KAMAK 2016

Rejštejn

September 18.–23.

Charles University, Prague

Organizers:

Pavel Dvořák
Dušan Knop
Robert Šámal

Brochure of open problems,
Prague, 2016.

Contents

Program

8:30 breakfast

9:30 morning session I

10:30 break

11:00 morning session II

12:30 lunch

15:00 afternoon session I

16:30 break

17:00 afternoon session II

18:30 progress reports

19:00 dinner

Wednesday is planned to be free to make an excursion in the neighbourhood with everyone who would like to come.

OPEN PROBLEMS

Problem 1. *Laplacian spectra consisting of distinct integers (suggested by Jan Bok)*

Source: Proposed by Fallat et al. [1].

Conjecture: There is no simple graph on n vertices whose Laplacian spectrum is given by $\{0, 1, \ldots, n-1\}$.

Related results:
• Conjecture holds for $n \geq 6{,}649{,}688{,}933$ (see [3]) and for $n \leq 11$ (see [1]).
• (from [2] and [4]) If we suppose that G is a (hypothetical?) graph on n vertices with Laplacian spectrum $\{0, 1, \ldots, n-1\}$, then we can say following:

- $n \equiv 0$ or $1 \pmod 4$,
- $(n-1)!$ is divisible by n,
- $n \geq 12$,
- G has $\frac{n(n-1)}{4}$,
- if G has degree sequence $d_1, \ldots, d_n$, then $2 \leq d_i \leq n-3$ for all i, and $\sum d_i^2 = n(n-1)(n-2)/3$,
- G and its complement G^C must have diameter 3.

References:

[1] Fallat, Shaun M., et al. *On graphs whose Laplacian matrices have distinct integer eigenvalues.* Journal of Graph Theory 50.2 (2005): 162-174.
[2] Stevanović, Dragan. *Research problems from the Aveiro workshop on graph spectra.* Linear Algebra and its Applications 423.1 (2007): 172-181.
[3] Goldberger, Assaf, and Michael Neumann. *On a Conjecture on a Laplacian Matrix with Distinct Integral Spectrum.* Journal of Graph Theory 72.2 (2013): 178-208.
[4] Das, Kinkar Ch, Sang-Gu Lee, and Gi-Sang Cheon. *On the conjecture for certain Laplacian integral spectrum of graphs.* Journal of Graph Theory 63.2 (2010): 106-113.

Problem 2. *Crushing Disks and Playing TRON (suggested by Martin Böhm)*

Source: Presented at IWOCA 2016.

Crushing Disks:

On input, we receive a set of n points (centers) in $\mathbb{R}^2$, each center i with associated radius r_i and priority p_i.

Imagine that we run a simulation where the disks grow linearly in time; namely, in time t, there is a disk with center i and radius $t \cdot r_i$. When two disks touch, the one with the lower priority is eliminated and the more important one continues to grow. This induces an order on the disks.

The task is to output the elimination order of the centers. The problem originates in map rendering – as you zoom out on a digital map, the map removes the labels which are less important.

A naive algorithm is able to compute the order in time $O(n^2 \log n)$. At IWOCA 2016, Funke, Krumpe and Storandt [1] showed how to do this order in expected time $O(n(\log^6 n + \Delta^2 \log^2 n + \Delta^4 \log n))$ where Δ is the ratio of the largest and smallest radius of the instance.

The open problem is to improve on this running time. One of the authors stated that they would welcome a running time that is close to $n \cdot \text{polylog}(n)$ without the Δ factor.

Playing TRON:

On input, we get information about a set of (light-)motorcycles, each motorcycle i starting at a point p_i in $\mathbb{R}^2$ riding in direction d_i with speed s_i. For simplicity, the motorcycles do not change direction or speed.

Each motorcycle leaves a barrier in its track. Whenever a motorcycle intersects a track of another motorcycle, it crashes into it and is eliminated. In this problem, several motorcycles can continue without ever crashing.

The task is to compute the elimination order of the motorcycles.

A simple algorithm can be designed in time $O(n^2 \log n)$, the currently known best algorithm is due to [2] with running time $O(n^{17/11+\varepsilon})$. The open problem is to again improve the running time.

References:

[1] Funke, S., Krumpe, F., Storandt, S.: Crushing Disks Efficiently. 27th International Workshop, IWOCA 2016, Helsinki, Finland, pp 43-54, LNCS, Springer (2016).

[2] Eppstein, D., Erickson, J.: Raising roofs, crashing cycles, and playing pool: appli- cations of a data structure for finding pairwise interactions. Discrete Comput. Geometry 22(4), 569–592 (1999).

Problem 3. *A zero-player graph game (suggested by Pavel Dvořák)*

Source: Proposed by Dohrau et al. [1].

Definitions.

• *A* switch graph *is a directed graph G in which every vertex has at most two outgoing edges. Formally, a switch graph is a* 4*-tuple* $G = (V, E, s_0, s_1)$ *where* $s_0, s_1 : V \to V$ *and* $E = \{(v, s_0(v)), (v, s_1(v)) : v \in V\}$*. Loops (i.e.,* $s_i(v) = v$*) and outdegree 1 (i.e.,* $s_1(v) = s_2(v)$*) are allowed.*

• *We want to analyze the following process on a switch graph G and vertices* $s, t \in V(G)$*.*

> **procedure** RUN (G, s, t)
>   for all $u \in V$: $\text{curr}[u] = s_0(u)$, $\text{next}[u] = s_1(u)$
> $v = s$
>   **while** $v \neq t$ **do**
> $w = \text{curr}[v]$
> $\text{swap}(\text{curr}[v], \text{next}[v])$
> $v = w$
>   **end while**
> **end procedure**

Informally, we analyze the process when we start at the vertex s and in each round we use an edge $(v, s_0(v))$ or $(v, s_1(v))$, if the number of visits of the vertex v is even or odd, respectively. The process ends when we get to the vertex t.

• *A problem* ARRIVAL *is to decide if the procedure* RUN(G, s, t) *terminates for a given switch graph* $G = (V, E, s_0, s_1)$ *and vertices* $s, t \in V$*.*

Question: Is the problem ARRIVAL in P?

Related results:

• Dohrau et al. [1] proved the problem ARRIVAL is in $\mathsf{NP} \cap co\mathsf{NP}$.

References:

[1] J. Dohrau, B. Gärtner, M. Kohler, J. Matoušek and E. Welzl. *A zero-player graph game in* $\mathsf{NP} \cap co\mathsf{NP}$. https://arxiv.org/abs/1605.03546.

Problem 4. *Fixed-Parameter Approximation of Planar p-Centre (suggested by Andreas E. Feldmann)*

This problem is about approximating the p-Centre problem in planar graphs within a factor better than 2 when allowing only a mild exponential running time, i.e. the running time should only be exponential in the number p of centres.

Definitions.
• *Let I be an instance of a minimization problem $\mathcal{P}$ with optimum solution OPT. An* α-approximation *of I is a solution S to I that is not worse than α times the optimum solution:* $\text{cost}(S) \leq \alpha \cdot \text{cost}(OPT)$.
• *Let $\mathcal{A}$ be an algorithm for a problem $\mathcal{P}$ that takes an instance $I \in \mathcal{P}$ and a parameter p of I as input. We say that $\mathcal{A}$ is a* fixed-parameter algorithm for p *if there exists a function $f(p)$ independent of the input size n, and a polynomial* $\text{poly}(n)$*, such that $\mathcal{A}$ halts after at most $f(p) \cdot \text{poly}(n)$ steps on any instance of $\mathcal{P}$ of size n with parameter p.*
• *A* fixed-parameter α-approximation (α-FPA) *algorithm $\mathcal{A}$ for a minimization problem $\mathcal{P}$ is a fixed-parameter algorithm for p that computes an α-approximation on any instance of $\mathcal{P}$ of size n with parameter p.*

• *Given a graph $G = (V, E)$, let* $\text{dist}(u, v)$ *denote the shortest-path distance between two vertices $u, v \in V$ in G.*
• *An instance to the* p-Centre problem *consists of an undirected graph $G = (V, E)$ and an integer p. A solution is a set $C \subseteq V$ of at most p* centres, *i.e. $|C| \leq p$. The cost of a centre set C is the maximum distance of any vertex of V to its closest centre:* $\text{cost}(C) = \max_{v \in V} \min_{u \in C} \text{dist}(u, v)$*. An optimum solution is a centre set that minimizes the cost among all centre sets of size at most p.*
• *A* planar graph *is a graph that can be drawn on the two-dimensional plane such that no two edges cross, i.e. the only points shared by any pair of edges may be their endpoints (their incident vertices).*

Question: Is there a factor $\alpha < 2$ for which the p-Centre problem on planar graphs has an α-FPA algorithm for parameter p, i.e. the number of centres?

Related results:
• It is not hard to obtain a 2-approximation for any input graph of p-Centre in polynomial time. (I can explain this result to you in 5 minutes, but see also [1] or [2].)

• Even for planar graphs there is no factor $\alpha < 2$ for which an α-approximation can be computed in polynomial time for the p-Centre problem (unless P=NP). (See [3].)
• On general graphs there is no factor $\alpha < 2$ for which the p-Centre problem has an α-FPA algorithm for parameter p (unless P=W[2]). (This is also easy to show; see [4].)
• For edge-weighted planar graphs there is no so-called *FPT algorithm*, which is a fixed-parameter algorithm for p computing the optimum solution (unless P=W[1]).
• For other special graph classes α-FPA algorithms are known for $\alpha < 2$. For example for graphs with *highway dimension* h it is possible to obtain $\alpha = 1.5$ (here the parameter is the combination of p and h), and for graphs with *doubling dimension* d it is even possible to obtain $\alpha = 1+\varepsilon$ for any $\varepsilon > 0$ (here the parameter is the combination of p, d, and ε). (For the former result see [4], the latter result is easy and can be explained in about 10 minutes.)

References:

1. V. V. Vazirani. *Approximation Algorithms.* Springer-Verlag New York, Inc., 2001.
2. D. S. Hochbaum and D. B. Shmoys. A unified approach to approximation algorithms for bottleneck problems. *JACM,* 33(3):533–550, 1986.
3. J. Plesník. On the computational complexity of centers locating in a graph. *Aplikace matematiky,* 25(6): 445–452, 1980.
4. A. E. Feldmann. Fixed-Parameter Approximations for k-Center Problems in Low Highway Dimension Graphs. *ICALP,* pages 588–600, 2015.

Problem 5. *Fixed-Parameter Approximation of Directed Steiner Networks (suggested by Andreas E. Feldmann)*

This problem is about approximating the Directed Steiner Network problem within a factor better than 2 when allowing only a mild exponential running time, i.e. the running time should only be exponential in the number of demands between terminals.

Definitions.

• *Let I be an instance of a minimization problem $\mathcal{P}$ with optimum solution OPT. An* α-approximation *of I is a solution S to I that is not worse than α times the optimum solution:* $\text{cost}(S) \leq \alpha \cdot \text{cost}(OPT)$.

• *Let $\mathcal{A}$ be an algorithm for a problem $\mathcal{P}$ that takes an instance $I \in \mathcal{P}$ and a parameter p of I as input. We say that $\mathcal{A}$ is a* fixed-parameter algorithm for p *if there exists a function $f(p)$ independent of the input size n, and a polynomial* $\text{poly}(n)$*, such that $\mathcal{A}$ halts after at most $f(p) \cdot \text{poly}(n)$ steps on any instance of $\mathcal{P}$ of size n with parameter p.*

• *A* fixed-parameter α-approximation (α-FPA) *algorithm $\mathcal{A}$ for a minimization problem $\mathcal{P}$ is a fixed-parameter algorithm for p that computes an α-approximation on any instance of $\mathcal{P}$ of size n with parameter p.*

• *Let a directed graph $D = (V, A)$ with arc set A and a set of p* demands *(or* terminal pairs*)* $\{(s_1, t_1), \ldots, (s_p, t_p)\} \subseteq V^2$ *be given. A* directed Steiner network *N is a subgraph of G that contains a directed path from s_i to t_i for every $i \in \{1, \ldots, p\}$. That is, the network N connects every terminal pair, and to do so it can contain some subset of the remaining* Steiner vertices $V \setminus \{s_i, t_i \mid 1 \leq i \leq p\}$ *(i.e. non-terminals).*

• *An instance to the* Directed Steiner Network problem *consists of a directed graph $D = (V, A)$ and a set of p terminal pairs. A solution is a directed Steiner network N, and its cost is its number of edges $|E(N)|$. An optimum solution is a Steiner network N that minimizes the cost among all solutions.*

Question: What is the smallest factor α for which the Directed Steiner Network problem has an α-FPA algorithm for parameter p, i.e. the number of demands? Even finding such algorithms for special cases is interesting.

Related results:

• The special case when the demands are of the form $\{(r, t_1), \ldots, (r, t_p)\}$ for a specified *root terminal* r is known as the *Directed Steiner Tree* problem, since the optimum solution will form a directed tree with root r (i.e. an *arborescence*). For this problem there is a so-called *FPT algorithm* for p,

i.e. a fixed-parameter algorithm for p computing the optimum solution in time $2^p \cdot \text{poly}(n)$. (See [1,2].)
• Even for the special case when the demands are of the form $\{(r, t_1), (t_1, r) \ldots, (r, t_p), (t_p, r)\}$ for a specified *root terminal* r (the so-called *Strongly Connected Steiner Subgraph* problem), there is no FPT algorithm for p, i.e. the optimum solution cannot be computed in time $f(p) \cdot \text{poly}(n)$ for any function $f(p)$ independent of n (unless P=W[1]). (See [3].)
• For the special case when the optimum solution is strongly connected it is not hard to obtain a 2-FPA by computing two optimal directed Steiner trees. (This can be explained in 5 minutes, but see also [4].)
References:

1. S. E. Dreyfus and R. A. Wagner. The Steiner problem in graphs. *Networks*, 1(3):195–207, 1971.

2. A. Björklund, T. Husfeldt, P. Kaski, and M. Koivisto. Fourier meets Möbius: fast subset convolution. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing*, pages 67–74, 2007.

3. J. Guo, R. Niedermeier, and O. Suchý. Parameterized complexity of arc-weighted directed Steiner problems. *SIAM J. Discrete Math.*, 25(2):583–599, 2011.

4. R. H. Chitnis, M. Hajiaghayi, and D. Marx. Tight bounds for planar strongly connected Steiner subgraph with fixed number of terminals (and extensions). In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1782–1801, 2014.

Problem 6. *Fixed-Parameter Approximation of Steiner Trees (suggested by Andreas E. Feldmann)*

Source: Proposed by Michael Lampis in 2014.

This problem is about approximating the Steiner Tree problem within a factor better than 1.39 when allowing only a mild exponential running time, i.e. the running time should only be exponential in the number of Steiner vertices of the optimum solution.

Definitions.
• *Let I be an instance of a minimization problem $\mathcal{P}$ with optimum solution OPT. An* α-approximation *of I is a solution S to I that is not worse than α times the optimum solution:* $\text{cost}(S) \leq \alpha \cdot \text{cost}(OPT)$.
• *Let $\mathcal{A}$ be an algorithm for a problem $\mathcal{P}$ that takes an instance $I \in \mathcal{P}$ and a parameter p of I as input. We say that $\mathcal{A}$ is a* fixed-parameter algorithm for p *if there exists a function $f(p)$ independent of the input size n, and a polynomial* $\text{poly}(n)$*, such that $\mathcal{A}$ halts after at most $f(p) \cdot \text{poly}(n)$ steps on any instance of $\mathcal{P}$ of size n with parameter p.*
• *A* fixed-parameter α-approximation (α-FPA) *algorithm $\mathcal{A}$ for a minimization problem $\mathcal{P}$ is a fixed-parameter algorithm for p that computes an α-approximation on any instance of $\mathcal{P}$ of size n with parameter p.*

• *Let an undirected graph $G = (V, E)$ and a set $R \subseteq V$ of* terminals *be given. A* Steiner tree *T is a connected acyclic subgraph of G containing R. That is, the tree T connects all terminals, and to do so it can contain some subset of the remaining* Steiner vertices *$V \setminus R$ (i.e. non-terminals).*
• *An instance to the* Steiner Tree problem *consists of an undirected graph $G = (V, E)$ and a set of* terminals*. A solution is a Steiner tree T, and its cost is its number $|E(T)|$ of edges. An optimum solution is a Steiner tree that minimizes the cost among all solutions.*
• *Let $p = |V(T) \setminus R|$ be the number of Steiner vertices contained in the optimum Steiner tree T.*

Question: Is there a factor $\alpha < 1.39$ for which the Steiner Tree problem has an α-FPA algorithm for parameter p, i.e. the number of needed Steiner vertices?

Related results:
• It is possible to obtain a 1.39-approximation for any instance of Steiner Tree in polynomial time. (See [1].)
• There is a (small) constant $\alpha > 1$ for which no polynomial time algorithm can compute an α-approximation for any instance of Steiner Tree (unless P=NP). (See [2].)

• For the Steiner Tree problem there is no so-called *FPT algorithm* for p, which is a fixed-parameter algorithm for parameter p computing the optimum solution (unless P=W[2]). (This can be explained in about 5 minutes.)

References:

1. J. Byrka, F. Grandoni, T. Rothvoß, and L. Sanità. Steiner tree approximation via iterative randomized rounding. *J. ACM*, 60(1):6, 2013.

2. M. Chlebík and J. Chlebíková. Approximation hardness of the Steiner tree problem on graphs. In *Proceedings, Scandinavian Workshop on Algorithm Theory*, pages 170–179, 2002.

Problem 7. *Complexity of Computing the Spectral Radius of an Interval Matrix (suggested by Milan Hladík)*

Source: Known to be an open problem for decades.

Definitions.
• *The spectral radius* $\rho(A)$ *of a square matrix is the largest absolute value of its eigenvalues.*
• *An interval matrix is defined as* $\mathbf{A} = [\underline{A}, \overline{A}] = \{A \mid \underline{a}_{ij} \leq a_{ij} \leq \overline{a}_{ij} \ \forall i, j\}$, *where* $\underline{A} \leq \overline{A}$ *are given.*
• *The spectral radius* $\rho(\mathbf{A})$ *of an interval matrix* $\mathbf{A}$ *is the interval* $[\underline{\rho}(\mathbf{A}), \overline{\rho}(\mathbf{A})]$, *where*

$$\underline{\rho}(\mathbf{A}) = \min_{A \in \mathbf{A}} \rho(A),$$
$$\overline{\rho}(\mathbf{A}) = \max_{A \in \mathbf{A}} \rho(A).$$

Question: What is the complexity of computing $\overline{\rho}(\mathbf{A})$?

Related results:
• It is posted as an open problem for decades, but probably not studied heavily, so there is a chance to do it.
• To the best of my knowledge the complexity of computing $\underline{\rho}(\mathbf{A})$ has not been investigated at all.
• I believe it is NP-hard as many related results are NP-hard (checking Hurwitz stability, regularity etc. of an interval matrix, as well as the spectral radius of an interval symmetric matrix).

Why is it interesting:
• For example: If $\mathbf{A}$ represents an uncertain matrix of a dynamical system, then $\overline{\rho}(\mathbf{A}) < 1$ implies that all matrices in $\mathbf{A}$ are Schur stable, so the system is stable whatever is the true matrix in $\mathbf{A}$.

References:

[1] A. Nemirovskii. Several NP-hard problems arising in robust stability analysis. *Math. Control Signals Syst.*, 6(2):99–105, 1993.

[2] J. Rohn. Checking Properties of Interval Matrices. Technical Report No. 686, Institute of Computer Science, Academy of Sciences of the Czech Republic, Prague 1996, `http://hdl.handle.net/11104/0123221`

Problem 8. *Extending matchings in Q_n for $n \geq 2$ to 2-factors (suggested by Tereza Hulcová)*

Source: Suggested by Jennifer Vandenbussche and Douglas B. West [1].

Definition. *An f-factor of a graph G for a function $f\colon V(G) \to N$ is a spanning subgraph H of G such that $deg_H(v) = f(v)$ for every $v \in V(G)$.*

Theorem 1. *(Ore-Ryser) A bipartite graph $G = (A \cup B, E)$ has an f-factor if and only if $f(A) = f(B)$ and for every $S \subseteq A$*

$$f(S) \leq \sum_{b \in N(S)} \min(f(b), |N_S(b)|).$$

The question is if every matching of the hypercube with dimension $n \geq 2$ can be extended to 2-factor. This is true for $n = 1, 2, 3, 4$. Vandenbussche and Douglas proved this for $n = 5$ applying the Ore-Ryser theorem.

References:

[1] Jennifer Vandenbussche, Douglas B. West: Extensions to 2-factors in bipartite graphs, 2013.
[2] Petr Gregor: Lecture notes for the subject Hypercube Problems.

Problem 9. *Majority coloring of digraphs (suggested by Tereza Klimošová)*

Source: Stephan Kreutzer, Sang-il Oum, Paul Seymour, Dominic van der Zypen, David R. Wood [1].

Definitions. *A* majority coloring *of a digraph is a function that assigns each vertex v a color, such that at most half of the out-neighbors of v receive the same color as v. In other words, majority of the out-neighbors of v receive a color different from v.*

A concept of majority coloring was recently introduced by Dominic van der Zypen, who posed as an open problem whether any digraph has a majority coloring by a bounded number of colors.

Very recently a very short proof (5 lines) of the following theorem appeared in [1].

Theorem 2. *Every digraph has a majority* 4*-coloring.*

It leads to the following conjecture.

Conjecture 3. *Every digraph has a majority* 3*-coloring.*

This would be best possible, since an odd directed cycles do not have a majority 2-coloring. (And there exist other, more complex examples of graphs requiring at least 3 colors.)

In [1], some evidence for Conjecture 3 is given. In particular, it is shown that a large minimum out-degree (greater than $72\ln(3v(G))$) implies the existence of a majority 3-coloring. The authors also propose a number of open problems related to Conjecture 3.

- Does every tournament have a majority 3-coloring?
- Does every Eulerian digraph have a majority 3-coloring?
- Provide a characterisation of digraphs that have a majority 2-coloring (or a polynomial time algorithm for their recognition).
- Consider a fractional version of Conjecture 3. Perhaps 3 can be replaced by something smaller.
- Generalization of Conjecture 3: For $k \geq 2$, every digraph has a $(k+1)$-coloring such that for each vertex v, at most $1/k$ of its out-neighbors receive the same color as v.

See [1] for more problems and background.

References:

[1] Kreutzer, Stephan, Sang-il Oum, Paul Seymour, Dominic van der Zypen, and David R. Wood. "Majority Colourings of Digraphs." arXiv preprint arXiv:1608.03040 (2016).

Problem 10. *The P_3-game on paths (suggested by Dušan Knop)*

Source: Proposed by Hon, Kloks, Liu, Liu, and Wang [1].

Definitions.
• *Let $G = (V, E)$ be a graph. A set $S \subseteq V$ is* P_3-closed *if*

$$\forall v \in V \qquad v \notin S \Rightarrow |N(v) \cap S| < 2.$$

• *Two players play the P_3-game on a graph by alternately selecting vertices. At the start of the game all vertices are unlabeled. During the game players label vertices. Prior to every move, the set of labeled vertices is P_3-closed. A move consists of labeling a, previously unlabeled, vertex v.*

Examples:
• When the graph is a clique with at least two vertices, then the second player wins the game.
• When the graph is a star $K_{1,l}$, then player one has a winning strategy if and only if l is even.

Related results:
• There exists an $O(n^2)$ algorithm that decides the P_3-game on paths.
• There exists a polynomial-time algorithm to decide the P_3-game on cycles.

Question: Characterise the number of vertices n for which player one wins the P_3-game on path P_n. What about cycle C_n?

References:
[1] Hon, W.-K., Kloks, T., Liu, F.-H., Liu, H.-H., & Wang, T.-M. 2016, arXiv:1608.05169

Problem 11. *Conductance of a Permutation (suggested by Karel Král)*

Source: Proposed by Yevgeniy Dodis in 2016.

Definitions.

• *Conductance of a permutation* $\pi\colon \{0,1\}^{wn} \to \{0,1\}^{wn}$ *at* q *queries is the maximum over all possible pairs of cartesian products* $(U_1 \times \cdots \times U_w, V_1 \times \cdots \times V_w)$*, where* $U_i, V_i \subseteq \{0,1\}^n$ *and* $|U_i| = |V_i| = q$ *for each* $1 \le i \le w$*, of the numbers of pairs* $(x,y) \in \{0,1\}^{wn} \times \{0,1\}^{wn}$ *such that*

$$\pi(x) = y \text{ and } (x,y) \in (U_1 \times \cdots \times U_w, V_1 \times \cdots \times V_w).$$

In other words:

$$cond_\pi(q) = \max_{U_1,\ldots,U_w,V_1,\ldots,V_w,|U_i|=|V_i|=q} |\pi(U_1 \times \cdots \times U_w) \cap [V_1 \times \cdots \times V_w]|.$$

Question: What is the smallest conductance of a linear permutation (that is a permutation that is a linear function)?

Related results:

• Conductance of any permutation $\pi\colon \{0,1\}^{wn} \to \{0,1\}^{wn}$ lies between q and q^w.

• A random permutation has conductance close to wqn.

• Linear permutations satisfy $\mathrm{Cond}_\tau(q) \ge \Omega(q^{2-\frac{1}{2w-1}})$.

References:

Indifferentiability of Confusion-Diffusion Networks
Yevgeniy Dodis, Martijn Stam, John Steinberger, Liu Tianren.
`https://www.cs.nyu.edu/~dodis/ps/cd-networks.pdf`

Problem 12. *Parity Matching problem (suggested by Martin Loebl)*

Source: Proposed by Marcos Kiwi and Martin Loebl in 2015.

Definitions.
• *Parity Matching problem* $P(G, w, p_1, \ldots, p_g, k)$ *is as follows: Given integer* k*, graph* G *with* n *vertices, positive integer weights* $w(e)$ *on edges and* g *disjoint pairs of edges* $p_1, \ldots, p_g$*, find out if there is a feasible perfect matching or total weight at least* k*; perfect matching* M *is feasible if it has an even number of edges from each* $p_i, i = 1, \ldots, g$.

Question: No deterministic or probabilistic algorithm for

$$P(G, w, p_2, \ldots, p_g, k)$$

has complexity less than $poly(n)2^g$.

Related results: I can explain that MaxCut problem for G with crossing number g can be reduced to the Parity Matching problem for $2g$. That can be solved by calculating linear combination of 2^{2g} Pfaffians. After attempts to find more straightforward algorithm, we start to think that no significantly better way exists. The Parity Matching problem pinpoints the complexity of MaxCut for graphs embedded on surfaces.

There is a straightforward reduction of $P(G, w, p_1, \ldots, p_g, k)$ to 2^g weighted perfect matching problems. I find it very surprising (but still we conjecture it) that no better algorithm should exist.

We have some very pleriminary results very weakly supposting the conjecture.

Problem 13. *Maximum Number of Minimal Connected Vertex Covers in Graphs–the lower bound (suggested by Tomáš Masařík)*

Source: This question was proposed by Golovach et all in 2015 on IWOCA conference.

Definitions.
• Minimal connected vertex cover *is a minimal connected set of vertices such that for every edge there is at least one endpoint in it.*

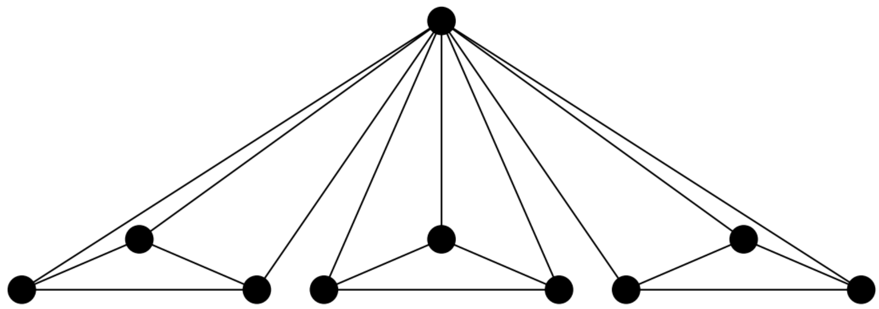

Figure 1: Actual lower bound.

Question: The maximum number of minimal connected vertex covers of an arbitrary graph on n vertices is at most 1.8668^n and also this can be enumerated in the same time. Is this result tight? Can we construct a graph with that many minimal connected vertex covers? The authors are able to construct a graph with $3^{(n-1)/3} \approx 1.4422$ minimal connected vertex covers.

References: Petr A. Golovach, Pinar Heggernes and Dieter Kratsch: Enumeration and Maximum Number of Minimal Connected Vertex Covers in Graphs, Combinatorial Algorithms - 26th International Workshop, IWOCA 2015, Verona, Italy, October 5-7, 2015, Revised Selected Papers, 235–247.

Problem 14. *Decomposibility of 3-edge-connected graphs (suggested by Jan Musílek)*

Source: Proposed by Barát and Thomassen in 2003.

Definitions.
• *Let us have connected graph G with count of edges $m = m_1 + \cdots + m_k$ where $m_1, \ldots, m_k \in \mathbb{Z}^+$. Then by* decomposition G according to $(m_1, \ldots, m_k)$ *we mean the set $G_1, \ldots, G_k$ of connected subgraphs G such that:*

1. $\forall i \in \{1, \ldots, k\} : |E(G_i)| = m_i$
2. *Graphs $G_1, \ldots G_k$ cover the set of edges of graph G.*

Alternatively, we can imagine that each edge of G is coloured by one of k colours in such way, that there are exactly m_i edges coloured by color i and that subgraph G induced by edges of color i is connected.
• *We say about graph G that it is* decomposible *if there exists $\forall k \in \{1, \ldots, m\}$ the decomposition of G according to every k-tuple $(m_1, \ldots, m_k)$ such that $m_1 + \cdots + m_k = m$ and $m_1, \ldots, m_k \in \mathbb{Z}^+$.*

Question: Is it true that every 3-edge-connected graph is decomposible?

Related results:
• Every 4-edge-connected graph is decomposible.
• There are 2-edge-connected graphs that are not decomposible.
• If G contains open dominating trail then G is decomposible.
• Every 2-edge-connected graph G is 2-decomposible (can be decomposed into 2 parts of prescribed size).

References:
[1] Barát, J.; Thomassen, C.: Dividing the edges of a graph into connected subgraphs of prescribed size. Abstrakt na konferenci Eurocomb, 2003.

[2] Györi, E.: On division of graphs to connected subgraphs. In Combinatorics (Proc. Fifth Hungarian Colloq., Keszthely, 1976), Vol. 1, 1976, s. 485–494.

[3] Musílek, J.: Decomposition of graphs into connected subgraphs. Master thesis, 2015.

Problem 15. *Trains on switch graphs (suggested by Veronika Slívová)*

Source: Proposed by Matoušek and Welzl in 2014 (Question A) and Michal Koucký in 2016 (Question B).

Definitions.
• *A* switch graph *is a directed graph G in which every vertex has two outgoing edges going to its* even *and* odd *successor (loops and multiple edges are allowed).*
• *A train drives from a given vertex o (origin) along the switch graph. If a train has visited a vertex even (resp. odd) number of times it leaves it along the even (resp. odd) edge from this vertex. The train stops only at a given vertex d (destination).*

Example:

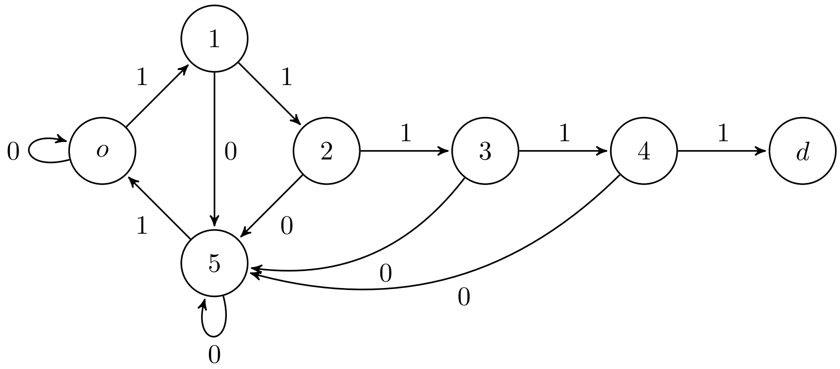

Figure 2: A graph with exponentially long route.

Question A: Can we decide in polynomial time if the train stops (i.e. arrives to the destination d)? Or is this problem as hard as some other problem in NP $\cap$ coNP (e.g. factoring, parity games, ...)?

Question B: What can we say about probability of arriving when the switches are chosen uniformly at random in the beginning? What is the expected length of the route?

This question can be restated as a question about deterministic random walks on directed graphs. See for example Friedrich and Sauerwald.

Related results:

• By the result of Dohrau et al. the problem of deciding whether the train arrives to the given destination is in NP ∩ coNP.

• There are results showing similarities between the expectation of a random walk and a deterministic random walk on undirected graphs. Does something like this hold also for directed graphs?

References:

A zero-player graph game in NP ∩ coNP
Jérôme Dohrau, Bernd Gärtner, Manuel Kohler, Jiří Matoušek, Emo Welzl
`https://arxiv.org/pdf/1605.03546v1.pdf`

The Cover Time of Deterministic Random Walks
Tobias Friedrich, Thomas Sauerwald
`http://arxiv.org/pdf/1006.3430.pdf`

Problem 16. *The NIM game on graphs (inspired by a poster in poster session at Mathematics of Jiří Matoušek conference – suggested by Jana Syrovátková)*

Source: Proposed by Criel Merino in 2016

Definitions.

- *A graph G, one vertex is a sink, on each vertex is some number of stones.*
- *One move of one player is to choose a vertex v and move one or more stones from v to each of his neighbors (on each of them the same number of stones).*
- *A game consists of moves by players. Who cannot play, looses the game.*

Example:
The usually played form of a Nim game is: "You have a number of heaps of stones and you can take away any number of stones from any but one heap. Who cannot play, loses." This could be realized as our game where G is a star and the center is the sink.

Question: You have a graph and some distribution of stones on its vertices. Who can win? How should he play to win?

Problem 17. *Compressing planar graphs (suggested by Robert Šámal)*

Source: Proposed by Thomas Böhme in 2016
Conjecture: Let G be a connected planar graph with maximal degree Δ. Then there exists $p = p(\Delta)$, a planar graph H and a mapping $f : V(G) \to V(H)$ such that

- $|f^{-1}(v)| \leq p$ for every vertex $v \in V(H)$,
- for all $x, y \in V(G)$ with $d_G(x, y) \leq 2$ we have $f(x) = f(y)$ or $\{f(x), f(y)\} \in E(H)$.

Problem 18. *Balancing two spanning trees (suggested by Robert Šámal)*

Source: Proposed by Matthias Kriesell in 2016 (and also earlier).
Conjecture: Suppose G is a union of two edge-disjoint spanning trees. Then we can write G as a union of two edge-disjoint spanning trees S, T such that for every $x \in V(G)$ we have $|\deg_S(x) - \deg_T(x)| \leq C$ for some absolute constant C.

A weaker version asks for $\deg_S(x)/\deg_T(x) \leq C$.

Problem 19. *Irregularity strength (suggested by Robert Šámal)*

Source: Proposed by Faudre and Lehel in 1987

Definition: A multigraph is irregular if no two vertex degrees are equal. A multigraph can be viewed as a weighted graph with nonnegative-integer weights on the edges. The degree of a vertex in a weighted graph is the sum of the incident weights. Chartrand et al. [CJLORS] defined the irregularity strength of a graph G, written $s(G)$, to be the minimum of the maximum edge weight in an irregular multigraph with underlying graph G.

Conjecture: Prove or disprove that if a tree T has n_1 leaves and n_2 vertices of degree 2, then $s(T) = n_1 + n_2/2$. If true, this would be sharp.

Conjecture: There is a constant c such that if G is a d-regular graph with n vertices, then $s(G) \leq (n/d) + c$.

Related results:

• Always $s(G)$ is at least the number of vertices with degree 1.

• Amar and Togni [AT] proved that equality holds for trees without vertices of degree 2.

• Bohman and Kravitz [BK] proved that there are trees (with 2-valent vertices) whose irregularity strength is greater than c times the number of leaves, where c is some constant greater than 1.

• Faudree and Lehel [FL] proved that $s(G) \leq \lceil n/2 \rceil + 9$ when G is 2-regular. For general d, the best result is by Przybylo [P]: $s(G) < 16(n/d) + 6$. This improves results by Frieze-Gould-Karoński-Pfender [FGKP] and Cuckler-Lazebnik [CL].

• Another result of Przybylo: $s(G) \leq (4 + o(1))\frac{n}{\delta} + 4$ if G has minimum degree $\delta \geq \sqrt{n} \log n$.

References:

`http://www.math.illinois.edu/~dwest/regs/irreg.html`

[AT] Amar, D.; Togni, O.; Irregularity strength of trees. Discrete Math. 190 (1998), no. 1–3, 15–38.

[BK] Bohman, Tom; Kravitz, David; On the irregularity strength of trees. J. Graph Theory 45 (2004), no. 4, 241–254.

[CJLORS] Chartrand, Gary; Jacobson, Michael S.; Lehel, Jenö; Oellermann, Ortrud R.; Ruiz, Sergio; Saba, Farrokh; Irregular networks. 250th Anniversary Conference on Graph Theory (Fort Wayne, IN, 1986). Congr. Numer. 64 (1988), 197–210.

[CL] Cuckler, Bill; Lazebnik, Felix; Irregularity Strength of Dense Graphs. J. Graph Theory (to appear).

[FL] Faudree, R. J.; Lehel, J.; Bound on the irregularity strength of regular graphs. Combinatorics (Eger, 1987), 247–256, Colloq. Math. Soc. János Bolyai, 52, North-Holland, Amsterdam, 1988.

[FGJW] Ferrara, M.; Gilbert, J.; Jacobson, M.; Whalen, T.; Irregularity Strength of Digraphs. Discrete Math (to appear).

[FGKP] Frieze, Alan; Gould, Ronald J.; Karoński, Michal; Pfender, Florian; On graph irregularity strength. J. Graph Theory 41 (2002), no. 2, 120–137.

[G] Gilbert, J.; Irregularity Strength of Digraphs, Ph.D. Dissertation, The University of Colorado Denver, May 2008.

[P] Przybylo, Jakub; Irregularity strength of regular graphs, Electr. J. Combin. 15 (2008), Paper #82, 10 pages.

Problem 20. *Path partitions (suggested by Robert Šámal)*

Source: Proposed by Gyárfáš in 1989
Conjecture: Every r-colored K_n has a partition of its edges into r monochromatic paths.
Related results:
• True for $r = 2$ (Gerencer and Gyárfáš 1967)
• True for $r = 3$ (Pokrovskiy 2014)

Problem 21. *When are random permutations k-universal (suggested by Matas Šileikis)*

Source: Proposed by Noga Alon [2] in 2016 at Oberwolfach workshop "Combinatorics and Probability".

Definitions.
• *A sequence* $(x_1, \ldots, x_n)$ *of distinct real numbers* defines *a permutation* $\sigma \in S_k$*, where* $\sigma(i) < \sigma(j)$ *whenever* $x_i < x_j$.
• *For* $n > k$*, a sequence* $(x_1, \ldots, x_n)$ *of* n *distinct real numbers* contains *a permutation* $\sigma \in S_k$ *if there is a subsequence* $(x_{i_1}, \ldots, x_{i_k})$ *defining* σ*. It is* k-universal *if it contains every* $\sigma \in S_k$.

Example:
If we denote a permutation by a vector $\sigma(1)\sigma(2)\ldots\sigma(n)$, we have that sequence $(0.9, 0.1, 0.2)$ defines the permutation 312 and is 2-universal, since $(0.9, 0.1)$ (or $(0.9, 0.2)$) defines 21 and $(0.1, 0.2)$ defines 12.

Definitions.
• *Let* $(x_1, \ldots, x_n)$ *be real numbers chosen independently and uniformly from interval* $[0, 1]$*. They define a permutation chosen uniformly from* S_n*. Let*

$$f(k) = \min\{n : S_n \text{ is } k\text{-universal with probability } \geq 1/2\}$$

Example:
Only monotone increasing and decreasing permutations are not 2-universal. The probability of 2-universality is thus $(n! - 2)/n!$, which is zero for $n = 2$ and larger than $1/2$ for $n \geq 3$, so $f(2) = 3$.

Related results:
• Arratia [1]: $f(k) \geq (1/4 + o(1))k^2$.
• Alon [2]: it is easy to show that $f(k) = O(k^2 \log k)$.

Question: In [2] Alon asked: is it true that $f(k) \leq 1000k^2$ for k large enough?

References:

[1] R. Arratia, On the Stanley-Wilf Conjecture for the Number of Permutations Avoiding a Given Pattern, *Electronic J. Combinatorics* 6 (1999), N1.

[2] N. Alon, Optimal induced universal graphs, Report of *Oberwolfach workshop 1616 "Combinatorics and Probability"*, April 2016 `www.mfo.de/document/1616/preliminary_OWR_2016_22.pdf`