9th Emléktábla Workshop

Matching Theory 06.17. - 06.21.2019.

Preliminary Schedule

Monday

Arrival in the hotel from 14:00.
15:30 Short presentations of problems
18:00 Dinner

Tuesday-Thursday

9:30 Short meeting after breakfast
9:35-17:00 Working in groups of 3-5
12:30 Lunch (except Wednesday)
17:00 Presentations of daily progress
18:00 (or whenever we are done) Dinner

Friday

Check-out at 10:00.

List of Participants

Katarina Cechlárová, Pavol Jozef Šafárik University in Košice
Tamás Fleiner, Budapest University of Technology and Economics
Tamás Király, Eötvös University, Budapest
Zoltán Király, Eötvös University, Budapest
Kavitha Telikepalli, Tata Institute of Fundamental Research, Mumbai

Martin Balko, Charles University, Prague
Erika Renáta Bérczi-Kovács, ELTE EGRES
Kristóf Bérczi, ELTE EGRES
Péter Bíró, MTA KRTK
Zoltán Blázsik, ELTE
Markus Brill, TU Berlin
Ágnes Cseh, MTA KRTK
Gábor Damásdi, MTA-ELTE CoGe
Dániel Gerbner, Rényi Institute
Zsuzsanna Jankó, Hamburg University
Balázs Keszegh, Rényi Institute and MTA-ELTE CoGe
Csaba Király, ELTE EGRES
Yusuke Kobayashi, Kyoto University
Dániel Lenger, ELTE
Jannik Matuschke, KU Leuven
Abhishek Methuku, École Polytechnique Fédérale de Lausanne
Dániel Nagy, Rényi Institute
Dömötör Pálvölgyi, MTA-ELTE CoGe
Balázs Patkós, Rényi Institute
Nika Salia, Rényi Institute
Ildiko Schlotter, BME
Ulrike Schmidt-Kraepelin, TU Berlin
Kenjiro Takazawa, Hoesi University
Casey Tompkins, Karlsruhe Institute of Technology
Lilla Tóthmérész, ELTE EGRES
Máté Vizer, Rényi Institute
Oscar Zamora Luna, Rényi Institute

The Young Physicists' Tournament

by Katarína Cechlárová

1 Young Physicists' Tournament

The International Young Physicists' Tournament (IYPT), sometimes referred to as "Physics World Cup", is a team-oriented scientific competition between secondary school students. The participating teams present their solutions to scientific problems they have prepared over several months and then discuss their solutions with other teams. More details can be found in the webpage `http://iypt.org/`.

Before taking part in the international competition, the teams compete in national tournaments. The rules I present here are valid in Slovakia.

The international jury publishes a set of 17 problems. Let us denote by $T = \{t_1, t_2, \ldots, t_m\}$ the set of teams that apply for the tournament. For simplicity, in what follows, we assume that the number of teams is a multiple of 3. Each team applying for participation submits a set of exactly 3 problems from the published set, which they like most and they will report on them. Let us call this set the team's *portfolio*. We shall denote the portfolio of team t_i by p_i^1, p_i^2, p_i^3. Table 1 gives an example of a set of applications.

Team	Portfolio	Team	Portfolio
A	1,2,3	D	3,4,5
B	1,2,4	E	1,5,6
C	2,3,4	F	4,5,7

Table 1: Applications

The organizers now prepare a schedule of the tournament. The tournament consists of 3 rounds. In each round, the set of teams is partitioned into groups of three. Each team in a group is assigned a problem from its portfolio. The group performs a so-called Fight.

The structure of the fight is as follows. First, the ordering of teams is selected by lot, assume this ordering is t_1, t_2, t_3. In the first stage of the Fight, team t_1 is the Reporter-Team; they present a report on the assigned problem. Team t_2 is the Opponent. After the presentation of the Reporter-Team, they present an estimation on the presentation. Afterwards the third team, the Reviewer-Team can ask questions to both other teams and then presents an overview of the performance of the Opponent. In the second stage of the Fight, the roles of Reporter, Reviewer and Opponent are performed by teams t_2, t_3 and t_1, respectively; and in the third stage by t_3, t_1 and t_2.

So as no team has an unjustified advantage, it is desirable that if team t presents problem p in a Fight, neither of the two other teams in its group have problem p in their portfolio. An example of a correct schedule for round 1 for the applications from Table 1 is given in Table 2. For easier reference, each team is shown with its complete portflio; the problem assigned for reporting is underlined and boldface.

A schedule is said to be *feasible* if

- each team is exactly in one group in each round;
- each team presents a different problem from its portfolio set in each round;

Group 1			Group 2		
A	D	E	B	C	F
1**2**3	3**4**5	15**6**	**1**24	2**3**4	45**6**

Table 2: A possible round 1

- if team t presents problem p being in group with two other teams then neither of them has problem p in its portfolio.

Questions:

- Can you design an algorithm to decide if a feasible schedule for a set of applications exists; and if the answer is positive, to find one?
- If a feasible schedule does not exist, can you propose a simple certificate to demonstrate this? For example, if all teams have the same portfolio, no feasible schedule is possible.
- Are there any necessary and/or sufficient conditions for the existence of a feasible schedule?

If teams are allowed to choose their portfolios completely arbitrary, the chances of a feasible schedule may be low. Let us therefore think about another approach. Suppose that instead of submitting a portfolio directly, each team submits a preference ordering of the problems, perhaps it might even be allowed to label some problems as unacceptable. We seek a matching of teams to triples of problems that enables a feasible schedule, and is in a sense optimal. Several optimality criteria can be thought of, my favourite one is to minimize the position of the worst problem in the portofio of each team.

House Allocation problem

by Tamás Fleiner

An instance I of the *House Allocation problem* (HA), also known as the Assignment problem, comprises a set $A = \{a_1, a_2, \ldots, a_{n_1}\}$ of *applicants* and a set $H = \{h_1, h_2, \ldots, h_{n_2}\}$ of *houses*. There is a set $E \subseteq A \times H$ of *acceptable* applicant–house pairs. Let $m = |E|$. Each applicant $a_i \in A$ has an *acceptable* set of houses $A(a_i)$, where $A(a_i) = \{h_j \in H : (a_i, h_j) \in E\}$. Similarly each house $h_j \in H$ has an acceptable set of applicants $A(h_j)$, where $A(h_j) = \{a_i \in A : (a_i, h_j) \in E\}$.

Each applicant $a_i \in A$ has a *preference list* which is a strict linear order $\prec_{a_i}$ over $A(a_i)$.[1] Given an applicant $a_i \in A$, and given two houses $h_j, h_k \in A(a_i)$, a_i is said to *prefer* h_j to h_k if $h_j \prec_{a_i} h_k$. For a given acceptable applicant–house pair (a_i, h_j), define $rank(a_i, h_j)$ to be 1 plus the number of houses that a_i prefers to h_j.

An *assignment* M is a subset of E. If $(a_i, h_j) \in M$, a_i and h_j are said to be *assigned to* one another. For each $p_k \in A \cup H$, the set of assignees of p_k in M is denoted by $M(p_k)$. If $M(p_k) = \emptyset$, p_k is said to be *unassigned*, otherwise p_k is *assigned*. A *matching* M is an assignment such that $|M(p_k)| \leq 1$ for each $p_k \in A \cup H$. For notational convenience, if p_k is assigned in M then where there is no ambiguity the notation $M(p_k)$ is also used to refer to the single member of the set $M(p_k)$. Let $\mathcal{M}$ denote the set of matchings in I.

The preferences of an applicant extend to $\mathcal{M}$ as follows. Given two matchings $M, M' \in \mathcal{M}$, we say that an applicant $a_i \in A$ *prefers* M' *to* M if either (i) a_i is assigned in M' and unassigned in M, or (ii) a_i is assigned in both M and M', and a_i prefers $M'(a_i)$ to $M(a_i)$. A matching $M \in \mathcal{M}$ is defined to be *Pareto optimal* if M is $\triangleleft$-minimal. Equivalently, M is Pareto optimal if and only if there is no other matching M' in I such that (i) some applicant prefers M' to M, and (ii) no applicant prefers M to M'.

2 House allocation via exchanges in a social network

In a well-studied and realistic variant of HA, each applicant a_i initially owns a house $M(a_i)$, and the goal of the applicants is to exchange these houses among themselves [8]. These exchanges can happen either in a centralised [8, 1, 6, 9, 2] or a decentralised manner [7, 3, 4, 5].

The latter case was studied in a social network by Gourvés et al. [5]. Two applicants in a social network either know each other, and they are capable of exchanging their houses, or they do not know each other, in which case no exchange can happen. More formally, each applicant a_i has an acceptable set $A(a_i)$ of houses, as in the other variants of HA discussed above, but in addition a_i has an acceptable set $A_a(a_i)$ of applicants. These $A_a(a_i)$ sets trivially define a *social network* N, which is a graph $N = (A, E^*)$, where $\{a_i, a_j\} \in E^*$ if and only if $a_j \in A_a(a_i)$ and $a_i \in A_a(a_j)$. As in the case of HA, applicant a_i ranks $A(a_i)$, including her initial endowment, in strict order, and she is only inclined to participate in an exchange if the house she receives is better than the one she currently owns.

In a social network it can happen that an exchange between applicants a_i and a_j that was infeasible earlier becomes feasible later. If $a_j \in A_a(a_i)$, and a_i finds $M(a_j)$ worse than her own house $M(a_i)$, then a_i will not accept $M(a_j)$ in any feasible exchange. However, if a_j participates in an exchange following which she receives a house $M'(a_j)$ that a_i ranks higher than her current house, then a_i suddenly becomes interested in accepting a_j's new house $M'(a_j)$.

[1] That is, $\prec_{a_i}$ is an irreflexive, transitive and linear binary relation over $A(a_i)$.

Gourvés et al. [5] restricted their attention to *swaps* in social networks. Swaps are exchanges of length 2, involving two applicants only, who swap their houses with each other, changing $\{(a_i, M(a_i)), (a_j, M(a_j))\}$ to $\{(a_i, M(a_j)), (a_j, M(a_i))\}$ in the new matching. If a matching M' is reachable from the initial matching M by a sequence of such swaps, then we call it a *reachable matching*. Similarly, houses that an applicant a_i can receive via swaps belong to the set of *reachable houses*. Pareto optimal matchings are also defined based on swaps. A matching M' is considered to be Pareto optimal if it is reachable from M and there is no other reachable matching M'' that Pareto-dominates M'. We summarise the main results from [5] in the following three theorems.

Theorem 1 ([5]). *The problem of deciding whether house h_j is reachable for applicant a_i is NP-complete even if the network N is a tree. The problem becomes polynomially solvable if N is a path and a_i is a leaf on this path, or when N is a star.*

Theorem 2 ([5]). *The problem of deciding whether matching M' is reachable from matching M is NP-complete. The problem becomes polynomially solvable if N is a tree.*

Theorem 3 ([5]). *The problem of finding a Pareto optimal matching is NP-hard even if the network N is a tree. The problem becomes polynomially solvable if N is a star.*

Table 3 contains a structured interpretation of the above results.

	path	star	tree	general
h REACHABLE	polynomial if a_i is a leaf	polynomial	NP-complete	NP-complete
M REACHABLE	polynomial	polynomial	polynomial	NP-complete
FIND PO MATCHING	polynomial	polynomial	NP-complete	NP-complete

Table 3: The complexity table summarising the three problems in [5]. The columns are the type of graph for which we pose the problem in the row, corresponding to Theorems 1, 2, and 3, in this order.

The most striking open question regarding exchanges in social networks involves longer cycles instead of swaps only. Strategyproofness is another topic worthy of investigation. Also, the hard cases in Theorems 1, 2, and 3 could be tackled from a parameterised complexity viewpoint.

References

[1] A. Abdulkadiroğlu and T. Sönmez. House allocation with existing tenants. *Journal of Economic Theory*, 88:233–260, 1999.

[2] H. Aziz. Algorithms for Pareto optimal exchange with bounded exchange cycles. Working paper, available from `http://www.cse.unsw.edu.au/~haziz/lexchange.pdf` (accessed 28 May 2019), 2018.

[3] Y. Chevaleyre, P. E. Dunne, U. Endriss, J. Lang, M. Lemaitre, N. Maudet, J. Padget, S. Phelps, J. A. Rodriguez-Aguilar, and P. Sousa. Issues in multiagent resource allocation. *Informatica*, 30(1):3–31, 2006.

[4] A. Damamme, A. Beynier, Y. Chevaleyre, and N. Maudet. The power of swap deals in distributed resource allocation. In *Proceedings of AAMAS '15: the 14th International Conference on Autonomous Agents and Multiagent Systems*, pages 625–633. International Foundation for Autonomous Agents and Multiagent Systems, 2015.

[5] L. Gourvès, J. Lesca, and A. Wilczynski. Object allocation via swaps along a social network. In *Proceedings of IJCAI '17: the 26th International Joint Conference on Artificial Intelligence*, pages 213–219. AAAI Press, 2017.

[6] C. G. Plaxton. A simple family of top trading cycles mechanisms for housing markets with indifferences. In *Proceedings of the 24th International Conference on Game Theory, Stony Brook*, pages 1–23. `https://www.cs.utexas.edu/users/plaxton/pubs/2013/icgt.pdf` (accessed 27 May 2019), 2013.

[7] T. Sandholm. Contract types for satisficing task allocation: I theoretical results. In *Proceedings of the AAAI Spring Symposium: Satisficing Models*, pages 68–75. AAAI Press, 1998.

[8] L. Shapley and H. Scarf. On cores and indivisibility. *Journal of Mathematical Economics*, 1:23–37, 1974.

[9] T. Sönmez and T. Switzer. Matching with (branch-of-choice) contracts at United States Military Academy. *Econometrica*, 81(2):451–488, 2013.

Three problems more-or-less related to matchings

by Tamás Király

3 A pairing approach to Woodall's conjecture

Let $D = (V, E)$ be a directed graph. An edge set $F \subseteq E$ is a *dijoin* if it contains at least one edge from every directed cut of D; it is a *k-dijoin* if it contains at least k edges from every directed cut.

Woodall [8, 9] conjectured that the maximum number of edge-disjoint dijoins in a digraph equals the minimum size of a directed cut. This conjecture is still open – there are only a few classes of graphs for which it is known to hold (source-sink connected graphs, series-parallel graphs, transitive closure of directed trees).

A possible generalization would be that a k-dijoin of a digraph always contains k edge-disjoint dijoins. However, Schrijver [7] showed that this is not true in general. His counterexample also shows that there is a directed graph $D = (V, E)$ and a 2-dijoin $F \subseteq E$ such that for any $F' \subseteq F$, there is a directed cut that contains at most 3 edges from F and is disjoint from either F' or $F - F'$.

In [6], we showed that for every $k \geq 2$, if $D = (V, E)$ is a directed graph and $F \subseteq E$ is a $2k$-dijoin, then there exists $F' \subseteq F$ such that both F' and $F - F'$ contain k edges from every directed cut that has at most $2k + 1$ edges in F. We conjectured that the following is also true: if $k \geq 2$, then any $2k$-dijoin can be partitioned into two k-dijoins.

Let $D = (V, E)$ be a directed graph and $F \subseteq E$ a subset of edges. We use the notation

$$d_F^{in}(X) := |\{uv \in F : \; u \in V - X, v \in X\},$$
$$d_F^{out}(X) := |\{uv \in F : \; u \in X, v \in V - X\},$$
$$d_F(X) := d_F^{in}(X) + d_F^{out}(X).$$

Let us define the following families of sets:

$$\mathcal{I} := \{\emptyset \neq X \subsetneq V : \; d_E^{out}(X) = 0\},$$
$$\mathcal{I}_j := \{X \in \mathcal{I} : \; d_F(X) = j\}.$$

The proof in [6] relies on the following lemma:

Lemma 1 ([6]). *Let $D = (V, E)$ be a digraph, $k \geq 2$ an integer, and $F \subseteq E$ a $2k$-dijoin. There is a pairing M of the nodes with $d_F(v)$ odd such that*

$$d_M(X) = 0 \quad \text{if } X \in \mathcal{I}_{2k},$$
$$d_M(X) = 1 \quad \text{if } X \in \mathcal{I}_{2k+1}.$$

Corollary 2. *Let $D = (V, E)$ be a digraph, $k \geq 2$ an integer, and $F \subseteq E$ a $2k$-dijoin. There is an edge set $F' \subseteq F$ so that*

$$\min\{d_{F'}(X), d_{F-F'}(X)\} \geq k \quad \text{for every } X \in \mathcal{I}_{2k} \cup \mathcal{I}_{2k+1}.$$

Proof. Let M be the pairing that exists according to Lemma 1, and let G be the even-degree graph obtained by taking the union of M and the edges of F without orientation. We select an arbitrary Eulerian orientation $\vec{G}$ of G, and denote by F' the set of edges in F that have the same orientation in $\vec{G}$ as in D. If $X \in \mathcal{I}_{2k}$, then $d_M(X) = 0$, so $d_{F'}(X) = d_{\vec{G}}^{in}(X) = k$. If $X \in \mathcal{I}_{2k+1}$, then $d_M(X) = 1$, so $d_{F'}(X) \leq d_{\vec{G}}^{in}(X) = k + 1$ and $d_{F'}(X) \geq d_{\vec{G}}^{in}(X) - d_M(X) = k$. $\square$

Note that Corollary 2 is not true for $k = 1$ by the counterexample of Schrijver [7]. It is an interesting open question whether Theorem 1 can be generalized by requiring $d_M(X) \leq d_F(X) - 2k$ for every $X \in \mathcal{I}$.

Question 1. *Let $D = (V, E)$ be a digraph, $k \geq 2$ an integer, and $F \subseteq E$ a $2k$-dijoin. Is there a pairing M of the nodes with $d_F(v)$ odd such that $d_M(X) \leq d_F(X) - 2k$ for every $X \subseteq V$ for which $d_E^{out}(X) = 0$?*

If true, this would imply the following relaxation of the capacitated version of Woodall's conjecture.

Question 2. *Let $D = (V, E)$ be a digraph, $k \geq 2$ an integer, and $F \subseteq E$ a $2k$-dijoin. Is it true that F can be partitioned into two k-dijoins?*

We note that it is open if there is a function $f(k)$ such that every $f(k)$-dijoin of a digraph can be partitioned into k dijoins. An affirmative answer to the above questions would confirm the existence of such a function.

4 Popular arborescences

The popular arborescence problem is related to the popular matching problem (see e.g. [1, 4, 5]), but there are no results on this topic yet, so there are plenty of open questions.

Let $D = (V + r, E)$ be a digraph with a specified root node r. For every $v \in V$, $\prec_v$ is a preference order defined on the incoming edges. We obtain different problems based on whether $\prec_v$ is required to be a total order or only a partial order. Given spanning arborescences A and A' of D rooted at r, a node $v \in V$ *prefers* A to A' if $\delta^{in}_A(v) \succ_v \delta^{in}_{A'}(v)$.

Definition. *A spanning arborescence A is* popular *(*strongly popular*) if for any spanning arborescence $A' \neq A$, the number of nodes preferring A' to A is less or equal (strictly less) than the number of nodes preferring A to A'.*

To decide whether a given arborescence A is popular, we can assign weights to the edges of the digraph such that the maximum weight of an arborescence is positive if and only if A is not popular. A similar technique works for deciding if A is strongly popular.

Exercise 3. *Show a digraph with preferences where no popular arborescence exists.*

Question 3. *Can we decide in polynomial time if there is a popular arborescence?*

Question 4. *Can we decide in polynomial time if there is a strongly popular arborescence?*

The same questions can be asked if each $v \in V$ has a weight w_v, and the vote of v is worth w_v when deciding popularity. It can still be decided whether a given arborescence is popular using the minimum weight arborescence algorithm.

5 Weighted bipartite edge colouring

Let $G = (S, T; E)$ be a bipartite graph, with weights $w : E \to [0, 1]$. A *proper weighted edge colouring* is a colouring of the edges such that at each vertex, the sum of weights of edges of the same colour is at most 1. A lower bound for the number of colours is the minimum number of unit bins needed to pack the weights incident to any vertex, denoted by b. Chung and Ross [2] asked the following question.

Question 5. *Is there always a proper weighted edge colouring using $2b - 1$ colours?*

Currently, the best bound is $20b/9 + o(b)$ by Khan and Singh [3]. Their proof uses the configuration LP of the bin packing problem.

References

[1] P. Biró, R. W. Irving, and D. F. Manlove, *Popular matchings in the marriage and roommates problems*, In Proceedings of the 7th International Conference on Algorithms and Complexity, volume 6078 of Lecture Notes in Computer Science (2010), 97–108.

[2] S.-P. Chung, K.W. Ross, *On nonblocking multirate interconnection networks*, SIAM Journal on Computing, 20(4):726–736, 1991

[3] A. Khan, M. Singh, *On Weighted Bipartite Edge Coloring*, 35th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2015). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2015.

[4] Á. Cseh, T. Kavitha, *Popular Matchings in Complete Graphs*, arXiv: 1807.01112

[5] Y. Faenza, T. Kavitha, V. Powers, X. Zhang, *Popular Matchings and Limits to Tractability*, Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms. 2019, 2790-2809

[6] T. Király, *A result on crossing families of odd sets*, EGRES Techincal Report 2007-10 (2007).

[7] A. Schrijver, *A conterexample to a conjecture of Edmonds and Giles*, Discrete Mathematics 32 (1980), 213–214.

[8] D.R. Woodall, *Menger and Kőnig systems*, in: Y. Alavi and D.R. Lick, eds, Theory and Applications of Graphs, Lecture Notes in Mathematics 642, Springer (1978), pp. 620–635.

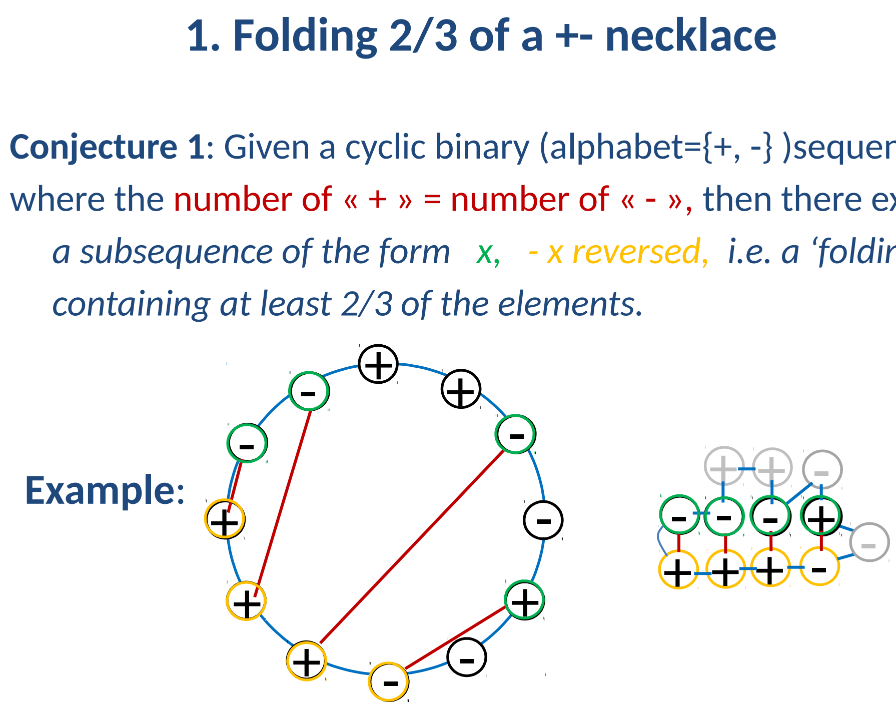
[9] D.R. Woodall, *Minimax theorems in graph theory*, in: L.W. Beineke and R.J. Wilson, es, Selected Topics in Graph Theory, Academic Press (1978), pp. 237–269.

Necklace folding and Exact matching

by Zoltán Király

6 Necklace folding problem

Given a necklace with n red $(+1)$ and n blue (-1) beads.

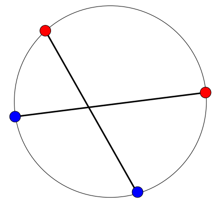


A slide of András Sebő

Given a cyclic sequence $c_0, \dots, c_{2n-1}$ where $c_i \in \{-1, +1\}$ and $\sum c_i = 0$.

A matching consists of mutually disjoint pairs $(c_{i_1}, c_{j_1}), (c_{i_2}, c_{j_2}), \dots, (c_{i_m}, c_{j_m})$ where for all $1 \leq k \leq m$ we have $c_{i_k} + c_{j_k} = 0$.

A matching is cross-free if no:



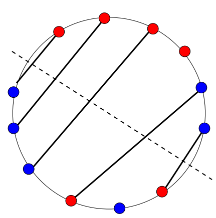
A crossing

Exercise 1. *Show that there is always a* perfect *cross-free matching.*

A secant is a pair of indices $0 \leq s_1 \neq s_2 \leq 2n-1$. We think that we cut the necklace along a secant, i.e., between c_{s_1} and c_{s_1+1}; and also between c_{s_2} and c_{s_2+1} (the indices of c are always meant modulo $2n$).

Definition 2. *A matching* $(c_{i_1}, c_{j_1}), (c_{i_2}, c_{j_2}), \ldots, (c_{i_m}, c_{j_m})$ *is* secant-respecting *if there is a secant* s_1, s_2 *such that* $s_1 \leq i_k \leq s_2$ *and* $s_2 \leq j_k \leq s_1$.

A matching is proper *if it is cross-free and secant-respecting.*



A proper matching

Exercise 3. *Show that there is always a proper matching of size* $\lceil n/2 \rceil$.

Exercise 4. *Show that the largest proper matching be found in* $O(n^3)$.

Exercise 5. *Show examples where the maximum size of a proper matching is* $(2n/3) + 1$.

Conjecture 1 (Lingsø-Pedersen (1999) and Giberti-Preissmann-Sebő (2004))**.** *There is always a proper matching of size* $2n/3$.

Definition 6. *An arc is* $c_i, c_{i+1}, \ldots, c_j$. *An arc is a half-circle if* $j = i + n - 1$. *A secant is a diameter if the two arcs it defines are both half-circles* ($s_2 = s_1 + n$)*.*

Conjecture 2 (Strong)**.** *There is always a proper matching of size* $2n/3$ *where the secant is a diameter.*

It would be interesting and instructive if someone has a counterexample to the Strong Conjecture.

Conjecture 3 (Weak)**.** *There is always a proper matching of size* $0.51 \cdot n$.

Definition 7. *An arc is unbalanced if* $\left|\sum_i^j c_k\right| > 0.51 \cdot n$.

Exercise 8. *If there is an unbalanced half-circle, then the weak conjecture is true.*

7 Exact matching

Definition 9. *Given $G = (V, E)$ with $|V| = 2n$, a two-coloring $E \to \{\text{red}, \text{blue}\}$ and positive integer k; a perfect matching of G consisting of exactly k red (and $n-k$ blue) edges is called an* exact matching.

Problem 4. *Decide in deterministic polynomial time whether an exact matching exists or not.*

Problem 5. *Suppose G is bipartite. Decide in deterministic polynomial time whether an exact matching exists or not.*

An edge-coloring is defined by a vertex-subset R if edge uv is red if and only if $u, v \in R$.

Problem 6. *Suppose the edge-coloring is defined by a vertex-subset R. Decide in deterministic polynomial time whether an exact matching exists or not.*

Problem 7. *Suppose G is bipartite and the edge-coloring is defined by a vertex-subset R. Decide in deterministic polynomial time whether an exact matching exists or not.*

An edge-coloring is defined by a two-patition $V = X \cup Y$ if edge uv is red if and only if $u \in X$ and $v \in Y$.

Problem 8. *Suppose the edge-coloring is defined by a two-patition $V = X \cup Y$. Decide in deterministic polynomial time whether an exact matching exists or not.*

Problem 9. *Suppose G is bipartite and the edge-coloring is defined by a two-patition $V = X \cup Y$. Decide in deterministic polynomial time whether an exact matching exists or not.*

Exercise 10. *Show that Problems 6 and 8 are 'equivalent'. Problems 7 and 9 are also 'equivalent'.*

Exercise 11. *Given an instance of Problem 8 (or Problem 9), and suppose we found a perfect matching having k' red edges. Show that if $k - k'$ is odd, then no exact matching exists.*

All of these are open! The first two were investigated several times while the latest four were not.

Theorem 12 (Karzanov, 1987)**.** *There are good characterizations yielding deterministic polynomial time algorithms for the Problems 4 and 5 in the two special cases when $G = K_{2n}$ or $G = K_{n,n}$.*

Theorem 13 (Lovász, 1979; and Mulmuley-Vazirani-Vazirani, 1987)**.** *The most general Problem 4 is in RP, moreover, it is in RNC^2. In RNC^2 the matching required can also be found if it exists.*

Interestingly enough, Yuster gave an 'approximation' for Problem 4 in the following strong sense.

Theorem 14 (Yuster, 2007 (APPROX), 2011 (Algorithmica))**.** *There is a deterministic polynomial time algorithm which either correctly asserts that no perfect matching contains exactly k red edges, or exhibits a matching of size at least $n-1$ with exactly k red edges.*

References

[1] L. Lovász, On Determinants, Matchings and Random Algorithms, *Fundamentals of Computing Theory*, Akademia-Verlag, Berlin, (1979).

[2] K. Mulmuley, U. V. Vazirani and V. V. Vazirani, Matching is as easy as matrix inversion, *Combinatorica*, **7**(1), (1987), pp. 105–113.

[3] A. V.. Karzanov, Maximum matching of given weight in complete and complete bipartite graphs, *Cybernetics and Systems Analysis*, **23**(1), (1987), pp. 8–13.

[4] R. Yuster, Almost Exact Matchings, *Algorithmica*, **63**(1-1), (2012), pp. 39–50.

Some problems in Popular / Pareto-optimal / Stable Matchings

by Kavitha Telikepalli

In all the problems listed here, the input is a bipartite graph $G = (A \cup B, E)$ where each vertex has a preference list ranking its neighbors. Such a graph G is typically called a *marriage* instance (with incomplete lists). Let $|A \cup B| = n$ and $|E| = m$. In the first 4 sections we assume preferences are *strict*.

8 Popular Matchings

Given a vertex u and a matching M such that u is matched in M, let $M(u)$ be u's partner in M. A vertex u prefers matching M to matching N if either (i) u is matched in M and unmatched in N or (ii) u is matched in both M, N and u prefers $M(u)$ to $N(u)$. For any two matchings M and N, let $\phi(M, N)$ be the number of vertices that prefer M to N.

Definition 1. *A matching M is popular if $\phi(M, N) \geq \phi(N, M)$ for all matchings N in G.*

Thus a popular matching never loses a head-to-head election against any matching where every vertex casts a vote for the matching that it prefers. It is easy to show that every stable matching is popular [4]. The question we consider here is:

- given an instance $G = (A \cup B, E)$, how efficiently can we decide if every popular matching in G is stable? That is, is $\{$popular matchings$\} \supset \{$stable matchings$\}$ or $\{$popular matchings$\} = \{$stable matchings$\}$?

It is known that every stable matching is a *min-size* popular matching [7]. We also know how to compute a *max-size* popular matching in linear time [10]. Thus it can be determined in linear time if G admits popular matchings of more than one size. If so, then G admits *unstable* popular matchings.

Suppose all popular matchings in G have the same size. Then how do we answer the above question? There is an $O(m^2)$ algorithm for this problem [1] and it works with *dominant matchings*. A dominant matching M is a popular matching that is *more popular* than all larger matchings. That is, (i) M is popular and (ii) if $|M'| > |M|$ then $\phi(M, M') > \phi(M', M)$. Dominant matchings always exist in G [7] and such a matching can be computed in linear time [10].

It was shown in [2] that if G admits an unstable popular matching then G has to admit an unstable dominant matching. The algorithm for deciding if G admits an unstable dominant matching or not checks for each $e \in E$, if there exists a dominant matching in G with e as a blocking edge. This can be done in linear time. If no such dominant matching exists for any edge e, then every dominant matching in G is stable and hence every popular matching in G is stable. The total time taken by this algorithm is $O(m \cdot m) = O(m^2)$.

Question 1. Is there a faster way to solve the above problem? In particular, can we decide in linear time if G has a popular matching that is not stable?

9 Weighted-Popular Matchings

This is a more general variant of the popular matching problem. The input here is a marriage instance $G = (A \cup B, E)$ where every vertex v has a weight w_v. For any two matchings M and N, let $\phi(M, N)$ be the *sum of weights* of vertices that prefer M to N. Thus in the first problem, $w_v = 1$ for all $v \in A \cup B$.

$*$ A matching M is *weighted-popular* if $\phi(M, N) \geq \phi(N, M)$ for all matchings N in G.

Note that a weighted-popular matching need not exist in a given instance. Consider the following simple example:

- Let $A = \{a_1, a_2, a_3\}$ and $B = \{b_1, b_2, b_3\}$. The edge set $E = A \times B$. Let $w_a = 2$ for each $a \in A$ and $w_b = 1$ for each $b \in B$.
- For $i \in \{1, 2, 3\}$, the preference list of a_i is $b_1 \succ b_2 \succ b_3$, i.e., b_1 is its first choice, b_2 is its second choice, and b_3 is its third choice. For $i \in \{1, 2, 3\}$, the preference list of b_i is $a_1 \succ a_2 \succ a_3$.

The matching $M_1 = \{(a_1, b_1), (a_2, b_2), (a_3, b_3)\}$ is stable, however it is not weighted-popular since the matching $M_2 = \{(a_2, b_1), (a_3, b_2), (a_1, b_3)\}$ defeats it. Observe that a_2, a_3, and b_3 prefer M_2 to M_1 while a_1, b_1, and b_2 prefer M_1 to M_2. So $\phi(M_2, M_1) = 2+2+1 = 5$ while $\phi(M_1, M_2) = 2+1+1 = 4$.

Similarly, M_2 is not weighted-popular since $M_3 = \{(a_3, b_1), (a_1, b_2), (a_2, b_3)\}$ defeats it. Observe that a_1, a_3, and b_2 prefer M_3 to M_2 while a_2, b_1, and b_3 prefer M_2 to M_3. Thus $\phi(M_3, M_2) = 2 + 2 + 1 = 5$ while $\phi(M_2, M_3) = 2 + 1 + 1 = 4$.

Similarly, M_3 is not weighted-popular as M_1 defeats it. Observe that a_1, a_2, and b_1 prefer M_1 to M_3 while a_3, b_2, and b_3 prefer M_3 to M_1. Thus $\phi(M_1, M_3) = 2 + 2 + 1 = 5$ while $\phi(M_3, M_1) = 2 + 1 + 1 = 4$. It is easy to check that no weighted-popular matching exists in this instance.

Question 2. Given a weighted-popular matching instance $G = (A \cup B, E)$, what is the complexity of deciding if a weighted-popular matching exists or not in G? Is it NP-hard or does it admit a polynomial time algorithm?

10 Popular roommates

Given a *roommates* instance $G = (V, E)$ (i.e., G need not be bipartite), the problem of deciding if G admits a popular matching or not is the *popular roommates* problem. Note that we are back to $w_v = 1$ for all $v \in V$. The popular roommates problem is NP-hard [3, 5].

The popular roommates problem admits a "simply" exponential time algorithm, i.e., one that runs in time $O^*(k^n)$ for some constant k. But the constant k we have for this problem is huge! Our algorithm is based on enumerating candidate matchings and testing each such matching for popularity. It uses the following facts:

- all stable matchings in a marriage instance can be enumerated with an algorithm given in [6] that takes $O^*(N)$ time where N is the number of stable matchings in this instance
- the number of stable matchings in a marriage instance with n vertices on each side is at most c^n for a constant c [9].

Our algorithm for popular roommates has running time $O^*(k^n)$, where $k \leq c^2 \leq 2^{34}$.

Question 3. Is there a *fast* exponential time algorithm for the popular roommates problem?

11 Pareto-optimal Matchings

We now move to a much larger class than the set of popular matchings. This is the set of *Pareto-optimal* matchings. As in the first two problems, the input is a marriage instance $G = (A \cup B, E)$.

Definition 2. *A matching M in G is Pareto-optimal if there is no matching N such that $\phi(N, M) > 0$ and $\phi(M, N) = 0$.*

So if there are one or more vertices that prefer some matching N to a Pareto-optimal matching M, then there has to exist at least one vertex that prefers M to N. Pareto-optimal matchings always exist in $G = (A \cup B, E)$ since every stable matching is Pareto-optimal.

There always exists a *max-size* matching in G that is Pareto-optimal and such a matching can be computed in polynomial time. Thus the max-size Pareto-optimal matching problem is tractable and this is so even in roommates instances. However the *max-weight* Pareto-optimal matching problem is NP-hard even in a marriage instance with a weight function $w : E \to \{1, 2\}$.

Question 4. What is the complexity of computing a min-size Pareto-optimal matching? What is the complexity of computing a Pareto-optimal matching in $G = (A \cup B, E)$ with a forced/forbidden edge?

12 Weakly Stable Matchings

We now assume that preference lists admit ties. As before, we have a marriage instance $G = (A \cup B, E)$ where each vertex in $A \cup B$ has a preference list (with ties allowed) ranking its neighbors.

Definition 3. *A matching M is* weakly stable *if M has no blocking edge, i.e., a pair of vertices that prefer each other to their respective assignments in M.*

Though computing weakly stable matchings is easy, computing a *max-size* weakly stable matching is NP-hard. The current best approximation algorithm for this problem is from 2009 with an approximation ratio of 3/2 [12]. There are more recent linear time algorithms that achieve this approximation ratio [11, 13].

The factor of 3/2 matches the integrality gap of a natural LP associated with this problem: in particular, the integrality gap is at least $(3\ell - 2)/(2\ell - 3)$ where ℓ is the maximum tie length [8]. It is known that getting an approximation ratio of $4/3 - \varepsilon$, for any constant $\varepsilon > 0$, is UGC-hard [14].

Question 5. Can we show an improved approximation algorithm for the max-size weakly stable matching problem? Or can we improve the lower bound?

References

[1] Á. Cseh, Y. Faenza, T. Kavitha, and V. Powers. *Understanding popular matchings via stable matchings.* `http://arxiv.org/abs/1811.06897`

[2] Á. Cseh and T. Kavitha. *Popular edges and dominant matchings.* Mathematical Programming, 172(1):209–229, 2018.

[3] Y. Faenza, T. Kavitha, V. Powers, and X. Zhang. *Popular Matchings and Limits to Tractability.* In SODA 2019.

[4] P. Gärdenfors. *Match making: assignments based on bilateral preferences.* Behavioural Science, 20:166–173, 1975.

[5] S. Gupta, P. Misra, S. Saurabh, and M. Zehavi. Popular matching in roommates setting is NP-hard. In SODA 2019.

[6] D. Gusfield and R. W. Irving. *The Stable Marriage Problem: Structure and Algorithms.* MIT Press, 1989.

[7] C.-C. Huang and T. Kavitha. *Popular matchings in the stable marriage problem.* Information and Computation, 222: 180–194, 2013.

[8] K. Iwama, S. Miyazaki, and H. Yanagisawa. *A 25/17-approximation algorithm for the stable marriage problem with one-sided ties.* Algorithmica 68(3): 758-775, 2014.

[9] A. Karlin, S. Oveis Gharan, and R. Weber. *A Simply Exponential Upper Bound on the Maximum Number of Stable Matchings.* In STOC 2018.

[10] T. Kavitha. *A size-popularity tradeoff in the stable marriage problem.* SIAM Journal on Computing, 43(1): 52–71, 2014.

[11] Z. Király. *Linear time local approximation algorithm for maximum stable marriage.* Algorithms, 6(3): 471–484, 2013.

[12] E. J. McDermid. *A 3/2-approximation algorithm for general stable marriage.* In ICALP 2009.

[13] K. Paluch. *Faster and simpler approximation of stable matchings.* Algorithms, 7(2): 189–202, 2014.

[14] H. Yanagisawa. *Approximation Algorithms for the stable marriage problem.* Ph.D. thesis, Graduate School of Informatics, Kyoto University, 2007.

Contributed Problems

Erdős-Ko-Rado for tilings

by Casey Tompkins

The classical Erdős-Ko-Rado theorem asserts that a pairwise intersecting r-uniform, n-vertex hypergraph with $2r \leq n$, contains at most $\binom{n-1}{r-1}$ hyperedges.

Butler, Horn and Tressler [1] introduced a variation of this problem for tilings of a 2 by n chessboard with dominos. Their theorem may be reformulated equivalently for tilings of a 1 by n chessboard with squares and dominos as follows:

Theorem 1 (Butler, Horn and Tressler reformulated)**.** *Let $\mathcal{T}$ be a collection of tilings of a 1 by n chessboard with 1 by 1 squares and 2 by 1 dominos. Assume moreover that for any $S, T \in \mathcal{T}$, we have that S and T have at least one square or domino in the same position. Then $|\mathcal{T}|$ is at most the Fibonacci number f_n, that is, the size of the set of tilings all containing a square in the first position.*

The proof of this theorem is very short and elegant and is included at the end. If instead of 1 by 1 and 2 by 1 dominos, we consider 1 by 1 and ℓ by 1 dominos (ℓ-ominos) the same proof goes through basically unchanged. However even for the case of tilings with 1 by 1, 2 by 1 and 3 by 1 dominos we don't know the answer.

Problem 1. *Let $\mathcal{L}$ be a set of tile lengths. Call a set of tilings $\mathcal{T}$ consisting of tiles of the form ℓ by 1 where $\ell \in \mathcal{L}$ intersecting if every pair of tilings from $\mathcal{T}$ has a tile occupying the same positions. Determine the maximum size of such a set of tilings $\mathcal{T}$. In particular if $\mathcal{L} = \{1, 2, 3\}$, is the optimum to take all tilings starting (or ending) with a square?*

Proof of Butler, Horn and Tressler: We simply introduce an injection ϕ from the set of tilings starting with a domino to those starting with a square in such a way that T and $\phi(T)$ are nonintersecting. The following injection works. Let us denote a square/domino tiling by a sequence of 2's and 1's in the obvious way. Partition any such sequence into blocks starting with a 2 followed by any number of 1's, say $21\ldots1$. If the number of 1's is even $\phi(21\ldots1) = 112\ldots2$, if the number of 1's is odd $\phi(21\ldots1) = 12\ldots2$. Performing this operation block-wise defines the required injection.

References

[1] Butler, Steve, Paul Horn, and Eric Tressler. "Intersecting domino tilings." The Fibonacci Quarterly 48 (2010): 114-120.

Ordered Ramsey numbers of matchings

by Martin Balko

An *ordered graph* $\mathcal{G}$ is a pair $(G, \prec)$ where G is a graph and $\prec$ is a total ordering of its vertices. The *ordered Ramsey number* $\overline{R}(\mathcal{G})$ is the minimum $N \in \mathbb{N}$ such that every 2-coloring of the edges of the ordered complete graph $\mathcal{K}_N$ on N vertices contains a monochromatic copy of $\mathcal{G}$. That is, we want to find G as a monochromatic subgraph with a fixed order of vertices; see Figure 1.

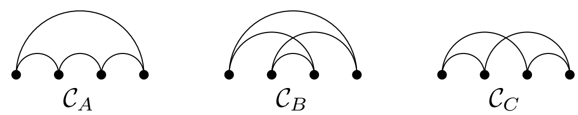


Figure 1: Pairwise non-isomorphic ordered cycles on four vertices. The ordered Ramsey numbers are $\overline{R}(\mathcal{C}_A) = 14$, $\overline{R}(\mathcal{C}_B) = 10$, and $\overline{R}(\mathcal{C}_C) = 11$. The Ramsey number of (unordered) C_4 is 6.

Let $R(G)$ be the Ramsey number of G. It is easy to see that $R(G) \leq \overline{R}(\mathcal{G})$ for each vertex-ordering $\mathcal{G}$ of G. We also have $\overline{R}(\mathcal{G}) \leq \overline{R}(\mathcal{K}_n) = R(K_n)$ and thus ordered Ramsey numbers are always finite and at most exponential in the number of vertices.

For an even positive integer n, a *matching* on n vertices is formed by $n/2$ pairwise disjoint edges. In the 1980s, Chvátal, Rödl, Szemerédi, and Trotter [3] showed that the Ramsey number $R(G)$ of every n-vertex graph G with constant maximum degree is linear in n. In sharp contrast to this result, Balko, Cibulka, Král, and Kynčl [1] and, independently, Conlon, Fox, Lee, and Sudakov [2] showed that there are ordered matchings $\mathcal{M}_n$ on n vertices with $\overline{R}(\mathcal{M}_n)$ superpolynomial in n. In particular, these results give $\overline{R}(\mathcal{M}_n) \geq n^{\Omega(\log n/\log\log n)}$. It follows from a result of Conlon et al. [2] that every ordered matching $\mathcal{M}$ on n vertices satisfies $\overline{R}(\mathcal{M}) \leq n^{O(\log n)}$. There is a gap between these lower and upper bounds and it would be interesting to close it.

Problem 1 ([2])**.** *Close the gap between the bounds on ordered Ramsey numbers of ordered matchings on n vertices.*

Very little is known for the case with more than two colors. For an ordered graph $\mathcal{G}$ and $q \in \mathbb{N}$, let $\overline{R}(\mathcal{G}; q)$ be the minimum $N \in \mathbb{N}$ such that each q-coloring of the edges of $\mathcal{K}_N$ contains a monochromatic copy of $\mathcal{G}$. Conlon et al. [2] showed $\overline{R}(\mathcal{M}; q) \leq n^{(2\log_2 n)^{q-1}}$ for each ordered matching $\mathcal{M}$ on n vertices and $q \geq 3$. However, they believe that a stronger bound should hold.

Problem 2 ([2])**.** *Show that, for $q \geq 3$, there exists a constant $c = c(q)$ such that $\overline{R}(\mathcal{M}; q) \leq n^{c\cdot\log_2 n}$ for every ordered matching $\mathcal{M}$ on n vertices.*

Plenty of other open problems about ordered Ramsey numbers can be found in [1, 2].

References

[1] M. Balko, J. Cibulka, K. Král, J. Kynčl. Ramsey numbers of ordered graphs, *Electron. Notes Discrete Math.*, **49**, 419–424, 2015.

[2] D. Conlon, J. Fox, C. Lee, B. Sudakov. Ordered Ramsey numbers, *J. Combin. Theory Ser. B* **122**, 353–383, 2017.

[3] V. Chvátal, V. Rödl, E. Szemerédi, W.T. Trotter Jr. The Ramsey number of a graph with bounded maximum degree, *J. Combin. Theory Ser. B* **34**(3), 239–243, 1983.

Face-free 2-matchings in plane graphs

by Kristóf Bérczi

Given a plane graph $G = (V, E)$, a subset $F \subseteq E$ is a 2-*matching* if $d_F(v) \leq 2$ for each $v \in V$. We call F *face-free* if it does not contain any of the faces as a connected component.

Problem 1. *Given a planar embedding of a graph $G = (V, E)$, find a maximum face-free 2-matching of G.*

Although the problem is probably difficult in general, it would be interesting to see the borderline between the tractable and non-tractable cases. For some reason -that is not well-understood yet- the complexity of some graph optimization problems show a strong connection to discrete convex analysis (more precisely, to whether the underlying structure is a jump system or not). For example, this is the case for restricted 2-matchings: finding a maximum 2-matching not containing cycles of length at most k is NP-complete for $k \geq 5$ and is solvable for $k = 3$ (the case $k = 4$ is open). Meanwhile, the degree sequences of 2-matchings not containing cycles of length at most k form a jump system for $k \leq 4$ and do not form a jump system for $k \geq 5$. There are other examples where a similar phenomenon appears. For the proposed problem, it is not difficult to find a planar graph in which the degree sequences of face-free 2-matchings do not form a jump system. Once again, this does not imply anything regarding the complexity of the problem, but at least calls for caution.

However, it would be enough to say something about maximal planar graphs. The motivation is the following. Given an undirected graph $G = (V, E)$, we call a subset S of vertices a *dominating set* if every vertex of G has at least one neighbour both in S and $V - S$. The *domatic number* of G is the maximum number of pairwise disjoint dominating sets in G. Determining the domatic number is NP-hard. However, Goddard and Henning [1] conjectured the following: *For every triangulated planar graph G on at least 4 vertices, the domatic number is at least 2.*

So let $G = (V, E)$ be a maximal plane graph. The conjecture of Goddard and Henning basically says that G has a bipartite subgraph intersecting the boundary of the wheel of every vertex (given a vertex v, let $v_1, \ldots, v_k$ denote its neighbours in a cyclic order; then the *boundary* consists of the edges $v_1v_2, \ldots, v_kv_1$). Indeed, if you have such a bipartite subgraph then its colour classes are defining a proper coloring.

If we take the dual graph G^*, then it is a simple 2-connected 3-regular plane graph. It is not difficult to see that if you can find a face-free 2-factor in this graph (note that a 2-factor exists by Petersen's theorem), then the corresponding edge set in the original graph form a bipartite graph which intersects the boundary of the wheel of every vertex, and so it is a certificate showing that the domatic number is at least 2. That is, the original conjecture would follow from the following.

Problem 2. *Prove that every 2-connected 3-regular simple plane graph has a face-free 2-factor.*

It is worth mentioning that there is a long list of results on restricted 2-factors in (not necessarily planar) graphs, namely on 2-factors not containing short cycles. The motivation is that this is, in some sense, a relaxation of the Hamiltonian cycle problem.

References

[1] W. Goddard, M. A. Henning. Thoroughly distributed colorings. arXiv preprint arXiv:1609.09684.

Independent sets in tangled grids

by Dömötör Pálvölgyi

A poset P is called an $n \times n$ *tangled grid* if it can be partitioned into chains $A_1, \dots, A_n$, and also into chains $B_1, \dots, B_n$, which have the additional property that $|A_i \cap B_j| \le 1$ for any i, j.

Problem 1. *What is the maximum number $f(n)$ of antichains that can occur in an $n \times n$ tangled grid?*

It was observed in [2] that $f(n)$ also gives an upper bound for the maximum possible number of stable matching among n men and n women. Here the A_i correspond to the men and the B_j to the women of the stable matching, and every intersection corresponds to an operation called *rotation*. In fact, since in each rotation there are at least two-two men and women, some elements of this poset should be contracted, but for an upper bound it will do.

It was proved in [2] that $f(n) \le C^n$ for some large enough C. The goal would be to determine the best possible C, which I conjecture to be 4 (with possibly some polynomial multiplicative factor). This is attained in the (untangled) $n \times n$ grid ordered as a diamond (with a unique smallest and largest element), there the answer is $\binom{2n}{n}$.

A possible approach to bound $f(n)$ is to denote the maximum number of antichains among $n \times n$ tangled grids with m elements by $F(m) = F_n(m)$ (note that $f(n) \le F(n^2)$) and apply the counting argument used also for the famous proof of the Crossing-lemma `https://en.wikipedia.org/wiki/Crossing_number_inequality#Proof`. This, however, doesn't give any good bounds on C. Nevertheless, I sketch it below.

We obviously have $F(m) \le 2^m$. Denote by r the number of $(x, y) \in \binom{P}{2}$ that are in *strict relation*, that is, for which $x <_P y$ and there is no i or j for which $x, y \in A_i$ or $\in B_j$ (i.e., they are not contained in the same chain). It is easy to see that $r \ge m - 2n$. If we keep every chain with probability $p = \frac{3n}{m}$, then the new poset will have $n' = pn$ chains[2], $m' = p^2 m$ elements and $r' = p^4 r$ strict relations. The inequality $r' \ge m' - 2n'$ is equivalent to $p^4 r \ge p^2 m - 2pn$, which gives $r \ge \frac{m^3}{27n^2}$. This means that for some element $p \in P$ is in (strict) relation with at least $\frac{m^2}{27n^2}$ other elements. Depending whether p is a part of the antichain of not, we get $F(m) \le F(m-1) + F(m - 1 - \frac{m^2}{27n^2})$ (using the convexity of F). This is practically the same recursion as the one obtained in [1], which finishes the proof. Unfortunately, the exponent is quite bad, and it has been improved very little, so this approach might not give any good bound.

In [2] they obtain the weaker recursion that some element $p \in P$ is in relation with $\Omega\left(\frac{m^{\frac{3}{2}}}{n^{\frac{3}{2}}}\right)$ other elements, but both from above and below, which gives a simpler but weaker recursion.

References

[1] Ajtai, M.; Chvátal, V.; Newborn, M. M.; Szemerédi, E. (1982), Crossing-free subgraphs, Theory and practice of combinatorics, North-Holland Mathematics Studies, 60, North-Holland, Amsterdam, pp. 9–12.

[2] Anna R. Karlin, Shayan Oveis Gharan, Robbie Weber: A Simply Exponential Upper Bound on the Maximum Number of Stable Matchings. `https://arxiv.org/abs/1711.01032`

[2]Here we cheat a bit, as the number of A_i and B_j chains can differ!

An enumerative analogue of Hall's theorem for bipartite graphs

by Nika Salia

Triesch (1997) [25] conjectured that Hall's classical theorem on matchings in bipartite graphs is a special case of a phenomenon of monotonicity for the number of matchings in such graphs.

Conjecture 1. *Suppose that $G = (V, E)$ and $G' = (V, E')$ are bipartite graphs on the same vertex set V and with the same 2-colouring $V = U \cup W$, where both colour classes U and W contain n vertices. Assume further that for all $A \subset U$ the number of neighbours in G is at least as large as in G',*

$$N_G(A) \geq N_{G'}(A).$$

Then the number of perfect matchings in G is at least as large as in G'.

The most recent work, in my knowledge, [2] settles the problem for 'very' dense graphs.

References

[1] Triesch E., *Conjectures on the uniqueness of Hamiltonian Cycles and the number of perfect matchings*, Martin Klazar Proceedings of the Fourth Prague Combinatorial Workshop, KAM Series **No.97–339,** (1997), 42–46.

[2] Lass B., The algebra of set functions II: An enumerative analogue of Hall's theorem for bipartite graphs, European Journal of Combinatorics, **Volume 33, Issue 2,** (2012), 199–214.

Perfect matching of monotone boolean function

from MathOverflow (Mikaël Monet, ed. by Dömötör Pálvölgyi)

Problem 1. *If for some downwards closed family $\mathcal{F} \subset \{0,1\}^n$ we have $\sum_i f_i(-1)^{|i|} = 0$, where f_i is the number of sets with i elements in $\mathcal{F}$, then is there a perfect matching in $\mathcal{F}$ or in $\{0,1\}^n \setminus \mathcal{F}$?*

The problem is taken from `https://cstheory.stackexchange.com/questions/42626/`, where you can find more comments. For example,

- it is not true that we can guarantee a perfect matching in $\mathcal{F}$,
- monotonicity is needed,
- there are no counterexamples for $n \leq 5$.

Discrete Optimization under Intersection Constraints

by Jannik Matuschke

We consider the following generic optimization problem:

Problem A. We are given a ground set E, two set families $\mathcal{B}_1, \mathcal{B}_2 \subseteq 2^E$, two weight functions $c_1, c_2 \in \mathbb{Q}^E$, and an integer k. Our goal is to find $B_1 \in \mathcal{B}_1$ and $B_2 \in \mathcal{B}_2$ minimizing $\sum_{e \in B_1} c_1(e) + \sum_{e \in B_2} c_2(e)$ under the constraint that $|M_1 \cap M_2| \geq k$.
Variants: Replace "$\geq k$" by "$\leq k$" or "$= k$", respectively.

Frank, Iwata, Peis, and Zenklusen [1] observed that for the case that $\mathcal{B}_1$ and $\mathcal{B}_2$ are base sets of two matroids, the "$\geq$"- and "$\leq$"-variant of the problem can be reduced to a matroid intersection problem. The "$=$"-variant can be solved using a Lagrangian relaxation approach.

Question: What is the complexity of Problem A when E is the edge set of a graph and $\mathcal{B}_1 = \mathcal{B}_2$ is the set of perfect matchings of that graph?

Remark. A special case of the "$\leq$"-variant is determining the existence of two disjoint perfect matchings. This is equivalent to finding a two-factor consisting of even-length cycles, which is NP-hard [2]. However, note that this hardness no longer holds when the graph is bipartite.

References

[1] Á. Frank, S. Iwata, B. Peis, and R. Zenklusen, personal communication at the HIM Follow-up Workshop "Combinatorial Optimization", 2018.

[2] P. Hell, D.G. Kirkpatrick, J. Kratochvíl, and I. Kříž, On restricted two-factors. *SIAM Journal on Discrete Mathematics* 1(4):472–484, 1988.