FPT Algorithms over Linear Delta-Matroids with Applications

Eduard Eiben
Dept. of Computer Science, Royal Holloway, University of London, UK

Tomohiro Koana
Graduate School of Information Science and Technology, The University of Tokyo, Japan

Magnus Wahlström
Dept. of Computer Science, Royal Holloway, University of London, UK

Abstract

Matroids, particularly linear matroids, have been a powerful tool for applications in parameterized complexity, both for algorithms and kernelization. In particular, they have been instrumental in speeding up or replacing dynamic programming. Delta-matroids are a generalization of matroids that further encapsulate structures such as non-maximum matchings in general graphs and various path-packing and topological structures. There is also a notion of linear delta-matroids (represented by skew-symmetric matrices) which carries significant expressive power and enables powerful algorithms. We investigate parameterized complexity aspects of problems defined over linear delta-matroids, or with delta-matroid constraints. Our initial analysis of basic intersection and packing problems reveals a different complexity landscape compared to the more familiar matroid case. In particular, there is a stark contrast in complexity between the *cardinality* parameter k and the *rank* parameter r. For example, finding an intersection of size k of three linear delta-matroids is W[1]-hard when parameterized by k, while far more general problems (such as finding a set packing of size k that is feasible in a given linear delta-matroid) are FPT when parameterized by the rank r of the delta-matroid. In fact, we extend the recent *determinantal sieving* procedure of Eiben, Koana, and Wahlström (TheoretiCS 2025) into a process that sieves a given polynomial for a monomial whose support is feasible in a given linear delta-matroid, parameterized by r. This is a direct generalization of determinantal sieving.

Second, we investigate a curious class of problems that turns out to be FPT parameterized by k, even on delta-matroids of unbounded rank. We begin with Delta-matroid Triangle Cover – find a feasible set of size k that can be covered by a vertex-disjoint packing of triangles (i.e., sets of size 3) out of a given triangle collection. For example, this allows us to find, in a graph, a packing of K_3's and K_2's with the maximum possible number of edges, parameterized above the matching number of the graph (note that this problem is NP-hard, unlike the problem of finding such a packing covering a maximum number of *vertices*, which is tractable). As applications, we resolve the FPT status of Cluster Subgraph and Strong Triadic Closure parameterized above the matching number.

2012 ACM Subject Classification Theory of computation → Parameterized complexity and exact algorithms; Mathematics of computing → Matroids and greedoids

Keywords and phrases Delta-matroids, parameterized algorithms, triangle packing

Digital Object Identifier 10.4230/LIPIcs.ESA.2026.89

Related Version *Full Version*: `https://arxiv.org/abs/2502.13654` [8]

Funding *Eduard Eiben*: Supported by Engineering and Physical Sciences Research Council (EPSRC) grant UKRI4530: Exploring Parameterized Complexity in Blockchain Systems.
Tomohiro Koana: Supported by JST CREST Grant Number JPMJCR24Q2 and JST ERATO Grant Number JPMJER2301.

1 Introduction

Matroids capture a notion of *independence* across many domains – from linearly independent sets in vector spaces to spanning forests in graphs. Formally, a matroid is a pair $M = (V, \mathcal{I})$ where $\mathcal{I} \subseteq 2^V$, called *independent sets*, satisfies $\emptyset \in \mathcal{I}$, downward closure, and the *augmentation axiom*:

$$\text{if } A, B \in \mathcal{I} \text{ and } |A| < |B| \text{ then there exists } x \in B \setminus A \text{ such that } A \cup \{x\} \in \mathcal{I}.$$

A maximal independent set is called a *basis*. Matroids date back to Whitney [33] and are ubiquitous in computer science. A central class is *linear matroids*, represented by a matrix over a field $\mathbb{F}$, where independence corresponds to linear independence of columns.

Matroids admit strong algorithmic meta-results. Two classics are Matroid Intersection, i.e. finding a common basis in two matroids, and Matroid Parity: given a matroid $M = (V, \mathcal{I})$ and a partition Π of V into pairs, find a maximum-cardinality independent set that is a union of pairs from Π. Matroid Intersection is polynomial-time solvable for arbitrary matroids, while Matroid Parity is NP-hard in general but polynomial-time on a class of matroids including linear matroids [23, 24]. For background, see Oxley [27] and Schrijver [28].

Linear matroids have also been a powerful tool in parameterized complexity, via the representative sets method [10, 25] and related kernelization results for graph separation problems [19, 20, 21, 31]. More recently, *extensor coding* [6] and *determinantal sieving* [9] provide algebraic sieving procedures for generating polynomials, filtering for monomials whose support forms a basis of a given linear matroid and generalizing earlier monomial sieving methods [3, 4].

Delta-matroids

Delta-matroids are generalizations of matroids, defined and initially investigated by Bouchet in the 1980s [5] although similar or equivalent structures were introduced independently by other researchers; see Moffatt [26] for a survey. A delta-matroid is a pair $D = (V, \mathcal{F})$ where $\mathcal{F} \subseteq 2^V$ is a collection of subsets called *feasible sets*, subject to the *symmetric exchange axiom*:

$$\text{for all } A, B \in \mathcal{F} \text{ and } x \in A \triangle B \text{ there exists } y \in A \triangle B \text{ such that } A \triangle \{x, y\} \in \mathcal{F}.$$

where $\triangle$ denotes symmetric difference. Unlike matroids, feasible sets of a delta-matroid can have different cardinalities and are typically not closed under taking subsets. Consequently, some standard problems are naturally phrased differently; for example, Delta-matroid Intersection asks for *any* common feasible set rather than one of maximum size.

Note that the symmetric-exchange axiom is a relaxation of the matroid basis exchange axiom, which requires choosing $x \in A \setminus B$ and $y \in B \setminus A$ (i.e., the two elements must come from different sets). In fact, when every feasible set has the same cardinality (as in a matroid's family of bases), the two axioms coincide. In addition, the family of independent sets of a matroid also satisfies the symmetric-exchange property. Consequently, both the bases and the independent sets of any matroid form delta-matroids. Another prominent example arises from graph matchings. Let $G = (V, E)$ be a graph and define

$$\mathcal{F} = \{ F \subseteq V \mid G[F] \text{ admits a perfect matching} \}.$$

Then $D = (V, \mathcal{F})$ is a delta-matroid, called the *matching delta-matroid* of G. A useful special case starts with a partition Π of V into disjoint pairs. Let G_Π be the graph with vertex set

V and edge set Π; its matching delta-matroid D_Π is the *pairing delta-matroid* of Π. Here a set $F \subseteq V$ is feasible if and only if it contains either both elements of every pair or none, i.e., no pair is *broken*. Pairing delta-matroids will serve as building blocks in our constructions.

Even this simple delta-matroid cannot be represented by any matroid. Take $V = \{x_1, x_1', x_2, x_2'\}$ with pairs $\{x_1, x_1'\}$ and $\{x_2, x_2'\}$. In D_Π the only feasible 2-element subsets are $\{x_1, x_1'\}$ and $\{x_2, x_2'\}$, while mixed pairs such as $\{x_1, x_2\}$ are infeasible. If these two sets were the bases of a matroid, the basis-exchange axiom would compel at least one mixed pair (e.g., $\{x_1, x_2\}$ or $\{x_1, x_2'\}$) to be a basis as well, contradicting the definition of D_Π. Hence pairing (and matching) delta-matroids lie strictly beyond matroids.

Analogously to matroids, there is also a notion of *linear delta-matroid*. A matrix A is *skew-symmetric* if $A^T = -A$. Let A be a skew-symmetric $V \times V$-matrix, and define

$$\mathcal{F} \;=\; \{\, F \subseteq V \mid A[F] \text{ is non-singular} \,\}.$$

Then $D = (V, \mathcal{F})$ is a delta-matroid, denoted by $D = \mathbf{D}(A)$. For $S \subseteq V$, define

$$\mathcal{F} \triangle S \;=\; \{\, F \triangle S \mid F \in \mathcal{F} \,\}.$$

Then $D \triangle S = (V, \mathcal{F} \triangle S)$ defines a delta-matroid, the *twist* of D by S. Twisting by the full ground set yields the *dual* of a delta-matroid: $D^* = D \triangle V$. In particular, for the matching delta-matroid of a graph $G = (V, E)$, the dual has feasible sets $S \subseteq V$ such that $G - S$ has a perfect matching; we call this the *dual matching delta-matroid* of G. It is straightforward to verify that twisting preserves the symmetric-exchange axiom, and hence the class of delta-matroids is closed under twists. A delta-matroid is called *linear* if it admits a representation $D = \mathbf{D}(A) \triangle S$ for some skew-symmetric matrix A. We call this a *twist representation*.

Examples of linear delta-matroids include basis delta-matroids of linear matroids, *twisted matroids* (being the twists of basis delta-matroids of linear matroids) and matching delta-matroids, the latter represented via the famous *Tutte matrix* of G. Again, see Moffatt [26] for more background and applications. In addition, slightly more generally, one may consider *projected* linear representations of delta-matroids, which includes independence delta-matroids of linear matroids.

Finally, as with matroids, delta-matroids come with algorithmic meta-results. There is a signed generalization of the greedy algorithm that finds a maximum-weight feasible set precisely for delta-matroids. On the other hand, while the natural generalizations Delta-matroid Intersection and Delta-matroid Parity are NP-hard in general, both become tractable on linear delta-matroids. More precisely, the problems are usually defined as follows. In Delta-matroid Intersection, the task is to find *any* common feasible set F in two delta-matroids D_1 and D_2, without any requirement on $|F|$, and in Delta-matroid Parity the task is to find a feasible set F in a delta-matroid such that a minimum number of pairs are broken in F. Since delta-matroids are not normally closed under taking subsets, this definition of Delta-matroid Intersection is already a non-trivial problem. These versions are tractable on linear delta-matroids due to a result of Geelen et al. [11], recently improved to $\tilde{O}(n^\omega)$ time by Koana and Wahlström [17]. The natural variant Maximum Delta-matroid Intersection, where $|F|$ is maximised, was raised by Kakimura and Takamatsu [16], and appears harder. It was only recently solved by a randomized algorithm [17]. See the latter for a deeper overview of algorithms for linear delta-matroids. In summary, although they are less known, delta-matroids (and especially linear delta-matroids) generalize many of the useful aspects of (linear) matroids.

In this paper, we investigate parameterized complexity aspects of problems on linear and projected linear delta-matroids, in the same vein as the work on linear matroids surveyed above. The representative sets theorem over linear matroids has recently been extended to linear delta-matroids, with some initial applications to kernelization [32], but the applicability of delta-matroid tools for FPT algorithms has so far not been pursued. Our purpose is to push this forward. In Section 2, we introduce the delta-matroid analogues of the classical intersection, parity, and set-packing problems such as q-Delta-matroid Intersection, q-Delta-matroid Parity, and Delta-matroid Set Packing, and investigate parameterized complexity with respect to the solution size k and the rank r. We develop a delta-matroid sieving algorithm generalizing the determinantal sieving [9]. We essentially show that a problem of this type is FPT when this sieving can be applied, and W[1]-hard otherwise. In Section 3, we discuss Delta-matroid Triangle Cover. While the delta-matroid sieving algorithm cannot be applied, we show by related methods that Delta-matroid Triangle Cover is FPT. We use this insight to settle open questions on Cluster Subgraph and Strong Triadic Closure [7, 12].

2 Delta-matroid intersection and packing problems

Subsection 2.1 formulates the delta-matroid variants of intersection, parity, and set packing and states our full parameterized complexity landscape. Subsection 2.2 recalls the algebraic encoding of set families by *multivariate generating polynomials*, and discusses polynomial constructions over delta-matroids. Subsection 2.3 introduces the *delta-matroid sieving* framework that filters these polynomials; it generalizes determinantal sieving from matroids to delta-matroids and is the key algorithmic ingredient behind our positive results.

2.1 Problem definitions and our results

We introduce the q-Delta-matroid Intersection and Delta-matroid Set Packing problems, generalizing Delta-matroid Intersection and Delta-matroid Parity. Among others, we consider the following problems.

- q-Delta-matroid Intersection: Given q delta-matroids $D_1 = (V, \mathcal{F}_1), \ldots, D_q = (V, \mathcal{F}_q)$ and an integer $k \in \mathbb{N}$, is there a set of cardinality k that is feasible in D_i for every $i \in [q]$?
- q-Delta-matroid Parity: Given a delta-matroid $D = (V, \mathcal{F})$, a partition $\mathcal{P}$ of V into blocks of size q, and an integer $k \in \mathbb{N}$, is there a feasible set that is the union of k blocks from $\mathcal{P}$?
- Delta-matroid Set Packing: Given a delta-matroid $D = (V, \mathcal{F})$, a partition $\mathcal{P}$ of V, and an integer $k \in \mathbb{N}$, is there a feasible set F in D, $|F| = k$, which is the union of blocks from $\mathcal{P}$?

Note that in q-Delta-matroid Parity all blocks have size q, so asking for k blocks is equivalent to asking for a feasible set of size qk. In Delta-matroid Set Packing blocks may have varying sizes, and we parameterize by $|F|$. We use the following shorthand: DDD Intersection refers to 3-Delta-matroid Intersection; DDM Intersection refers to the special case of 3-Delta-matroid Intersection where D_3 is a matroid; and DΠM Intersection refers to the case where D_1 is an arbitrary delta-matroid, D_2 is a pairing delta-matroid for a partition Π, and D_3 is a matroid. In other words, DΠM Intersection is Delta-matroid Parity with an additional matroid constraint. Throughout, we assume that all matroids and delta-matroids are linear (or projected linear) with representations given in the input over some common field $\mathbb{F}$.

The classical problems Linear Delta-matroid Intersection and Linear Delta-matroid Parity correspond to the special case where $q = 2$ and where there is in addition no cardinality constraint k on the solution, and were solved by Geelen et al. [11]. For the variant where $q = 2$ and the cardinality constraint is present, only a randomized polynomial-time algorithm is known [17] (even if we replace the constraint $|F| = k$ by asking $|F|$ to be maximized).

The matroid versions of these problems are all FPT for linear matroids when parameterized by the solution cardinality k for any q (and more generally FPT when parameterized by $k + q$) [6, 9, 25]. The most powerful method here, at least for representations over fields of characteristic 2, is the recent method of *determinantal sieving* [9], which forms the inspiration for our algorithms; see below. In particular, for matroids represented over a field of characteristic 2, q-Matroid Intersection can be solved in $O^*(2^{(q-2)k})$ time, q-Matroid Parity in $O^*(2^{qk})$ time, and Matroid Set Packing in $O^*(2^k)$ time [9]. Results over other fields are somewhat slower, but still single-exponential.

For delta-matroids, we find a more intricate picture. The *rank* of a delta-matroid $D = (V, \mathcal{F})$ is the cardinality of a largest feasible set in D. We consider the above problems both parameterized by k and by the rank r. For matroids, this distinction makes no difference – given a linear matroid $M = (V, \mathcal{I})$ of rank r and a parameter k, via a *truncation* operation one can construct a linear matroid $M' = (V, \mathcal{I}')$ of rank k, which retains all independent sets of cardinality at most k from M. Furthermore, this can be done efficiently with a maintained linear representation [22, 25]. Thus, over matroids, parameterizing by r is no easier than parameterizing by k. However, for delta-matroids, the distinction turns out to be highly significant. Unlike matroids, delta-matroids are not closed under taking subsets, and the naive analogue of truncation – keeping only feasible sets of size at most k – does not in general yield a delta-matroid. Thus, bounding the rank interacts with the structure in a fundamentally different way than for matroids.

Parameterized by k, we find a dividing line where some problems, including DDM Intersection, are FPT given representations over a common field $\mathbb{F}$, whereas others, including DDD Intersection, are W[1]-hard. This holds even for very simple delta-matroids.

▶ **Theorem 1.** *DDD Intersection is W[1]-hard parameterized by k even for linear delta-matroids represented over the same field.*

On the other hand, if we parameterize by the rank r, then we recover all positive cases from the matroid case, and with matching running times.

▶ **Theorem 2.** *The following problems are FPT over linear delta-matroids and matroids provided as representations over a common field.*

- *DDD Intersection parameterized by $r = \mathrm{rank}(D_3)$, with a running time of $O^*(2^r)$ over characteristic 2 and $O^*(2^{O(r)})$ otherwise.*
- *More generally, q-Delta-matroid Intersection parameterized by r, where r is the maximum rank of D_i, $i \geq 3$, with a running time of $O^*(2^{(q-2)r})$ over characteristic 2 and $O^*(2^{O(qr)})$ otherwise.*
- *q-Delta-matroid Parity and Delta-matroid Set Packing parameterized by the rank r of the delta-matroid, with a running time of $O^*(2^r)$ over characteristic 2 and $O^*(2^{O(r)})$ in general.*
- *DDM Intersection and DΠM Intersection, parameterized by the cardinality k, with a running time of $O^*(2^k)$ over characteristic 2 and $O^*(2^{O(k)})$ in general.*
- *More generally, Delta-matroid Intersection and Delta-matroid Parity with an additional $q - 2$ matroid constraints, parameterized by q and the cardinality k, with a running time of $O^*(2^{(q-2)k})$ over characteristic 2 and $O^*(2^{O(qk)})$ in general.*

Intuitively, we treat the first two delta-matroid layers by encoding them into a Pfaffian-based generating polynomial, and then use sieving to enforce feasibility (in rank r or size k) for the remaining matroid and delta-matroid constraints. In all cases, the algorithms are randomized and use polynomial space over characteristic 2 but exponential space otherwise.

Finally, we note a problem of Colorful Delta-matroid Matching that will be used later in the paper. In this problem, we are given a delta-matroid $D = (V, \mathcal{F})$, a graph $G = (V, E)$ with edges colored with k colors, and an integer $k \in \mathbb{N}$. The task is to find a feasible set $F \in \mathcal{F}$, $|F| = 2k$ such that $G[F]$ has a matching with k distinct edge colors. This reduces to DΠM Intersection.

▶ **Theorem 3.** *Colorful Delta-matroid Matching on linear delta-matroids can be solved in $O^*(2^k)$ time if the representation is over a field of characteristic 2 and in $O^*(2^{O(k)})$ time otherwise.*

2.2 Multivariate generating polynomials

Our algorithms rely on an algebraic encoding of combinatorial families by *(multivariate) generating polynomials.* Let V be a ground set and $\mathcal{F} \subseteq 2^V$ a family of sets. Associate an indeterminate x_v with every $v \in V$ and write $X = \{x_v \mid v \in V\}$. Over a field $\mathbb{F}$ we define

$$P(X) \;=\; \sum_{S\in\mathcal{F}} c_S \prod_{v\in S} x_v,$$

where each coefficient $c_S \in \mathbb{F} \setminus \{0\}$. Here we do *not* insist on $c_S = 1$: for decision problems only the support of P matters, and allowing arbitrary non-zero coefficients sometimes allows for an efficient evaluation of P.

Suppose P can be evaluated in polynomial time. (This can be true even if there are exponentially many terms in the monomial expansion of P.) By the Schwartz-Zippel lemma, a degree-d polynomial over $\mathbb{F}$ is zero at a uniformly random evaluation point with probability at most $d/|\mathbb{F}|$. Hence, by working in a sufficiently large extension field of $\mathbb{F}$ one can test whether $\mathcal{F} = \emptyset$ by evaluating $P(X)$ at random points.

As a simple example, we discuss perfect matchings in bipartite graphs. Let $G = (U \cup W, E)$ be bipartite and fix a field $\mathbb{F}$. Introduce a variable x_e for every edge $e \in E$ and define the *Edmonds matrix* by

$$A[u, w] \;=\; \begin{cases} x_{uw} & \text{if } uw \in E, \\ 0 & \text{otherwise.} \end{cases}$$

Expanding the determinant,

$$\det A \;=\; \sum_{\pi\colon U\to W \text{ bijective}} \operatorname{sgn}(\pi) \prod_{u\in U} x_{u\pi(u)},$$

we obtain a generating polynomial whose monomials correspond exactly to perfect matchings in G. Because determinants are computable in polynomial time, evaluating $\det A$ at random points yields a simple randomized algorithm for testing the existence of a perfect matching. Here, c_S can be $+1$ or -1. Forcing $c_S = 1$ would require computing the permanent, which is #P-hard.

Generating polynomials on linear matroids and delta-matroids

The example of bipartite perfect matchings can also be phrased as a matroid intersection problem: a perfect matching is a set of edges that picks exactly one edge incident to each vertex in U and in W, i.e., a common basis of two partition matroids on the edge set E. More generally, we can build generating polynomials for intersections of linear matroids. Recall that a linear matroid on a ground set V of rank k is represented by a matrix $A \in \mathbb{F}^{k \times V}$; a subset $I \subseteq V$ is independent if and only if the columns indexed by I are linearly independent. To build a generating polynomial for the intersection of two such matroids over the same field $\mathbb{F}$, let $A_1, A_2 \in \mathbb{F}^{k \times V}$ be their representations. By the Cauchy-Binet formula,

$$\det(A_1 A_2^T) \;=\; \sum_{B \in \binom{V}{k}} \det A_1[\cdot, B]\; \det A_2[\cdot, B],$$

where $\binom{V}{k}$ is the family of k-element subsets of V and $A[\cdot, B]$ denotes the submatrix consisting of the columns in B. Introduce the diagonal matrix $X = \operatorname{diag}(x_v)_{v \in V}$. Then $\det(A_1 X A_2^T)$ is a generating polynomial whose monomials correspond exactly to the common bases of the two matroids. Indeed, expanding the determinant via Cauchy–Binet gives

$$\det(A_1 X A_2^T) \;=\; \sum_{B \in \binom{V}{k}} \det A_1[\cdot, B]\; \det A_2[\cdot, B] \prod_{v \in B} x_v.$$

We further extend this construction to two linear delta-matroid intersections, building on the work of Koana and Wahlström [17]. As mentioned above, a delta-matroid D is linear if it admits a twist representation $D = \mathbf{D}(A) \triangle S$, where $A \in \mathbb{F}^{V \times V}$ is a skew-symmetric matrix, and $S \subseteq V$. They gave an alternative representation based on *contraction* as $\mathbf{D}(A')/T$. For a delta-matroid $D = (V, \mathcal{F})$ and $T \subseteq V$, the *contraction* by T, denoted by D/T, is defined as $(V \setminus T, \mathcal{F}/T)$, where

$$\mathcal{F}/T \;=\; \{\, F \setminus T \mid F \in \mathcal{F}, T \subseteq F \,\}.$$

They showed that twist and contraction representations are polynomial-time interconvertible. While they are equivalent, the contraction representation seems more suitable for algorithmic tasks. Among other things, they developed a randomized polynomial-time algorithm for Linear Delta-matroid Intersection, essentially by showing that a generating polynomial for two linear delta-matroid intersections can be evaluated in polynomial time. In this paper, we observe that their linear representation constructions give rise to a generating polynomial that is useful for all algorithmic tasks listed in Theorem 2. In fact, the result is remarkably generic. Given two linear delta-matroids $D_1 = (V, \mathcal{F}_1)$ and $D_2 = (V, \mathcal{F}_2)$, represented over a common field and a set $S \subseteq V$, define variables $X = \{x_{v,i} \mid v \in V, i \in \{1,2,3\}\}$. In polynomial time, via a Pfaffian construction, we can evaluate a polynomial

$$P_S(X) = \sum_{F_1 \in \mathcal{F}_1, F_2 \in \mathcal{F}_2 \colon F_1 \triangle F_2 = S} f(F_1, F_2) \prod_{v \in F_1 \cap S} x_{v,1} \prod_{v \in F_2 \cap S} x_{v,2} \prod_{v \in F_1 \cap F_2} x_{v,3},$$

where $f(F_1, F_2)$ is a non-zero constant for each F_1 and F_2. For our purposes, the key point is that P_S can be evaluated in polynomial time, and its monomials encode pairs of feasible sets with $F_1 \triangle F_2 = S$. For example, if we are interested in intersection problems, then we can evaluate this for $S = \emptyset$ and get a generating polynomial in the variables $x_{v,3}$.

2.3 Sieving over generating polynomials

Multivariate generating polynomials are useful not only for randomized polynomial-time algorithms but also for designing (randomized) FPT algorithms. A canonical example in the literature is k-Path: given a graph G and $k \in \mathbb{N}$, decide whether G contains a simple path on k vertices. Since k-Path generalizes Hamiltonicity, we cannot hope to evaluate a generating polynomial that encodes *only* k-paths in polynomial time. We consider a relaxation to *k-walks*. For each $i \in [k]$, construct the $|V| \times |V|$ matrix A_i with $A_i[u,v] = y_{(u,v),i}$ if $(u,v) \in E(G)$ and $A_i[u,v] = 0$ otherwise. Then the (u,v)-entry of $W = A_1 A_2 \cdots A_{k-1}$, denoted $W[u,v]$, is a generating polynomial (in the y-variables) for all walks of length $k-1$ from u to v. Now replace each variable $y_{(u,v),i}$ by $y_{(u,v),i}\, x_v$, and finally multiply the result by x_u. A simple k-path exists if and only if, in the polynomial, there is a *multilinear* monomial of degree k in the x-variables (i.e. no variable x_v appears with exponent greater than 1). This "multilinear detection" framework was introduced by Koutis [18] and refined by Williams [34], who obtained an $O^*(2^k)$-time algorithm using group-algebra techniques. Subsequently, Björklund et al. [2, 3] gave an arguably simpler approach over characteristic-2 fields:

▶ **Lemma 4** (Multilinear sieving [2, 3])**.** *Let $P(X)$ be a polynomial over a set of variables X over a field $\mathbb{F}$ of characteristic 2. There is a randomized algorithm that, in time $O^*(2^k)$ and polynomial space, tests if $P(X)$ contains a multilinear monomial of degree k.*

Recently, this multilinear detection approach has been extended to the linear matroid via the technique called *determinantal sieving* [9]. At a high level, the sieve is a polynomial transformation that filters monomials by making those with "bad" supports cancel out over characteristic 2 fields. Given a generating polynomial P for $\mathcal{F}$, the Schwartz-Zippel lemma allows us to test whether $\mathcal{F}$ is empty in randomized polynomial time. Given a linear matroid of rank k, determinantal sieving essentially allows us to test whether $\mathcal{F}$ contains a basis in randomized $O^*(2^k)$ time, when P is defined over a field of characteristic 2. We call the *support* of a monomial m the set of variables that appear in m (equivalently, those with non-zero exponents).

▶ **Lemma 5** (Basis sieving [9])**.** *Let $P(X)$ be a polynomial in variables X over a field $\mathbb{F}$ of characteristic 2, and let $M = (X, \mathcal{I})$ be a rank-k linear matroid represented over $\mathbb{F}$. There is a randomized algorithm that, in time $O^*(2^k)$ and polynomial space, decides whether the monomial expansion*[1] *of $P(X)$ contains a multilinear term of degree k whose support is a basis of M.*

Applying basis sieving with a uniform matroid of rank k, where all sets of size at most k are independent, we recover the multilinear sieving. The striking feature of determinantal sieving is that this step from the uniform matroid to an arbitrary rank-k linear matroid incurs no extra exponential time. Prior to [9], only partition matroids were known to admit such algorithms [4].

From matroids to delta-matroids

We extend Lemma 5 to the richer setting of *linear delta-matroids*. Given a generating polynomial for $\mathcal{F}$ over a field of characteristic 2 and a linear delta-matroid D whose rank is r, our new algorithm decides, again in randomized $O^*(2^r)$ time and polynomial space, whether $\mathcal{F}$ contains a feasible set in D. Remarkably, this additional expressive power costs only a polynomial factor over the matroid case.

▶ **Theorem 6** (Delta-matroid sieving)**.** *Let $P(X)$ be a polynomial in variables X over a field $\mathbb{F}$ of characteristic 2, and let $D = (X, \mathcal{F})$ be a rank-r linear delta-matroid represented over $\mathbb{F}$. There is a randomized algorithm that, in time $O^*(2^r)$ and polynomial space, decides whether the monomial expansion of $P(X)$ contains a multilinear term whose support is a feasible set of D.*

---

[1] A monomial expansion is the expression obtained after fully expanding every product into a sum of monomials.

To prove Theorem 6, we introduce a *sparse representation* for rank-r linear delta-matroids, in which the underlying matrix A has only $O(rn)$ non-zero entries. We work with the contraction representation $D = (V, \mathcal{F}) = \mathbf{D}(A)/T$ [17], which has the useful property that $A[V] = O$, so the most non-trivial structure is confined to T. As in determinantal sieving [9], we construct a carefully designed polynomial and apply the inclusion-exclusion principle to filter desired terms. However, since we are working with skew-symmetric matrices, we must use Pfaffians instead of determinants to express feasibility. To achieve this, we use a Pfaffian-based construction (a delta-matroid analogue of determinantal sieving) together with inclusion-exclusion.

Lemma 5 and Theorem 6 also extend to fields of characteristic other than 2, but with three caveats: (1) the process uses exponential space, (2) the process works over an arithmetic circuit, not via black-box access, and (3) the running time increases to $O^*(2^{O(r)})$. As in the characteristic 2 case, we use another ad-hoc argument to obtain Pfaffians instead of determinants.

Summary

All results in Theorem 2 over characteristic-2 fields are obtained by applying Theorem 6 to generating polynomials. A rule of thumb for packing and intersection problems is:

If the problem can be reduced to Theorem 6 then it is FPT; otherwise W[1]-hard.

Thus, for example, DDD Intersection is hard with parameter k because there is no rank bound on D_1, D_2, D_3, but becomes FPT when parameterized by $r = \text{rank}(D_3)$, whereas DDM Intersection is FPT since the matroid layer can be truncated to rank k. See full version [8] for details.

3 Delta-matroid triangle cover with applications

Motivated by graph-theoretic applications, we study a relaxation of 3-Delta-matroid Parity that turns out to be FPT. Let $D = (V, \mathcal{F})$ be a linear delta-matroid and let $\mathcal{T} = \{T_1, \ldots, T_m\} \subseteq \binom{V}{3}$ be a family of triples (triangles). We say that $S \subseteq V$ is *covered* by triangles if some disjoint packing $\mathcal{T}' \subseteq \mathcal{T}$ satisfies $S \subseteq V(\mathcal{T}') := \bigcup_{T \in \mathcal{T}'} T$. We do not require S to be partitioned by $\mathcal{T}'$: triangles in $\mathcal{T}'$ may intersect S in one or two vertices. Given $k \in \mathbb{N}$, Delta-matroid Triangle Cover (DMTC) asks for a feasible set $F \in \mathcal{F}$ of size k that can be covered by triangles.

▶ **Theorem 7.** *DMTC can be solved in $O^*(k^{\mathcal{O}(k)})$ time over a linear delta-matroid.*

The most obvious structural simplification of DMTC would seem to be to assume that we are looking for a set $F \in \mathcal{F}$ that is precisely partitioned by triangles. However, this defines 3-Delta-matroid Parity, which is W[1]-hard. Indeed, the approach of applying delta-matroid sieving (Theorem 6) fails. Hence Delta-matroid Triangle Cover inhabits an interesting spot of being "just barely" tractable. Indeed, it can be shown that Delta-matroid K_4 Cover, where F is to be covered by sets of cardinality 4, is W[1]-hard.

3.1 High-level Overview of our FPT Algorithm

Our algorithm for Delta-matroid Triangle Cover seamlessly combines combinatorial and algebraic techniques, ultimately reducing the problem to Colorful Delta-matroid Matching, which can be solved via delta-matroid sieving. On the algebraic side, we introduce a new delta-matroid operation, termed *ℓ-projection*, which generalizes the classical matroid truncation operation and appears promising for kernelization [32]. On the combinatorial side, we use arguments reminiscent of the sunflower lemma to effectively reduce triangles to (colored) pairs.

Technical overview

Assume there exists a triangle packing $\mathcal{T} = \{T_1, \ldots, T_p\}$ that covers an unknown feasible set F. For illustration, we assume that F is partitioned by $\mathcal{T}$, i.e., $F = \bigcup_{T \in \mathcal{T}} T$. Classical color coding [1] lets us find (with probability $1/k^{O(k)}$) a coloring of V into p blocks $V_1, \ldots, V_p$ such that each packed triangle T_i is contained in its block V_i. We further refine every block into three parts $V_i = V_{i,1} \cup V_{i,2} \cup V_{i,3}$ so that T_i intersects $V_{i,j}$ for all $j \in [3]$. Henceforth, we will assume that each triangle lies inside V_i for some i and intersects each part $V_{i,j}$. Let $v_{i,j}$ denote the unique vertex of $T_i \cap V_{i,j}$. This refinement step costs $2^{O(k)}$ time.

We then classify the vertices as *flowery* or *non-flowery*. A vertex is flowery if it belongs to many pairwise disjoint triangles; otherwise it is non-flowery. Because $|\bigcup \mathcal{T}| \in O(k)$ in general (the triangles are disjoint and $|\mathcal{T}| \leq |F| = k$), we can afford to guess whether $v_{i,j}$ is flowery or not in $2^{O(k)}$ time. By definition, a flowery vertex can always be covered by a triangle. This is an argument often used in the kernelization for Hitting Set and Set Packing. Unlike these classical problems, however, we cannot simply delete a flowery vertex, since it may or may not be part of F. Even worse, there can be $\Omega(n)$ flowery vertices (within each block $V_{i,j}$), and we cannot afford to choose which of the flowery vertices should be part of F. To circumvent this issue, we introduce a delta-matroid operation called ℓ-projection defined as follows. Let $D = (V, \mathcal{F})$ be a delta-matroid, $X \subseteq V$, and $\ell \in \mathbb{N}$. The ℓ-projection of D to $V \setminus X$ is defined by $D|_\ell X = (V \setminus X, \mathcal{F}|_\ell X)$, where

$$\mathcal{F}|_\ell X \ = \ \{\, F \setminus X \mid F \in \mathcal{F}, |F \cap X| = \ell \,\}$$

We also show that a linear representation of ℓ-projection can be computed in randomized polynomial time. With ℓ-projection at hand, we can handle a block $V_{i,j}$ containing a flowery vertex $v_{i,j}$ by applying 1-projection with $X = V_{i,j}^f$, where $V_{i,j}^f$ is the set of flowery vertices v in $V_{i,j}$. This is essentially correct because a flowery vertex can be covered "for free" and the fact that $v_{i,j}$ is part of F is now encoded in the delta-matroid. Note that this is possible without knowing which particular flowery vertex is selected – thanks to ℓ-projection!

On the other hand, assume that $v_{i,3}$ is non-flowery. Then the triangles in $\mathcal{T}$ that cover $v_{i,3}$ form a graph with bounded vertex cover S, and since $v_{i,3} \in F$, one member of S must be in $\bigcup \mathcal{T}$. This allows us to make a local guess for every vertex $v \in V_{i,3}$ of the form *if* $v \in \bigcup \mathcal{T}$ *then* $w_v \in \bigcup \mathcal{T}$, for some second vertex w_v. This leaves some case work (e.g., we guess whether $v_{i,3}$ is flowery; and if not, then for the case $v = v_{i,3}$ we guess whether $w_v = v_{i,1}$ or $w_v = v_{i,2}$, and whether $w_v \in F$ or not), but for every case we can use ℓ-projection to eliminate one class $V_{i,j}$ for every i from the delta-matroid. At this point, the remaining problem can be reduced to Colorful Delta-matroid Matching, which can be solved by sieving methods as above.

3.2 Graph-theoretic applications

We showcase two graph-theoretic applications. The first is Cluster Subgraph: given a graph G and integer ℓ, decide whether G contains a cluster subgraph (each component a clique) with at least ℓ edges. Although NP-hard [29], it admits an $O^*(9^\ell)$ algorithm [14]. We examine the sharper *above-guarantee* parameter $k = \ell - \mathsf{MM}(G)$, where $\mathsf{MM}(G)$ is the matching number of G (every matching is a cluster subgraph). The parameterized complexity of this variant was open [7].

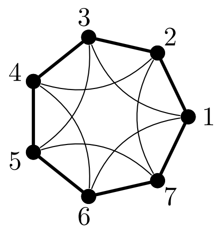

Figure 1 A maximum strong set (thick lines) which does not induce a cluster graph.

▶ **Theorem 8.** *Cluster Subgraph can be solved in $O^*(k^{O(k)})$ time for $k = \ell - \mathsf{MM}(G)$.*

To see the connection to DMTC, assume that G is K_4-free and that G has a perfect matching. Then every clique has size at most 3, and the goal is to find a vertex-disjoint packing of K_2's and K_3's with as many edges as possible, i.e., a collection of K_3's whose deletion leaves a graph with a large matching number.[2] Intuitively, the vertices covered by triangles form a set whose deletion leaves a perfect matching, and this feasibility condition is captured by the dual matching delta-matroid. We show that this reduces to solving DMTC over the dual matching delta-matroid of G. On the other hand, if G contains a large packing of K_4's, then the instance is immediately positive. Thus, we may assume that G has a set K of $O(k)$ vertices such that $G - K$ is K_4-free, and Theorem 8 follows by guessing its interaction with the solution.

The second application is to Strong Triadic Closure, formalized by Sintos [30]. Let G be a graph. We say that a 3-path (u, v, w) is *closed* if the edge uw exists in G. For a graph G, an edge set S is said to be *strong* if every 3-path in S is closed. Note that the shortcut edge uw is not required to exist in S, only in G. Given $G = (V, E)$ and ℓ, the Strong Triadic Closure problem is to find a strong edge set $S \subseteq E$ with $|S| \geq \ell$. This is based on the notion of triadic closure [13]. This problem is closely related to Cluster Subgraph as an edge set that induces a cluster graph is strong. However, not every strong set induces a cluster subgraph (see Figure 1). It is known to be NP-hard [30] and FPT when parameterized by ℓ [12, 14] with the fastest algorithm running in $O^*(\ell^{O(\ell)})$ time [14]. Again, we parameterize by $k = \ell - \mathsf{MM}(G)$.

Again, the K_4-free case turns out to be tractable: if G is K_4-free, then the solution S must be a collection of strong paths and cycles, and in fact, it can be shown that there exists an optimum where S consists only of strong cycles and K_2's. Similarly to above, we solve the problem by looking for a feasible set F in the dual matching delta-matroid of G such that F can be covered by a collection of strong cycles, possibly of length $\Omega(k)$. This is significantly more complicated; e.g., as mentioned, Delta-matroid K_c Cover is W[1]-hard for any $c \geq 4$. However, there are several cases, such as a cycle of length $3t$, $t \in \mathbb{N}$, where strong cycles can be replaced by triangles. We show that when such triangle replacement fails, pairing constraints are powerful enough to solve the problem, and they can be encoded in Colorful Delta-matroid Matching.

▶ **Theorem 9.** *Strong Triadic Closure on K_4-free graphs can be solved in $O^*(k^{O(k)})$ time for $k = \ell - \mathsf{MM}(G)$.*

[2] The problem becomes polynomial-time solvable when the goal is to maximize the number of covered *vertices* [15]. However, the edge maximization problem is NP-hard as it generalizes the perfect triangle packing problem.

Furthermore, as above, any K_4 in G immediately contributes to $\ell - \mathsf{MM}(G)$, and we can find a small modulator K so that $G - K$ is K_4-free. Unfortunately, a full FPT algorithm remained elusive; guided by the precise structure of solutions where our insights fail, we show that the problem is W[1]-hard. Somewhat unexpectedly, the source of hardness stems not from strong cycles but from a different aspect.

▶ **Theorem 10.** *Strong Triadic Closure is W[1]-hard when parameterized by $k = \ell - \mathsf{MM}(G)$.*

On the other hand, a counting argument shows that, although we do not get an FPT algorithm, we can get a constant-factor FPT approximation simply by packing a modulator K and ignoring all edges $\delta(K)$ (those with exactly one endpoint in K) between K and $G - K$ in the solution, solving $G[K]$ (exhaustively) and $G - K$ separately. This yields an algorithm that either shows that G has no strong edge set of size $\mathsf{MM}(G) + k$ or finds a strong edge set of size $\mathsf{MM}(G) + \Omega(k)$ in FPT time.

▶ **Theorem 11.** *There is a $O^*(k^{O(k)})$-time algorithm for Strong Triadic Closure that approximates the above matching parameter $\ell - \mathsf{MM}(G)$ within a factor of 7.*

In addition, we generalize a result of Golovach et al. [12], who gave an FPT algorithm for maximum degree 4, by showing that Strong Triadic Closure can be solved in time $O^*(\Delta^{O(k)})$ for graphs of maximum degree Δ:

▶ **Theorem 12.** *Strong Triadic Closure can be solved in $O^*(\Delta^{O(k)})$ time, where Δ is the maximum degree of G and $k = \ell - \mathsf{MM}(G)$.*

Lastly, we remark that the question whether Strong Triadic Closure is FPT has been asked repeatedly in the literature [7, Open Problem 5.4], [12]. The connection to Cluster Subgraph has also been observed [14]. We settle the status of both of these problems.

References

1 Noga Alon, Raphael Yuster, and Uri Zwick. Color-coding. *J. ACM*, 42(4):844–856, 1995. doi:10.1145/210332.210337.

2 Andreas Björklund. Determinant sums for undirected Hamiltonicity. *SIAM J. Comput.*, 43(1):280–299, 2014. doi:10.1137/110839229.

3 Andreas Björklund, Thore Husfeldt, Petteri Kaski, and Mikko Koivisto. Narrow sieves for parameterized paths and packings. *J. Comput. Syst. Sci.*, 87:119–139, 2017. doi:10.1016/j.jcss.2017.03.003.

4 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.

5 André Bouchet. Greedy algorithm and symmetric matroids. *Math. Program.*, 38(2):147–159, 1987. doi:10.1007/BF02604639.

6 Cornelius Brand, Holger Dell, and Thore Husfeldt. Extensor-coding. In *STOC*, pages 151–164. ACM, 2018. doi:10.1145/3188745.3188902.

7 Christophe Crespelle, Pål Grønås Drange, Fedor V. Fomin, and Petr A. Golovach. A survey of parameterized algorithms and the complexity of edge modification. *Comput. Sci. Rev.*, 48:100556, 2023. doi:10.1016/j.cosrev.2023.100556.

8 Eduard Eiben, Tomohiro Koana, and Magnus Wahlström. FPT algorithms over linear delta-matroids with applications. *arXiv preprint arXiv:2502.13654*, 2025. doi:10.48550/arXiv.2502.13654.

9 Eduard Eiben, Tomohiro Koana, and Magnus Wahlström. Determinantal sieving. *TheoretiCS*, Volume 4, October 2025. doi:10.46298/theoretics.25.21.

10 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.

11 James F. Geelen, Satoru Iwata, and Kazuo Murota. The linear delta-matroid parity problem. *J. Comb. Theory, Ser. B*, 88(2):377–398, 2003. doi:10.1016/S0095-8956(03)00039-X.

12 Petr A. Golovach, Pinar Heggernes, Athanasios L. Konstantinidis, Paloma T. Lima, and Charis Papadopoulos. Parameterized aspects of strong subgraph closure. *Algorithmica*, 82(7):2006–2038, 2020. doi:10.1007/s00453-020-00684-9.

13 Mark S Granovetter. The strength of weak ties. *Am. J. Sociol.*, 78(6):1360–1380, 1973.

14 Niels Grüttemeier and Christian Komusiewicz. On the relation of strong triadic closure and cluster deletion. *Algorithmica*, 82(4):853–880, 2020. doi:10.1007/s00453-019-00617-1.

15 Pavol Hell and David G. Kirkpatrick. Packings by cliques and by finite families of graphs. *Discrete Math.*, 49(1):45–59, 1984. doi:10.1016/0012-365X(84)90150-X.

16 Naonori Kakimura and Mizuyo Takamatsu. Matching problems with delta-matroid constraints. *SIAM J. Discrete Math.*, 28(2):942–961, 2014. doi:10.1137/110860070.

17 Tomohiro Koana and Magnus Wahlström. Faster algorithms on linear delta-matroids. In *STACS*, volume 327 of *LIPIcs*, pages 62:1–62:19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025. doi:10.4230/LIPIcs.STACS.2025.62.

18 Ioannis Koutis. Faster algebraic algorithms for path and packing problems. In *ICALP (1)*, volume 5125 of *Lecture Notes in Computer Science*, pages 575–586. Springer, 2008. doi:10.1007/978-3-540-70575-8_47.

19 Stefan Kratsch. A randomized polynomial kernelization for vertex cover with a smaller parameter. *SIAM J. Discrete Math.*, 32(3):1806–1839, 2018. doi:10.1137/16M1104585.

20 Stefan Kratsch and Magnus Wahlström. Compression via matroids: A randomized polynomial kernel for odd cycle transversal. *ACM Trans. Algorithms*, 10(4):20:1–20:15, 2014. doi:10.1145/2635810.

21 Stefan Kratsch and Magnus Wahlström. Representative sets and irrelevant vertices: New tools for kernelization. *J. ACM*, 67(3):16:1–16:50, 2020. doi:10.1145/3390887.

22 Daniel Lokshtanov, Pranabendu Misra, Fahad Panolan, and Saket Saurabh. Deterministic truncation of linear matroids. *ACM Trans. Algorithms*, 14(2):14:1–14:20, 2018. doi:10.1145/3170444.

23 László Lovász. On determinants, matchings, and random algorithms. In *FCT*, pages 565–574. Akademie-Verlag, Berlin, 1979.

24 László Lovász and Michael D Plummer. *Matching theory*, volume 29. North-Holland, 1986.

25 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.

26 Iain Moffatt. Delta-matroids for graph theorists. In Allan Lo, Richard Mycroft, Guillem Perarnau, and Andrew Treglown, editors, *Surveys in Combinatorics 2019*, London Mathematical Society Lecture Note Series, pages 167–220. Cambridge University Press, 2019. doi:10.1017/9781108649094.007.

27 James Oxley. *Matroid Theory.* Oxford University Press, 2011.

28 A. Schrijver. *Combinatorial Optimization: Polyhedra and Efficiency.* Algorithms and combinatorics. Springer, 2003.

29 Ron Shamir, Roded Sharan, and Dekel Tsur. Cluster graph modification problems. *Discrete Appl. Math.*, 144(1-2):173–182, 2004. doi:10.1016/J.DAM.2004.01.007.

30 Stavros Sintos and Panayiotis Tsaparas. Using strong triadic closure to characterize ties in social networks. In *KDD*, pages 1466–1475. ACM, 2014. doi:10.1145/2623330.2623664.

31 Magnus Wahlström. Quasipolynomial multicut-mimicking networks and kernels for multiway cut problems. *ACM Trans. Algorithms*, 18(2):15:1–15:19, 2022. doi:10.1145/3501304.

32 Magnus Wahlström. Representative set statements for delta-matroids and the Mader delta-matroid. In *SODA*, pages 780–810. SIAM, 2024. doi:10.1137/1.9781611977912.31.

33 Hassler Whitney. On the abstract properties of linear dependence. *Amer. J. Math.*, 57:509–533, 1935.

34 Ryan Williams. Finding paths of length k in $O^*(2^k)$ time. *Inf. Process. Lett.*, 109(6):315–318, 2009. doi:10.1016/J.IPL.2008.11.004.