Graph-separation problems: introduction and a glimpse on the history

Marcin Pilipczuk

Institute of Informatics
University of Warsaw

9 April 2013

Prehistory

Feedback Vertex Set

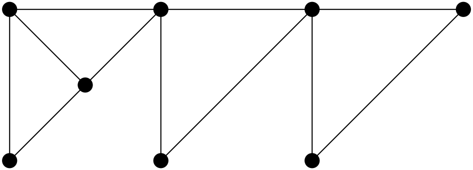
Feedback Vertex Set (FVS)

Input: A graph $G = (V, E)$ and an integer k.
Question: Does there exist $X \subseteq V$, $|X| \leq k$, such that $G \setminus X$ is a forest?
In other words, X hits all cycles of G.

Feedback Vertex Set

Feedback Vertex Set (FVS)

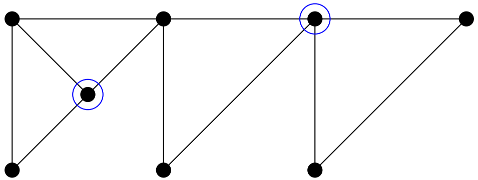
Input: A graph $G = (V, E)$ and an integer k.
Question: Does there exist $X \subseteq V$, $|X| \leq k$, such that $G \setminus X$ is a forest?
In other words, X hits all cycles of G.



Feedback Vertex Set

Feedback Vertex Set (FVS)

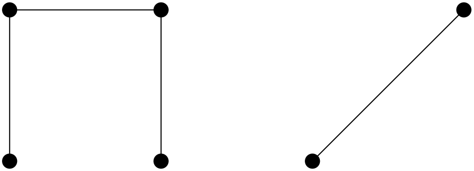
Input: A graph $G = (V, E)$ and an integer k.
Question: Does there exist $X \subseteq V$, $|X| \leq k$, such that $G \setminus X$ is a forest?
In other words, X hits all cycles of G.



Feedback Vertex Set

Feedback Vertex Set (FVS)

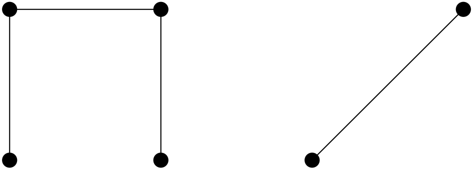
Input: A graph $G = (V, E)$ and an integer k.
Question: Does there exist $X \subseteq V$, $|X| \leq k$, such that $G \setminus X$ is a forest?
In other words, X hits all cycles of G.



Feedback Vertex Set

Feedback Vertex Set (FVS)

Input: A graph $G = (V, E)$ and an integer k.
Question: Does there exist $X \subseteq V$, $|X| \leq k$, such that $G \setminus X$ is a forest?
In other words, X hits all cycles of G.



- Classical problem with multiple applications. On Karp's list of 21 NP-hard problems.

Feedback Vertex Set

- One motivation: circuit design.

Feedback Vertex Set

- One motivation: circuit design.
  - Sensors on a feedback vertex set control all closed circuits.

Feedback Vertex Set

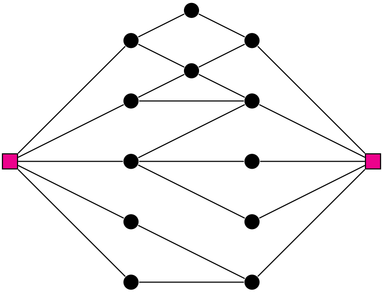
- One motivation: circuit design.
  - Sensors on a feedback vertex set control all closed circuits.
- Expect a small solution $\Rightarrow$ parameterize by solution size!

Feedback Vertex Set

- One motivation: circuit design.
  - Sensors on a feedback vertex set control all closed circuits.
- Expect a small solution $\Rightarrow$ parameterize by solution size!

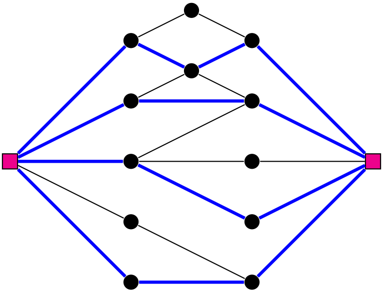
Bodlaender Downey and Fellows	$\mathcal{O}(17(k^4)!n^{\mathcal{O}(1)})$	1994
Downey and Fellows	$\mathcal{O}((2k+1)^k n^2)$	1999
Becker, Bar-Yehuda, Geiger	$\mathcal{O}(4^k kn)$ randomized	2000
Raman, Saurabh, Subramanian	$\mathcal{O}(\max\{12^k, (4\log k)^k\} n^\omega)$	2002
Kanj, Pelsmajer, Schaefer	$\mathcal{O}((2\log k + 2\log\log k + 18)^k n^2)$	2004
Raman, Saurabh, Subramanian	$\mathcal{O}((12\log k/\log\log k + 6)^k n^\omega)$	2006
Guo, Gramm, Hüffner, Niedermeier, Wernicke	$\mathcal{O}(37.7^k n^2)$	2006
Dehne, Fellows, Langston, Rosamond, Stevens	$\mathcal{O}(10.6^k n^3)$	2005
Chen, Fomin, Liu, Lu, Villanger	$\mathcal{O}(5^k kn^2)$	2008
Cao, Chen, Liu	$\mathcal{O}(3.83^k kn^2)$	2010
Cygan, Nederlof, P., Pilipczuk, Rooij, Wojtaszczyk	$\mathcal{O}(3^k n^{\mathcal{O}(1)})$ randomized	2011

Flows and cuts



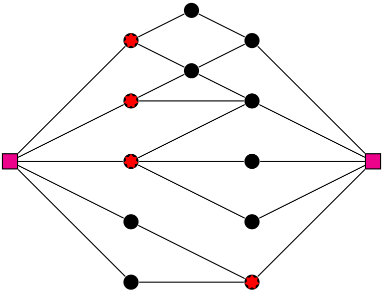
- Given G, s and t,

Flows and cuts



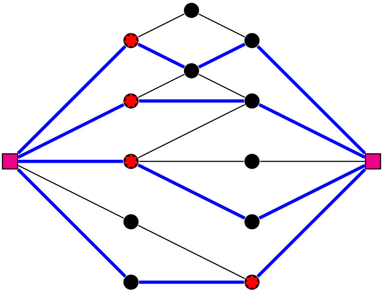
- Given G, s and t,
- finding a maximum flow between s and t is in P,

Flows and cuts



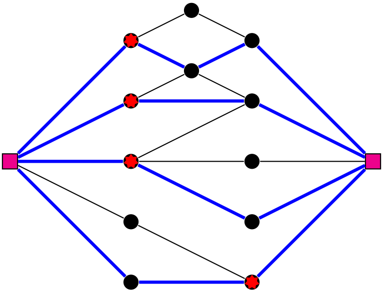
- Given G, s and t,
- finding a maximum flow between s and t is in P,
- as well as finding a minimum cut between s and t.

Flows and cuts



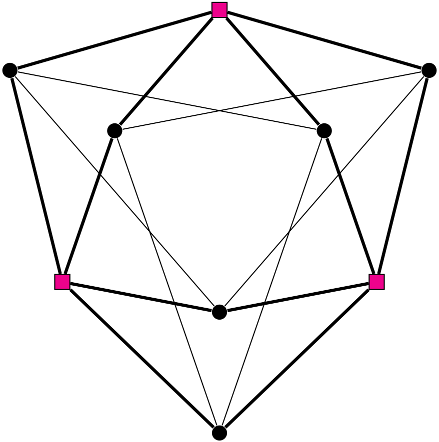
- Given G, s and t,
- finding a maximum flow between s and t is in P,
- as well as finding a minimum cut between s and t.
- Moreover, these two values are equal.

Flows and cuts

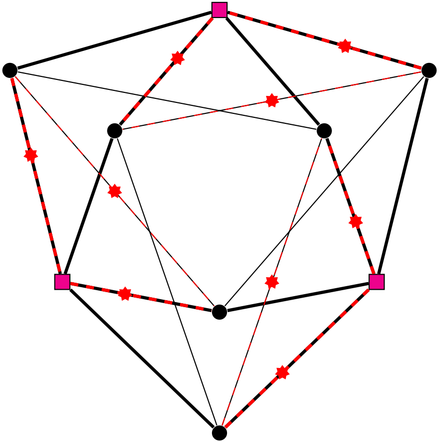


- Given G, s and t,
- finding a maximum flow between s and t is in P,
- as well as finding a minimum cut between s and t.
- Moreover, these two values are equal.
- What happens if we have more terminals?

Non-submodularity example



Non-submodularity example



Multiway Cut

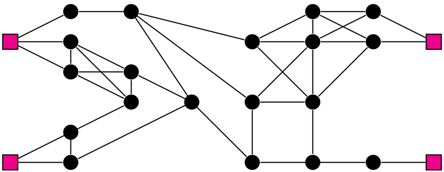
Multiway Cut (MWC)

Input: $G = (V, E)$, a set $T \subseteq V$ of terminals and k.
Question: Does there exist $X \subseteq V \setminus T$, $|X| \leq k$, such that each terminal is in different connected component of $G \setminus X$?

Multiway Cut

Multiway Cut (MWC)

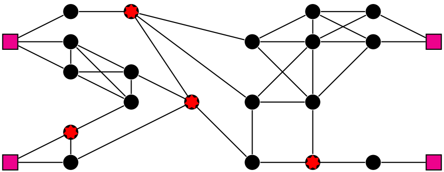
Input: $G = (V, E)$, a set $T \subseteq V$ of terminals and k.
Question: Does there exist $X \subseteq V \setminus T$, $|X| \leq k$, such that each terminal is in different connected component of $G \setminus X$?



Multiway Cut

Multiway Cut (MWC)

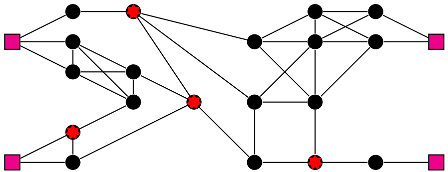
Input: $G = (V, E)$, a set $T \subseteq V$ of terminals and k.
Question: Does there exist $X \subseteq V \setminus T$, $|X| \leq k$, such that each terminal is in different connected component of $G \setminus X$?



Multiway Cut

Multiway Cut (MWC)

Input: $G = (V, E)$, a set $T \subseteq V$ of terminals and k.
Question: Does there exist $X \subseteq V \setminus T$, $|X| \leq k$, such that each terminal is in different connected component of $G \setminus X$?



- Dahlhaus, Johnson, Papadimitriou, Seymour, Yannakakis: NP-hard for 3 terminals (1994).

Important separators

Parameterized Graph Separation Problems⋆

Dániel Marx

Department of Computer Science and Information Theory,
Budapest University of Technology and Economics
Budapest, H-1521, Hungary
dmarx@cs.bme.hu

Abstract. We consider parameterized problems where some separation property has to be achieved by deleting as few vertices as possible. The following five problems are studied: delete k vertices such that (a) each of the given ℓ terminals is separated from the others, (b) each of the given ℓ pairs of terminals are separated, (c) exactly ℓ vertices are cut away from the graph, (d) exactly ℓ connected vertices are cut away from the graph, (e) the graph is separated into ℓ components, We show that if both k and ℓ are parameters, then (a), (b) and (d) are fixed-parameter tractable, while (c) and (e) are W[1]-hard.

Important separators

- Submodularity $\Rightarrow$ unique minimum st-cut 'closest' to t.

Important separators

- Submodularity $\Rightarrow$ unique minimum *st*-cut 'closest' to t.
- Important separators: at most 4^k cuts of size $\leq k$ that are 'closest' to t.

Important separators

- Submodularity $\Rightarrow$ unique minimum *st*-cut 'closest' to t.
- Important separators: at most 4^k cuts of size $\leq k$ that are 'closest' to t.
- Applicable if you can greedily take as much as you can on one side of the cut.

Applicable!

Multicut

Input: $G = (V, E)$, a set $T \subseteq V \times V$ of pairs of terminals and k.
Question: Does there exist $X \subseteq V$ of non-terminals, $|X| \leq k$, such that each pair of terminals in T is separated in $G \setminus X$?

Theorem (Marx, 2004)

Multicut is FPT when parameterized by $|T| + k$.

Applicable!

Directed FVS

Input: A directed graph G and an integer k.
Question: Can we delete at most k vertices from G to make it acyclic?

Theorem (Chen, Liu, Lu, O'Sullivan, Razgon, 2008)

Directed FVS is FPT.

Applicable!

Almost 2-SAT

Input: A 2-CNF-SAT formula ϕ and k.
Question: Can we delete at most k clauses from ϕ to make it satisfiable?

Theorem (O'Sullivan, Razgon, 2008)

Almost 2-SAT is FPT.

Limitations of important separators

Unknown source

- Sometimes we know that part of the solution can be assumed to be an important separator.

Unknown source

- Sometimes we know that part of the solution can be assumed to be an important separator.
- But we do not know where the source is!

Unknown source

- Sometimes we know that part of the solution can be assumed to be an important separator.
- But we do not know where the source is!
- Trick: randomly select some sources.

Unknown source

- Sometimes we know that part of the solution can be assumed to be an important separator.
- But we do not know where the source is!
- Trick: randomly select some sources.
  - Like colour-coding

Unknown source

- Sometimes we know that part of the solution can be assumed to be an important separator.
- But we do not know where the source is!
- Trick: randomly select some sources.
  - Like colour-coding
- **Shadow removal technique.**

Shadow removal: applications

Theorem (Marx, Razgon, 2011)

Multicut is FPT when parameterized by the cutsize k.

Shadow removal: applications

Theorem (Marx, Razgon, 2011)

Multicut is FPT when parameterized by the cutsize k.

Theorem (Chitnis, Hajiaghayi, Marx, 2012)

Directed Multiway Cut is FPT when parameterized by the cutsize k.

Shadow removal: applications

Theorem (Marx, Razgon, 2011)

Multicut is FPT when parameterized by the cutsize k.

Theorem (Chitnis, Hajiaghayi, Marx, 2012)

Directed Multiway Cut is FPT when parameterized by the cutsize k.

Theorem (Chitnis, Cygan, Hajiaghayi, Marx, 2012)

Directed Subset FVS is FPT when parameterized by the cutsize k.

Shadow removal: applications

Theorem (Marx, Razgon, 2011)

Multicut is FPT when parameterized by the cutsize k.

Theorem (Chitnis, Hajiaghayi, Marx, 2012)

Directed Multiway Cut is FPT when parameterized by the cutsize k.

Theorem (Chitnis, Cygan, Hajiaghayi, Marx, 2012)

Directed Subset FVS is FPT when parameterized by the cutsize k.

Theorem (Kratsch, P., Pilipczuk, Wahlström, 2012)

Directed Multicut in DAGs is FPT when parameterized by the cutsize k.

Shadow removal: applications

Clustering with Local Restrictions

Daniel Lokshtanov[1] and Dániel Marx[2,⋆]

[1] University of California, San Diego, USA
dlokshtanov@cs.ucsd.edu
[2] Humboldt-Universität zu Berlin, Berlin, Germany
dmarx@cs.bme.hu

Abstract. We study a family of graph clustering problems where each cluster has to satisfy a certain local requirement. Formally, let μ be a function on the subsets of vertices of a graph G. In the (μ, p, q)-Partition problem, the task is to find a partition of the vertices into clusters where each cluster C satisfies the requirements that (1) at most q edges leave C and (2) $\mu(C) \leq p$. Our first result shows that if μ is an *arbitrary* polynomial-time computable monotone function, then (μ, p, q)-Partition can be solved in time $n^{O(q)}$, i.e., it is polynomial-time solvable *for every fixed* q. We study in detail three concrete functions μ (number of nonedges in the cluster, maximum number of non-neighbours a vertex has in the cluster, the number of vertices in the cluster), which correspond to natural clustering problems. For these functions, we show that (μ, p, q)-Partition can be solved in time $2^{O(p)} \cdot n^{O(1)}$ and in randomized time $2^{O(q)} \cdot n^{O(1)}$, i.e., the problem is fixed-parameter tractable parameterized by p or by q.

Ensuring properties of the cut I

- Go back to Multiway Cut.

Ensuring properties of the cut I

- Go back to Multiway Cut.
- What if we want the cut to be, say, independent?

Ensuring properties of the cut I

- Go back to Multiway Cut.
- What if we want the cut to be, say, independent?
- Greedy argument of important separators does not work!

Treewidth reduction

Theorem (Marx, O'Sullivan, Razgon, 2010)

All minimal st separators of size $\leq k$ live in a part of the graph with treewidth bounded by $2^{\mathrm{poly}(k)}$.

- Computable in FPT time.

Treewidth reduction

Theorem (Marx, O'Sullivan, Razgon, 2010)

All minimal st separators of size $\leq k$ live in a part of the graph with treewidth bounded by $2^{\mathrm{poly}(k)}$.

- Computable in FPT time.
- Independent Multiway Cut: reduce treewidth, perform standard DP.

Treewidth reduction

Theorem (Marx, O'Sullivan, Razgon, 2010)

All minimal st separators of size $\leq k$ live in a part of the graph with treewidth bounded by $2^{\mathrm{poly}(k)}$.

- Computable in FPT time.
- Independent Multiway Cut: reduce treewidth, perform standard DP.
- Drawback: double-exponential dependency on k.

Treewidth reduction

Theorem (Marx, O'Sullivan, Razgon, 2010)

All minimal st separators of size $\leq k$ live in a part of the graph with treewidth bounded by $2^{\mathrm{poly}(k)}$.

- Computable in FPT time.
- Independent Multiway Cut: reduce treewidth, perform standard DP.
- Drawback: double-exponential dependency on k.
- Drawback: keeps cuts only between bounded number of terminals.

Ensuring properties of the cut II

- What about cutting paths of certain parity?

Ensuring properties of the cut II

- What about cutting paths of certain parity?

Parameterized Tractability of Multiway Cut with Parity Constraints

Daniel Lokshtanov[1] and M.S. Ramanujan[2]

[1] University of California, San Diego, USA
daniello@ii.uib.no
[2] The Institute of Mathematical Sciences, Chennai, India
msramanujan@imsc.res.in

Abstract. In this paper, we study a parity based generalization of the classical MULTIWAY CUT problem. Formally, we study the PARITY MULTIWAY CUT problem, where the input is a graph G, vertex subsets T_e and T_o ($T = T_e \cup T_o$) called terminals, a positive integer k and the objective is to test whether there exists a k-sized vertex subset S such that S intersects all odd paths from $v \in T_o$ to $T \setminus \{v\}$ and all even paths from $v \in T_e$ to $T \setminus \{v\}$. When $T_e = T_o$, this

Unknown terminals

Unknown terminals

s-Way Cut

Input: $G = (V, E)$, and integers s and k.
Question: Does there exist $X \subseteq E$ of edges, $|X| \leq k$, such $G \setminus X$ contains at least s connected components?

Unknown terminals

s-Way Cut

Input: $G = (V, E)$, and integers s and k.
Question: Does there exist $X \subseteq E$ of edges, $|X| \leq k$, such $G \setminus X$ contains at least s connected components?

Theorem (Kawarabayashi, Thorup, 2011)

s-Way Cut is FPT when parameterized by k.

Unknown terminals

s-Way Cut

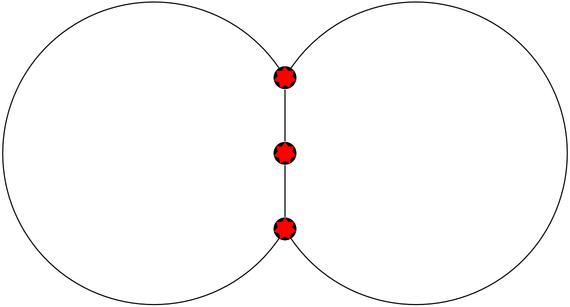
Input: $G = (V, E)$, and integers s and k.
Question: Does there exist $X \subseteq E$ of edges, $|X| \leq k$, such $G \setminus X$ contains at least s connected components?

Theorem (Kawarabayashi, Thorup, 2011)

s-Way Cut is FPT when parameterized by k.

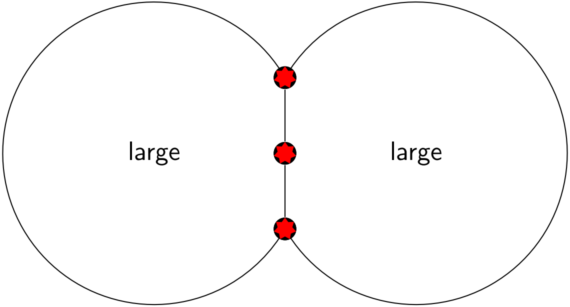
Remark: all other reasonable variants and parameterizations are W[1]-hard (Marx 2004).

Good cuts



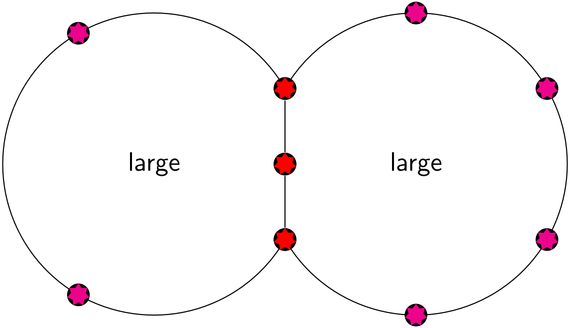
- Assume that we can cut the graph into two parts,

Good cuts



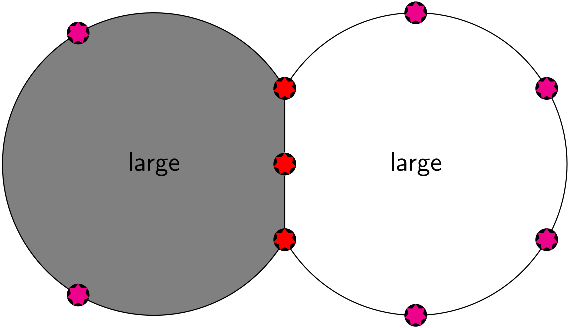
- Assume that we can cut the graph into two parts,
- such that both parts are large.

Good cuts



- Assume that we can cut the graph into two parts,
- such that both parts are large.
- Recursively solve ($2k$-boundaried) problem on one half, and simplify it.

Good cuts

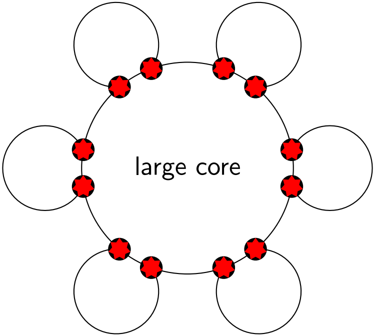


- Assume that we can cut the graph into two parts,
- such that both parts are large.
- Recursively solve ($2k$-boundaried) problem on one half, and simplify it.
  - Choose half with $\leq k$ elements of previous boundary.

No good cuts

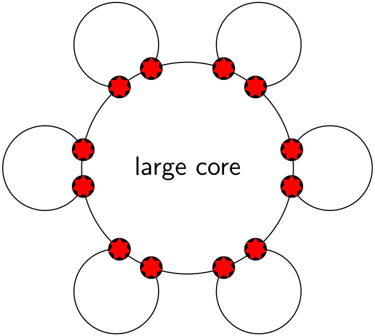
- If there are no good cuts,

No good cuts



- If there are no good cuts,
- the solution needs to contain a huge core and small petals.

No good cuts



- If there are no good cuts,
- the solution needs to contain a huge core and small petals.
- Do problem-specific tricks.

Randomized contractions

2012 IEEE 53rd Annual Symposium on Foundations of Computer Science

Designing FPT Algorithms for Cut Problems Using Randomized Contractions

New Brunswick, NJ, USA USA

October 20-October 23

ISBN: 978-1-4673-4383-1

Rajesh Chitnis

Marek Cygan

MohammadTaghi Hajiaghayi

Marcin Pilipczuk

Michal Pilipczuk

DOI Bookmark: http://doi.ieeecomputersociety.org/10.1109/FOCS.2012.29

- Simpler and faster finding good cuts and final case.

Randomized contractions

2012 IEEE 53rd Annual Symposium on Foundations of Computer Science

Designing FPT Algorithms for Cut Problems Using Randomized Contractions

New Brunswick, NJ, USA USA
October 20-October 23
ISBN: 978-1-4673-4383-1

Rajesh Chitnis
Marek Cygan
MohammadTaghi Hajiaghayi
Marcin Pilipczuk
Michal Pilipczuk

DOI Bookmark: http://doi.ieeecomputersociety.org/10.1109/FOCS.2012.29

- Simpler and faster finding good cuts and final case.
- More robust: easier to use for a new problem.

Randomized contractions

2012 IEEE 53rd Annual Symposium on Foundations of Computer Science

Designing FPT Algorithms for Cut Problems Using Randomized Contractions

New Brunswick, NJ, USA USA
October 20-October 23
ISBN: 978-1-4673-4383-1

Rajesh Chitnis
Marek Cygan
MohammadTaghi Hajiaghayi
Marcin Pilipczuk
Michal Pilipczuk

DOI Bookmark: http://doi.ieeecomputersociety.org/10.1109/FOCS.2012.29

- Simpler and faster finding good cuts and final case.
- More robust: easier to use for a new problem.
- Unique Label Cover: core problem in hardness of approximation.

Randomized contractions: application

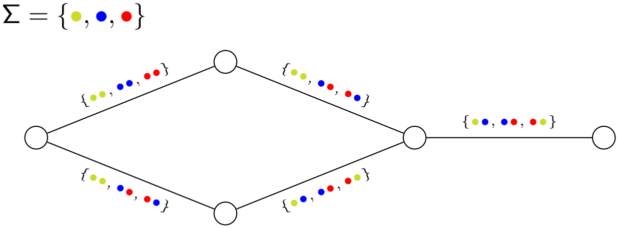
Unique Label Cover (ULC)

Input: Graph $G = (V, E)$, alphabet Σ, an integer k and for each $e \in E$ a constraint ψ_e being a bijection $\Sigma \to \Sigma$.
Question: Can one label the vertices of the graph with elements of Σ such that at most k constraints are not satisfied?

Randomized contractions: application

Unique Label Cover (ULC)

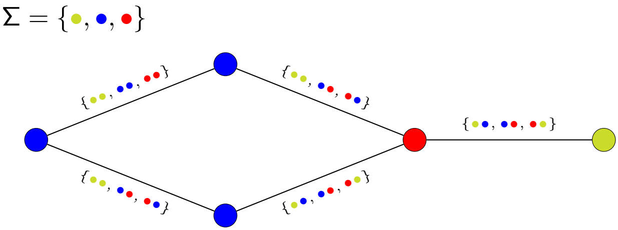
Input: Graph $G = (V, E)$, alphabet Σ, an integer k and for each $e \in E$ a constraint ψ_e being a bijection $\Sigma \to \Sigma$.
Question: Can one label the vertices of the graph with elements of Σ such that at most k constraints are not satisfied?



Randomized contractions: application

Unique Label Cover (ULC)

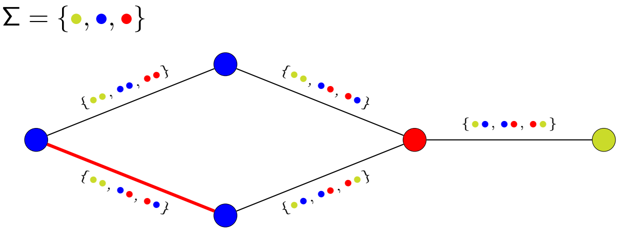
Input: Graph $G = (V, E)$, alphabet Σ, an integer k and for each $e \in E$ a constraint ψ_e being a bijection $\Sigma \to \Sigma$.
Question: Can one label the vertices of the graph with elements of Σ such that at most k constraints are not satisfied?



Randomized contractions: application

Unique Label Cover (ULC)

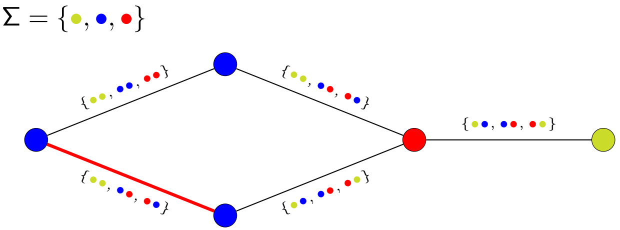
Input: Graph $G = (V, E)$, alphabet Σ, an integer k and for each $e \in E$ a constraint ψ_e being a bijection $\Sigma \to \Sigma$.
Question: Can one label the vertices of the graph with elements of Σ such that at most k constraints are not satisfied?



Randomized contractions: application

Unique Label Cover (ULC)

Input: Graph $G = (V, E)$, alphabet Σ, an integer k and for each $e \in E$ a constraint ψ_e being a bijection $\Sigma \to \Sigma$.
Question: Can one label the vertices of the graph with elements of Σ such that at most k constraints are not satisfied?



Theorem (Chitnis, Cygan, Hajiaghayi, P., Pilipczuk, 2012)

Unique Label Cover *is FPT when parameterized by* $|\Sigma|$ *and* k.

Faster than important separators

- Recall: there are at most 4^k important *st*-cuts of size $\leq k$.

Faster than important separators

- Recall: there are at most 4^k important st-cuts of size $\leq k$.
- This bound is tight (up to poly factors).

Faster than important separators

- Recall: there are at most 4^k important *st*-cuts of size $\leq k$.
- This bound is tight (up to poly factors).
- Can you do faster algorithms than 4^k?

Faster than important separators

On Multiway Cut Parameterized above Lower Bounds*

Marek Cygan[1], Marcin Pilipczuk[1],
Michał Pilipczuk[1], and Jakub Onufry Wojtaszczyk[2]

[1] Institute of Informatics, University of Warsaw, Poland
{cygan@,malcin@,mp248287@students.}mimuw.edu.pl
[2] Google Inc., Cracow, Poland
onufry@google.com

Abstract. In this paper we consider two *above lower bound* parameterizations of the Node Multiway Cut problem — above the maximum separating cut and above a natural LP-relaxation — and prove them to be fixed-parameter tractable. Our results imply $O^*(4^k)$ algorithms for Vertex Cover above Maximum Matching and Almost 2-SAT as well as an $O^*(2^k)$ algorithm for Node Multiway Cut with a standard parameterization by the solution size, improving previous bounds for these problems.

LP can be a cure for Parameterized Problems

N.S. Narayanaswamy[1], Venkatesh Raman[2], M.S. Ramanujan[2], and Saket Saurabh[2]

1 **Department of Computer Science and Engineering, IIT Madras, Chennai, India.**
swamy@cse.iitm.ernet.in

2 **The Institute of Mathematical Sciences, Chennai 600113, India.**
{vraman|msramanujan|saket}@imsc.res.in

Abstract

We investigate the parameterized complexity of Vertex Cover parameterized above the optimum value of the linear programming (LP) relaxation of the integer linear programming formulation of the problem. By carefully analyzing the change in the LP value in the branching steps, we argue that even the most straightforward branching algorithm (after some preprocessing) results in an $O^*(2.6181^r)$ algorithm for the problem where r is the excess of the vertex cover size over the LP optimum. We write $O^*(f(k))$ for a time complexity of the form $O(f(k)n^{O(1)})$, where $f(k)$ grows exponentially with k.

Faster than important separators

Theorem (Cygan, P., Pilipczuk, Wojtaszczyk, 2011)

Multiway Cut can be solved in time $\mathcal{O}^\star(2^k)$.

Faster than important separators

Theorem (Cygan, P., Pilipczuk, Wojtaszczyk, 2011)

Multiway Cut can be solved in time $\mathcal{O}^\star(2^k)$.

Theorem (Lokshtanov, Narayanaswamy, Raman, Ramanujan, Saurabh, 2012)

Almost 2-SAT can be solved in time $\mathcal{O}^\star(2.3146^k)$.

Thank you

Questions?