y) \ : \ Y \in \mathcal{A}^d\}$
7. **return** $\mathcal{X}_n$

Fig. 2. Sequential method for finding minimal transversals

Let $f(p, i)$ be the running time of algorithm DUALIZE1 when given a hypergraph with p edges on i vertices. Consider the i-th iteration. From Proposition 3.1 we have $|\mathcal{X}_{i-1}| \leq |\mathcal{F}^d|$ and since we only compute $\mathcal{A}^d$ in step 5 if not already computed, there are at most $\min\{2^p, |\mathcal{F}^d|\}$ calls to DUALIZE1. The size of $\mathcal{A}^d$ can also be bounded by Proposition 3.1, which gives us $|\mathcal{A}^d| \leq |\mathcal{F}^d|$. Furthermore it is easy to see that the minimal transversal in step 6 can be found in time $\mathcal{O}(n|\mathcal{F}|)$ by removing the extra vertices (at most n). Thus the time spent in the i-th iteration can be bounded by $\mathcal{O}(\min\{2^p, |\mathcal{F}^d|\} \cdot f(p, n) + n|\mathcal{F}| \cdot |\mathcal{F}^d|^2)$.

Theorem 3.2. *Let $\mathcal{F} \subseteq 2^V$ be a hypergraph on $|V| = n$ vertices in which the degree of each vertex $v \in V$ is bounded by p. Then all minimal transversals of $\mathcal{F}$ can be found in time $\mathcal{O}\left(n^2 m m' \cdot \left(\min\{2^p, m'\} \cdot np^2 e^{p/e} + m'\right)\right)$, where $m = |\mathcal{F}|$ and $m' = |\mathcal{F}^d|$.*

4 Vertex Complementary Degree as Parameter

For a hypergraph $\mathcal{F} \subseteq 2^V$ and a vertex $v \in V$, consider the number of edges in $\mathcal{F}$ not containing v for some vertex $v \in V$. Let q be maximum such number, i.e., $q = \max_{v \in V} |\{F \in \mathcal{F} \ : \ v \notin F\}|$. We show that $\textsc{Dual}(\mathcal{F}, \mathcal{G})$ is fixed-parameter tractable with parameter q and improve the running time of [Hag07].

The following proposition gives a decomposition rule originally due to [FK96] which for a vertex $v \in V$ divides the problem into two subproblems not containing v.

Proposition 4.1 (cf. [FK96]). *Let $\mathcal{F}, \mathcal{G} \subseteq 2^V$ be two hypergraphs satisfying* (1)*, and $v \in V$ be a given vertex. Then $\mathcal{F}$ and $\mathcal{G}$ are dual if and only if the pairs $(\mathcal{F}_{V \setminus v}, \mathcal{G}^{V \setminus v})$ and $(\mathcal{F}^{V \setminus v}, \mathcal{G}_{V \setminus v})$ are dual.*

For a vertex $v \in V$, one of the subproblem $(\mathcal{F}_{V \setminus v}, \mathcal{G}^{V \setminus v})$ involves a hypergraph $\mathcal{F}_{V \setminus v}$ with at most q edges. The algorithm solves it by calling DUALIZE1 resulting in time $\mathcal{O}(n^2 q^2 e^{(q/e)} \cdot |(\mathcal{F}_{V \setminus v})^d|)$. The other subproblem $(\mathcal{F}^{V \setminus v}, \mathcal{G}_{V \setminus v})$ is solved recursively. Since at least one vertex is reduced at each step of the algorithm, there are at most $n = |V|$ recursive steps.

Theorem 4.2. *Let* $\mathcal{F}, \mathcal{G} \subseteq 2^V$ *be two hypergraphs with* $|\mathcal{F}| = m, |\mathcal{G}| = m'$ *and* $|V| = n$. *Let* $q = \max_{v \in V} |\{H \in \mathcal{F} \ : \ v \notin H\}|$. *Then* $\mathcal{F}^d = \mathcal{G}$ *can be decided in time* $\mathcal{O}(n^3 q^2 e^{(q/e)} \cdot m')$.

5 Results Based on the Apriori Technique

Gunopulos et al. [GKM+03] showed (Theorem 23, page 156) that generating minimal transversals of hypergraphs $\mathcal{F}$ with edges of size at least $n - c$ can be done in time $\mathcal{O}(2^c poly(n, m, m'))$, where $n = |V|$, $m = |\mathcal{F}|$ and $m' = |\mathcal{F}^d|$. This is a fixed-parameter algorithm for c as parameter. Furthermore, this result shows that the transversals can be generated in polynomial time for $c \in \mathcal{O}(\log n)$. The computation is done by an Apriori (level-wise) algorithm [AS94].

Using the same approach, we shall show below that we can compute all the minimal transversals in time $\mathcal{O}(\min\{2^c (m')^k poly(n, m), e^{k/e} n^{c+1} poly(m, m')\})$ if the union of any k distinct minimal transversals has size at least $n - c$. Equivalently, if any k distinct maximal independent sets of a hypergraph $\mathcal{F}$ intersect in at most c vertices, then all maximal independent sets can be computed in the same time bound. As usual, an *independent set* of a hypergraph $\mathcal{F}$ is a subset of its vertices which does not contain any hyperedge of $\mathcal{F}$.

And again using the same idea, we show that the maximal frequent sets of an $m \times n$ database can be computed in $\mathcal{O}(2^c (nm')^{2^{k-1}+1} poly(n, m))$ time if any k rows of it intersect in at most c items, where m' is the number of such sets.

Note that for $c \in \mathcal{O}(\log n)$ we have incremental polynomial-time algorithms for all four problems.

5.1 The Generalized Apriori Algorithm

Let $f : V \mapsto \{0, 1\}$ be a monotone Boolean function, that is, for which $f(X) \geq f(Y)$ whenever $X \supseteq Y$. We assume that f is given by a polynomial-time evaluation oracle requiring maximum time T_f, given the input. The Apriori approach for finding all maximal subsets X such that $f(X) = 0$ (maximal false sets of f), works by traversing all subsets X of V, for which $f(X) = 0$, in increasing size, until all maximal such sets have been identified. The procedure is given in Figure 3.

Lemma 5.1. *If any maximal false set of* f *contains at most* c *vertices, then APRIORI finds all such sets in* $\mathcal{O}(2^c m' n T_f)$ *time, where* $n = |V|$ *and* m' *is the number of maximal false sets.*

Proof. The correctness of this Apriori style method can be shown straightforwardly (cf. e.g. [AS94, GKM+03]). To see the time bound, note that for each maximal false set we check at most 2^c candidates (all the subsets) before adding it to $\mathcal{C}$. For each such candidate we check whether it is a false set and whether it cannot be extended by adding more vertices. □

Procedure APRIORI(f, V)**:**
Input: a monotone Boolean function $f : V \mapsto \{0, 1\}$
Output: the maximal sets $X \subseteq V$ such that $f(X) = 0$

```
C ← ∅; C_1 ← {{v} : v ∈ V}; i ← 1; C_j ← ∅ ∀j = 2, 3, ...
while C_i ≠ ∅
    for X, Y ∈ C_i, |X ∩ Y| = i − 1
        Z ← X ∪ Y
        if f(Z) = 0 then
            if f(Z ∪ {v}) = 1, for all v ∈ V \ Z then
                C ← C ∪ {Z}
            else
                C_{i+1} ← C_{i+1} ∪ {Z}
    i ← i + 1
return C
```

Fig. 3. The generalized Apriori algorithm

5.2 Maximal Independent Sets

Let $\mathcal{F} \subseteq 2^V$ be a hypergraph. An *independent set* of $\mathcal{F}$ is a subset of V which does not contain any hyperedge of $\mathcal{F}$. It is easy to see that the hypergraph of maximal independent sets $\mathcal{F}^{dc}$ of $\mathcal{F}$ is the complementary hypergraph of the dual $\mathcal{F}^d$: $\mathcal{F}^{dc} = \{V \setminus T : T \in \mathcal{F}^d\}$.

Let k and c be two positive integers. We consider hypergraphs $\mathcal{F} \subseteq 2^V$ satisfying the following condition:

(C1) Any k distinct maximal independent sets $I_1, \dots, I_k$ of $\mathcal{F}$ intersect in at most c vertices, i. e., $|I_1 \cap \dots \cap I_k| \le c$.

We shall derive below fixed-parameter algorithms with respect to either c or k. We note that condition (C1) can be checked in polynomial time for $c = \mathcal{O}(1)$ and $k = \mathcal{O}(\log n)$. Indeed, (C1) holds if and only if every set $X \subseteq V$ of size $|X| = c + 1$ is contained in at most $k - 1$ maximal independent sets of $\mathcal{F}$. The latter condition can be checked in time $n^{c+1} \operatorname{poly}(n, m, k) e^{k/e}$ as follows from the following lemma.

Lemma 5.2. *Given a hypergraph $\mathcal{F}$ with vertex set V and a subset $S \subseteq V$ of vertices, we can check in polynomial time whether S is contained in k different maximal independent sets. Furthermore k such sets can be generated in time $\mathcal{O}(\operatorname{poly}(n, m, k) e^{k/e})$.*

Proof. Clearly, this check is equivalent to checking if S does not contain an edge of $\mathcal{F}$ and if the truncated hypergraph $\mathcal{F}^{\bar{S}}$ has k maximal independent sets, or equivalently k minimal transversals. By Corollary 2.5, this can be done in $\mathcal{O}(\operatorname{poly}(n, m, k) e^{k/e})$ time. □

For a set $S \subseteq V$, denote by $\mathcal{F}^{dc}[S]$ the set of maximal independent sets of $\mathcal{F}$ containing S.

Theorem 5.3. *If any k distinct maximal independent sets of a hypergraph $\mathcal{F}$ intersect in at most c vertices, then all maximal independent sets can be computed in time $\mathcal{O}(\min\{2^c(m')^k poly(n,m), e^{k/e}n^{c+1}poly(m,m')\})$, where $n = |V|$, $m = |\mathcal{F}|$ and $m' = |\mathcal{F}^{dc}|$.*

Proof. (i) c as a parameter: we first use APRIORI to find the set $\mathcal{X}$ of all maximal subsets contained in at least k distinct maximal independent sets of $\mathcal{F}$. By (C1) the size of each such subset is at most c. To do this we use APRIORI with the monotone Boolean function defined by $f(X) = 0$ if and only if $X \subseteq I_1 \cap \cdots \cap I_k$, for k distinct maximal independent sets $I_1, \ldots, I_k$. The procedure is given in Figure 4. By Lemmas 5.1 and 5.2, all the intersections in $\mathcal{X}$ can be found in time $2^c \operatorname{poly}(n,m,k)e^{k/e}|\mathcal{X}|$. Thus the total running time can be bounded by $2^c \operatorname{poly}(n,m,k)e^{k/e}(m')^k$ since $|\mathcal{X}| \leq (m')^k$. It remains to argue that any maximal independent set $I \in \mathcal{F}^{dc}$ is generated by the procedure. To see this, let Y be a maximal subset such that $Y = I \cap I_1 \cap \ldots \cap I_r$, where $I, I_1, \ldots, I_r$, are distinct maximal independent sets of $\mathcal{F}$ with $r \geq k-1$, and let $v \in I \setminus (\cap_{j\in[r]} I_j)$. Note that such v exists since $I \not\subseteq \cap_{j\in[r]} I_j$ since $I, I_1, \ldots, I_r$ are distinct maximal independent sets. Then by maximality of Y, $Y \cup \{v\}$ is contained in at most $k-1$ maximal independent sets, one of which is I, and hence will be considered by the procedure in Step 7.

(ii) k as a parameter: Let $\mathcal{I}_1 = \{I \in \mathcal{F}^{dc} : \ |I| \leq c\}$ and $\mathcal{I}_2 = \mathcal{F}^{dc} \setminus \mathcal{I}_1$. Elements of $\mathcal{I}_1$ can be found using the APRIORI procedure with the monotone Boolean function, defined as $f(X) = 0$ if and only if $X \subseteq V$ is independent and has size at most c (or by testing all subsets of size at most c for maximal independence). Elements of $\mathcal{I}_2$ can be found by noting that each of them contains a set of size $c+1$, and that each such set is contained in at most $k-1$ elements of $\mathcal{I}_2$ by (C1). Thus for each set X of size $c+1$, we can use Lemma 5.2 to find all maximal independent sets containing X. $\square$

Corollary 5.4. *Let $\mathcal{F} \subseteq 2^V$ be a hypergraph on $n = |V|$ vertices, and k, c be positive integers.*

Procedure MAX-INDP-GEN($\mathcal{F}, V$)**:**

Input: a hypergraph $\mathcal{F} \subseteq 2^V$

Output: the set of maximal independent sets of $\mathcal{F}$

1. $\mathcal{C} \leftarrow \emptyset$
2. Use APRIORI to find the set of maximal k-independent set intersections $\mathcal{X}$
3. **for** each $X \in \mathcal{X}$ **do**
4.   **for** each $Y \subseteq X$ **do**
5.   **for** each $v \in V \setminus Y$ **do**
6.   **if** $|\mathcal{F}^{dc}[Y \cup \{v\}]| \leq k-1$
7. $\mathcal{C} \leftarrow \mathcal{C} \cup \mathcal{F}^{dc}[Y \cup \{v\}]$ (obtained using Corollary 2.5)
8. **return** $\mathcal{C}$

Fig. 4. The fixed parameter algorithm for finding all maximal independent sets

(i) If any k distinct minimal transversals of $\mathcal{F}$ have a union of at least $n-c$ vertices, we can compute all minimal transversals in $\mathcal{O}(\min\{2^c(m')^k poly(n,m), e^{k/e}n^{c+1}poly(m,m')\})$ *time, where $m = \mathcal{F}$ and $m' = |\mathcal{F}^d|$.*

(ii) If any k distinct hyperedges of $\mathcal{F}$ have a union of at least $n-c$ vertices, we can compute all minimal transversals in time $\mathcal{O}(\min\{2^c m^k poly(n,m'), e^{k/e}n^{c+1}poly(m,m')\})$*, where $m = \mathcal{F}$ and $m' = |\mathcal{F}^d|$.*

Proof. Both results are immediate from Theorem 5.3. (i) follows by noting that each minimal transversal is the complement of a maximal independent set, and hence any k maximal independent sets are guaranteed to intersect in at most c vertices. (ii) follows by maintaining a partial list $\mathcal{G} \subseteq \mathcal{F}^d$, and switching the roles of $\mathcal{F}$ and $\mathcal{G}$ in (i) to compute the minimal transversals of $\mathcal{G}$ using Theorem 5.4. Since condition (i) is satisfied with respect to $\mathcal{G}$, we can either verify duality of $\mathcal{F}$ and $\mathcal{G}$, or extend $\mathcal{G}$ by finding a witness for the non-duality (in a way similar to Corollary 2.5). □

5.3 Maximal Frequent Sets

Consider the problem of finding the maximal frequent item sets in a collection of m transactions on n items, stated in the Introduction. Here, a transaction simply is a set of items. An item set is maximal frequent for a frequency t if it occurs in at least t of the transactions and none of its supersets does. As another application of the approach of the previous subsection we obtain the following.

Theorem 5.5. *If any k distinct maximal frequent sets intersect in at most c items, we can compute all maximal frequent sets in $\mathcal{O}(2^c(nm')^k poly(n,m))$ time, where m' is the number of maximal frequent sets.*

Proof. The proof is analogous to that of Theorem 5.3. Just note that the set of transactions forms a hypergraph and replace "independent" by "frequent". To complete the proof, we need the following procedure to find k maximal frequent sets containing a given set. For $1 \le i \le k$ and frequent set X, let $F_1, \ldots, F_{i-1}$ be the maximal frequent sets containing X and let Y be the set with the property that $X \cup Y$ is frequent and $\forall j < i, \exists y \in Y : y \notin F_j$. Then any maximal frequent set containing $X \cup Y$ is different from $F_1 \ldots F_{i-1}$ by construction and thus giving us a new maximal frequent set. The running time of the above procudure can be bounded by $\mathcal{O}(n^k poly(n,m))$. Combining it with Lemma 5.1 gives us the stated running time. □

Corollary 5.6. *If any k distinct transactions intersect in at most c items, then all maximal frequent sets can be computed in time $\mathcal{O}(2^c(nm')^{2^{k-1}+1}poly(n,m))$, where m' is the number of maximal frequent sets.*

Proof. Note that if $t \ge k$ then every maximal frequent set has size at most c which in turn implies $\mathcal{O}(2^c poly(n,m) \cdot m')$ time algorithm using straightforward Apriori approach, so we may assume otherwise. Consider the intersection X of l distinct maximal frequent sets and let $|X| > c$, we bound the maximum such l. Since the intersection size is more then c, at most $k-1$ transactions define these l disticnt maximal frequent sets and so $l \le \sum_{j=t}^{k-1} \binom{k-1}{j} \le 2^{k-1}$. □

6 Concluding Remarks

Giving an FPT algorithm for Dual with respect to the parameter size l of a largest edge remains open. Nevertheless, proving that Dual is not FPT with respect to l seems to be tough as this would imply that there is no polynomial time algorithm for Dual assuming $\mathsf{W}[1] \neq \mathsf{FPT}$. Furthermore, this would be a strong argument for a separation of polynomial and quasi-polynomial time in "classical" computational complexity.

References

[AIS93] Agrawal, R., Imielinski, T., Swami, A.N.: Mining association rules between sets of items in large databases. In: Proc. SIGMOD 1993, pp. 207–216 (1993)

[AS94] Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: Proc. VLDB 1994, pp. 487–499 (1994)

[BEGK00] Boros, E., Elbassioni, K.M., Gurvich, V., Khachiyan, L.: An efficient incremental algorithm for generating all maximal independent sets in hypergraphs of bounded dimension. Parallel Processing Letters 10(4), 253–266 (2000)

[BEGK03] Boros, E., Elbassioni, K.M., Gurvich, V., Khachiyan, L.: An inequality for polymatroid functions and its applications. Discrete Applied Mathematics 131(2), 255–281 (2003)

[BEGK04] Boros, E., Elbassioni, K.M., Gurvich, V., Khachiyan, L.: Generating maximal independent sets for hypergraphs with bounded edge-intersections. In: Farach-Colton, M. (ed.) LATIN 2004. LNCS, vol. 2976, pp. 488–498. Springer, Heidelberg (2004)

[BGH98] Boros, E., Gurvich, V., Hammer, P.L.: Dual subimplicants of positive Boolean functions. Optimization Methods and Software 10(2), 147–156 (1998)

[BGKM02] Boros, E., Gurvich, V., Khachiyan, L., Makino, K.: On the complexity of generating maximal frequent and minimal infrequent sets. In: Alt, H., Ferreira, A. (eds.) STACS 2002. LNCS, vol. 2285, pp. 133–141. Springer, Heidelberg (2002)

[BI95] Bioch, J.C., Ibaraki, T.: Complexity of identification and dualization of positive Boolean functions. Inf. Comput. 123(1), 50–63 (1995)

[Dam06] Damaschke, P.: Parameterized enumeration, transversals, and imperfect phylogeny reconstruction. Theoret. Comput. Sci. 351(3), 337–350 (2006)

[Dam07] Damaschke, P.: The union of minimal hitting sets: Parameterized combinatorial bounds and counting. In: Thomas, W., Weil, P. (eds.) STACS 2007. LNCS, vol. 4393, pp. 332–343. Springer, Heidelberg (2007)

[DF99] Downey, R.G., Fellows, M.R.: Parameterized Complexity. Springer, Heidelberg (1999)

[EG95] Eiter, T., Gottlob, G.: Identifying the minimal transversals of a hypergraph and related problems. SIAM J. Comput. 24(6), 1278–1304 (1995)

[EG02] Eiter, T., Gottlob, G.: Hypergraph transversal computation and related problems in logic and AI. In: Flesca, S., Greco, S., Leone, N., Ianni, G. (eds.) JELIA 2002. LNCS (LNAI), vol. 2424, pp. 549–564. Springer, Heidelberg (2002)

[EGM03] Eiter, T., Gottlob, G., Makino, K.: New results on monotone dualization and generating hypergraph transversals. SIAM J. Comput. 32(2), 514–537 (2003)

[Eit94] Eiter, T.: Exact transversal hypergraphs and application to Boolean μ-functions. J. Symb. Comput. 17(3), 215–225 (1994)

[FK96] Fredman, M.L., Khachiyan, L.: On the complexity of dualization of monotone disjunctive normal forms. J. Algorithms 21(3), 618–628 (1996)

[GB85] Garcia-Molina, H., Barbará, D.: How to assign votes in a distributed system. J. ACM 32(4), 841–860 (1985)

[GKM+03] Gunopulos, D., Khardon, R., Mannila, H., Saluja, S., Toivonen, H., Sharm, R.S.: Discovering all most specific sentences. ACM Trans. Database Syst. 28(2), 140–174 (2003)

[Hag07] Hagen, M.: On the fixed-parameter tractability of the equivalence test of monotone normal forms. Inf. Process. Lett. 103(4), 163–167 (2007)

[JPY88] Johnson, D.S., Papadimitriou, C.H., Yannakakis, M.: On generating all maximal independent sets. Inf. Process. Lett. 27(3), 119–123 (1988)

[KS03] Kavvadias, D.J., Stavropoulos, E.C.: Monotone Boolean dualization is in $\mathsf{coNP}[\log^2 \mathsf{n}]$. Inf. Process. Lett. 85(1), 1–6 (2003)

[LLK80] Lawler, E.L., Lenstra, J.K., Rinnooy Kan, A.H.G.: Generating all maximal independent sets: NP-hardness and polynomial-time algorithms. SIAM J. Comput. 9(3), 558–565 (1980)

[Mak03] Makino, K.: Efficient dualization of $O(\log n)$-term monotone disjunctive normal forms. Discrete Applied Mathematics 126(2–3), 305–312 (2003)

[MR92] Mannila, H., Räihä, K.-J.: On the complexity of inferring functional dependencies. Discrete Applied Mathematics 40(2), 237–243 (1992)

[Nie06] Niedermeier, R.: Invitation to Fixed-Parameter Algorithms. Oxford University Press, Oxford (2006)

[RT75] Read, R.C., Tarjan, R.E.: Bounds on backtrack algorithms for listing cycles, paths, and spanning trees. Networks 5, 237–252 (1975)

[SS98] Sarkar, S., Sivarajan, K.N.: Hypergraph models for cellular mobile communication systems. IEEE Transactions on Vehicular Technology 47(2), 460–471 (1998)