A parameterized view on matroid optimization problems

Dániel Marx

Humboldt-Universität zu Berlin

`dmarx@informatik.hu-berlin.de`

ICALP 2006

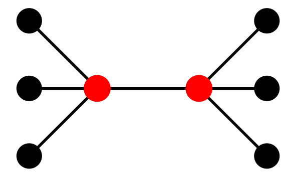
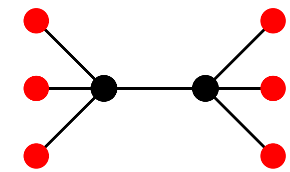
Venice, Italy

July 14, 2006

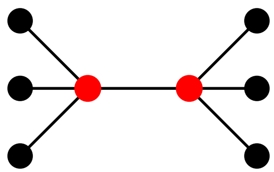
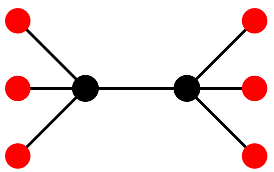
Overview

- Parameterized complexity
- Matroid basics
- Main result
- Applications

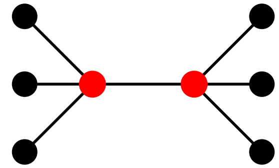
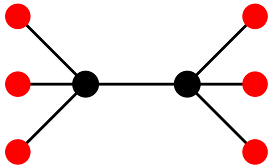
Parameterized complexity

Problem:	MINIMUM VERTEX COVER	MAXIMUM INDEPENDENT SET
Input:	Graph G, integer k	Graph G, integer k
Question:	Is it possible to cover the edges with k vertices?	Is it possible to find k independent vertices?
		
Complexity:	NP-complete	NP-complete

Parameterized complexity

Problem:	MINIMUM VERTEX COVER	MAXIMUM INDEPENDENT SET
Input:	Graph G, integer k	Graph G, integer k
Question:	Is it possible to cover the edges with k vertices? 	Is it possible to find k independent vertices? 
Complexity:	NP-complete	NP-complete
Complete enumeration:	$O(n^k)$ possibilities	$O(n^k)$ possibilities

Parameterized complexity

Problem:	MINIMUM VERTEX COVER	MAXIMUM INDEPENDENT SET
Input:	Graph G, integer k	Graph G, integer k
Question:	Is it possible to cover the edges with k vertices?	Is it possible to find k independent vertices?
		
Complexity:	NP-complete	NP-complete
Complete enumeration:	$O(n^k)$ possibilities	$O(n^k)$ possibilities
	$O(2^k n^2)$ algorithm exists	No $n^{o(k)}$ algorithm known

Parameterized problems

A problem is **fixed-parameter tractable (FPT)** with parameter k if it has an $f(k) \cdot n^c$ time algorithm, where c is independent of k.

For a large number of NP-hard problems, the parameterized version is fixed-parameter tractable. For many other problems, we have evidence that they are not FPT (W[1]-hardness).

Fixed-parameter tractable problems:

- MINIMUM VERTEX COVER
- LONGEST PATH
- DISJOINT TRIANGLES
- GRAPH GENUS
- ...

W[1]-hard problems:

- MAXIMUM INDEPENDENT SET
- MINIMUM DOMINATING SET
- LONGEST COMMON SUBSEQUENCE
- SET PACKING
- ...

Matroids

Definition: A set system $\mathcal{M}$ over E is a **matroid** if

(1) $\emptyset \in \mathcal{M}$.

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

(3) If $X, Y \in M$ and $|X| > |Y|$, then $\exists e \in X$ such that $Y \cup \{e\} \in \mathcal{M}$.

Example: $\mathcal{M} = \{\emptyset, 1, 2, 3, 12, 13\}$ is a matroid.
Example: $\mathcal{M} = \{\emptyset, 1, 2, 12, 3\}$ is not a matroid.

If $x \in \mathcal{M}$, then we say that X is **independent** in matroid $\mathcal{M}$.

Matroids—Examples

Cycle matroid: Given a graph G, let $\mathcal{M}$ contain those subsets $E' \subseteq E$ that are acyclic. $\mathcal{M}$ is a matroid.

Partition matroid: Let $E_1, \ldots, E_k$ be a partition of E, and let $a_1, \ldots, a_k$ be integers. Let $X \in \mathcal{M}$ if and only if $|X \cap E_i| \leq a_i$ for every i. Then $\mathcal{M}$ is a matroid.

Linear matroid: Let A be matrix and let E be the set of column vectors in A. The subsets $E' \subseteq E$ that are linearly independent form a matroid.

The matrix A is the **representation** of the linear matroid.

Matroids—Examples

Cycle matroid: Given a graph G, let $\mathcal{M}$ contain those subsets $E' \subseteq E$ that are acyclic. $\mathcal{M}$ is a matroid.

Partition matroid: Let $E_1, \ldots, E_k$ be a partition of E, and let $a_1, \ldots, a_k$ be integers. Let $X \in \mathcal{M}$ if and only if $|X \cap E_i| \leq a_i$ for every i. Then $\mathcal{M}$ is a matroid.

Linear matroid: Let A be matrix and let E be the set of column vectors in A. The subsets $E' \subseteq E$ that are linearly independent form a matroid.

The matrix A is the **representation** of the linear matroid.

Fact: If the elements have weights, then the greedy algorithm finds an independent set of maximum weight.

Matroid intersection

Given two matroids $\mathcal{M}_1$ and $\mathcal{M}_2$, the **intersection** $\mathcal{M}_1 \wedge \mathcal{M}_2$ contains those sets that are independent in both matroids.

Fact: [Edmonds] It is possible to find in polynomial time a set of maximum size in $\mathcal{M}_1 \wedge M_2$ (if $\mathcal{M}_1$ and $\mathcal{M}_2$ are given by an independence oracle).

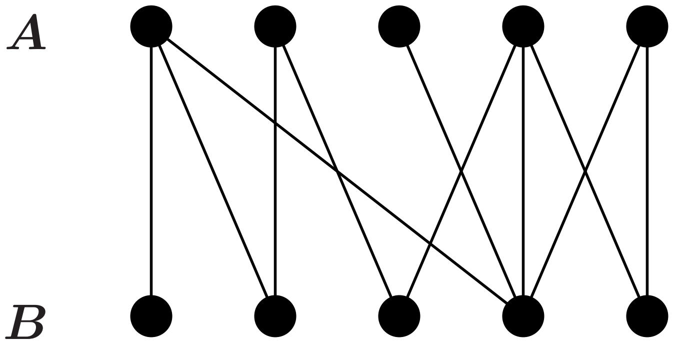
$\mathcal{M}_1 \wedge \mathcal{M}_2$ is not necessarily a matroid!

Bipartite matching

Bipartite matching can be solved with matroid intersection.

We define two partition matroids on the edge set of $G(A, B; E)$:

- $E' \in \mathcal{M}_1$ if E' contains at most one edge incident to each $v \in A$.
- $E' \in \mathcal{M}_2$ if E' contains at most one edge incident to each $v \in B$.

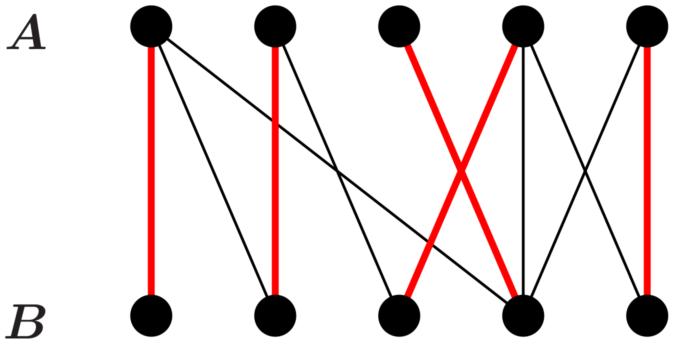


Bipartite matching

Bipartite matching can be solved with matroid intersection.

We define two partition matroids on the edge set of $G(A, B; E)$:

- $E' \in \mathcal{M}_1$ if E' contains at most one edge incident to each $v \in A$.
- $E' \in \mathcal{M}_2$ if E' contains at most one edge incident to each $v \in B$.



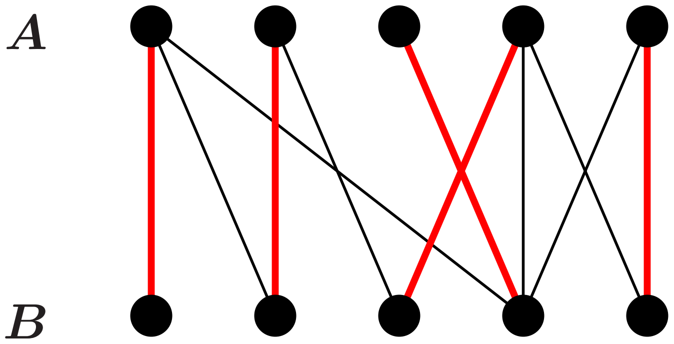
$$E' \in \mathcal{M}_1 \wedge \mathcal{M}_2 \quad \Leftrightarrow \quad E' \text{ is a matching}$$

Bipartite matching

Bipartite matching can be solved with matroid intersection.

We define two partition matroids on the edge set of $G(A, B; E)$:

- $E' \in \mathcal{M}_1$ if E' contains at most one edge incident to each $v \in A$.
- $E' \in \mathcal{M}_2$ if E' contains at most one edge incident to each $v \in B$.



$$E' \in \mathcal{M}_1 \wedge \mathcal{M}_2 \quad \Leftrightarrow \quad E' \text{ is a matching}$$

Fact: Finding a set of maximum size in the intersection of 3 matroids is NP-hard (reduction from 3-dimensional matching).

Matroid parity

Assume that E is partitioned into pairs and $\mathcal{M}$ is a matroid over E.

Task: Find an independent set of $\mathcal{M}$ that is the union of k pairs.

Can be used to solve (nonbipartite) matching.

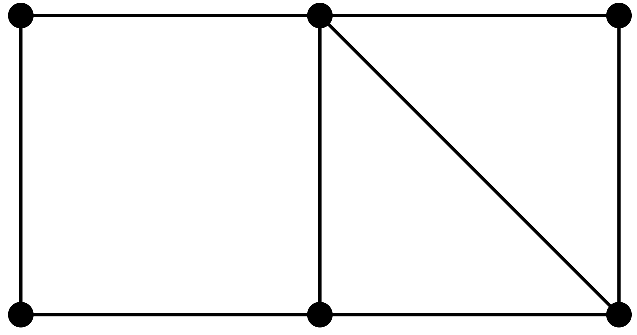
Matroid parity

Assume that E is partitioned into pairs and $\mathcal{M}$ is a matroid over E.

Task: Find an independent set of $\mathcal{M}$ that is the union of k pairs.

Can be used to solve (nonbipartite) matching.

Replace each edge with a pair of elements, $\mathcal{M}$ is a partition matroid where the classes correspond to the vertices, and at most one element can be selected from each class.



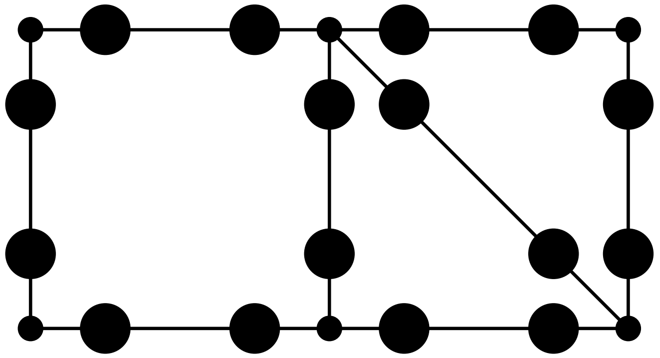
Matroid parity

Assume that E is partitioned into pairs and $\mathcal{M}$ is a matroid over E.

Task: Find an independent set of $\mathcal{M}$ that is the union of k pairs.

Can be used to solve (nonbipartite) matching.

Replace each edge with a pair of elements, $\mathcal{M}$ is a partition matroid where the classes correspond to the vertices, and at most one element can be selected from each class.



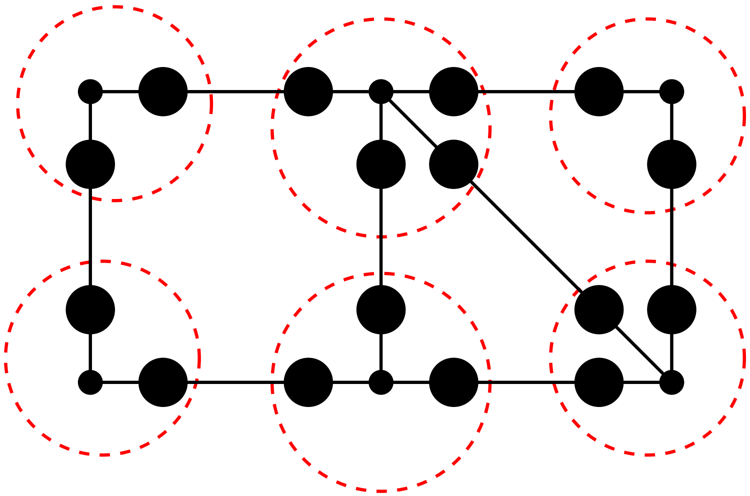
Matroid parity

Assume that E is partitioned into pairs and $\mathcal{M}$ is a matroid over E.

Task: Find an independent set of $\mathcal{M}$ that is the union of k pairs.

Can be used to solve (nonbipartite) matching.

Replace each edge with a pair of elements, $\mathcal{M}$ is a partition matroid where the classes correspond to the vertices, and at most one element can be selected from each class.



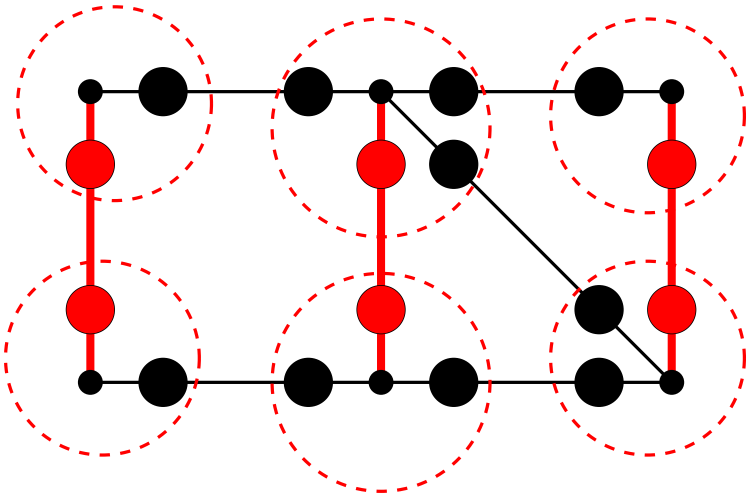
Matroid parity

Assume that E is partitioned into pairs and $\mathcal{M}$ is a matroid over E.

Task: Find an independent set of $\mathcal{M}$ that is the union of k pairs.

Can be used to solve (nonbipartite) matching.

Replace each edge with a pair of elements, $\mathcal{M}$ is a partition matroid where the classes correspond to the vertices, and at most one element can be selected from each class.



Matroid parity

Fact: Matroid parity is NP-hard.

Fact: [Lovász] Matroid parity is polynomial-time solvable if $\mathcal{M}$ is given by a linear representation.

Fact: If we have triples instead of pairs, then the problem is NP-hard even for linear matroids (reduction from 3-dimensional matching).

Matroid parity

Fact: Matroid parity is NP-hard.

Fact: [Lovász] Matroid parity is polynomial-time solvable if $\mathcal{M}$ is given by a linear representation.

Fact: If we have triples instead of pairs, then the problem is NP-hard even for linear matroids (reduction from 3-dimensional matching).

Main result: Let $\mathcal{M}$ be a linear matroid over E, given by a representation A. If E is partitioned into blocks of size ℓ, then it can be decided in randomized time $f(k, \ell) \cdot n^{O(1)}$ whether $\mathcal{M}$ has an independent set that is the union of k blocks.

That is, the problem is (randomized) fixed-parameter tractable with parameters k and ℓ.

The algorithm

Inspired by Monien's algorithm for finding paths of length k.

Let $\mathcal{S}_i$ be the set of all independent sets in $\mathcal{M}$ that arise as the union of i blocks.

- Set $\mathcal{S}_0 := \{\emptyset\}$.
- Assume that $\mathcal{S}_i$ is determined. For every $S \in \mathcal{S}_i$ and every block B, if S and B are disjoint and $S \cup B$ is independent in $\mathcal{M}$, then add $S \cup B$ to $\mathcal{S}_{i+1}$.
- Check whether $\mathcal{S}_k$ is empty or not.

The algorithm

Inspired by Monien's algorithm for finding paths of length k.

Let $\mathcal{S}_i$ be the set of all independent sets in $\mathcal{M}$ that arise as the union of i blocks.

- Set $\mathcal{S}_0 := \{\emptyset\}$.
- Assume that $\mathcal{S}_i$ is determined. For every $S \in \mathcal{S}_i$ and every block B, if S and B are disjoint and $S \cup B$ is independent in $\mathcal{M}$, then add $S \cup B$ to $\mathcal{S}_{i+1}$.
- Check whether $\mathcal{S}_k$ is empty or not.

Problem: The size of $\mathcal{S}_i$ can be n^k — running time is not fpt!

The algorithm

Inspired by Monien's algorithm for finding paths of length k.

Let $\mathcal{S}_i$ be the set of all independent sets in $\mathcal{M}$ that arise as the union of i blocks.

- Set $\mathcal{S}_0 := \{\emptyset\}$.
- Assume that $\mathcal{S}_i$ is determined. For every $S \in \mathcal{S}_i$ and every block B, if S and B are disjoint and $S \cup B$ is independent in $\mathcal{M}$, then add $S \cup B$ to $\mathcal{S}_{i+1}$.
- Check whether $\mathcal{S}_k$ is empty or not.

Problem: The size of $\mathcal{S}_i$ can be n^k — running time is not fpt!

Solution: Retain only a small part of $\mathcal{S}_i$, and throw away all the other sets. (But be careful!)

Representative systems

Definition: A subsystem $\mathcal{S}_i^* \subseteq \mathcal{S}_i$ is r**-representative** if whenever some member of $\mathcal{S}_i$ can be extended with r new blocks, then $\mathcal{S}_i^*$ has a member that can be extended with the same blocks.

Representative systems

Definition: A subsystem $\mathcal{S}_i^* \subseteq \mathcal{S}_i$ is r**-representative** if whenever some member of $\mathcal{S}_i$ can be extended with r new blocks, then $\mathcal{S}_i^*$ has a member that can be extended with the same blocks.

Formal definition: A subsystem $\mathcal{S}_i^* \subseteq \mathcal{S}_i$ is r**-representative** if for every X that is the union of r blocks

$$\exists S \in \mathcal{S}_i : S \cap X = \emptyset, S \cup X \in \mathcal{M}$$
$$\Downarrow$$
$$\exists S' \in \mathcal{S}_i^* : S' \cap X = \emptyset, S' \cup X \in \mathcal{M}.$$

Instead of the set $\mathcal{S}_i$, it is sufficient to have a $(k-i)$-representative subsystem.

Representative systems

How small representative system can we find?

Lemma: $\mathcal{S}_i$ has a $(k-i)$-representative subsystem of size at most $\binom{k\ell}{i\ell}$.

Proof is based on the following generalization of Bollobás' Inequality:

Lemma: [Lovász 1977] If $A_1, \ldots, A_n$ are a-dimensional subspaces, $B_1, \ldots, B_n$ are b-dimensional subspaces of a space of dimension $a+b$ and

(1) $A_j \cap B_{j'} = \emptyset$ for $j = j'$,

(2) $A_j \cap B_{j'} \neq \emptyset$ for $j \neq j'$,

holds, then $n \leq \binom{x+y}{x}$.

Truncation

Definition: The t**-truncation** of a matroid $\mathcal{M}$ is a matroid $\mathcal{M}'$ such that $S \in \mathcal{M}'$ iff $S \in \mathcal{M}$ and $|S| \le t$.

The answer does not change if we replace $\mathcal{M}$ with the $k\ell$-truncation $\mathcal{M}'$.

For technical reasons, we have to use the truncated matroid $\mathcal{M}'$ in the algorithm.

A representation of $\mathcal{M}'$ can be obtained in randomized polynomial time from a representation of $\mathcal{M}$ (we need the Schwartz-Zippel Lemma here).

The algorithm

- Compute a representation A' of the $k\ell$-truncation.
- Set $\mathcal{S}_0^* := \emptyset$.
- For every $S \in \mathcal{S}_i^*$ and every block B, if S and B are disjoint and $S \cup B$ is independent in $\mathcal{M}$, then add $S \cup B$ to $\mathcal{S}_{i+1}^*$.
- Reduce the size of S_i^* to $\binom{k\ell}{i\ell}$.
- Check whether $\mathcal{S}_k$ is empty or not.

As the size of S_i^* can be bounded by $f(k, \ell)$, the running time is $f(k, \ell) \cdot n^{O(1)}$.

Applications

- Matroid intersection.
- Packing problems.
- A terminal location problem.

Matroid intersection

Reminder: Finding a set of maximum size in the intersection of 2 matroids is polynomial-time solvable, but becomes NP-hard for the intersection of 3 matroids.

Can we find an independent set of size k in fpt-time?

Matroid intersection

Reminder: Finding a set of maximum size in the intersection of 2 matroids is polynomial-time solvable, but becomes NP-hard for the intersection of 3 matroids.

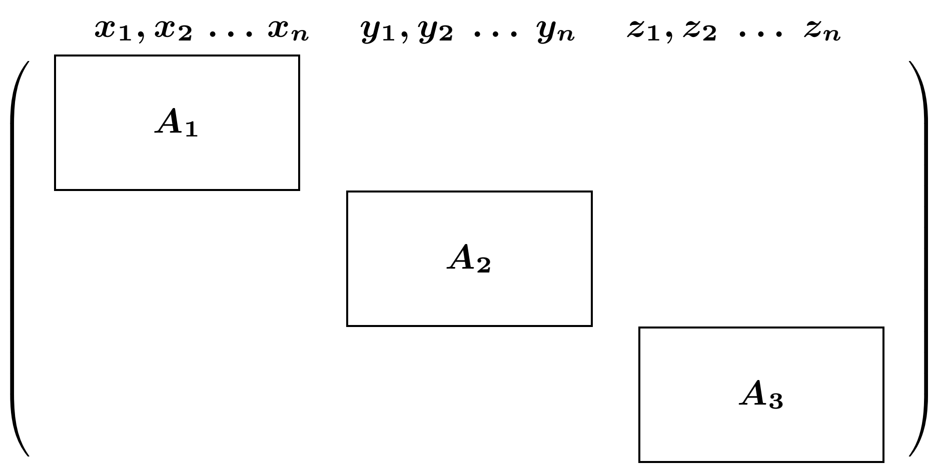
Can we find an independent set of size k in fpt-time?

Theorem: Given ℓ matroids with representations $A_1, \ldots, A_\ell$, we can determine in randomized time $f(k, \ell) \cdot n^{O(1)}$ whether the intersection contains a set of size k.

Matroid intersection

Theorem: Given ℓ matroids with representations $A_1, \ldots, A_\ell$, we can determine in randomized time $f(k, \ell) \cdot n^{O(1)}$ whether the intersection contains a set of size k.

Consider the matrix A with the partition $\{x_1, y_1, z_1\}, \ldots, \{x_n, y_n, z_n\}$:



Union of k blocks is in independent in A

$$\Updownarrow$$

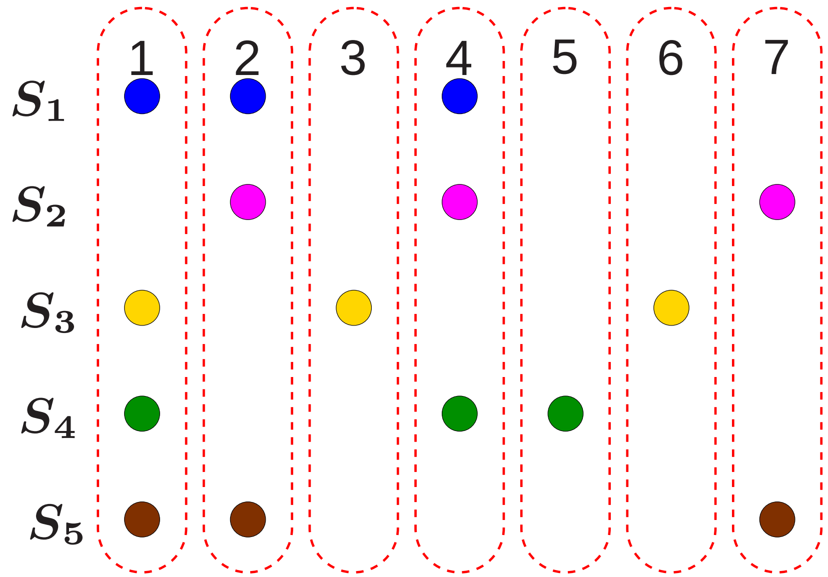
$\mathcal{M}_1 \wedge \mathcal{M}_2 \wedge \mathcal{M}_3$ has a set of size k

Disjoint subsets

Let $S_1, \ldots, S_n \subseteq E$ be subsets of size ℓ.

Task: Find k pairwise disjoint subsets.

Example: ($\ell = 3$) Let $S_1 = \{1, 2, 4\}, S_2 = \{2, 4, 7\}, S_3 = \{1, 3, 6\}, S_4 = \{1, 4, 5\}, S_5 = \{1, 2, 7\}$. Consider the following partition matroid (at most one element in each class):

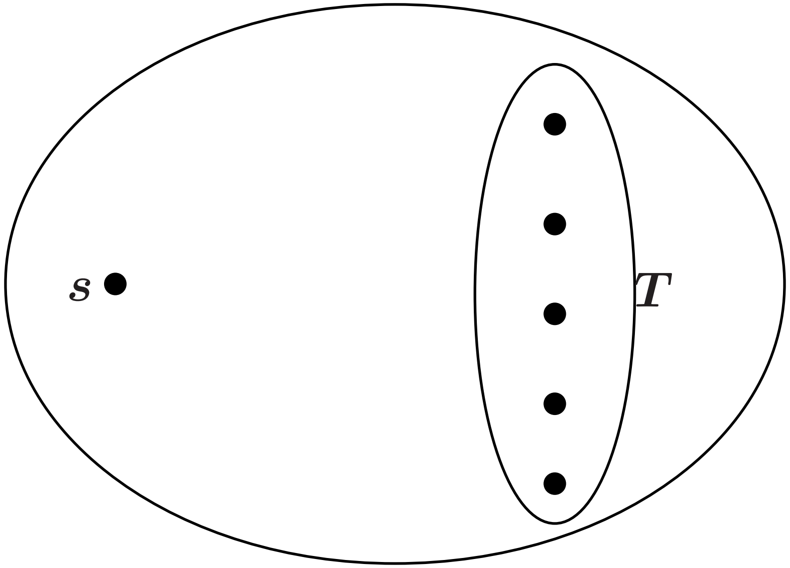


union of k triples is independent $\Leftrightarrow$ there are k disjoint triples

Reliable terminals

Let D be a directed graph with a source vertex s and a subset T of vertices.

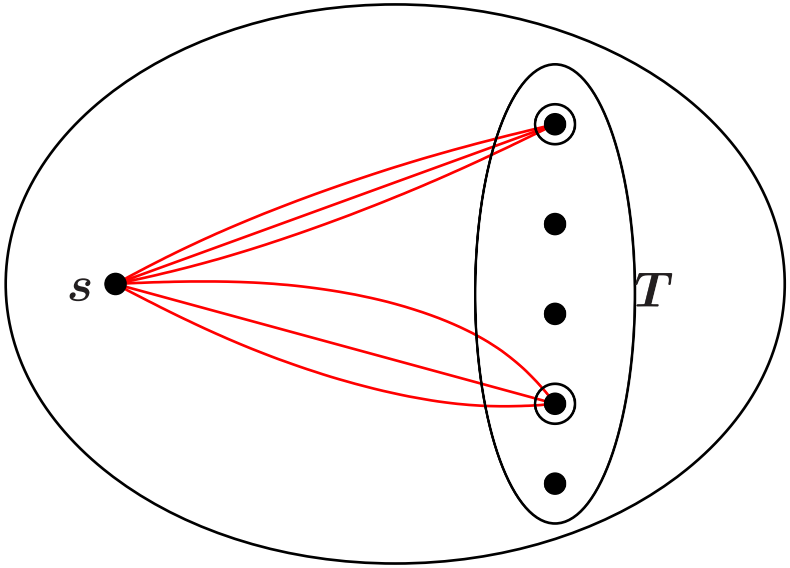
Task: Select k terminals $t_1, \ldots, t_k \in T$, and ℓ paths from s to each t_i such that these $k \cdot \ell$ paths are pairwise internally vertex disjoint.



Reliable terminals

Let D be a directed graph with a source vertex s and a subset T of vertices.

Task: Select k terminals $t_1, \ldots, t_k \in T$, and ℓ paths from s to each t_i such that these $k \cdot \ell$ paths are pairwise internally vertex disjoint.

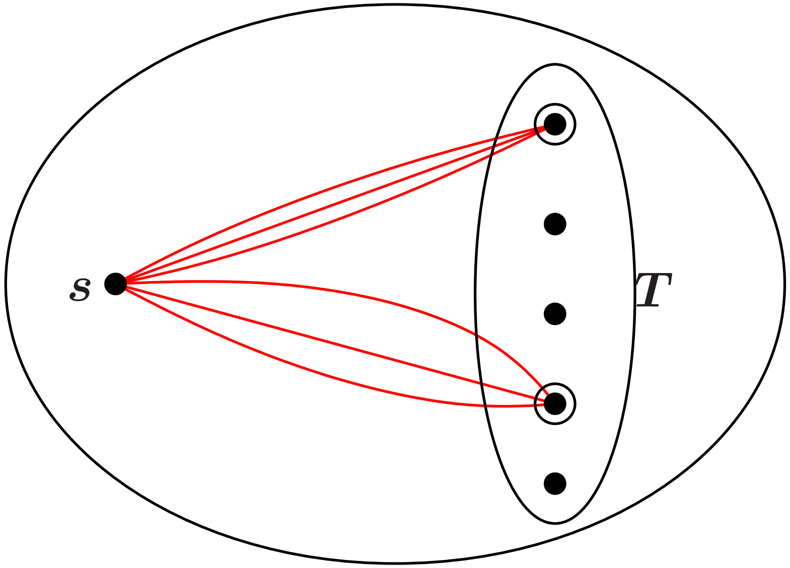


$k = 2, \ell = 3$

Reliable terminals

Let D be a directed graph with a source vertex s and a subset T of vertices.

Task: Select k terminals $t_1, \ldots, t_k \in T$, and ℓ paths from s to each t_i such that these $k \cdot \ell$ paths are pairwise internally vertex disjoint.

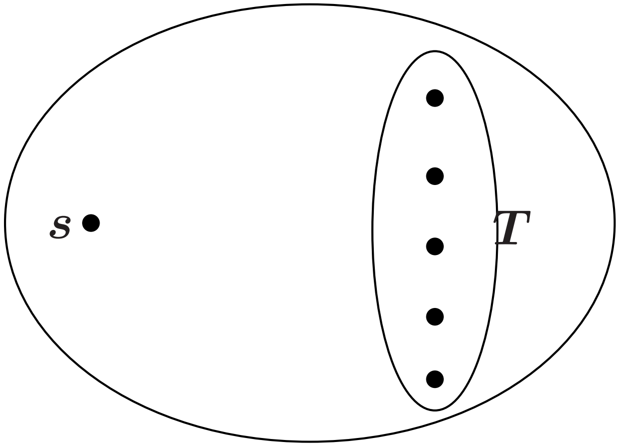


$$k = 2, \ell = 3$$

Theorem: The problem can be solved in randomized time $f(k, \ell) \cdot n^{O(1)}$.

Reliable terminals

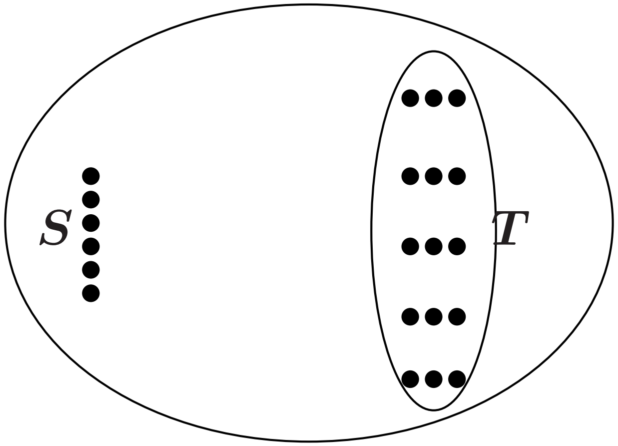
A technical trick: replace each $t \in T$ with ℓ copies, and replace s with a set S of $k \cdot \ell$ copies.



$k = 2, \ell = 3$

Reliable terminals

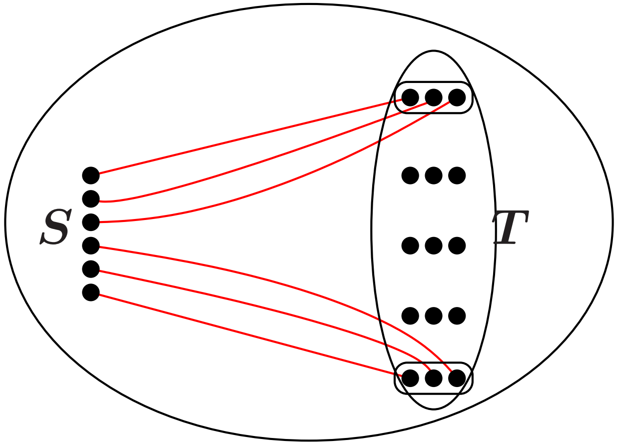
A technical trick: replace each $t \in T$ with ℓ copies, and replace s with a set S of $k \cdot \ell$ copies.



$$k = 2, \ell = 3$$

Reliable terminals

A technical trick: replace each $t \in T$ with ℓ copies, and replace s with a set S of $k \cdot \ell$ copies.

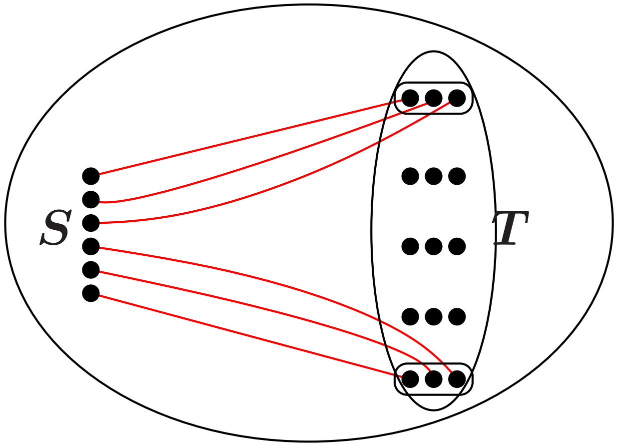


$$k = 2, \ell = 3$$

Now if a terminal t is selected, then we should connect the ℓ copies of t with ℓ different vertices of S.

Reliable terminals

A technical trick: replace each $t \in T$ with ℓ copies, and replace s with a set S of $k \cdot \ell$ copies.



$k = 2, \ell = 3$

Now if a terminal t is selected, then we should connect the ℓ copies of t with ℓ different vertices of S.

Fact: [Perfect] Let D be a directed graph and S a subset of vertices. Those subsets X that can be reached from S by disjoint paths form a matroid.

Conclusions

- Randomized fixed-parameter tractability of a general matroid problem.
- Operations on representations.
- Applications.