Matroid parameters for fixed-parameter tractability

Sarolta Oláh
Supervisor: Kristóf Bérczi

May 19, 2024

1 Introduction

This semester I studied matroid parameters, which are essential tools in the design of parameterized algorithms.

Fixed-parameter tractability focuses on specific parameters to handle computationally hard problems. It enables the development of efficient algorithms, with running time which is exponential only in the size of a fixed parameter while polynomial in the size of the input.

This initiated research into practical parameters for combinatorial optimization. Among the various graph parameters are tree-width and tree-depth. Tree-width measures how close a graph is to being a tree, allowing many NP-hard problems, like the traveling salesman problem and vertex cover, to be solved in polynomial time on graphs with bounded tree-width using dynamic programming. Tree-depth measures how close a graph is to being a star, defined as the minimum height of a rooted forest whose closure contains the graph. Similarly, many hard problems become polynomial with bounded tree-depth, such as the mixed Chinese postman problem.

As for many graph attributes, a natural idea is to try and generalize these types of parameters to the field of matroids. The generalization is not that simple, since these definitions include vertices, which concepts cannot be generalized into matroids. A perception to overcome this problem was to create a definition for parameters using only the notion of connectivity-functions. The obtained parameters are in 3.

2 Basics of matroids and connectivity functions

The notion of matroid was introduced by Hassler Whitney in 1933. There are many ways to define a matroid axiomatically, the following definition contains the independence axioms. A matroid is given by a pair $(S, \mathcal{F})$, where S is a set of elements, the so-called **ground-set**, and $\mathcal{F}$ contains certain subsets of S. These subsets should satisfy the next three axioms.

Definition 2.1. A set-system $M = (S, \mathcal{F})$ is called a **matroid** if it satisfies the following properties, called **independence axioms**.

- $\emptyset \in \mathcal{F}$.
- If $X \subseteq Y \in \mathcal{F}$, then $X \in \mathcal{F}$.
- For every subset $X \subseteq S$, the maximal subsets of X which are in $\mathcal{F}$ have the same cardinality.

The members of $\mathcal{F}$ are called **independent**, the other subsets of S are called **dependent**. The maximal independent subsets of S are the **basis**, and the minimal dependent sets are called **circuits**.

The **rank of** X, where $X \subseteq S$ denoted by $r(X)$ is the size of the maximal independent set in X, here r is the **rank-function** and $r(S)$ is the rank of the matroid. An element s, which has $r(\{s\}) = 0$ is a **loop**, and an element s such that $r(M \setminus \{s\}) = r(M) - 1$ is a **bridge**. The **connected components** of a matroid are inclusion-wise maximal sets, that for every two element of a component there exists a circuit which contains them.

The **dual** of a matroid M, signed by M^*, is defined on S and the bases of M^* are the complements of the bases of M.

The two most important matroid operations are **deletion** and **contraction**. For a set $Z \subseteq S$, $M \setminus Z$ means the **deletion** of Z from the matroid. This way we get a new matroid with ground-set $S \setminus Z$, a set is independent here if and only if it is independent in M. M/Z means the **contraction** of Z, the matroid obtained from this has ground-set $S \setminus Z$ and a set Z' is independent here, if $r(Z \cup Z') = r(Z) + r(Z')$, and Z' is independent in M. A matroid obtained by deletions and contractions is called a **minor**.

The two most notable matroids are called **vector** or **linear** matroids, and **graphic** or **circuit** matroid. A vector matroid is where S is defined to be a finite set of vectors over an arbitrary field $\mathbf{F}$, $\mathcal{F}$ contains the linear independent subsets of S. A matroid is **representable** over a finite field F, if there exists a vector matroid isomorphic to it. A graphic matroid is defined as follows. Let S be the edge-set of an undirected graph, and let $\mathcal{F}$ contain the subsets of edges, that contains no circuit. The independent sets form trees and forests in the graph.

Let S be a finite set of elements. The next definition describes connectivity functions.

Definition 2.2. $\lambda : 2^S \to \mathbb{Z}$ is a **connectivity function** if it satisfies the following three properties.

- $\lambda(\emptyset) = 0$,
- $\lambda(X) = \lambda(S \setminus X)\ \forall X \subseteq S$, symmetry,
- $\lambda(X) + \lambda(Y) \geq \lambda(X \cap Y) + \lambda(X \cup Y)\ \forall X, Y \subseteq S$, submodularity.

For a graph, the following is a connectivity function. For $G = (V, E)$, for $X \subseteq E$, $\lambda_G(X)$ is the number of vertices, which have edges from both X and $E \setminus X$ sets.

For a matroid $M = (S, \mathcal{F})$, a connectivity function $\lambda_M : 2^S \to \mathbb{Z}$ is defined as $\lambda_M(X) = r_M(X) + r_M(S \setminus X) - r_M(S)$, for all $X \subseteq S$, where r is the rank-function of M.

3 The parameters

For the analogue of tree-width, Robertson and Seymour [13] introduced the branch-width parameter and demonstrated that its tied to tree-width for graphs. Geelen, Gerards, and Whittle [6] extended the study of branch-width to matroids. Branch-width became a crucial parameter partly because Hliněný [7] proved that any property definable in the monadic second-order logic of matroids can be tested in polynomial time for matroids represented over a fixed finite field with bounded branch-width.

Definition 3.1. A **branch-decomposition** of H (where H can be an edge set of a graph, hypergraph, the domain of a function or a matroid) is a (T, L) pair, where T is a sub-cubic tree (all nodes have at most 3 neighbours), and L is a bijection between the elements of H, and the leaves of T. (T, L) is a **partial branch-decomposition** if L is only surjective.

An edge $e \in T$, splits it into two connected components. This gives a partition (E_1, E_2) in H. Using this, we can specify the width of the decomposition, along with the branch-width.

Definition 3.2. For a connectivity function λ the **width of an edge** e is $\lambda(E_1) = \lambda(E_2)$, where (E_1, E_2) is the partition induced by e. For a graph $G = (V, E)$ it is $\lambda_G(E_1) = \lambda_G(E_2)$, where λ_G is the connectivity function of G. For a matroid M, it is $\omega_T(e) = \lambda_M(E_1) + 1 = \lambda_M(E_2) + 1$, where λ_M is the connectivity function of matroids.

Definition 3.3. The **width of T**, if T is a branch-decomposition, is the maximum edge-width for all $e \in T$. **Branch-width** is the minimum width over all branch-decompositions. Its notation for graphs, matroids and connectivity-functions is $\mathrm{bw}(G)$, $\mathrm{bw}(M)$ and $\mathrm{bw}(\lambda)$, respectively. T is **tight**, if there is no other branch-decomposition with less branch-width.

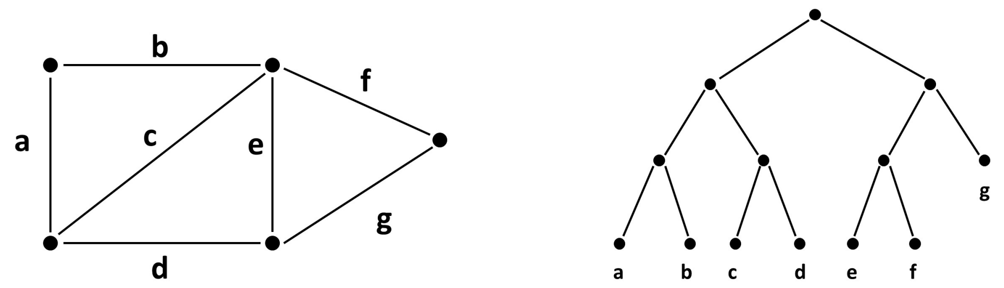

Figure 1: Example of a branch-decomposition. The graph on the left is viewed as a cycle matroid, the right picture shows an optimal branch-decomposition with $\mathrm{bw}(M) = 3$.

Computing branch-width is NP-hard, but, it was shown by Hliněňy [8] with an algorithm, that for matroids, represented over a fixed finite field, branch-width is fix-parameter tractable. Sang-il Oum and Paul Seymour [10] created a 3-approximating algorithm for creating branch-decompositions and in [5] Fomin and Korhonen made a 2-approximating one for the same problem.

Many researchers defined parameters to generalize tree-depth. The notion branch-depth as a generalization, was introduced by DeVos Kwon and Oum [4]. Their definition produced various parameters, such as rank-depth of a graph, by substituting different kinds of connectivity-functions.

The branch-depth of a matroid is the same as the branch-depth of its connectivity function. The definition is the following.

Definition 3.4. Let S be a finite set of elements. A **depth-decomposition** of a connectivity function $\lambda : 2^S \to \mathbb{Z}$ is a (T, L) pair, where T is a tree with at least one internal node, which is a node that has child nodes.

The **radius** of a (T, L) decomposition is the radius of the tree T. It is the smallest number r, so there exists a node with distance at most r from every node.

Definition 3.5. Let (T, L) be a decomposition of a connectivity function λ. For an internal node $v \in V(T)$, the connected components of the graph $T \setminus \{v\}$ give a partition $\mathcal{P}_v$ on E by L. The **width of** v is defined to be $\lambda(\mathcal{P}_v)$, where $\lambda(\mathcal{P}_v) = \max_{\mathcal{P} \subseteq \mathcal{P}_v} \lambda_M \left(\bigcup_{X \in \mathcal{P}} X\right)$. The **width of the decomposition** (T, L) is the maximum width of an internal node of T. We say that a decomposition (T, L) is a (k, r)-decomposition of λ if the width is at most k and the radius is at most r. The **branch-depth** of λ, denoted by $\mathrm{bd}(\lambda)$, is the minimum k such that there exists a (k, k)-decomposition of λ.

Figure 1 is also an excellent example of branch-depth. Since the root has distance at most 3 from every edge and every nodewidth is at most 2, it is a $(3, 3)$-decomposition.

The same computational properties go for branch-depth as for branch-width, because of a connection between them. If a matroid has bounded branch-depth, it has bounded branch-width, hence the fpt algorithm for branch-width can be used for computing branch-depth, with a little modification [4].

Other depth parameters are contraction-depth, deletion-depth and contraction-deletion-depth. The first two notions were first researched by Robertson and Seymour under the names C-type and D-type [12]. Similarly, contraction-deletion-depth was investigated by, Ding Oporowski and Oxley with the name type. These names were given by DeVos, Kwon and Oum [4].

Definition 3.6.

- If $E(M) = \emptyset$, then $\mathrm{dd}(M) = \mathrm{cd}(M) = \mathrm{cdd}(M) = 0$.
- If M is not connected, then $\mathrm{dd}(M)$, $\mathrm{cd}(M)$, $\mathrm{cdd}(M)$ is the maximum respective depth of the matroid's components.
- If M is connected, and $E(M) \neq \emptyset$, then:
  - $\mathrm{dd}(M) = 1 + \min_{e\in M}\{\mathrm{dd}(M \setminus e)\}$.
  - $\mathrm{cd}(M) = 1 + \min_{e\in M}\{\mathrm{cd}(M/e)\}$
  - $\mathrm{cdd}(M) = \min\{\min_{e\in M}\{\mathrm{dd}(M \setminus e)\}, \min_{e\in M}\{\mathrm{cd}(M/e)\}\}$

These parameters can also be set, by the height of their decomposition trees. From definition 3.6 it follows trivially, that contraction-deletion-depth of a matroid is at most its contraction-depth, and at most its deletion-depth. An observation is that contraction-depth and deletion-depth are dual notions, for a matroid M, $\mathrm{cd}(M) = \mathrm{dd}(M^*)$, where M^* is the dual matroid.

Kardos, Král', Liebenau and Mach [9] introduced contraction*-depth, as another analogue for graph tree-depth. They introduced it with the name branch-depth, around the same time DeVos, Kwon and Oum introduced their parameter, hence they changed the name. The last parameter that I paid attention to is contraction*-deletion-depth. It was first introduced and studied in [3]. These parameters gained importance due to their connection to preconditioners in combinatorial optimization.

Definition 3.7. Contraction*-depth decomposition is a pair (T, f), where T is a tree with $r(M)$ edges and f maps the elements to the leaves, so for every set of elements $X \subseteq S$, the number of edges in the rooted subtree induced by $f(X)$, denoted by $||T^*(X)||$, is at least $r(X)$. **Contraction*-depth** is the minimum depth of a contraction*-depth-decomposition of M.

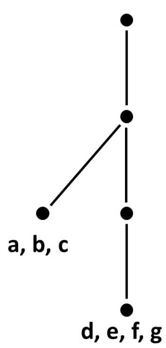

Figure 2: Example of a contraction*-decomposition, of the same matroid as in Figure 1. The consequence of this decomposition is that $\mathrm{c^*d}(M) = 3$, since it is relatively easy to see, that no other decomposition with less depth would fulfill the rank requirements.

For matroids represented over a fixed finite field $\mathbb{F}$, there exists a recursive definition, that doesn't include the decomposition.

The definition of contraction*-deletion-depth in [11] is recursive for all types of matroids. It is as follows.

Definition 3.8.

- If $r(M) = 0$ then $\mathrm{c}^*\mathrm{dd}(M) = 0$.
- If $r(M) = 1$ then $\mathrm{c}^*\mathrm{dd}(M) = 1$.
- If M is disconnected, then $\mathrm{c}^*\mathrm{dd}(M)$ is the maximum contraction*-deletion-depth of components of M.
- If M is connected then $\mathrm{c}^*\mathrm{dd}(M)$ is the minimum contraction*-deletion-depth of $(M \setminus e)$ increased by 1.

Contraction-depth, deletion-depth, contraction-deletion-depth, contraction*-depth, and contraction*-deletion-depth can be extended from matroids to matrices. For a given matrix A, these parameters are defined as the corresponding parameters of its associated column matroid.

These parameters do not share the computational properties of branch-width and branch-depth. In fact, even the questions whether for a matroid these types of parameters are less than k, are shown to be NP-complete [1].

Some trivial connections between the parameters were mentioned earlier, now let us explore some more complex ones.

In [4] it was shown, that the branch-depth of a matroid is at most its contraction-deletion-depth and branch-width of a matroid is at most its branch-depth. Moreover, in [3] Briański, Král' and Pekárková presented, that branch-depth is the minor closure of contraction-deletion-depth, making them functionally equivalent for representable matroids when minor closures are considered.

Briański, Král' and Lamaison [2] proved another functional equality, it is between the parameters contraction-depth and contraction*-depth.

4 Open questions

Matroid parameters opened the door to many new research topics. For graphs with bounded tree-width, it was shown that it is tractable to decide whether they are isomorphic. The question raise, can we do something similar for matroids with some bounded parameter? Král' and Pekárková are currently working on this problem, more specifically they are trying to design a parameterized algorithm, that for given two represented matroids with bounded branch-width, can decide whether they are isomorphic. For contraction*-depth there exists an approximation algorithm computing a depth-decomposition, only issue, is that the approximation factor is not constant. It is open to improve the approximation ratio, and to find approximation algorithms for the other parameters.

References

[1] M. Briański, M. Koutecký, D. Král, K. Pekárková, and F. Schröder. Characterization of matrices with bounded graver bases and depth parameters and applications to integer programming. *Mathematical Programming*, pages 1–35, 2024.

[2] M. Brianski, D. Kral, and A. Lamaison. Closure property of contraction-depth of matroids. *arXiv preprint arXiv:2311.01945*, 2023.

[3] M. Briański, D. Král', and K. Pekárková. Branch-depth is minor closure of contraction-deletion-depth. *arXiv preprint arXiv:2402.16215*, 2024.

[4] M. DeVos, O.-j. Kwon, and S.-i. Oum. Branch-depth: Generalizing tree-depth of graphs. *European Journal of Combinatorics*, 90:103186, 2020.

[5] F. V. Fomin and T. Korhonen. Fast fpt-approximation of branchwidth. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 886–899, 2022.

[6] J. F. Geelen, A. M. Gerards, and G. Whittle. Branch-width and well-quasi-ordering in matroids and graphs. *Journal of Combinatorial Theory, Series B*, 84(2):270–290, 2002.

[7] P. Hliněnỳ. On matroid properties definable in the mso logic. In *Mathematical Foundations of Computer Science 2003: 28th International Symposium, MFCS 2003, Bratislava, Slovakia, August 25-29, 2003. Proceedings 28*, pages 470–479. Springer, 2003.

[8] P. Hlinenỳ. A parametrized algorithm for matroid branch-width. *SIAM Journal on Computing*, 35(2):259–277, 2005.

[9] F. Kardoš, A. Liebenau, L. Mach, et al. First order convergence of matroids. *European Journal of Combinatorics*, 59:150–168, 2017.

[10] S.-i. Oum and P. Seymour. Approximating clique-width and branch-width. *Journal of Combinatorial Theory, Series B*, 96(4):514–528, 2006.

[11] K. PEKÁRKOVÁ. Matroid algorithms and their applications in optimization.

[12] N. Robertson and P. D. Seymour. Graph minors—a survey. *Surveys in combinatorics*, 103:153–171, 1985.

[13] N. Robertson and P. D. Seymour. Graph minors. x. obstructions to tree-decomposition. *Journal of Combinatorial Theory, Series B*, 52(2):153–190, 1991.