Quasipolynomial Representation of Transversal Matroids with Applications in Parameterized Complexity

Daniel Lokshtanov[1], **Pranabendu Misra**[2], **Fahad Panolan**[3], **Saket Saurabh**[4], **and Meirav Zehavi**[5]

1 **University of Bergen, Bergen, Norway**
`daniello@ii.uib.no`
2 **The Institute of Mathematical Sciences, HBNI, Chennai, India**
`pranabendu@imsc.res.in`
3 **University of Bergen, Bergen, Norway**
`fahad.panolan@ii.uib.no`
4 **University of Bergen, Bergen, Norway and The Institute of Mathematical Sciences, HBNI, Chennai, India**
`saket@imsc.res.in`
5 **Ben-Gurion University, Beersheba, Israel**
`meiravze@bgu.ac.il`

Abstract

Deterministic polynomial-time computation of a representation of a transversal matroid is a longstanding open problem. We present a deterministic computation of a so-called union representation of a transversal matroid in time quasipolynomial in the rank of the matroid. More precisely, we output a collection of linear matroids such that a set is independent in the transversal matroid if and only if it is independent in at least one of them. Our proof directly implies that if one is interested in preserving independent sets of size at most r, for a given $r \in \mathbb{N}$, but does not care whether larger independent sets are preserved, then a union representation can be computed deterministically in time quasipolynomial in r. This consequence is of independent interest, and sheds light on the power of union representation.

Our main result also has applications in Parameterized Complexity. First, it yields a fast computation of representative sets, and due to our relaxation in the context of r, this computation also extends to (standard) truncations. In turn, this computation enables to efficiently solve various problems, such as subcases of subgraph isomorphism, motif search and packing problems, in the presence of color lists. Such problems have been studied to model scenarios where pairs of elements to be matched may not be identical but only similar, and color lists aim to describe the set of compatible elements associated with each element.

1998 ACM Subject Classification I.1.2 Algorithms, F.2.2 Nonnumerical Algorithms and Problems

Keywords and phrases Travserval matroid, matroid representation, union representation, representative set

Digital Object Identifier 10.4230/LIPIcs.ITCS.2018.32

1 Introduction

Matroids are widely-studied mathematical objects. In the context of computer science, these objects are of particular importance to algorithm design, combinatorial optimization and computational complexity. Specifically, from the viewpoint of algorithm design, analysis of these objects often leads to the discovery of algorithmic meta theorems. Such theorems unify classical results such as polynomial-time solvability of a wide-variety of problems as central as Minimum Weight Spanning Tree and Perfect Matching. In fact, if a problem admits a greedy algorithm, then it can be embedded in a matroid – solutions are thus associated with maximum independent sets in the matroid. Recently, matroids also stand in the forefront of studies of approximation algorithms, parameterized algorithms and kernels.

A matroid is a pair $M = (E, \mathcal{I})$, where $\mathcal{I}$ is a family of subsets of E (called *independent sets*), that satisfy three conditions called *matroid axioms* (see Section 2). As the size of $\mathcal{I}$ can be exponential in the size of E, the explicit listing of all independent sets is often rendered prohibitive. Then, it is necessary to have an *independence oracle* that, given a subset I of E, determines (in polynomial time) whether I is present in $\mathcal{I}$. For a wide class of matroids, known as *linear matroids*, such oracle is given by a matrix called a *representation*. Roughly speaking, the columns of the matrix are in bijection with the elements in E, and a set of columns is linearly independent if and only if the set of corresponding elements is independent. Unfortunately, for several important linear matroids, efficient computations of the desired representations are not known.

Specific well-known classes of matroids are those of uniform matroids, partition matroids, graphic matroids, cographic matroids, transversal matroids and gammoids. A common property of all of these classes is that all of them are contained in the wider class of linear matroids. However, for the last two classes in this list a polynomial-time deterministic computation of a representation is not known. Developing such a computation is a longstanding open problem. In this paper, we make significant progress towards the resolution of this question. We remark that as the dual of a transversal matroid is a gammoid and vice versa, a polynomial-time computation of a representation for one also yields such a computation for the other [13]. We specifically focus on the class of transversal matroids. Formally, a *transversal matroid* is a matroid derived from a bipartite graph G with a fixed bipartition (A, B) as follows: the ground set E is simply A, and a subset $X \subseteq A$ is independent if and only if G has a matching that saturates it. Matching constraints are ubiquitous in problems arising in all fields of research. Indeed, such constraints model scenarios where some set of objects relevant to our solution should be partitioned into pairs. Transversal matroids are precisely the translation of these constraints (in the bipartite setting) into the language of matroids.

To tackle the question above, we introduce the notion of *union representation*, which we believe to be worthy of independent study. For algorithmic purposes, such representation is as good as standard representation, given that the number of members in the union is small, and it may be useful also in cases where not only an efficient computation of a standard representation is not known, but a standard representation simply does not exist. Before we further discuss the power of this notion, let us first present it properly. Roughly speaking, a union representation of a matroid $M = (E, \mathcal{I})$ is a collection of matrices such that a subset X of E is independent in M if and only if for at least one of the matrices, the set of columns corresponding to X is linearly independent. Standard representation is precisely union representation where the size of the collection is one. While only linear matroids admit standard representations, note that all matroids admit union representations: to see this, for every base of the matroid, create one matrix with a set of linearly independent columns corresponding to the base, and vectors having only 0 entries as the rest of the columns. However, this procedure may create a huge number of matrices, and in order to make the notion of union representation relevant to algorithmic purposes, we desire the number of matrices to be as small as possible.

In this work, we present a deterministic computation of a union representation of a transversal matroid consisting of a quasipolynomial (in the rank of the matroid) number of matrices in quasipolynomial time. Prior to our work, the fastest such computation was only slightly better than trivial brute-force. More precisely, Misra et al. [14] showed that given a bipartite graph G with a fixed bipartition (A, B), a representation of the transversal matroid can be computed deterministically in (exponential) time $\binom{|A|}{r}|A|^{\mathcal{O}(1)}$ where r is the rank of the matroid, which equals the maximum size of a matching in G. In this context, it is important to note that a randomized polynomial-time algorithm to compute a representation of a transversal matroid is well known (see, e.g., [13, 17]). Here, randomization means that with some (low) probability, the algorithm may output a matrix that is not a representation of the matroid. This algorithm utilizes the Schwartz-Zippel lemma [2, 22, 25], and hence it is inherently randomized. The above mentioned trivial brute-force, which runs in time $2^{\mathcal{O}(|A|^2|B|)}$ (see [14]), refers to a loop through all choices made by the randomized algorithm.

Our technique builds upon recent powerful derandomization tools, particularly a construction given by Fenner et al. [4]. This construction is essentially a (quasipolynomial-time) derandomization of a special case of the isolation lemma [15], namely, the isolation of a perfect matching (if one exists). Roughly speaking, given a positive integer $n \in \mathbb{N}$, the construction is a collection of $2^{\mathcal{O}(\log^2 n)}$ weight functions such that for any bipartite graph G on $2n$ vertices that has a perfect matching, there exists a weight function w in the collection such that, when the edges of G are assigned weights according to w,[1] G has a unique perfect matching of minimum weight. Fenner et al. [4] utilized this construction to prove that Perfect Matching on bipartite graphs is in quasi-NC. Soon after this paper was published, significant generalizations of it followed [10, 7, 24]. Briefly, Gurjar et al. [10] showed that Linear Matroid Intersection is in quasi-NC, Goldwasser et al. [7] showed that Perfect Matching on bipartite graphs is in pseudo-deterministic NC, and Svensson et al. [24] showed that Perfect Matching on general graphs is in quasi-NC.

We introduce the above derandomization tools (specifically, the construction of [4]) to the context of representation, incorporating a flavor of Parameterized Complexity to the representation itself. Indeed, the computation we derive can be viewed as a (quasipolynomial-time) fixed-parameter tractable algorithm with respect to r. Consequently, we also introduce these tools to Parameterized Complexity, as we observe that our union representation computation can be incorporated in the method of *representative sets* by Fomin et al. [5] (see below). On a high-level, our proof consists of the use of a splitter [16] to "color" vertices of the input graph G using "small" integers. Then, we view a weight function not as a function that assigns weights to edges of some specific graph, but as a function that assigns weights to pairs of integers (that are simultaneously associated to a possibly exponential number of induced subgraphs of G). This allows us to compose splitters functions with weight functions. For each composition, we are then able to define a matrix, in the spirit of [15], that is one member of our union representation. The crux of the correctness is that both a splitter and a collection of [4] are *universal* in the sense that neither of them is tailored to a specific input graph to highlight structures of that graph (such as a perfect matching, in the case of the collection). Specifically, the same splitter and collection are relevant simultaneously to an exponential number of graphs that are of interest to our purpose, namely, all the induced

[1] Informally, each weight function assigns weights to the edges of a complete bipartite graph on $2n$ vertices that has a perfect matching, and the assignment of weights to the edges of G can be derived from this.

subgraphs of our input graph G that are sufficient to witness the independence of all sets in $\mathcal{I}$ (the family of independent sets of the transversal matroid at hand).

Our main theorem is in fact slightly stronger than described above. Suppose that for algorithmic purposes, we would like to obtain a union representation of a structure where all independent sets are also independent in our transversal matroid of interest (i.e., we do not introduce false independent sets), and where all independent sets *of size up to k* in our transversal matroid are also independent in our structure. In other words, we are pleased with a structure that may "throw away" some large independent sets. For this purpose, we introduce the appropriate notion of a structure that is in fact a weakening of the well-known notion of a k-truncation of a matroid. Such a structure is useful for applications to problems where solution size is at most k, which can be significantly smaller than the rank of the matroid. Then, our computation of union representation runs in time quasipolynomial in k rather than the rank, and thereby enables the design of parameterized algorithms with respect to k.

Applications. Our main result also has applications in Parameterized Complexity. First, it yields a fast computation of representative sets (integrated in the framework of [5, 12]). Formally, given a matroid $M = (E, \mathcal{I})$ and a family $\mathcal{S}$ of subsets of E, a subfamily $\widehat{\mathcal{S}} \subseteq \mathcal{S}$ is *q-representative* for $\mathcal{S}$ if the following holds: for every set $Y \subseteq E$ of size at most q, if there is a set $X \in \mathcal{S}$ disjoint from Y with $X \cup Y \in \mathcal{I}$, then there is a set $\widehat{X} \in \widehat{\mathcal{S}}$ disjoint from Y with $\widehat{X} \cup Y \in \mathcal{I}$. Fomin et al. [5, 12] showed that if one is given a representation of the matroid M, then small representative sets can be computed efficiently (see Section 4). This computation has led, since its introduction in 2013, to the development of dozens of parameterized algorithms that are the current state-of-the-art for their respective problems. We observe that our computation of a union representation of a transversal matroid can be composed with the algorithm of Fomin et al. [5, 12] to obtain small representative sets efficiently also with respect to transversal matroids. As matching constraints naturally arise in various scenarios, we find it important that the powerful tool of representative sets can now be employed to handle them as well.

To illustrate the usefulness of the computation above using a simple didactic example, we consider the List k-Path problem, informally defined as follows. The input consists of an undirected graph G such that each of its vertices has its own list of *compatible* colors, and the objective is to determine whether G has a (simple) path on k vertices such that one can assign a compatible color to each vertex on this path to make it colorful. This problem is a natural generalization of the classical k-Path problem to the presence of color lists, and it was studied (in a slightly more general form) in [18] in the setting of randomized algorithms. Previously, to solve this problem using representative sets, we remark that one would have to use the direct sum of two uniform matroids, one to ensure distinctness of vertices and one to ensure distinctness of colors, which would result in running time $\mathcal{O}(6.86^k \cdot n^{\mathcal{O}(1)})$ – this would be, to the best of our knowledge, the state-of-the-art. We show that simply by using a transversal matroid rather than a direct sum of two uniform matroids, one obtains a running time of $\mathcal{O}(5.18^k \cdot n^{\mathcal{O}(1)})$.

We stress that the choice of List k-Path is only done for illustrative purposes. Indeed, by *only* considering transversal matroids rather than direct sums of two uniform matroids, a wide variety of problems can now immediately be solved in time $\mathcal{O}(5.18^k \cdot n^{\mathcal{O}(1)})$ rather than $\mathcal{O}(6.86^k \cdot n^{\mathcal{O}(1)})$ in the presence of color lists. Such problems have been studied to model scenarios where pairs of elements to be matched may not be identical but only similar, and color lists aim to describe the set of compatible elements associated with each element. This includes, for example, graph problems such as subcases of subgraph isomorphism, which are of relevance (in the presence of color lists) to bioinformatics [19, 21, 23, 3]. We remark that this approach is applicable not only to graph problems, but also to various packing and matching problems (such as those studied in [8]) in the presence of list colors. The Graph Motif problem, in particular, was extensively studied also in the presence of color lists (see [20, 1, 9, 11] and references therein), where the previous fastest deterministic algorithm run in time $\mathcal{O}(6.86^k \cdot n^{\mathcal{O}(1)})$ [20]. By simply using a transversal matroid rather than a direct sum of two uniform matroids in the algorithm of [20], we immediately derive an improved running time of $\mathcal{O}(5.18^k \cdot n^{\mathcal{O}(1)})$.

Proofs of results marked by an asterisk ('*') are omitted.

2 Preliminaries

Given $t \in \mathbb{N}$, we use $[t]$ as a shorthand for $\{1, 2, \ldots, t\}$. Given a function $f : A \rightarrow B$ and a subset $A' \subseteq A$, we denote $f(A') = \{f(a) : a \in A'\}$, and we define $f|_{A'}$ as the restriction of f to A'. We slightly abuse notation, and given a function $g : A \rightarrow \mathbb{N}$, called a *weight function*, and a subset $A' \subseteq A$, we denote $g(A') = \sum_{a \in A'} g(a)$. Whenever we refer to a function that is a weight function, we use the second notation.

Given a graph G, we say that (A, B) is a *vertex bipartition* of G if it is a partition of $V(G)$ such that A and B are independent sets. Moreover, we say that G is a *bipartite graph* if it has a vertex bipartition. A *matching* μ is a family of pairwise-disjoint subsets of $E(G)$.

2.1 Matroids

Let us begin by presenting the definition of an independence system.

▶ **Definition 1** (Independence System). A pair $P = (\mathcal{I}, E)$, where E is a ground set and $\mathcal{I}$ is a family of subsets of E (called *independent sets*), is an *independence system* if it satisfies the following conditions:

(I1) $\phi \in \mathcal{I}$.

(I2) If $X \subseteq Y$ and $Y \in \mathcal{I}$, then $X \in \mathcal{I}$.

A matroid is an independence system with an additional property, formally defined as follows.

▶ **Definition 2** (Matroid). An independence system $M = (\mathcal{I}, E)$ is a *matroid* if it satisfies the following condition:

(I3) If $X, Y \in \mathcal{I}$ and $|X| < |Y|$, then there exists $e \in (Y \setminus X)$ such that $X \cup \{e\} \in \mathcal{I}$.

The *rank* of M is the maximum size of a set in $\mathcal{I}$.

We remark that conditions (I1), (I2) and (I3) are called *matroid axioms*. We say that two independence systems $P = (E, \mathcal{I})$ and $P' = (E', \mathcal{I}')$ are *isomorphic* if there exists a bijection $\varphi : E \rightarrow E'$ such that for every $X \subseteq E$, $X \in \mathcal{I}$ if and only if $\varphi(X) \in \mathcal{I}'$. In this paper, we are specifically interested in transversal matroids, defined as follows.

▶ **Definition 3** (Transversal Matroid). Let G be a bipartite graph with a fixed vertex bipartition (A, B). The *transversal matroid* M of G is the pair $(A, \mathcal{I})$ where $\mathcal{I}$ is the family that consists of every subset $X \subseteq A$ such that there exists a matching that saturates X.

It is well-known that a transversal matroid is indeed a matroid [17]. Having a *representation* of a matroid, given by a matrix that compactly encodes the matroid, is a central component in many algorithmic applications. Matroids having a representation are called *linear*, as formally defined below.

▶ **Definition 4.** Let A be a matrix over an arbitrary field $\mathbb{F}$, and let C be the set of columns of A. The *matroid represented by* A is the pair $M = (C, \mathcal{I})$, where a subset $X \subseteq C$ belongs to $\mathcal{I}$ if and only if the columns in X are linearly independent over $\mathbb{F}$.

It is well-known that the pair $M = (C, \mathcal{I})$ in Definition 4 indeed defines a matroid [17].

▶ **Definition 5** (Linear Matroid, Representation)**.** A matroid $M = (E, \mathcal{I})$ is a *linear matroid* if there exists a matrix A, called a *representation* of M, such that M and the matroid represented by A are isomorphic. Furthermore, M is *representable over a field* $\mathbb{F}$ if it has a representation A over $\mathbb{F}$.

We introduce a generalization of the concepts above, resulting in the notion of *union representation*, which is sufficient for many algorithmic purposes and may be of independent interest.

▶ **Definition 6.** Let $A_1, A_2, \ldots, A_t$ be t matrices over an arbitrary field $\mathbb{F}$, let E be a ground set, and for all $i \in [t]$, let $\varphi_i : E \to C_i$ be a bijection, where C_i is the set of columns of A_i. The *independence system represented by* $(E, \{A_i, \varphi_i\}|_{i\in[t]})$ is given by $P = (E, \mathcal{I})$, where a subset $X \subseteq E$ belongs to $\mathcal{I}$ if and only if there exists $i \in [t]$ such that the columns in $\varphi_i(X)$ are linearly independent over $\mathbb{F}$.

It should be clear that $P = (E, \mathcal{I})$ in Definition 6 is indeed an independence system, but we remark that it might not be a matroid since it may not satisfy axiom (I3) in Definition 2. If the bijective functions φ_i, $i \in [t]$, are clear from context, we do not specify them explicitly.

▶ **Definition 7** (Union Representation)**.** Let $P = (E, \mathcal{I})$ be an independence system. Let $(E, \{A_i, \varphi_i\}|_{i\in[t]})$ be defined as in Definition 6. Then, $(E, \{A_i, \varphi_i\}|_{i\in[t]})$ is a *t-union representation* of P if the independence system represented by $(E, \{A_i, \varphi_i\}|_{i\in[t]})$ is isomorphic to P. Furthermore, we say that $(E, \{A_i, \varphi_i\}|_{i\in[t]})$ is defined over $\mathbb{F}$ if $\mathbb{F}$ is the field over which $A_1, A_2, \ldots, A_t$ are defined.

Finally, we present the definition of a *k-truncation* of a matroid, which comes in handy in various algorithmic applications. Our main result directly captures structures that we call *weak k-truncations of transversal matroids* rather than only transversal matroids, and hence we present it in this context (see Section 3). Let us first present the standard definition of truncation.

▶ **Definition 8** (Truncation)**.** Let $M = (E, \mathcal{I})$ be a matroid, and let $k \in \mathbb{N}$. The *k-truncation* of M is the matroid $M' = (E, \mathcal{I}')$ where $\mathcal{I}' = \{I \in \mathcal{I} : |I| \leq k\}$.

We now turn the present the definition with whom we will be working. This definition is sufficient for our algorithmic purposes, since the motivation underlying the use of a k-truncation is to obtain a matrix of small rank (i.e. $k^{\mathcal{O}(1)}$), which is also attainable by a weak k-truncation.

▶ **Definition 9** (Weak Truncation)**.** Let $M = (E, \mathcal{I})$ be a matroid, and let $k \in \mathbb{N}$. A *weak k-truncation* of M is an independence system $P' = (E, \mathcal{I}')$ where $\{I \in \mathcal{I} : |I| \leq k\} \subseteq \mathcal{I}' \subseteq \mathcal{I}$.

2.2 Isolation

For the sake of clarity, let us first introduce the following notation. Given $n \in \mathbb{N}$, let G be a bipartite graph with a fixed bipartition (A, B) such that $|A|, |B| \leq n$, and fixed injective functions $\gamma_A : A \to [n]$ and $\gamma_B : B \to [n]$. Given a weight function $w : [n] \times [n] \to \mathbb{N}$, we define the weight of an edge $\{a, b\} \in E(G)$, where $a \in A$ and $b \in B$, by $\widetilde{w}(\{a, b\}) = w(\gamma_A(a), \gamma_B(b))$. Thus, $\widetilde{w}$ can be thought of as a function from $E(G)$ to $\mathbb{N}$. Let us remind that for a subset $U \subseteq E(G)$, $\widetilde{w}(U) = \sum_{e \in U} \widetilde{w}(e)$.

We remark that we need to define a weight function via injective functions of the form γ_A and γ_B as above (rather than letting the domain directly be an edge set) in order to prove the correctness of our main result, particularly in its general form. Now, for perfect matchings, isolating weight functions are defined as follows.

▶ **Definition 10** (Isolating Weight Function)**.** Let G be a bipartite graph with a fixed bipartition (A, B) such that $|A|, |B| \leq n$, and fixed injective functions $\gamma_A : A \to [n]$ and $\gamma_B : B \to [n]$. A weight function $w : [n] \times [n] \to \mathbb{N}$ is *isolating* if it satisfies the following condition: If G has a perfect matching, then G also has a unique perfect matching μ of minimum weight (i.e. for every perfect matching $\mu' \neq \mu$, $\widetilde{w}(\mu) < \widetilde{w}(\mu')$).

Such isolating weight functions are particularly relevant to the detection of a perfect matching. To see this, we first need to define the matrix associated with an isolating weight function.

▶ **Definition 11.** Let G be a bipartite graph with a fixed bipartition (A, B) such that $|A|, |B| \leq n$, and fixed injective functions $\gamma_A : A \to [n]$ and $\gamma_B : B \to [n]$. In addition, let $w : [n] \times [n] \to \mathbb{N}$ be a weight function. Then, $W_{(G,w)}$ is the matrix on $|A|$ columns indexed by the vertices in A and $|B|$ rows indexed by the vertices in B, where

$$W_{(G,w)}[b, a] = \begin{cases} 2^{\widetilde{w}(\{b,a\})} & \text{if } \{b, a\} \in E(G) \\ 0 & \text{otherwise} \end{cases}$$

for all $a \in A$ and $b \in B$.

The following well-known result, due to Mulmuley et al. [15], reveals a connection between isolating weight functions, determinants and perfect matchings.

▶ **Proposition 1** ([15])**.** *Let G be a bipartite graph with a fixed bipartition (A, B) such that $|A| = |B| \leq n$, and fixed injective functions $\gamma_A : A \to [n]$ and $\gamma_B : B \to [n]$. In addition, let $w : [n] \times [n] \to \mathbb{N}$ be a weight function. If $\mathsf{det}(W_{(G,w)}) \neq 0$, then G has a perfect matching. Moreover, if w is isolating and G has a perfect matching, then $\mathsf{det}(W_{(G,w)}) \neq 0$.*

Fenner et al. [4] presented a (deterministic) computation of a collection of weight functions that, for any bipartite graph, has at least one isolating weight function. Formally,

▶ **Definition 12** (Isolating Collection)**.** Let $n \in \mathbb{N}$. An *n-isolating collection* is a set $\mathcal{W}_n$ of weight functions $w : [n] \times [n] \to \mathbb{N}$ with the following property: For any bipartite graph G with a fixed bipartition (A, B) such that $|A|, |B| \leq n$, and fixed bijective functions $\gamma_A : A \to [n]$ and $\gamma_B : B \to [n]$, there exists a weight function $w \in \mathcal{W}_n$ such that w is isolating.

▶ **Proposition 2** ([4])**.** *Let $n \in \mathbb{N}$. An n-isolating collection $\mathcal{W}_n$ of $2^{\mathcal{O}(\log^2 n)}$ weight functions with the following property can be obtained in time $2^{\mathcal{O}(\log^2 n)}$: For any weight function $w \in \mathcal{W}_n$, every weight assigned by w can be represented (in binary) using $\mathcal{O}(\log^2 n)$ bits.*

2.3 Splitters, Representative Families

Splitters are well-known tools in derandomization, formally defined as follows.

▶ **Definition 13** (Splitter). Let $n, k, \ell \in \mathbb{N}$ where $k \leq \ell$. An (n, k, ℓ)-*splitter* is a family $\mathcal{F}$ of functions from $[n]$ to $[\ell]$ such that for every $S \subseteq [n]$ of size k, there is a function $f \in \mathcal{F}$ that satisfies $f(i) \neq f(j)$ for all distinct $i, j \in S$.

We are specifically interested in an (n, k, k^2)-splitter. The following lemma asserts that such a small splitter can be computed efficiently.

▶ **Proposition 3** ([16]). *Given $n, k \in \mathbb{N}$, an (n, k, k^2)-splitter of size $k^{\mathcal{O}(1)} \log n$ can be constructed in time $k^{\mathcal{O}(1)} n \log n$.*

The notion of a representative family (implicitly linked to that of a splitter), introduced by Fomin et al. [5], plays a central role in the design of fast deterministic parameterized algorithms.

▶ **Definition 14** (Representative Family). Given a matroid $M = (E, \mathcal{I})$ and a family $\mathcal{S}$ of subsets of E, a subfamily $\widehat{\mathcal{S}} \subseteq \mathcal{S}$ is q-*representative* for $\mathcal{S}$, denoted by $\widehat{\mathcal{S}} \subseteq^q_{rep} \mathcal{S}$, if the following holds: for every set $Y \subseteq E$ of size at most q, if there is a set $X \in \mathcal{S}$ disjoint from Y with $X \cup Y \in \mathcal{I}$, then there is a set $\widehat{X} \in \widehat{\mathcal{S}}$ disjoint from Y with $\widehat{X} \cup Y \in \mathcal{I}$.

3 Representation

The purpose of this section is to compute a union representation of a transversal matroid consisting of a quasipolynomial (in the rank of the matroid) number of matrices. As our proof directly works for weak truncations of transversal matroids rather than only transversal matroids, we present the statement of our result in the following more general form, and the objective above as a corollary.

▶ **Theorem 15.** *Let G be an n-vertex bipartite graph with a fixed vertex bipartition (A, B), and let $r \in \mathbb{N}$. A t-union representation $(E, \{A_i, \varphi_i\}|_{i\in[t]})$ of some weak r-truncation of the transversal matroid of G over $\mathbb{Q}$, where $t = 2^{\mathcal{O}(\log^2 r)} \log n$ and every entry in A_i, $i \in [t]$, is an integer of bit-length $2^{\mathcal{O}(\log^2 r)}$, can be computed in time $2^{\mathcal{O}(\log^2 r)} n \log n$.*

Let us remind that the maximum size of a matching in a graph G is denoted by $\kappa(G)$, and that it upper bounds the rank of the transversal matroid of G. In the theorem above, if $r = \kappa(G)$, then any weak r-truncation of the transversal matroid of G is equal to the transversal matroid of G. Hence, we have the following corollary.

▶ **Corollary 16.** *Let G be an n-vertex bipartite graph with a fixed vertex bipartition (A, B), and denote $r = \kappa(G)$. A t-union representation $(E, \{A_i, \varphi_i\}|_{i\in[t]})$ of the transversal matroid of G over $\mathbb{Q}$, where $t = 2^{\mathcal{O}(\log^2 r)} \log n$ and every entry in A_i, $i \in [t]$, is an integer of bit-length $2^{\mathcal{O}(\log^2 r)}$, can be computed in time $2^{\mathcal{O}(\log^2 r)} n \log n$.*

For the sake of clarity, we first analyze the special case where $|A|, |B| \leq (2r)^2$. More precisely, we prove a weaker version of Corollary 16, but it is conceptually convenient to think of this proof as the above special case given that we later map integers in $[2n]$ to integers in $[(2r)^2]$. Then, we present a more involved construction that handles the general case.

3.1 Special Case

For our analysis of the special case, we introduce the following definition.

▶ **Definition 17.** Let G be a bipartite graph with a fixed vertex bipartition (A, B) such that $|A|, |B| \leq n$, and fixed injective functions $\gamma_A : A \to [n]$ and $\gamma_B : B \to [n]$. A weight function $w : [n] \times [n] \to \mathbb{N}$ is *good for a subset* $X \subseteq A$ if $\mathsf{det}(W_{(G,w)}[Y, X]) \neq 0$ for some $Y \subseteq B$.

The heart of the proof of the special case is based on the two following lemmas.

▶ **Lemma 18 (*).** *Let G be a bipartite graph with a fixed vertex bipartition (A, B) such that $|A|, |B| \leq n$, and fixed injective functions $\gamma_A : A \to [n]$ and $\gamma_B : B \to [n]$. In addition, let $\mathcal{W}_n$ be an n-isolating collection. For every subset $X \subseteq A$, if X is independent in the transversal matroid of G, then there exists $w \in \mathcal{W}_n$ that is good for X.*

▶ **Lemma 19 (*).** *Let G be a bipartite graph with a fixed vertex bipartition (A, B) such that $|A|, |B| \leq n$, and fixed injective functions $\gamma_A : A \to [n]$ and $\gamma_B : B \to [n]$. In addition, let $\mathcal{W}_n$ be an n-isolating collection. For every subset $X \subseteq A$, if there exists $w \in \mathcal{W}_n$ that is good for X, then X is independent in the transversal matroid of G.*

Lemmas 18 and 19 lead us to the following result.

▶ **Lemma 20 (*).** *Let G be a bipartite graph with a fixed vertex bipartition (A, B) such that $|A|, |B| \leq n$, and fixed injective functions $\gamma_A : A \to [n]$ and $\gamma_B : B \to [n]$. In addition, let $\mathcal{W}_n$ be an n-isolating collection. Then, $(A, \{W_{(G,w)}\}|_{w \in \mathcal{W}_n})$ is a t-union representation of the transversal matroid of G over $\mathbb{Q}$, where $t = |\mathcal{W}_n|$.*

Due to Proposition 2, we have the following consequence of Lemma 20.

▶ **Lemma 21 (*).** *Let G be an n-vertex bipartite graph with a fixed vertex bipartition (A, B). A t-union representation $(E, \{A_i, \varphi_i\}|_{i \in [t]})$ of the transversal matroid of G over $\mathbb{Q}$, where $t = 2^{\mathcal{O}(\log^2 n)}$ and every entry in A_i, $i \in [t]$, is an integer of bit-length $2^{\mathcal{O}(\log^2 n)}$, can be computed in time $2^{\mathcal{O}(\log^2 n)}$.*

3.2 General Case

We begin by adapting the definition of the matrix $W_{(G,w)}$ to the presence of a "splitter functions", which is a function from $[2n]$ to $[(2r)^2]$ where $n, r \in \mathbb{N}$ will be clear from context.

▶ **Definition 22.** Let G be a bipartite graph with a fixed bipartition (A, B) such that $|A|, |B| \leq n$, and fixed injective functions $\gamma_A : A \to [n]$ and $\gamma_B : B \to [n]$. In addition, let $w : [(2r)^2] \times [(2r)^2] \to \mathbb{N}$ be a weight function and $f : [2n] \to [(2r)^2]$ be a splitter function for some $r \in \mathbb{N}$. Then, $W_{(G,w,f)}$ is the matrix on $|A|$ columns indexed by the vertices in A and $|B|$ rows indexed by the vertices in B, where

$$W_{(G,w,f)}[b, a] = \begin{cases} 2^{w(f(\gamma_A(a)), f(n+\gamma_B(b)))} & \text{if } \{b, a\} \in E(G) \\ 0 & \text{otherwise} \end{cases}$$

for all $a \in A$ and $b \in B$.

In order to proceed, we need to generalize Definition 17 to pairs of a weight function and a splitter function.

▶ **Definition 23.** Let G be a bipartite graph with a fixed vertex bipartition (A, B) such that $|A|, |B| \leq n$, and fixed injective functions $\gamma_A : A \to [n]$ and $\gamma_B : B \to [n]$. In addition, let $r \in \mathbb{N}$. For a weight function $w : [(2r)^2] \times [(2r)^2] \to \mathbb{N}$ and a splitter function $f_A : [2n] \to [(2r)^2]$, the pair (w, f) is *good for a subset* $X \subseteq A$ if $\mathsf{det}(W_{(G,w,f)}[Y, X]) \neq 0$ for some $Y \subseteq B$.

We first need to establish the following lemma.

▶ **Lemma 24 (*).** *Let G be a bipartite graph with a fixed vertex bipartition (A, B) such that $|A|, |B| \leq n$, and fixed injective functions $\gamma_A : A \to [n]$ and $\gamma_B : B \to [n]$. In addition, let $\mathcal{W}$ be a $(2r)^2$-isolating collection, and $\mathcal{F}$ be a $(2n, 2r, (2r)^2)$-splitter for some $r \in \mathbb{N}$. For every subset $X \subseteq A$ of size at most r, if X is independent in the transversal matroid of G, then there exist $w \in \mathcal{W}$ and $f \in \mathcal{F}$ such that (w, f) is good for X.*

▶ **Lemma 25 (*).** *Let G be a bipartite graph with a fixed vertex bipartition (A, B) such that $|A|, |B| \leq n$, and fixed injective functions $\gamma_A : A \to [n]$ and $\gamma_B : B \to [n]$. In addition, let $\mathcal{W}$ be a $(2r)^2$-isolating collection, and $\mathcal{F}$ be a $(2n, 2r, (2r)^2)$-splitter for some $r \in \mathbb{N}$. For every subset $X \subseteq A$, if there exist $w \in \mathcal{W}$ and $f \in \mathcal{F}$ such that (w, f) is good for X, then X is independent in the transversal matroid of G.*

Lemmas 24 and 25 lead us to the following result.

▶ **Lemma 26 (*).** *Fix $r \in \mathbb{N}$. Let G be a bipartite graph with a fixed vertex bipartition (A, B) such that $|A|, |B| \leq n$, and fixed injective functions $\gamma_A : A \to [n]$ and $\gamma_B : B \to [n]$. In addition, let $\mathcal{W}$ be a $(2r)^2$-isolating collection, and $\mathcal{F}$ be a $(2n, 2r, (2r)^2)$-splitter. Then, $(A, \{W_{(G,w,f)}\}|_{w\in\mathcal{W}, f\in\mathcal{F}})$ is a t-union representation of some weak r-truncation of the transversal matroid of G over $\mathbb{Q}$, where $t = |\mathcal{W}| \cdot |\mathcal{F}|$.*

We are now ready to prove Theorem 15.

Proof. First, we apply Proposition 2 to obtain a $(2r)^2$-isolating collection $\mathcal{W}$ of size $2^{\mathcal{O}(\log^2 r)}$ in time $2^{\mathcal{O}(\log^2 r)}$. Second, we apply Proposition 13 to obtain a $(2n, 2r, (2r)^2)$-splitter $\mathcal{F}$ of size $r^{\mathcal{O}(1)} \log n$ in time $r^{\mathcal{O}(1)} n \log n$. We select arbitrary bijective functions $\gamma_A : A \to [|A|]$ and $\gamma_B : B \to [|B|]$. By Lemma 26, $(A, \{W_{(G,w,f)}\}|_{w\in\mathcal{W}, f\in\mathcal{F}})$ is a t-union representation of some weak r-truncation of the transversal matroid of G over $\mathbb{Q}$, where $t = |\mathcal{W}| \cdot |\mathcal{F}| = 2^{\mathcal{O}(\log^2 r)} \log n$. By Proposition 2 and Definition 11, every entry in $W_{(G,w,f)}$, $w \in \mathcal{W}$ and $f \in \mathcal{F}$, is an integer of bit-length $2^{\mathcal{O}(\log^2 r)}$. Thus, the time to construct $(A, \{W_{(G,w,f)}\}|_{w\in\mathcal{W}, f\in\mathcal{F}})$ is bounded by $2^{\mathcal{O}(\log^2 r)} n \log n$. This concludes the proof. ◀

4 Representative Families

In this section, we give applications of Theorem 15 in the design of parameterized algorithms. First, we give a fast deterministic algorithm to compute representative families over a transversal matroid. Prior to our work, only randomized algorithms were known from the works of Fomin et al. [5] and Lokshtanov et al. [12], since no fast deterministic algorithm was known for computing a linear representation of transversal matroids. Later in this section, we will use this deterministic algorithm to give a deterministic parameterized algorithm for the List k-Path problem. We remind that, as explained in Introduction, we selected List k-Path for illustrative purposes, and that the approach described to solve it is readily applicable to problems such as Graph Motif, d-Dimensional k-Matching and d-Set k-Packing in the presence of color lists.

We begin by stating the following known results about the computation of representative families over linear matroids.

▶ **Proposition 4** ([12]). *Let $M = (E, \mathcal{I})$ be a linear matroid of rank n, and let $\mathcal{S}$ be a family of ℓ independent sets, each of size p. Let A be an $n \times |E|$ matrix representing M over a field $\mathbb{F}$, and let $\omega < 2.373$ be the exponent of matrix multiplication [6]. Then, there are deterministic algorithms computing $\widehat{\mathcal{S}} \subseteq^q_{rep} \mathcal{S}$ as follows.*

(i) *A family $\widehat{\mathcal{S}}$ of size $\binom{p+q}{p}$ in $\mathcal{O}\left(\binom{p+q}{p}^2 \ell p^3 n^2 + \ell \binom{p+q}{q}^\omega np\right) + (n + |E|)^{\mathcal{O}(1)}$ operations over $\mathbb{F}$.*

(ii) *A family $\widehat{\mathcal{S}}$ of size $np\binom{p+q}{p}$ in $\mathcal{O}\left(\binom{p+q}{p} \ell p^3 n^2 + \ell \binom{p+q}{q}^{\omega-1} (pn)^{\omega-1}\right) + (n + |E|)^{\mathcal{O}(1)}$ operations over $\mathbb{F}$.*

▶ **Proposition 5** ([5]). *Let $M = (E, \mathcal{I})$ be a matroid and $\mathcal{S}$ be a subset of $\mathcal{I}$. If $\mathcal{S}' \subseteq^q_{rep} \mathcal{S}$ and $\widehat{\mathcal{S}} \subseteq^q_{rep} \mathcal{S}'$, then $\widehat{\mathcal{S}} \subseteq^q_{rep} \mathcal{S}$.*

Now we will apply the above results to prove the following theorem.

▶ **Theorem 27** (*). *Let $M = (E, \mathcal{I})$ be a transversal matroid of rank n and let $\mathcal{S}$ be a family of ℓ independent sets, each of size p. Let q be a positive integer, $r = p + q$ and ω be the exponent of matrix multiplication. Then, there are deterministic algorithms computing $\widehat{\mathcal{S}} \subseteq^q_{rep} \mathcal{S}$ as follows.*

(i) *A family $\widehat{\mathcal{S}}$ of size $2^{\mathcal{O}(\log^2 r)}\binom{r}{p} \log n$ in time $\left(\binom{r}{p}^2 \ell p^3 n^2 + \ell \binom{r}{q}^\omega np\right) 2^{\mathcal{O}(\log^2 r)} \log n + (n + |E|)^{\mathcal{O}(1)} 2^{\mathcal{O}(\log^2 r)}$.*

(ii) *A family $\widehat{\mathcal{S}}$ of size $2^{\mathcal{O}(\log^2 r)}\binom{r}{p} np \log n$ in time $\left(\binom{r}{p} \ell p^3 n^2 + \ell \binom{r}{q}^{\omega-1} (pn)^{\omega-1}\right)$ $2^{\mathcal{O}(\log^2 r)} \log n + (n + |E|)^{\mathcal{O}(1)} 2^{\mathcal{O}(\log^2 r)}$.*

Now we will use Theorem 27 to design a deterministic algorithm for List k-Path, which is defined as follows.

List k-Path	**Parameter:** k
Input: A graph G, a set of colors C, a function $L : V(G) \to 2^C$, and $k \in \mathbb{N}$,	
Question: Is there a path P on k vertices and an injective map $g : V(P) \to C$ such that $g(v) \in L(v)$ for all $v \in V(P)$?	

Note that, unlike k-Path where the objective is to check whether there is a path P on k vertices in a given input graph, in List k-Path we must also find an injective map g that assigns distinct colors to the vertices on P from their respective lists of colors. Towards that, we create an auxiliary bipartite graph H with bipartition $V(G) \uplus C$, such that for each $v \in V(G)$, the neighborhood $N_H(v)$ of v in H is $L(v)$. The following lemma states that any solution to the instance (G, C, L, k) is also an independent set in the transversal matroid of H.

▶ **Lemma 28** (*). *Let (G, C, L, k) be an instance of* List k-Path*. Let $H = (V(G) \uplus C, F)$ be a bipartite graph such that $N_H(v) = L(v)$ for all $v \in V(G)$. Let P be a path on k vertices in G. Then, P is a solution to* List k-Path *if and only if $V(P)$ is an independent set in the transversal matroid M of H over the ground set $V(G)$.*

Using Lemma 28, a dynamic programming (DP) algorithm for List k-Path can be designed using representative families. This algorithm will follow the outline of the algorithm of Fomin et al. [5] for k-Path. In the rest of this section, we will present this DP algorithm for List k-Path. Recall that (G, C, L, k) is the input and $M = (V(G), \mathcal{I})$ is the transversal matroid of H over the ground set $V(G)$, where $H = (V(G) \uplus C, F)$ is the bipartite graphs such that $N_H(v) = L(v)$ for all $v \in V(G)$. For any $i \in [k]$ and $v \in V(G)$, define the following.

$$\mathcal{P}_v^i = \Big\{ X \;\Big|\; X \subseteq V(G),\ v \in X,\ |X| = i, X \in \mathcal{I},\ \text{and } G \text{ has a path of length } i-1 \text{ whose vertex set is precisely } X \text{ and whose end vertex is } v \Big\}$$

The following lemma gives an efficient computation of $\widehat{\mathcal{P}_v^i} \subseteq_{rep}^{k-i} \mathcal{P}_v^i$ for all $i \in [k]$ where the underlying matroid is M, i.e. the transversal matroid of H over the ground set $V(G)$.

▶ **Lemma 29 (*).** *For every $i \in [k]$ and $v \in V(G)$, $\widehat{\mathcal{P}_v^i} \subseteq_{rep}^{k-i} \mathcal{P}_v^i$ of size $2^{\mathcal{O}(\log^2 k)} \binom{k}{i} n \cdot i \log n$ can be computed in time $2^{\omega k} 2^{\mathcal{O}(\log^2 k)} n^{\mathcal{O}(1)}$.*

▶ **Theorem 30 (*).** List k-Path *can be solved in time $2^{\omega k} 2^{\mathcal{O}(\log^2 k)} n^{\mathcal{O}(1)}$, where $\omega < 2.373$ is the exponent of matrix multiplication [6].*

We remark that in the theorem above, $2^{\omega k} 2^{\mathcal{O}(\log^2 k)} n^{\mathcal{O}(1)} = 5.18^k n^{\mathcal{O}(1)}$. If the computation in Proposition 4 is sped-up or the bound on ω is improved, then our algorithm is automatically sped-up as well.

References

1 Andreas Björklund, Petteri Kaski, and Lukasz Kowalik. Constrained multilinear detection and generalized graph motifs. *Algorithmica*, 74(2):947–967, 2016. doi:10.1007/s00453-015-9981-1.

2 R A DeMillo and R J Lipton. A probabilistic remark on algebraic program testing. *Inform. Process Lett.*, 7(4):193–195, 1978.

3 Banu Dost, Tomer Shlomi, Nitin Gupta, Eytan Ruppin, Vineet Bafna, and Roded Sharan. Qnet: A tool for querying protein interaction networks. *Journal of Computational Biology*, 15(7):913–925, 2008. doi:10.1089/cmb.2007.0172.

4 Stephen A. Fenner, Rohit Gurjar, and Thomas Thierauf. Bipartite perfect matching is in quasi-nc. In Daniel Wichs and Yishay Mansour, editors, *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 754–763. ACM, 2016. doi:10.1145/2897518.2897564.

5 Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, and Saket Saurabh. Efficient computation of representative families with applications in parameterized and exact algorithms. *J. ACM*, 63(4):29:1–29:60, 2016. doi:10.1145/2886094.

6 François Le Gall. Powers of tensors and fast matrix multiplication. In Katsusuke Nabeshima, Kosaku Nagasaka, Franz Winkler, and Ágnes Szántó, editors, *International Symposium on Symbolic and Algebraic Computation, ISSAC '14, Kobe, Japan, July 23-25, 2014*, pages 296–303. ACM, 2014. doi:10.1145/2608628.2608664.

7 Shafi Goldwasser and Ofer Grossman. Bipartite perfect matching in pseudo-deterministic NC. In Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl, editors, *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10-14, 2017, Warsaw, Poland*, volume 80 of *LIPIcs*, pages 87:1–87:13. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017. doi:10.4230/LIPIcs.ICALP.2017.87.

8 Prachi Goyal, Neeldhara Misra, Fahad Panolan, and Meirav Zehavi. Deterministic algorithms for matching and packing problems based on representative sets. *SIAM J. Discrete Math.*, 29(4):1815–1836, 2015. doi:10.1137/140981290.

9 Sylvain Guillemot and Florian Sikora. Finding and counting vertex-colored subtrees. *Algorithmica*, 65(4):828–844, 2013. doi:10.1007/s00453-011-9600-8.

10 Rohit Gurjar and Thomas Thierauf. Linear matroid intersection is in quasi-nc. In Hamed Hatami, Pierre McKenzie, and Valerie King, editors, *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 821–830. ACM, 2017. doi:10.1145/3055399.3055440.

11 Ioannis Koutis. Constrained multilinear detection for faster functional motif discovery. *Inf. Process. Lett.*, 112(22):889–892, 2012. doi:10.1016/j.ipl.2012.08.008.

12 Daniel Lokshtanov, Pranabendu Misra, Fahad Panolan, and Saket Saurabh. Deterministic truncation of linear matroids. In Magnús M. Halldórsson, Kazuo Iwama, Naoki Kobayashi, and Bettina Speckmann, editors, *Automata, Languages, and Programming - 42nd International Colloquium, ICALP 2015, Kyoto, Japan, July 6-10, 2015, Proceedings, Part I*, volume 9134 of *Lecture Notes in Computer Science*, pages 922–934. Springer, 2015. doi:10.1007/978-3-662-47672-7_75.

13 Dániel Marx. A parameterized view on matroid optimization problems. *Theor. Comput. Sci.*, 410(44):4471–4479, 2009. doi:10.1016/j.tcs.2009.07.027.

14 Pranabendu Misra, Fahad Panolan, M. S. Ramanujan, and Saket Saurabh. Linear representation of transversal matroids and gammoids parameterized by rank. In Yixin Cao and Jianer Chen, editors, *Computing and Combinatorics - 23rd International Conference, COCOON 2017, Hong Kong, China, August 3-5, 2017, Proceedings*, volume 10392 of *Lecture Notes in Computer Science*, pages 420–432. Springer, 2017. doi:10.1007/978-3-319-62389-4_35.

15 Ketan Mulmuley, Umesh V. Vazirani, and Vijay V. Vazirani. Matching is as easy as matrix inversion. *Combinatorica*, 7(1):105–113, 1987. doi:10.1007/BF02579206.

16 Moni Naor, Leonard J. Schulman, and Aravind Srinivasan. Splitters and near-optimal derandomization. In *36th Annual Symposium on Foundations of Computer Science, Milwaukee, Wisconsin, 23-25 October 1995*, pages 182–191. IEEE Computer Society, 1995. doi:10.1109/SFCS.1995.492475.

17 J.G. Oxley. *Matroid Theory*. Oxford graduate texts in mathematics. Oxford University Press, 2006.

18 Fahad Panolan and Meirav Zehavi. Parameterized algorithms for list k-cycle. In Akash Lal, S. Akshay, Saket Saurabh, and Sandeep Sen, editors, *36th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2016, December 13-15, 2016, Chennai, India*, volume 65 of *LIPIcs*, pages 22:1–22:15. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016. doi:10.4230/LIPIcs.FSTTCS.2016.22.

19 Ron Y. Pinter, Oleg Rokhlenko, Esti Yeger Lotem, and Michal Ziv-Ukelson. Alignment of metabolic pathways. *Bioinformatics*, 21(16):3401–3408, 2005. doi:10.1093/bioinformatics/bti554.

20 Ron Y. Pinter, Hadas Shachnai, and Meirav Zehavi. Deterministic parameterized algorithms for the graph motif problem. *Discrete Applied Mathematics*, 213:162–178, 2016. doi:10.1016/j.dam.2016.04.026.

21 Ron Y. Pinter and Meirav Zehavi. Algorithms for topology-free and alignment network queries. *J. Discrete Algorithms*, 27:29–53, 2014. doi:10.1016/j.jda.2014.03.002.

22 J T Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *J. Assoc. Comput. Mach.*, 27(4):701–717, 1980.

23 Tomer Shlomi, Daniel Segal, Eytan Ruppin, and Roded Sharan. Qpath: a method for querying pathways in a protein-protein interaction network. *BMC Bioinformatics*, 7:199, 2006. doi:10.1186/1471-2105-7-199.

24 Ola Svensson and Jakub Tarnawski. The matching problem in general graphs is in quasi-nc. *CoRR*, abs/1704.01929, 2017. arXiv:1704.01929.

25 R Zippel. Probabilistic algorithms for sparse polynomials. In *EUROSAM*, pages 216–226, 1979.