Experiments with Parameterized Approaches to Hard Graph Problems

Falk Hüffner

joint work with

Nadja Betzler   Rolf Niedermeier   Sebastian Wernicke
Thomas Zichner

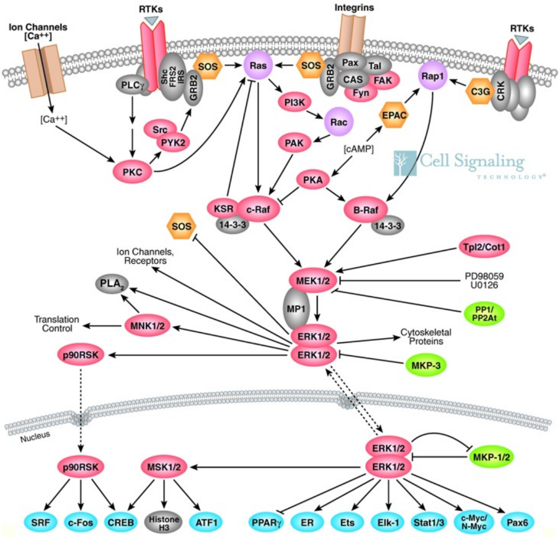
Friedrich-Schiller-Universität Jena
Institut für Informatik

Dagstuhl Seminar N° 07281
Structure Theory and FPT Algorithmics for
Graphs, Digraphs and Hypergraphs

Outline

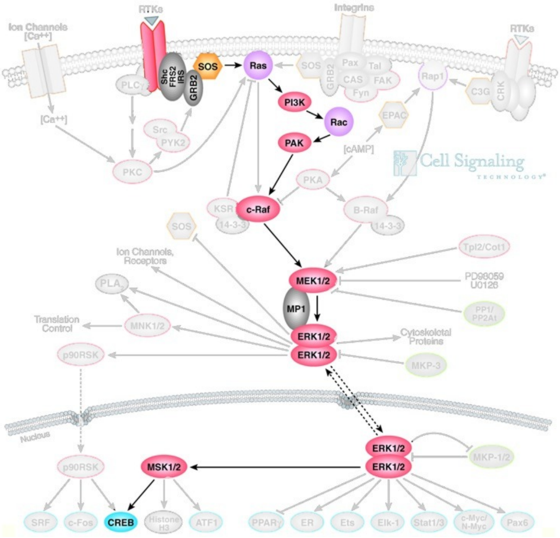
1. Minimum-Weight Path
    - Application: protein interaction networks
    - Color-coding
    - Speedups
2. Minimum-Weight Path experiments
3. Balanced Subgraph
    - Applications
    - Data reduction
    - Iterative compression
4. Balanced Subgraph experiments

Signaling pathways



[www.cellsignal.com]

Signaling pathways



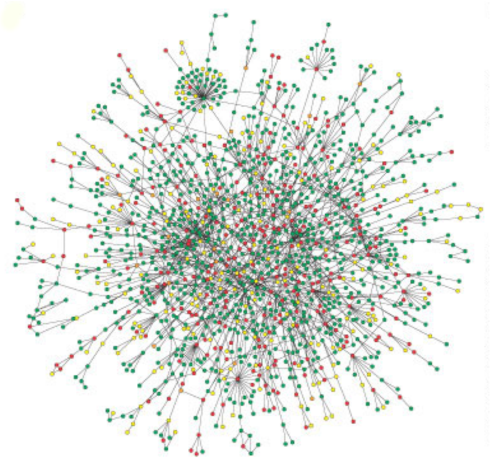
[www.cellsignal.com]

Signaling pathways

Minimum-Weight Path

Input: Graph $G = (V, E)$, weights $w : E \rightarrow \mathbb{R}^+$, integer $k > 0$.
Task: Find a non-overlapping path $v_1, \ldots, v_k$ of length k in G that minimizes $w(v_1, v_2) + \cdots + w(v_{k-1}, v_k)$.

Example: yeast network



4 400 proteins, 14 300 interactions, looking for paths of length 5–15

Minimum-Weight Path

Theorem

MINIMUM-WEIGHT PATH *is NP-hard* [GAREY & JOHNSON 1979].

Idea

Exploit the fact that the paths sought for are rather short ($\approx$ 5–15): parameter k.

Color-coding

Color-coding [Alon, Yuster & Zwick J. ACM 1995]

- randomly color each vertex of the graph with one of k colors
- hope that all vertices in the subgraph searched for obtain different colors (colorful)
- solve the Minimum-Weight Path under this assumption (which is much quicker)
- repeat until it is reasonably certain that the path was colorful at least once

Result: FPT algorithm

Dynamic programming for Minimum-Weight Colorful Path

Idea

Table entry $W[v, C]$ stores the minimum-weight path that ends in v and uses exactly the colors in C.

- Each table entry can be calculated in $O(n)$ time
- $n \cdot 2^k$ table entries

$\rightsquigarrow$ Running time per trial: $O(2^k \cdot n^2)$

To obtain error probability ε, one needs $O(-\ln \varepsilon \cdot e^k)$ trials

Theorem ([Alon et al. JACM 1995])

Minimum-Weight Path *can be solved in* $O(-\ln \varepsilon \cdot 5.44^k |G|)$ *time.*

Implementations of color-coding

- Find minimum-weight paths of length 10 in the yeast protein interaction networks within 3 hours ($n = 4\,400$, $k = 10$)
  [Scott et al., RECOMB 2005]
- Pathway queries
  [Shlomi et al., BMC Bioinformatics 2006]
- Protein docking
  [Mayrose et al., Nucleic Acids Research 2007]
- Balanced paths
  [Cappanera & Scutellà, INOC 2007]
- Automated text headline generation
  [Deshpande et al., NAACL HLT 2007]

Increasing the number of colors

Idea

Use $k + x$ colors instead of k colors.

Trial runtime:

$$O(2^k|G|) \to O(2^{k+x}|G|)$$

Increasing the number of colors

Idea

Use $k + x$ colors instead of k colors.

Trial runtime:

$$O(2^k|G|) \to O(2^{k+x}|G|)$$

Probability P_c for colorful path ($k = 8$, $\varepsilon = 0.001$):

x	0	1	2	3	4	5
P_c	0.0024	0.0084	0.0181	0.0310	0.0464	0.0636
trials	2871	816	378	220	146	106

Increasing the number of colors

Idea

Use $k + x$ colors instead of k colors.

Trial runtime:

$$O(2^k|G|) \to O(2^{k+x}|G|)$$

Probability P_c for colorful path ($k = 8$, $\varepsilon = 0.001$):

x	0	1	2	3	4	5
P_c	0.0024	0.0084	0.0181	0.0310	0.0464	0.0636
trials	2871	816	378	220	146	106

Theorem

MINIMUM-WEIGHT PATH *can be solved in* $O(-\ln \varepsilon \cdot 4.32^k|G|)$ *time by choosing* $x = 0.3k$.

Increasing the number of colors

Idea

Use $k + x$ colors instead of k colors.

Trial runtime:

$$O(2^k|G|) \to O(2^{k+x}|G|)$$

Probability P_c for colorful path ($k = 8$, $\varepsilon = 0.001$):

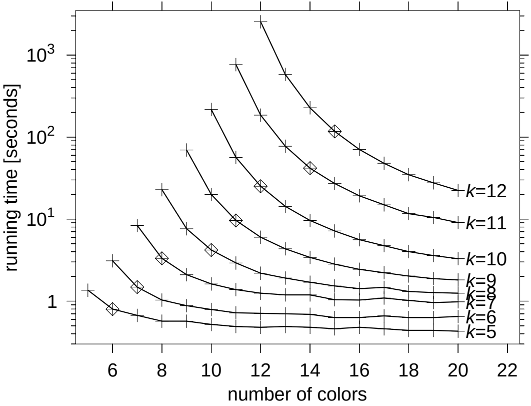
x	0	1	2	3	4	5
P_c	0.0024	0.0084	0.0181	0.0310	0.0464	0.0636
trials	2871	816	378	220	146	106

Theorem

Minimum-Weight Path *can be solved in* $O(-\ln \varepsilon \cdot 4.32^k|G|)$ *time by choosing* $x = 0.3k$.

But: Higher memory usage

Increasing the number of colors



Runtimes for the yeast protein interaction network (highlighted point of each curve marks worst-case optimum)

Exploiting lower bounds

Idea

Use a known solution to prune “hopeless” table entries.

- Discard entries that already have a weight higher than the known solution.

Exploiting lower bounds

Idea

Use a known solution to prune "hopeless" table entries.

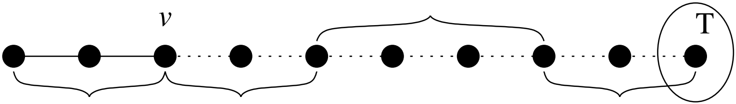
- Discard entries that already have a weight higher than the known solution.
- Discard entries when

$$\text{weight} + (\text{minimum edge weight} \cdot \text{edges left})$$

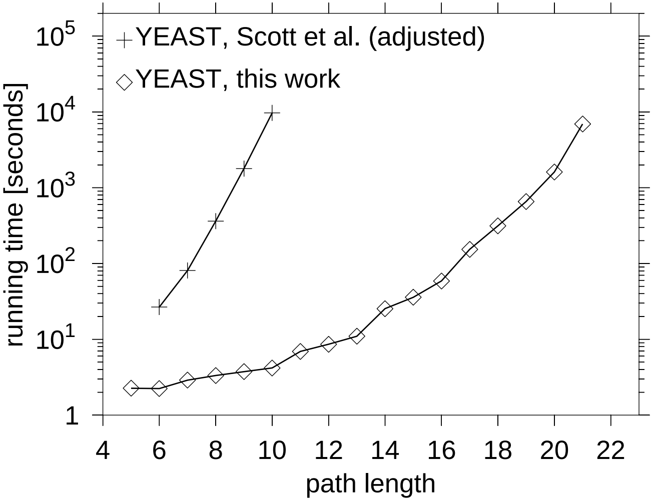
is higher than the weight of the known solution.

Precalculated lower bounds

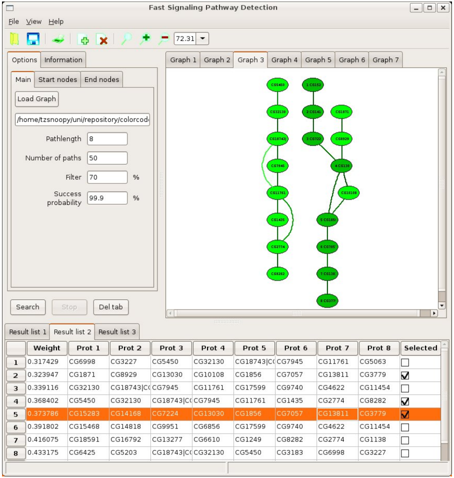
For each vertex u and a range of lengths $1 \leq i \leq d$, determine the minimum weight of a path of i edges that starts at u.



Yeast network



Graphical user interface: FASPAD



Free software, available at
http://theinf1.informatik.uni-jena.de/faspad/

Conclusion & Outlook

Color-coding, with some algorithm engineering, is a practical method for finding signaling pathways in protein interaction networks.

Conclusion & Outlook

Color-coding, with some algorithm engineering, is a practical method for finding signaling pathways in protein interaction networks.

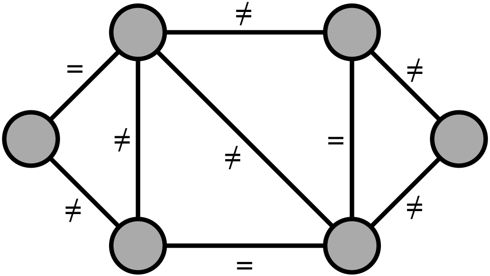
Future work:

- Pathway queries
- Richer motifs (cycles, trees, . . .)
- "Divide-and-color" [Kneis et al., WG 2007; Chen et al., SODA 2007]: Improvement from 4.32^k to 4^k. But: "$\Omega(4^k)$"

Balanced graphs

Definition

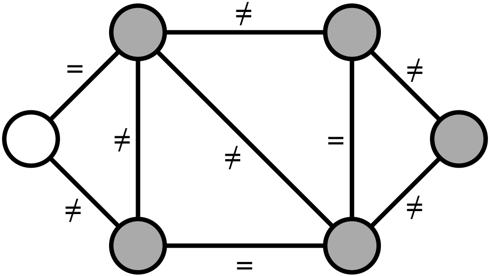
A graph with edges labeled by $=$ or $\neq$ (signed graph) is balanced if the vertices can be colored with two colors such that the relation on each edge holds.



Balanced graphs

Definition

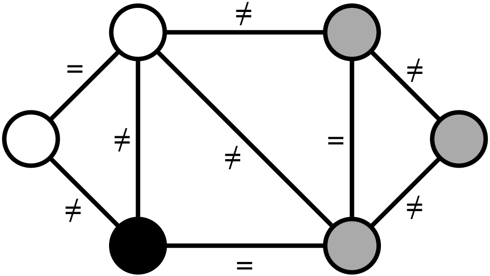
A graph with edges labeled by $=$ or $\neq$ (signed graph) is balanced if the vertices can be colored with two colors such that the relation on each edge holds.



Balanced graphs

Definition

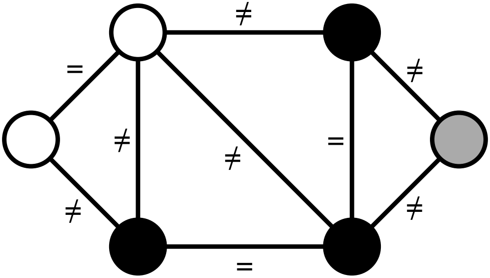
A graph with edges labeled by $=$ or $\neq$ (signed graph) is balanced if the vertices can be colored with two colors such that the relation on each edge holds.



Balanced graphs

Definition

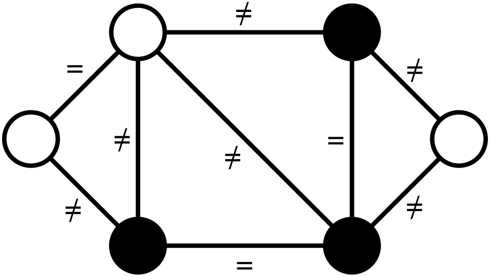
A graph with edges labeled by $=$ or $\neq$ (signed graph) is balanced if the vertices can be colored with two colors such that the relation on each edge holds.



Balanced graphs

Definition

A graph with edges labeled by $=$ or $\neq$ (signed graph) is balanced if the vertices can be colored with two colors such that the relation on each edge holds.



Characterization of balance

Special case

Bipartite graphs are balanced graphs that contain only $\neq$-edges.

Characterization of balance

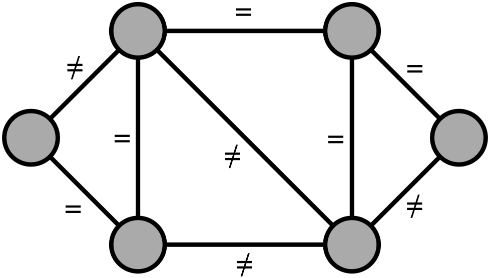
Special case

Bipartite graphs are balanced graphs that contain only $\neq$-edges.

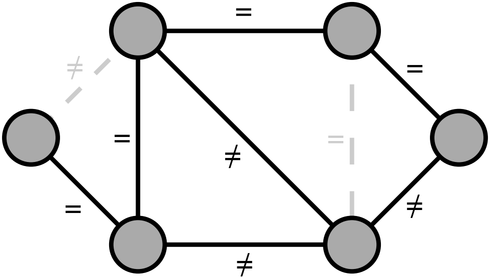
Theorem (Kőnig 1936)

A signed graph is balanced iff it contains no cycle with an odd number of $\neq$-edges.

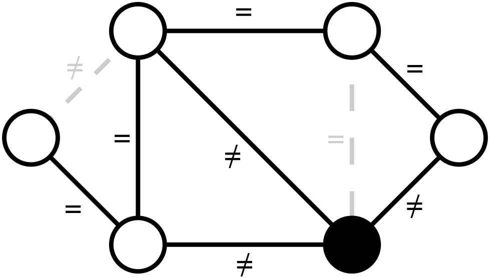
Balanced Subgraph



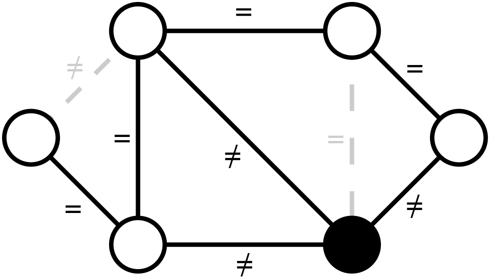
Balanced Subgraph



Balanced Subgraph



Balanced Subgraph



Definition (Balanced Subgraph)

Input: A graph with edges labeled by $=$ or $\neq$.
Task: Find a minimum set of edges to delete such that the graph becomes balanced.

Balanced Subgraph: known results

- Balanced Subgraph is NP-hard, since it is a generalization of Max-Cut (Max-Cut is the special case where all edges are $\neq$)
- A solution that keeps at least 87.8 % of the edges can be found in polynomial time [DasGupta et al., WEA 2006]
- A solution that deletes at most c times the edges that need to be deleted can probably not be found in polynomial time [Khot, STOC 2002]

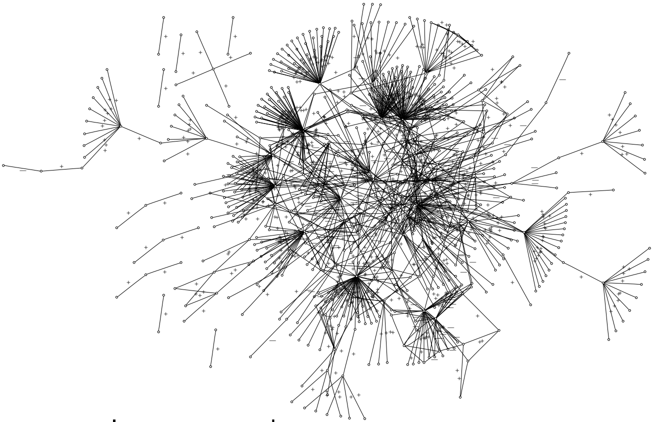
Applications of Balanced Subgraph

- "Monotone subsystems" in gene regulatory networks
  [DasGupta et al., WEA 2006]
- Balance in social networks
  [Harary, Mich. Math. J. 1953]
- Portfolio risk analysis
  [Harary et al., IMA J. Manag. Math. 2002]
- Minimum energy state of magnetic materials (spin glasses)
  [Kasteleyn, J. Math. Phys. 1963]
- Stability of fullerenes
  [Došlić & Vikičević, Discr. Appl. Math. 2007]
- Integrated circuit design
  [Chiang et al., IEEE Trans. CAD of IC & Sys. 2007]

Graph structure

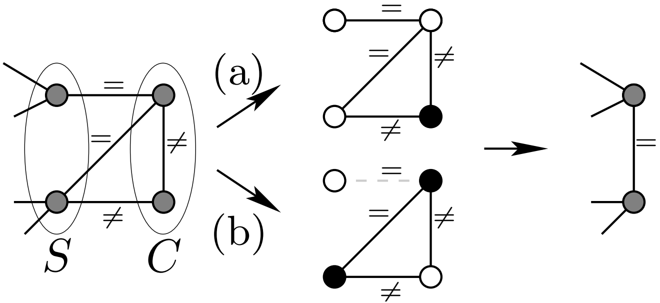
Idea

Exploit the structure of the relevant networks



Yeast gene regulatory network

Vertex cut-based data reduction



Data reduction scheme

Data reduction scheme

- Find cut S that cuts off small component C
- For each of the (up to symmetry) $2^{|S|-1}$ colorings of S, determine the size of an optimal solution for $G[S \cup C]$
- Replace in G the subgraph $G[S \cup C]$ by an equivalent smaller gadget

Data reduction scheme

Data reduction scheme

- Find cut S that cuts off small component C
- For each of the (up to symmetry) $2^{|S|-1}$ colorings of S, determine the size of an optimal solution for $G[S \cup C]$
- Replace in G the subgraph $G[S \cup C]$ by an equivalent smaller gadget

Subsumes all 8 data reduction rules given by [Wernicke, 2003] for Edge Bipartization

Filling in the data reduction scheme

- Need to restrict both $|S|$ and $|C|$: we use $|S| \leq 4$ and $|C| \leq 32$

Filling in the data reduction scheme

- Need to restrict both $|S|$ and $|C|$: we use $|S| \leq 4$ and $|C| \leq 32$
- How to construct gadgets that behave equivalently to $S \cup C$?

Gadget construction

Idea

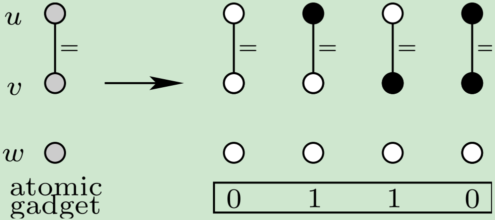
Use atomic gadgets and describe their effect by cost vectors.

Gadget construction

Idea

Use atomic gadgets and describe their effect by cost vectors.

Example



Gadget construction

Theorem

With 10 atomic gadgets, we can emulate the behavior of any component behind a 3-vertex cut.

Gadget construction

Theorem

With 10 atomic gadgets, we can emulate the behavior of any component behind a 3-vertex cut.

Theorem

All cuts with $|S| = 2$ *and* $|C| \geq 1$ *and and all cuts with* $|S| = 3$ *and* $|C| \geq 2$ *are subject to data reduction.*

Gadget construction

Theorem

With 10 atomic gadgets, we can emulate the behavior of any component behind a 3-vertex cut.

Theorem

All cuts with $|S| = 2$ and $|C| \geq 1$ and and all cuts with $|S| = 3$ and $|C| \geq 2$ are subject to data reduction.

- 4-cuts: 2948 atomic gadgets

Gadget construction

Problem

How to determine an appropriate set of atomic cost vectors for a given cost vector?

Gadget construction

Problem

How to determine an appropriate set of atomic cost vectors for a given cost vector?

Vector Sum Problem

Given a set S of n vectors of length l with nonnegative integer components and a target vector t of length l, find a sub-(multi)-set of vectors from S that sums to t.

Gadget construction

Problem

How to determine an appropriate set of atomic cost vectors for a given cost vector?

Vector Sum Problem

Given a set S of n vectors of length l with nonnegative integer components and a target vector t of length l, find a sub-(multi)-set of vectors from S that sums to t.

- "Equality-constrained multidimensional knapsack"

Gadget construction

Problem

How to determine an appropriate set of atomic cost vectors for a given cost vector?

Vector Sum Problem

Given a set S of n vectors of length l with nonnegative integer components and a target vector t of length l, find a sub-(multi)-set of vectors from S that sums to t.

- "Equality-constrained multidimensional knapsack"
- In our implementation: simple branch & bound

Gadget construction

Problem

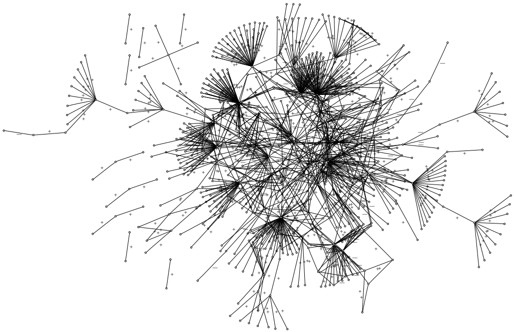
How to determine an appropriate set of atomic cost vectors for a given cost vector?

Vector Sum Problem

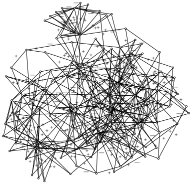
Given a set S of n vectors of length l with nonnegative integer components and a target vector t of length l, find a sub-(multi)-set of vectors from S that sums to t.

- “Equality-constrained multidimensional knapsack”
- In our implementation: simple branch & bound
- Sometimes this is a bottleneck!

Reduction. . . and then?

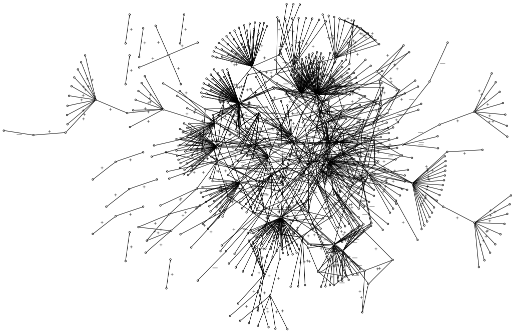


$n = 690,\ m = 1082$

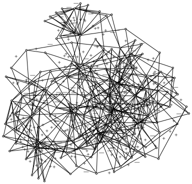


$n = 144,\ m = 405$

Reduction… and then?



$n = 690,\ m = 1082$



$n = 144,\ m = 405$

After data reduction, a hard "core" remains.

Fixed-parameter tractability

Theorem

BALANCED SUBGRAPH *can be solved in* $O(2^k \cdot m^2)$ *time by a reduction to* EDGE BIPARTIZATION *and using an algorithm based on iterative compression* [GUO et al., JCSS 2006].

Fixed-parameter tractability

Theorem

BALANCED SUBGRAPH can be solved in $O(2^k \cdot m^2)$ time by a reduction to EDGE BIPARTIZATION and using an algorithm based on iterative compression [Guo et al., JCSS 2006].

A heuristic speedup trick can give large speedups over this worst-case running time.

Experimental results

			Approximation			Exact alg.	
Data set	n	m	$k \geq$	$k \leq$	t [min]	k	t [min]
EGFR	330	855	196	219	7	210	108
Yeast	690	1082	0	43	77	41	1
Macr.	678	1582	218	383	44	374	1

Experimental results

			Approximation			Exact alg.	
Data set	n	m	$k \geq$	$k \leq$	t [min]	k	t [min]
EGFR	330	855	196	219	7	210	108
Yeast	690	1082	0	43	77	41	1
Macr.	678	1582	218	383	44	374	1

- Yeast is not solvable without reducing 4-cuts
- A real-world network with 688 vertices and 2208 edges could not be solved

Outlook

- Kernel for Balanced Subgraph?
- Directed case of Balanced Subgraph (delete minimum number of edges to remove all unbalanced cycles): FPT?
    - Problem: Characterization by two-coloring does not work
- The data reduction scheme is applicable to all graph problems where a coloring or a subset of the vertices is sought. For example:
    - Vertex Cover
    - Dominating Set
    - 3-Coloring
    - Feedback Vertex Set

  but: need small cuts (e. g., small-world networks)

References

- Falk Hüffner, Sebastian Wernicke, and Thomas Zichner:
  Algorithm engineering for color-coding with applications to signaling pathway detection.
  Algorithmica.
  Accepted for publication, June 2007.
- Falk Hüffner, Nadja Betzler, and Rolf Niedermeier:
  Optimal edge deletions for signed graph balancing.
  Proc. 6th Workshop on Experimental Algorithms (WEA '07), 2007.
  LNCS 4525, pp. 297–310.
- Falk Hüffner:
  Algorithms and Experiments for Parameterized Approaches to Hard Graph Problems.
  PhD thesis, Universität Jena, to appear.