KAMAK 2024

Erlebachova Bouda, Špindlerův Mlýn

September 22 – 27

Charles University, Prague

Organizers:

David Mikšaník
Robert Šámal
Tung Anh Vu

Brochure of open problems,
Prague, 2024.

Contents

OPEN PROBLEMS

Problem 1. *Planar graph orientation (suggested by Sudatta Bhattacharya)*

Source: Suggested by Ben Moore.

Definitions.
• *The orientation of a graph refers to the process of assigning a direction to the edges of an undirected graph, transforming it into a directed graph (also called a digraph).*

Question: Is it true that every planar graph has an orientation with maximum outdegree 4 and no directed odd cycles.

Related results:
• Every planar graph has an orientation with maximum outdegree 5 with no directed cycles.
• Every planar graph has an orientation with maximum outdegree 3.
• Every planar graph has an orientation with maximum outdegree 4 such that the number of directed odd cycles is not equal to the number of directed even cycles.

Problem 2. *Non-repetitive coloring of Planar graphs (suggested by Sudatta Bhattacharya)*

Source: Proposed by Jaroslaw Grytczuk in 2019.

Definitions.
• *A coloring of a graph G is any function from the set of vertices to some alphabet Σ. The language $L(G)$ of a colored graph G is the set of all words that appear on simple paths of G.*
• *A square is a word of the form XX, where X is any nonempty word.*
• *A coloring of a graph is square-free if $L(G)$ does not contain squares.*
• *The non-repetitive chromatic number of a graph G, denoted by $\pi(G)$ is the least number of colors needed for a square-free coloring of G.*

Example:
$\pi(C_5) = 4$.

Question: Conjectured by Jaroslaw Grytczuk: There is a number k such that every planar graph has a 4-coloring such that $L(G)$ does not contain squares of length greater than k.

Related results:
• If G is a path, then $\pi(G) \leq 3$. This was shown by Axel Thue in 1906.
• Every path graph P has a 2-coloring such that $L(P)$ does not contain squares of length greater than 4.
• For any planar graph G, $11 \leq \pi(G) \leq 768$.

References:
• Nonrepetitive Graph Colouring, by David R. Wood, 2020.
• Planar graphs have bounded nonrepetitive chromatic number by Vida Dujmović, Louis Esperet, Gwenaël Joret, Bartosz Walczak, David R. Wood, 2019.
• Infinite words containing the minimal number of repetitions by Golnaz Badkobeh, 2013.

Problem 3. *Are Various Regular Restrictions of Resolution over Parities the Same? (suggested by Pavel Koblich Dvořák)*

A proof in a propositional proof system starts from a set of clauses Φ, called axioms, that is purportedly unsatisfiable. It generates a proof by deriving the empty clause from the axioms, using inference rules. If we can derive the empty clause from the original set Φ then it proves the set Φ is unsatisfiable.

Resolution over parities (Res($\oplus$)) is a generalization of the standard resolution, using linear clauses (disjunction of linear equations in $\mathbb{F}_2$) to express lines of a proof. It consists of two rules:

Resolution Rule: From linear clauses $A \vee (\ell = 0)$ and $B \vee (\ell = 1)$ derive a linear clause $A \vee B$.

Weakening Rule: From a linear clause A derive a linear clause B that is semantically implied by A (i.e., any assignment satisfying A also satisfies B).

We can represent a Res($\oplus$) refutation Π of a contradiction Φ as a DAG $G(\Pi, \Phi)$, where each node is labeled by a linear clause, sources are labeled by clauses of Φ, and G has exactly one target t and t is labeled by the empty clause. Nodes with two parents represent the application of the resolution rule, and nodes with exactly one parent represent the application of the weakening rule.

If we change the direction of all edges of $G(\Pi, \Phi)$ and "negate" all labels of the nodes $G(\Pi, \Phi)$ we get a structure that is called an *affine DAG G*:

- Each node v is associated with an affine space $A_v \subseteq \mathbb{F}_2^m$.
- For every node v with two children u and w, it holds that $A_u = A_v^0$ and $A_w = A_v^1$, where $A_v^c = \{x \in A_v \mid \langle f_v, x\rangle = c\}$ for a linear query $f_v = \mathbb{F}_2$ and $c \in \{0, 1\}$.
- For every node v with exactly one child u is called a *forget node*, it holds that $A_v \subseteq A_u$.

• For the root r, it holds $A_r = \mathbb{F}_2^n$, where n is the number of variables of Φ.
• Each target t is associated with a cube C_t such that all elements of C_t falsify one clause of Φ.

Thus, the affine DAG G computes a search problem *Search*(Φ) where given an assignment $\alpha \in \{0,1\}^n$ of the variables of Φ, one needs to find a clause of Φ that is falsified by α (such clause always exists as we assume that Φ is a contradiction).

There are several notions of regularity for Res($\oplus$) in the literature [1, 2].

Definitions.
• *Let v be a node of an affine DAG G. Let $Pre(v)$ be the space spanned by all linear functions queried on any path from the source of G to v. Let $Post(v)$ be the space spanned by all linear functions queried on any path from v to any sink of G.*
• *Let Π be a Res($\oplus$) refutation of Φ and G be the affine DAG corresponding to Π.*

• *The refutation Π is* bottom-regular *if for each edge (v, u) of G such that v is a query node holds that $f_v \notin Post(u)$.*

• *The refutation Π is* top-regular *if for each query node v of G, we have $f_v \notin Pre(v)$.*

• *The refutation Π is* strongly regular *if for each query node v of G, we have $Pre(v) \cap Post(v) = \{0\}$.*

Questions:

1. Are the various regular fragments of Res($\oplus$) equivalent, or strongly regular Res($\oplus$) is weaker than bottom-regular or top-regular?

2. Are bottom-regular and top-regular Res($\oplus$) equivalent?

Related results:
• There are several lower bounds for bottom-regular Res($\oplus$) [2, 3, 4],

• There is a lower bound for strongly regular affine DAG for a function (not for a search problem) [1].

References:
[1] S. Gryaznov, P. Pudlák, and N. Talebanfard. *Linear branching programs and directional affine extractors*, CCC '22.
[2] K. Efremenko, M. Garlík, and D. Itsykson. *Lower bounds for regular resolution over parities*, STOC '24.
[3] S. K. Bhattacharya, A. Chattopadhyay, and P. Dvořák. *Exponential Separation Between Powers of Regular and General Resolution Over Parities*, CCC '24.
[4] Y. Alekseev and D. Itsykson. *Lifting to regular resolution over parities via games*, ECCC '24.

Problem 4. *Min/Max label process on vertex copy graphs (suggested by Tomáš Hons)*

Definitions.

• *Fix $p \in [0,1]$. A vertex copy graph on n vertices with copying probability p, denoted by $G_n(p)$, is the following random graph. The graph $G_1(p)$ is the single-vertex graph $(\{v_1\}, \emptyset)$. For $n \geq 1$, the graph $G_n(p)$ is created from $G_{n-1}(p)$ by the steps: (i) pick a vertex v of $G_{n-1}(p)$ uniformly at random, (ii) add a new vertex v_n to $G_{n-1}(p)$, (iii) for each neighbor u of v add the edge $v_n u$ independently at random with probability p, (iv) add the edge $v_n v$. The resulting graph is $G_n(p)$.*

• *Consider a sequence of graphs $G_1, G_2, \ldots$ that are subsequently built when iterating the vertex copy mechanism, i.e. $V(G_n) = \{v_1, \ldots, v_n\}$. Give label 0 the first vertex v_1. When the vertex v_n is introduced in G_n, it gets a new label determined as the minimum of its neighbors' labels (in G_n) increased by 1. This process is called the* MinLabel *process.*

• *If instead of the minimum of neighbors' labels, we take their maximum and increase it by 1, we speak of the* MaxLabel *process.*

• *In both processes, we are interested in the maximum label assigned to a vertex. For a particular vertex copy graph G, we denote by $m(G)$ the maximum label by the* MinLabel *process. Similarly, $M(G)$ stands for the maximal label assigned by the* MaxLabel *process.*

• *For the random vertex copy graph $G_n(p)$, we denote by $m_n(p)$ the expected value of $m(G)$ over all realizations of $G_n(p)$. We analogously define $M_n(p)$ as the expectation of $M(G)$.*

Examples:

• The graph $G_n(1)$ is (deterministically) the clique on n vertices, so $m_n(1) = 1$ and $M_n(1) = n - 1$.

• Each realization G of $G_n(0)$ is a tree. Thus, $m(G) = M(G) = \max\{\operatorname{dist}(v, v_1) : v \in V(G)\}$ for each G. Moreover, the random graph $G_n(0)$ is exactly the Random Rrecursive Tree on n vertices, whose expected depth is known to be $e \cdot \log n$ [1]. That is, $m_n(0) = M_n(0) = e \cdot \log n$.

Question: How does the asymptotics of $m_n(p)$ and $M_n(p)$ depend on p?

Motivation and related results:
• The graphs produces by the vertex copy mechanism are known as *dismantlable graphs*, which are exactly those graphs where a single Cop can capture a Robber in a certain pursuit-evasion game. Thus, they are also known as cop-win graphs [2].
• The greater goal of this project is to examine vertex copy graphs as a useful model of random cop-win graphs with nontrivial distribution of the parameter called *capture time*, which is the maximum time that the Cop needs to capture the Robber on the given graph [3]. It is not difficult to show that the capture time of G can be bounded from below by $m(G)/2$ and from above by $M(G)$.
• By our best knowledge, the only distribution on cop-win graph that was studied is the restriction of Erdős-Rényi model. It turns out that such cop-win graphs have a.a.s. a universal vertex [4]. Therefore, they trivially have the capture time equal to 1, which makes them quite boring from this perspective.

References:

[1] Random recursive tree. (2024, January 9). Wikipedia, the free encyclopedia.
`https://en.wikipedia.org/wiki/Random_recursive_tree`
[2] Cop-win graph. (2023, May 23). Wikipedia, the free encyclopedia. `https://en.wikipedia.org/wiki/Cop-win_graph`
[3] Bonato, A., Golovach, P., Hahn, G., & Kratochvíl, J. (2009). The capture time of a graph. Discrete Mathematics, 309(18), 5588-5595.
[4] Bonato, A., Kemkes, G., & Prałat, P. (2012). Almost all cop-win graphs contain a universal vertex. Discrete Mathematics, 312(10), 1652-1657.

Problem 5. *Maximizing entropy of vertex copy graphs (suggested by Tomáš Hons)*

Definitions.
• *Fix* $p \in [0,1]$. *A vertex copy graph on* n *vertices with copying probability* p*, denoted by* $G_n(p)$*, is the following random graph. The graph* $G_1(p)$ *is the single-vertex graph* $(\{v_1\}, \emptyset)$*. For* $n \geq 1$*, the graph* $G_n(p)$ *is created from* $G_{n-1}(p)$ *by the steps: (i) pick a vertex* v *of* $G_{n-1}(p)$ *uniformly at random, (ii) add a new vertex* v_n *to* $G_{n-1}(p)$*, (iii) for each neighbor* u *of* v *add the edge* $v_n u$ *independently at random with probability* p*, (iv) add the edge* $v_n v$*. The resulting graph is* $G_n(p)$*.*
• *Let* α *be a probability distribution on a set* Ω*. The entropy* $H(\alpha)$ *of* α *is defined as* $-\sum_{x \in \Omega} \alpha(x) \cdot \log_2 \alpha(x)$*.*

Examples:
• The graph $G_n(1)$ is (deterministically) the clique on n vertices, $G_n(0)$ is the Random Recursive Tree on n vertices [1].
• Note that for all $p \in (0,1)$ is the set of possible realizations of $G_n(p)$ the same, but the probabilities differ.

Question: Which $p \in (0,1)$ maximizes the entropy of $G_n(p)$ as n tends to infinity?

Motivation and related results:
• The graphs produces by the vertex copy mechanism are known as *dismantlable graphs*, which are exactly those graphs where a single Cop can capture a Robber in a certain pursuit-evasion game. Thus, they are also known as cop-win graphs [2].
• The greater goal of this project is to examine vertex copy graphs as an useful model of random cop-win graphs. We essentially ask which of the $G_n(p)$ models (for different p) is the most rich.

References:
[1] Random recursive tree. (2024, January 9). Wikipedia, the free encyclopedia.
`https://en.wikipedia.org/wiki/Random_recursive_tree`

[2] Cop-win graph. (2023, May 23). Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Cop-win_graph

Problem 6. *Constructing Verifiable Memory-Hard Functions (suggested by Charlotte Hoffmann)*

Source: Not explicitly proposed by anyone. Got the idea when reading [1].

Definitions.
• *The cumulative memory complexity (CMC) of an algorithm is the sum of its memory consumption at every point in time, i.e., the area under the memory usage curve.*
• *A memory-hard function (MHF) is a function that requires a lot of CMC to evaluate.*
• *A verifiable memory-hard function is an MHF with an output that, possibly given a proof, can be verified with much less CMC than needed to evaluate it.*

Example:
[2] is an example of a memory-hard function.
[1] is a memory-hard function that can be verified using a trapdoor.

Question: Can we construct a memory-hard function that does not require a trapdoor to be verified?

Related results:
• One starting point might be the verifiable delay function (VDF) by Pietrzak [3] since computing the proof of the VDF can be sped up by storing a lot of group elements. Computing the proof of the VDF can therefore be seen as a verifiable MHF with very bad parameters. We could try to improve this.

Bibliography

[1] Auerbach, B., Günther, C.U., Pietrzak, K. (2024). Trapdoor Memory-Hard Functions. In: Joye, M., Leander, G. (eds) Advances in Cryptology – EUROCRYPT 2024. EUROCRYPT 2024. Lecture Notes in Computer Science, vol 14653. Springer, Cham.

[2] Percival, Colin. Stronger key derivation via sequential memory-hard functions. In BSDCan 2009, 2009.

[3] Pietrzak, Krzysztof. Simple Verifiable Delay Functions. IACR Cryptol. ePrint Arch. 2018 (2018): 627.

Problem 7. *On extractability of pairing-based polynomial commitments (suggested by Pavel Hubáček)*

Source: Recent papers by Helger Lipmaa and his coauthors.

Lipmaa, Parisella, and Siim (EUROCRYPT 2024) gave the first meaningful proof for extractability of the KZG polynomial commitment scheme by Kate, Zaverucha, and Goldberg (ASIACRYPT 2010). For their proof, they introduced and used a computational hardness assumption they call the Adaptive Rational Strong Diffie-Hellmann (ARSDH) assumption. However, the ARSDH assumption seems highly tailored to the specifics of the univariate variant of the KZG scheme, and, thus, the proof of extractability from Lipmaa et al. does not apply to other variants of the KZG polynomial commitment that have been proposed in the literature since its introduction.

Question: Can we prove extractability of all the known variants of the KZG polynomial commitment under the ARSDH assumption?

References:

- Helger Lipmaa, Roberto Parisella, Janno Siim: Constant-Size zk-SNARKs in ROM from Falsifiable Assumptions. IACR Cryptol. ePrint Arch. 2024: 173 (2024) https://eprint.iacr.org/2024/173
- Helger Lipmaa, Roberto Parisella, Janno Siim: On Knowledge-Soundness of Plonk in ROM from Falsifiable Assumptions. IACR Cryptol. ePrint Arch. 2024: 994 (2024) https://eprint.iacr.org/2024/994
- Aniket Kate, Gregory M. Zaverucha, Ian Goldberg: Constant-Size Commitments to Polynomials and Their Applications. ASIACRYPT 2010: 177-194

Problem 8. *Condition for independent transversals in tripartite graphs (suggested by Karolína Hylasová)*

Definitions.

• *Let G be a graph with maximum degree Δ whose vertex set is partitioned into parts $V(G) = V_1 \cup \cdots \cup V_n$. A transversal is a subset of $V(G)$ containing exactly one vertex from each part V_i. If this set is also independent, then it is called an independent transversal.*

Question:

It was proved [1] that for bipartite graph $G = (V_A \cup V_B, E)$ in which any vertex in V_A (resp. in V_B) has degree at most D_A (resp. D_B) and for which there is a partition of V that is a refinement of the bipartition $V_A \cup V_B$ such that each part in V_A (resp. V_B) has size at least k_A (resp. k_B) the condition $D_A/k_A + D_B/k_B \leq 1$ is sufficient for the existence of an independent transversal of G.
Is there a similar condition for tripartite graphs?

Related results:

• For $D_A = D_B$, $k_A = k_B$ the condition for the existence of an independent transversal in the bipartite grpah gives the optimal condition proved by Penny Haxell [2]:

Theorem 1. *If G is a graph with maximum degree at most Δ whose vertex set is partitioned into parts $V(G) = V_1 \cup V_2 \cup \cdots \cup V_r$ of size $|V_i| \geq 2\Delta$, then G has an independent transversal.*

References:

[1] Cambie, S., Haxell, P., Kang, R., and Wdowinski, R. A precise condition for independent transversals in bipartite covers. *Proceedings of the 12th European Conference on Combinatorics, Graph Theory and Applications* 9 (2023), pp. 263-269.

[2] Haxell, P. A note on vertex list colouring. *Combinatorics, Probability and Computing* 10 (2001), pp. 345–347.

Problem 9. *Beyond 1-2-3 Conjecture (suggested by Tereza Klimošová)*

Source: Proposed by Przybyło and Woźniak in 2010.

Inspired by a recent proof of the 1-2-3 conjecture I propose to work on one (or more) of its variants. The 1-2-3 conjecture, now theorem, states the following.

Theorem 2 (Keusch'24). *Let $G = (V, E)$ be a graph without connected components isomorphic to K_2. Then there exists an edge-weighting f such that for each edge $\{u, v\}$, $f(\{u, v\}) \in \{1, 2, 3\}$ and*

$$\sum_{w \in N(u)} f(\{u, w\}) \neq \sum_{w \in N(v)} f(\{v, w\}).$$

That is, one can always assign weights from the set $\{1, 2, 3\}$ to the edges of G in such a way, that if you label the vertices of G by the sum of weights of incident edges, you get a proper vertex coloring. Such weighting is called *neighbor distinguishing.*

A *k-total-weighting* of a graph G is an assignment of an integer weight, $f(e), f(v) \in \{1, \dots k\}$ to each edge e and each vertex v of G. Przybyło and Woźniak [4] conjectured the following

Conjecture: *Every simple graph permits a neighbour-distinguishing 2-total-weighting.*

The conjecture holds when G is a 3-colourable, complete or 4-regular graph [4].

The following theorem is actually a simple corollary of Keusch's theorem, but there is a short stand-alone proof of it in the survey [1].

Theorem 3 (Kalkowski'09). *Every graph has a neighbor distinguishing weighting with vertices with weights from the set $\{1, 2\}$ and edges with weights from the set $\{1, 2, 3\}$.*

References:

[1] Jarosław Grytczuk: *From the 1-2-3 conjecture to the Riemann hypothesis*, European Journal of Combinatorics, Volume 91, 2021.
[2] Maciej Kalkowski: *A Note on the 1, 2-Conjecture*, Ph.D. Thesis (2009).
[3] Ralph Keusch: *A solution to the 1-2-3 conjecture*, Journal of Combinatorial Theory, Series B, Volume 166, 2024, 183–202.
[4] Jakub Przybylo, Mariusz Woźniak: *On a 1, 2 Conjecture*, Discrete Mathematics & Theoretical Computer Science, 2010.

Problem 10. *Improving the bounds on* $\mathrm{ex}(n, P_5^{1342})$ *(suggested by Gaurav Kucheriya)*

Source: See the reference below.

Definitions.
• *An edge-ordered graph is a pair* $(G, <)$*, where* $G = (V, E)$ *is a finite simple graph and* $<$ *is a linear order on* E *given by some injective labeling* $L : E \to \mathbb{R}$*. The so obtained edge-ordered graph is denoted by* G^L*.*
• *For a positive integer* n *and an edge-ordered graph* H*, the Turán number of* H *is the maximal number of edges an edge-ordered graph on* n *vertices can have such that it avoids* H*. Let this maximum be denoted by* $\mathrm{ex}(n, H)$*. For a fixed* H *this is a function of* n*, called the extremal number of* H*.*
• *Let* P_k *be the path on* k *vertices. The edge-ordered path* P_k^L *is called a monotone path if the labels increase or decrease monotonically along the path.*

Question: The order of magnitude of the extremal numbers for all 4-edge-ordered paths are known except for $\mathrm{ex}(n, P_5^{1342})$. Gerbner et al. have proved $\mathrm{ex}(n, P_5^{1342} = \Omega(n \log n)$ and $\mathrm{ex}(n, P_5^{1342} = O(n \log^2 n)$. Is $\mathrm{ex}(n, P_5^{1342}) = \Theta(n \log n)$?

Related results:
• Let P be an edge-ordered path with a vertex v that cuts it into two monotone paths P' and P'', such that all labels of P' are smaller than all labels of P''. Then $\mathrm{ex}(n, P) = O(n \log n)$.

References: D. Gerbner, A. Methuku, D. Nagy, D. Pálvölgyi, G. Tardos, M. Vizer. *Turán problems for Edge-ordered graphs.* arXiv:2001.00849 (Preliminary version appeared in Acta Math. Univ. Comenianae, Vol. 88, 3, 717-722 (2019))

Problem 11. *List packing coloring (suggested by David Mikšaník)*

Definitions. *Let G be a graph.*
• *k-list assignment for G is a function $L\colon V(G) \to 2^{\mathbb{N}}$ such that $|L(v)| = k$ for every vertex $v \in V(G)$.*
• *L-coloring of G is a function $\varphi\colon V(G) \to \mathbb{N}$ such that*

(i) $\varphi(v) \in L(v)$ for every vertex $v \in V(G)$ and

(ii) $\varphi(u) \neq \varphi(v)$ for every edge $\{u, v\} \in E(G)$.

• *L-packing of G is a set $\{\varphi_1, \dots, \varphi_k\}$ of mutually disjoint L-colorings of G, that is $\varphi_i(v) \neq \varphi_j(v)$ for every $i \neq j$ and every vertex $v \in V(G)$.*
• *List chromatic number of G, denoted by $\chi_\ell(G)$, is the smallest k such that G admits an L-coloring for every k-list assignment L.*
• *List packing number of G, denoted by $\chi_\ell^\star(G)$, is the smallest k such that G admits an L-packing for every k-list assignment L.*

Examples:
• $\chi_\ell(G) \leq \chi_\ell^\star(G)$
• $\chi_\ell^\star(G) \leq |V(G)|$, and the inequality become an equality if and only if G is a complete graph (no short proof of this fact is known)
• $\chi_\ell^\star(C_n) = 3$ for every $n \geq 3$

Question 1: Is it possible to find a graph G satisfying $\chi_\ell^\star(G) > \chi_\ell(G) + 1$? (Source: [1])

Related results:
• A negative answer would imply a major conjecture in this area, which says that the list packing number is linearly bounded by the list chromatic number [2]. The best known upper bound is exponential [2]. Not only for this reason do we expect a positive answer.

Question 2: Let G be a planar bipartite 3-regular graph. Is it possible to find an L-packing of G for any 3-list assignment L?

(Source: Stijn Cambie at Workshop on Cycles and Colourings 2024, in more generality also in [3])

Related results:
• It should be possible to find at least two disjoint L-colorings of G [personal communication with Stijn Cambie].
• Alon and Tarsi proved that every planar bipartite graph G satisfies $\chi_\ell(G) \leq 3$ [4]. An affirmative answer to Question 2 would generalize the theorem for planar bipartite 3-regular graphs.
• Question 2 remains open if the 3-regularity assumption is omitted [3].

Question 3: Let G be a planar graph and $k \in \{5, 6, 7\}$. Is it possible to find at least $k-3$ disjoint L-colorings of G for any k-list assignment L? (Source: David Mikšaník)

Related results:
• Since $\chi_\ell^\star(G) \leq 8$ [4], it is possible to find 8 disjoint L-colorings for any 8-list assignment L.
• Although there are exponentially many L-colorings of G for any 5-list assignment L [5], we do not know how to find two disjoint L-colorings. For a 6-list assignment L, it is trivial to find two disjoint L-colorings, but we do not know how to find three of them.
• It is an open question whether the upper bound on the list packing number of planar graphs can be improved to 5.

References:
[1] S. Cambie, W. Cames van Batenburg, E. Davies, and R. J. Kang: *List packing number of bounded degree graphs.* `https://arxiv.org/abs/2303.01246` (2023)
[2] S. Cambie, W. Cames van Batenburg, E. Davies, and R. J. Kang: *Packing list-colorings.* Random Structures & Algorithms, Vol. 64 (1), pp. 62–93 (2024)
[3] S. Cambie, W. Cames van Batenburg, and X. Zhu: *Disjoint list-colorings for planar graphs.* `https://arxiv.org/abs/2312.17233` (2023)

[4] N. Alon and M. Tarsi: *Colorings and orientations of graphs.* Combinatorica, Vol. 12 (2), pp. 125–134 (1992)
[5] C. Thomassen: *Exponentially many 5-list-colorings of planar graphs.* Journal of Combinatorial Theory, Series B, Vol. 97 (4), pp. 571–583 (2007)

Problem 12. *Construction of constant gap sequences (suggested by Daniela Opočenská)*

Source: An equivalent problem was introduced by Paul Erdős in the early 1930s under the name *exact covering systems*, more info can be found in [3]. We met constant gap sequences when solving problems in combinatorics on words [1]. An efficient way of construction of constant gap sequences plays an important role there and seems to be an unsolved problem.

Definitions.

- *An* alphabet $\mathcal{A}$ *is a finite set of symbols that are called* letters.
- *A* sequence *over an alphabet* $\mathcal{A}$ *is an infinite sequence of letters from the alphabet, we write* $\mathbf{y} = y_0y_1y_2\cdots$*, where* $y_n \in \mathcal{A}$ *for all* $n \in \mathbb{N}_0$.
- *A sequence* $\mathbf{y} = y_0y_1y_2\cdots$ *is* periodic*, if there exists* $P \in \mathbb{N}$ *such that for all* $n \in \mathbb{N}_0$*,* $y_n = y_{n+P}$*. The smallest possible* P *fulfilling this condition is called the* period*. In this case, we often write* $\mathbf{y} = (y_0y_1y_2\cdots y_{P-1})^\omega$ *to symbolize infinite repetition of the first* P *letters.*
- *A sequence* $\mathbf{y}$ *over an alphabet* $\mathcal{A}$ *is a* constant gap sequence *if for each letter* $i \in \mathcal{A}$ *there is a positive integer denoted by* p_i *such that the distance between any two consecutive occurrences of* i *in* $\mathbf{y}$ *is* p_i.
- *We denote* n_i *the first occurrence of the letter* i *in* $\mathbf{y}$.
- *Let us consider* k *sequences over disjoint alphabets. Their* shuffling *is a sequence obtained when reading step by step their first letters, second letters, etc.*

Example:

- All constant gap sequences are periodic.
- $\mathbf{y} = (0102)^\omega = 01020102\cdots$ is a constant gap sequence, where the letter periods are $p_0 = 2$, $p_1 = 4 = p_2$, the first occurrences are $n_0 = 0$, $n_1 = 1$, $n_2 = 3$ and the period is 4.
- $\mathbf{v} = (0122)^\omega = 012201220122\cdots$ is periodic but not constant gap because of the occurences of 2.

• The constant gap sequence can be described by a list of pairs (n_i, p_i) for all $i \in \mathcal{A}$. For example, $\mathbf{y} = (010203)^\omega$ can be rewritten as $(0, 2)$, $(1, 6)$, $(3, 6)$, $(5, 6)$.
• We can shuffle the constant gap sequences $\mathbf{u} = (0)^\omega$, $\mathbf{v} = (12)^\omega$, $\mathbf{w} = (3435)^\omega$ to obtain a sequence over the alphabet $\mathcal{A} = \{0, 1, 2, 3, 4, 5\}$ in the form

$$\mathbf{y} = (013024013025)^\omega.$$

Question: Find an algorithm to generate all constant-gap sequences over a given alphabet such that it works well for alphabets with at least 13 letters.

Related results:
• Let $\mathbf{y}$ be a constant gap sequence over an alphabet $\mathcal{A} = \{0, 1, 2, 3, \ldots, d-1\}$. Then

$$\sum_{i=0}^{d-1} \frac{1}{p_i} = 1.$$

• Let $n, n' \in \mathbb{N}_0$, and $p, p' \in \mathbb{N}$. We say that (n, p) and (n', p') are *in collision* if $\gcd(p, p')$ divides $n - n'$, Let the pairs $(n_0, p_0), (n_1, p_1), \ldots, (n_{d-1}, p_{d-1})$ satisfy $n_i < p_i$ for all $i \in \mathcal{A}$, Then the pairs $(n_0, p_0), (n_1, p_1), \ldots, (n_{d-1}, p_{d-1})$ give rise to a constant gap sequence iff no two of them are in collision and

$$\sum_{i=0}^{d-1} \frac{1}{p_i} = 1.$$

• The period of the letter i in the shuffling of k sequences equals $k \cdot \hat{p}_i$, where $\hat{p}_i$ is the period of the same letter in its original sequence.

Moreover, if there exists $k \in \mathbb{N}, k > 1$, such that k divides p_i for all $i \in \mathcal{A}$, then the constant gap sequence $\mathbf{y} = y_0 y_1 y_2 \cdots$ is the shuffling of k constant gap sequences in the form

$$\begin{aligned}
&y_0 y_k y_{2k} y_{3k} \cdots \\
&y_1 y_{k+1} y_{2k+1} y_{3k+1} \cdots \\
&y_2 y_{k+2} y_{2k+2} y_{3k+2} \cdots \\
&\vdots \\
&y_{k-1} y_{2k-1} y_{3k-1} y_{4k-1} \cdots
\end{aligned}$$

Therefore a constant gap sequence over a d-letter alphabet that cannot be generated by shuffling has $\gcd(p_0, p_1, \ldots, p_{d-1}) = 1$. The smallest d such that there exists a d-letter constant gap sequence, which is not obtained by shuffling, is 13. The sequence is

$$(021364051782031495061 2A30415BC)^\omega$$

which can be rewritten to offsets and periods as

letter	0	1	2	3	4	5	6	7	8	9	A	B	C
n_i	0	2	1	3	5	7	4	9	10	16	22	28	29
p_i	6	6	10	10	10	10	15	30	30	30	30	30	30

• A brute force search algorithm to find all constant gap sequences over a d-letter alphabet, where d is user input, is described in [4]. The algorithm runs well on standard user computer up to $d = 12$, but the computational complexity becomes too much for bigger alphabets.
• This problem is equivalent to a problem of finding an exact covering system with a given number of equivalence classes. The *exact covering system* is a system of congruence classes in the form

$$n \equiv n_i \mod p_i, \qquad i = 1, 2, 3, \ldots, k$$

such that any $n \in \mathbb{N}_0$ belongs to exactly one congruence class.

The duality is as follows: There exists a constant gap sequence defined by pairs $(n_0, p_0), (n_1, p_1), \ldots, (n_{d-1}, p_{d-1})$, if and only if

$$n \equiv n_i \mod p_i, \qquad i = 0, 1, \ldots, d-1$$

is an exact covering system. There are several theoretical results concerning covering systems, some of them can be found in [2, 5, 6]

References:

[1] L. Dvořáková, D. Opočenská, E. Pelantová, *Asymptotic repetitive threshold of balanced sequences*, Mathematics of Computation, Vol. 92, (2023), pp . 1403–1429.

[2] J. Fabrykowski, T. Smotzer, *Covering Systems of Congruences*, Mathematics Magazine, (2005), 78:3, pp. 228-231

[3] R. K. Guy, *Unsolved Problems in Number Theory*, Springer Science & Business Media, (2004), pp. 386–390

[4] A. Kasalová, *Konstrukce slov s konstantními mezerami* (Czech). Rozhledy matematicko-fyzikální, vol. 97 (2022), issue 3, pp. 1-12

[5] Š. Porubský, *Generalization of Some Results for Exactly Covering Systems*, Matematický časopis, Vol. 22 (1972), No. 3, 208–214

[6] Š. Porubský, J. Schönheim, *Old and new necessary and sufficient conditions on* (a_i, m_i) *in order that* $n \equiv a_i (\mathrm{mod}\ m_i)$ *be a covering system*, Mathematica Slovaca, Vol. 53 (2003), No. 4, 341–349

Problem 13. *Graphs with full-support equilibrium (suggested by Lluís Sabater Rojas)*

Source: Proposed[1] by David Kempe et al. in 2013 in the context of equilibria of opinion diffusion processes.

Definitions.
• *Given a vector* $w \in \mathbb{R}^n$, *the* mass *at vertex* i *is the sum of weights in the closed neighborhood of* i, *that is,* $w(N[i]) = \sum_{j \in N[i]} w(j)$.
• *A graph* G *is* infeasible *if the system* $(A+I)w = 1$, $w > 0$ *has no solution. Where* A *is the adjacency matrix of* G *and* $w > 0$ *means that every entry of* w *is positive. We call a solution to this system a* full-support equilibrium.

Example:
Some graphs can not have a full-support equilibrium.

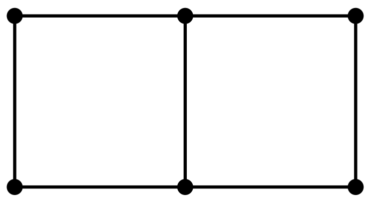

Figure 1: Example of an infeasible graph.

Question: Is there any characterization for infeasible graphs?

[1]It was not proposed but there was no answer to the case with $\alpha = 1$ in the local model, which is equivalent to our problem.

Related results:

• Restatement: a full-support equilibria must satisfy $w(N[u])$
$= 1 \quad \forall\, u \in V$

• Necessary (not sufficient) condition: $N[u] \not\subset N[v] \quad \forall\, (u, v) \in E$

– There is no full-support equilibrium for graphs that have leaves.

• There always are full-support equilibria for k-regular graphs (cycles, complete graphs...) and complete bipartite graphs (except star graphs).

• There are graphs with multiple full-support equilibria (e.g. K_n, C_{3n}).

References:

David Kempe, Jon Kleinberg, Sigal Oren, and Aleksandrs Slivkins. Selection and influence in cultural dynamics. https://doi.org/10.1145/2482540.2482566

Problem 14. *Improper (1,1)-coloring of planar graphs of girth 6 (suggested by Felix Schröder)*

Definitions.
• *The girth of a graph is the minimum length of any cycle in the graph.*
• *An improper (a, b)-coloring is a partition of the vertices of the graph into two subgraphs A and B, such that the maximum degree is of A is a and the maximum degree of B is b.*

Example: The result cannot be generalized to girth 5, as those graphs are not even (3,1)-colorable. For a counterexample, consider a P_3, each of whose vertices connects to all vertices of a P_3 via 7 copies of the gadget below.

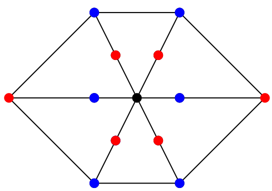

Figure 2: If the left- and rightmost vertex are in A (red) and have 3 neighbors in A outside the gadget, then this gadget cannot be (3,1)-colored.

Question: Is every planar graph of girth 6 improperly (1,1)-colorable?

Related results:
• Planar graphs of girth 7 have maximum average degree $\frac{14}{5}$ and are therefore $(1, 1)$-colorable by a result of Borodin, Kostochka and Yancey [1]. The bound on the maximum average degree is tight.

• A recent result shows that (1,1)-colorable graphs of girth 4 have a linear size universal point set. This is a set of $2n - 2$ points in the plane, such that every n-vertex graph of that class can be embedded using points from the set and straight-line edges.

References:

• On 1-improper 2-coloring of sparse graphs, Borodin, Kostochka and Yancey, Discrete Mathematics 313 (2013)

• Near-colorings: non-colorable graphs and NP-completeness, Montassier and Ochem, Electronic Journal of Combinatorics 22 (2015)

• Linear Size Universal Point Sets for Classes of Planar Graphs, Felsner, Schrezenmaier, S, Steiner , SOCG 2023 Proceedings, Dagstuhl

Problem 15. *Sunflower Recognition of Interval Graphs (suggested by Peter Stumpf)*

Source: Proposed by me in my PhD thesis this year.

Definitions.
• *A* sunflower (simultaneous) graph *is a graph where each vertex has a label in* $\{0, 1, \ldots, k\}$ *for some* $k \in \mathbb{N}$. *We say that a sunflower graph* $\mathcal{G}$ encodes *the graphs* $G_1, \ldots G_k$ *where for* $i \in \{1, \ldots, k\}$ *graph* G_i *is the subgraph of* $\mathcal{G}$ *induced by the vertices labeled with 0 or* i. *We call the subgraph induced by the vertices labeled with* 0 *the* shared graph S.
• *An* intersection graph *for a family* $\mathcal{F}$ *of geometric objects is a graph where each vertex can be assigned a geometric object of* $\mathcal{F}$ *such that the assigned objects have a non-empty intersection.*
• Interval graphs *are intersection graphs of intervals. That is, an* interval graph *is a graph* $G = (V, E)$ *where each vertex* v *can be assigned an interval* I_v *such that two vertices* u, v *are adjacent if and only if* $I_v \cap I_u \neq \emptyset$. *We call such an assignment an* interval representation *of* G.

Question: Given a sunflower simultaneous graph $\mathcal{G}$, is it possible to decide in linear time whether the graphs $G_1, \ldots, G_k$ encoded by $\mathcal{G}$ are all interval graphs?

Related results:
• In the *simultaneous representation problem* for interval graphs the question is whether for multiple input graphs $G_1, \ldots, G_k$ (that may share subgraphs) there are interval representations such that each vertex is assigned the same interval in each interval representation (Jampani, Lubiw; JGAA, 2012). The most studied case is the sunflower case where the shared subgraph is the same for any pair of the given graphs. While for many intersection graph classes (including interval graphs (Rutter, Stumpf; ESA 2023)) linear-time algorithms for the sunflower case are known, these algorithms assume that the input graphs are given separately. However, by encoding the input

graphs with a sunflower simultaneous graph, faster times might be possible if the shared graph S is relatively large. This motivates the given question as a stepping stone.

• Interval graphs can be characterized by having valid clique orderings – orderings of their maximal cliques where for each vertex v the cliques containing v are consecutive. Interval graphs can be recognized by using *PQ-trees* to check for the existence of such a clique ordering (for this one uses that interval graphs are chordal and thus a list of the maximal cliques can be efficently computed) (Booth, Lueker; JCSS, 1976).

Korte and Möhring introduced modified PQ-trees which can be constructed incrementally, by adding the vertices of the graph one by one (Korte, Möhring; SICOMP 1989). If modified PQ-trees can be made to some extend persistent, such that the PQ-tree for the shared graph can be reused for each input graph, this would support a positive answer.

References.

• Kellogg S. Booth and George S. Lueker: Testing for the Consecutive Ones Property, Interval Graphs, and Graph Planarity Using PQ-Tree Algorithms, JCSS; 1976

• Krishnam Raju Jampani and Anna Lubiw: The Simultaneous Representation Problem for Chordal, Comparability, JGAA; 2012

• Ignaz Rutter and Peter Stumpf: Simultaneous Representation of Interval Graphs in the Sunflower Case, ESA; 2023

• Norbert Korte and Rolf H. Möhring: An Incremental Linear-Time Algorithm for Recognizing Interval Graphs, SICOMP; 1989

Problem 16. *Flow reconfiguration (suggested by Robert Šámal)*

Source: Dvořák and Feghali

Question: Given two nowhere-zero 6-flows on a graph, is one always a reconfiguration of the other?

• A flow in a digraph G is a mapping $\varphi : E(G) \to A$ (for an abelian group A) such that at every vertex, flow in equals flow out, that is for every v

$$\sum_{u \in N^+(v)} \varphi(vu) = \sum_{u \in N^-(v)} \varphi(uv).$$

All flows we care about are nowhere-zero (nz): $\varphi(e) \neq 0$ for all $e \in E(G)$.
• A single-step reconfiguration is a change from φ to $\varphi' = \varphi + \varphi_K$, where K is any cycle in the undirected graph G and φ_K a flow having values ± 1 along the cycle K and 0 elsewhere. (Both φ and φ' are nz flows.)
• A flow φ can be reconfigured to ψ if there is a sequence of singe-step reconfigurations that transform φ to ψ (using only nz flows in the process).
• A dual version to this is coloring reconfiguration: changing a color of one vertex at a time.

One question is listed above. Others:
• Given a graph and an abelian group A, find an efficient algorithm to decide, if one given nz flow can be reconfigured to another given nz flow.
• Given a graph and an abelian group A, can any nz flow be reconfigured to any other?
• Estimate how many steps are needed for the reconfiguration.

Generalize to flows with a given (nonzero) boundary.

References: [DF] Z. Dvořák, C. Feghali: Flow reconfiguration, unpublished manuscript

Problem 17. *Shortest coordinated paths for two robots (suggested by Tung Anh Vu)*

Let A and B be two unit-disc robots ("roombas") moving among polygonal obstacles $\mathcal{O}$ in the plane. We specify the configuration of each robot by the coordinates of its center: When robot R is placed with its center at the point $p \in \mathbb{R}^2$, we let $R(p)$ denote $\{x \in \mathbb{R}^2 \mid \|x - p\| < 1\}$. Our goal is to plan a collision-free motion for the two robots from given free start configurations marked A_0 and B_0, to given free target configurations A_1 and B_1. Throughout the motion the robots should not collide with the obstacles, nor with one another. In a valid plan, robots may touch but they should not overlap each other or touch obstacles.

Question 1: Give an algorithm which produces a pair of roomba paths that minimizes the sum of paths or prove hardness of doing so. (Source: [Abrahamsen, Halperin: Ten problems in geobotics; arXiV 2024])

Related results:

- [Sharir, Sifrony; AMAI 1991] give an algorithm for producing feasible paths (not necessarily optimal) in time $\mathcal{O}(n^2)$.
- The case of no obstacles was not settled until recently by [Kirkpatrick, Liu; CCCG 2016].
- A related problem exists where instead of round roombas we have square roombas. Again, it is not known whether the problem is in P or whether it is NP-hard. But [Agarwal, Halperin, Sharir, Steiger; SODA 2024] give a PTAS for the problem.
- Additional measures of optimality exist aside from sum of paths, e.g. makespan.

Problem 18. *Reconstructing your string (suggested by Hadi Zamani)*

Source: Problem 92 from Ben Green's book "100 Open Problems".

I have a string $x \in \{0,1\}^n$. Let $\tilde{x}$ be the random string obtained by deleting bits from x independently at random with probability $\frac{1}{2}$ (thus, for example, if $n = 8$ and $x = 00110110$, it might be the case that $\tilde{x} = 0111$, generated by deleting bits $2, 3, 5, 8$.) An instance of $\tilde{x}$ is called a 'trace'. How many independent traces $\tilde{x}_1, \ldots, \tilde{x}_m$ are needed before one can reconstruct x with probability 0.9 ?

References:

- **Best known Upper bound:**
  Z. Chase, New upper bounds for trace reconstruction.

- **Best known Lower bound:**
  N. Holden and R. Lyons, Lower bounds for trace reconstruction, Ann. Appl. Probab. 30 (2020), no. 2, 503–525. (see also Erratum to 'Lower bounds for trace reconstruction', Ann. Appl. Probab. 32 (2022), no. 4, 3201–3203.