Spring School 2017

Dušan Knop, Tomáš Masařík, Veronika Slívová (eds.)

Preface

Spring school on Combinatorics has been a traditional meeting organized for faculty and students participating in the Combinatorial Seminar at Charles University for over 30 years. It is internationally known and regularly visited by students, postdocs and teachers from our cooperating institutions in the DIMATIA network. As it has been the case for several years, this Spring School is generously supported by Computer Science Institute (IÚUK) of Charles University, the Department of Applied Mathematics (KAM), and Center of Excellence — Institute for Theoretical Computer Science (ITI) of Charles University.

The Spring Schools are entirely organized and arranged by our students (mostly undergraduates). The lecture subjects are selected by supervisors from the Department of Applied Mathematics (KAM) and Computer Science Institute (IÚUK) of Charles University as well as from other participating institutions. In contrast, the lectures themselves are almost exclusively given by students, both undergraduate and graduate. This leads to a unique atmosphere of the meeting which helps the students in further studies and their scientific orientation.

This year the Spring School is organized in Sklené u Fryšavy, a mountain village in Žďárské vrchy in middle part of Czech republic with a great variety of possibilities for outdoor activities.

Ondřej Pangrác, Robert Šámal, Martin Tancer

Contents

Dependent random choice: Introduction: Dependent random choice (Zdeněk Dvořák) 7

Dependent random choice: Dependent random choice (Tomáš Toufar) 8

Network Coding: Fundamental Theorem of Network Coding (Veronika Slívová) 10

Network Coding: Constructing Network Coding (Karel Král) 11

Beyond trilateration: On the localizability of wireless ad-hoc networks (Onur Çağırıcı) . . . 12

Edge-coloring of 3-uniform hypergraphs (Radovan Červený) 13

The topology of competitively constructed graphs (Andrej Dedík) 15

Integer 4-flows and cycle covers (Anna Dresslerová) 16

The communication complexity of addition (Pavel Dvořák) 18

Parameterized Approximations of Symmetric and Planar Directed Steiner Networks (Andreas Feldmann) 20

Highway Dimension and Provably Efficient Shortest Path Algorithms (Tomáš Gavenčiak) . . 21

De-Bruijn-Erdős-type theorems for graphs and posets (Jaroslav Hančl) 23

Euler's polyhedral formula (Petr Hliněný) 24

Approximating bounded-degree spanning trees and connected factors with leaves (Dušan Knop) 25

On hardness of some planar SAT problems (Tomáš Masařík) 27

Cops and Robbers ordinals of cop-win trees (Ondřej Mička) 28

Fractional triangle decompositions in graphs with large minimum degree (Jana Novotná) . . 30

Contraction Hierarchies & Hub Labeling (Jitka Novotná) 31

A Short Proof That χ Can be Bounded ε Away from $\Delta + 1$ toward ω (Petra Pelikánová) . . 33

Searching for knights and spies: A majority/minority game (Jakub Svoboda) 34

Combinatorial species and graph enumeration (Josef Svoboda) 36

On the Shortest Path Game (Jana Syrovátková) 38

Note on terminal-pairability in complete grid graphs (Jakub Tětek) 40

Schedule of talks

	9:00 – 10:30	10:30 – 12:00	12:00
Saturday *room A* *chair: Jakub Svoboda*	The topology of competitively constructed graphs Andrej Dedík 9:00 – 10:30 15	Beyond Trilateration: On the Localizability of Wireless Ad-hoc Networks Onur Çağırıcı 10:30 – 12:00 12	Lunch 12:00 – 13:00
Sunday *room A* *chair: Milan Hladík*	Fundamental Theorem of Network Coding: *Network Coding* Veronika Slívová 9:00 – 10:30 10	Constructing Network Coding: *Network Coding* Karel Král 10:30 – 12:00 11	Lunch 12:00 – 13:00
Sunday *room B* *chair: Tomáš Masařík*	Combinatorial species and graph enumeration Josef Svoboda 9:00 – 10:30 36	Fractional triangle decompositions in graphs with large minimum degree Jana Novotná 10:30 – 12:00 30	
Monday *room A* *chair: Veronika Slívová*	Highway Dimension and Provably Efficient Shortest Path Algorithms Tomáš Gavenčiak 9:00 – 10:30 21	Highway dimension, shortest paths, and provably efficient algorithms Jitka Novotná 10:30 – 12:00 31	Lunch 12:00 – 13:00
Tuesday *room A* *chair: Jaroslav Hančl*	Integer 4-flows and cycle covers Anna Dresslerová 9:00 – 10:30 16	Edge-coloring of 3-uniform hypergraphs Radovan Červený 10:30 – 12:00 13	Lunch 12:00 – 13:00
Wednesday	Trip Day 9:00 –		
Thursday *room A* *chair: Robert Lukoťka*	The communication complexity of addition Pavel Dvořák 9:00 – 10:30 18	Introduction: *Dependent random choice* Zdeněk Dvořák 10:30 – 12:00 7	Lunch 12:00 – 13:00
Friday *room A* *chair: Ondřej Pangrác*	A short Proof that χ Can be Bounded ε Away from $\Delta + 1$ toward ω Petra Pelikánová 19:00 – 20:30 33	Approximating bounded-degree spanning trees and connected factors with leaves Dušan Knop 10:30 – 12:00 25	Lunch 12:00 – 13:00

	18:00		19:00	20:00	21:00
Friday *room A* *chair: Karel Král*	Arrival – 18:00	Dinner 18:00 – 19:00	Searching for knights and spies: A majority/minority game Jakub Svoboda 19:00 – 20:30 34		
Saturday *room A* *chair: Tereza Klimošová*	The Code Game – 19:00		Dinner 19:00 – 19:40	Euler's polyhedral formula Petr Hliněný 19:40 – 20:30 24	On hardness of some SAT problems Tomáš Masařík 20:30 – 21:30 27
Sunday *room A* *chair: Andreas Feldmann*		Dinner 18:00 – 19:00	Cops and Robbers ordinals of cop-win trees Ondřej Mička 19:00 – 20:30 28		
Monday *room A* *chair: Tomáš Gavenčiak*		Dinner 18:00 – 19:00	Parameterized Approximations of Symmetric and Planar Directed Steiner Networks Andreas Feldmann 19:00 – 20:30 20		
Tuesday *room A* *chair: Jana Syrovátková*		Dinner 18:00 – 19:00	De-Bruijn-Erdős-type theorems for graphs and posets Jaroslav Hančl 19:00 – 20:30 23		
Wednesday *room A* *chair: Tomáš Toufar*	Trip Day – 18:00	Dinner 18:00 – 19:00	Note on terminal-pairability in complete grid graphs Jakub Tětek 19:00 – 20:30 40		
Thursday *room A* *chair: Pavel Dvořák*		Dinner 18:00 – 19:00	Dependent random choice II. *Dependent random choice* Tomáš Toufar 19:00 – 20:30 8		
Thursday *room B* *chair: Jitka Novotná*			On the Shortest Path Game Jana Syrovátková 9:00 – 10:30 38		

Typical day

8:00 – 9:00	Breakfast
9:00 – 10:30	Talk 1
10:30 – 12:00	Talk 2
12:00 – 13:00	Lunch
13:00 – 18:00	Free time
18:00 – 19:00	Dinner
19:00 – 20:30	Talk 3

Series Talks

Zdeněk Dvořák
rakdver@iuuk.mff.cuni.cz

Introduction: Dependent random choice
as part of a serie Dependent random choice

Abstract

We give a brief introduction into extremal graph theory, focusing on the problem of determining the extremal function for bipartite graphs. In this context, we note the usefulness of finding (in a sufficiently dense graph) a set of vertices such that each small tuple of them has many common neighbors. We introduce the basic idea of dependent random choice method and use it to show a bound on the extremal function for 1-subdivision of multigraphs.

Tomáš Toufar
toufi@iuuk.mff.cuni.cz
Presented paper by Jacob Fox, Benny Sudakov

Dependent random choice *as part of a serie* Dependent random choice

(https://arxiv.org/pdf/0909.3271.pdf)

Introduction

In this talk we present a simple yet surprisingly useful probabilistic technique which shows that in a dense graph there exists a large subset of vertices in which all small subsets have many common neighbors.

Recently, this technique had several applications in Extremal Graph Theory, Ramsey Theory, Combinatorial Geometry, and Additive Combinatorics. We discuss some of the applications in Extremal graph theory and Ramsey Theory.

Notation

By $N(v)$ we denote *neighborhood of* v (the set $\{u : uv \in E\}$). By $N(U)$ we denote the *common neighbors of* U, i.e., the set

$$N(U) = \bigcap_{v \in U} N(v).$$

The *Turán number* $\mathrm{ex}(n, H)$ denotes the maximum number of edges of a graph with n vertices that does not contain H as a subgraph.

The *Ramsey number* $r(H)$ denotes the smallest positive integer N such that every 2-coloring of K_N contains a monochromatic copy of a graph H.

Key lemma

In Extremal Graph theory, we often want to embed small or sparse graph into a dense graph. To obtain such an embedding, it is useful to have a large vertex subset U in which all small subsets have many common neighbors. We can then use U to greedily embed the desired subgraph.

The condition for existence of such a subset is established by the next lemma.

Lemma 1 (Key lemma) *Let n, d, a, r, m be positive integers. Let $G = (V, E)$ be a graph on n vertices with average density $d = 2|E|/|V|$. If there exists a positive integer t such that*

$$\frac{d^t}{n^{t-1}} - \binom{n}{r}\left(\frac{m}{n}\right)^t \geq a,$$

then G contains a subset $U \subseteq V$ of size at least a such that every r vertices in U have at least m common neighbors.

Turán number of bipartite graphs

The central problem in Extremal Graph Theory is to determine or estimate $\mathrm{ex}(n, H)$. A classical result by Turán determines the $\mathrm{ex}(n, H)$ when H is a complete graph. Another well-known result by Erdõs, Stone, and Simonovits determines the asymptotic behavior for graphs of chromatic number at least three.

However, for bipartite graph estimating $\mathrm{ex}(n, H)$ is more complicated; there are only few nontrivial bipartite graph for which the order of magnitude of $\mathrm{ex}(n, H)$ is known. The following result gives an upper bound on $\mathrm{ex}(n, H)$ for H having small degree in one partite; the result is best possible for every fixed r.

Theorem 2 *If $H = (A \cup B, F)$ is a bipartite graph in which every vertex in B has degree at most r, then $\mathrm{ex}(n, H) \le cn^{2-1/r}$, where the constant c depends on H only.*

Degenerate graphs

To deal with degenerate graphs we need the following twist of the Key lemma.

Lemma 3 (Two-sided variant of key lemma) *Let r, s be integers with $r, s \ge 2$ and let $G = (V, E)$ be a graph with N vertices with at least $N^{2-1/(s^3 r)}$ edges. The graph G contains subsets $U_1, U_2 \subseteq V$ such that, for $k \in \{1, 2\}$, every r-tuple in U_k has at least $N^{1-1.8/s}$ common neighbors in U_{3-k}.*

The next result has two quick corollaries in Ramsey theory and Extremal Graph theory.

Theorem 4 *Let $r, s \ge 2$ and let G be a graph with N vertices and at least $N^{2-1/(s^3 r)}$ edges. Then the graph G contains every r-degenerate bipartite graph with $N^{1-1.8/s}$ vertices.*

Erdõs and Burr conjectured that for every r there is a constant c_r such that for every r-degenerate graph H on n vertices we have $r(H) \le c_r n$. We show a nearly linear upper bound for degenerate bipartite graphs.

Corollary 5 *The Ramsey number of every r-degenerate bipartite graph H on n vertices satisfies*

$$r(H) \le 2^{8r^{1/3}(\log n)^{2/3}} n$$

for n sufficiently large.

Erdõs conjectured that $\mathrm{ex}(n, H) \in O(n^{2-1/r})$ for every r-degenerate bipartite graph. As a corollary of Theorem 4, we obtain a slightly weaker bound.

Corollary 6 *Let H be an r-degenerate bipartite graph on h vertices and let n be an integer satisfying $n > h^{10}$. Then*

$$\mathrm{ex}(n, H) < n^{2-\frac{1}{8r}}.$$

Veronika Slívová
slivova@iuuk.mff.cuni.cz
Presented paper by Christina Fragouli and Emina Soljanin

Fundamental Theorem of Network Coding *as part of a serie* Network Coding

(http://ect.bell-labs.com/who/emina/papers/NCF.pdf)

Network coding is a novel technique improvng network throughput and performance. Contrary to the previous approach (network flows) network coding allows the internal nodes to not only pass the information but also to process it. The fundamental theorem of network coding says that we can broadcast all information from a source to all receivers if and only if each receiver can get all the information.

Definition 1 *A* cut *between S and R is a set of graph edges whose removal disconnects S from R. A* min-cut *is a cut with the minimal value. The* value *of the cut is the sum of the capacities of the edges in the cut.*

Theorem 2 ([1]) *Consider a graph $G = (V, E)$ with unit capacity edges, a source vertex X and a receiver vertex R. If the min-cut between S and R equals h, then the information can be send from S to R at a maximum rate of h. Equivalently, there exists exactly h edge-disjoint paths between S and R.*

Theorem 3 (Main NC Thm [1]) *Consider a directed acyclic graph $G = (V, E)$ with unit capacity edges, h unit rate sources located on the same vertex of the graph and N receivers. Assume that the value of the min-cut to each receiver is h. Then there exists a multicast transmission scheme over a large enough finite field $\mathbb{F}_q$, in which intermediate network nodes linearly combine their incoming information symbols over $\mathbb{F}_q$, that delivers the information from the sources simultaneously to each receiver at a rate equal to h.*

Definition 4 *The* local coding *vector $c^\ell(e)$ associated with an edge e is the vector of coefficients over $\mathbb{F}_q$ with which we multiply the incoming symbols to edge e. The dimension of $c^\ell(e)$ is $1 \times |In(e)|$, where In(e) is the set of incoming edges to the parent node of e.*

Definition 5 *The* global coding vector *$c(e)$ associated with an edge e is the vector of coefficients of the source symbols that flow (linearly combined) through edge e. The dimension of $c(e)$ is $1 \times h$.*

Theorem 6 (Algebraic version of the Main Thm [1]) *In linear network coding, there exist values in some large enough finite field $\mathbb{F}_q$ for the components $\{\alpha_k\}$ of the local coding vectors, such that all matrices A_j, $1 \le j \le N$, defining the information that the receivers observe, are full rank.*

Lemma 7 (Sparse Zeros Lemma [1]) *Let $f(\alpha_1, \ldots, \alpha_\eta)$ be a polynomial in variables $\alpha_1, \ldots, \alpha_\eta$, with maximum degree in each variable of at most d. Then, in every finite field $\mathbb{F}_q$ of size $q > d$ on which $f(\alpha_1, \ldots, \alpha_\eta)$ is not identically equal to zero, there exist values $p_1, \ldots, p_\eta$, such that $f(\alpha_1 = p_1, \ldots, \alpha_\eta = p_\eta) \neq 0$.*

Bibliography

[1] Christina Fragouli, and Emina Soljanin. *Network Coding Fundamentals.* The Essence of Knowledge, 2007.

Karel Král
kralka@iuuk.mff.cuni.cz

Constructing Network Coding *as part of a serie* Network Coding

We are going to describe the Linear Information Flow Algorithm which for a known network produces a network flow in polynomial time.

Definition 1 Coding points *are the edges of the graph G where we need to perform network coding operations.*

Lemma 2 *Consider a coding point δ with $m \leq h$ parents and a receiver $R_j \in R(\delta)$. Let $V(\delta)$ be the m-dimensional space spanned by the coding vectors of the parents of δ, and $V(R_j, \delta)$ be the $(h-1)$-dimensional space spanned by the elements of B_j after removing $c(f^j_{\leftarrow}(\delta))$. Then $\dim\{V(\delta) \cap V(R_j, \delta)\} = m - 1$. Where $f^j_{\leftarrow}(\delta)$ denotes the predecessor coding point to δ along the source-target path (S_i, R_j) and c denotes the coding vector.*

Theorem 3 *The LIF algorithm identifies a valid network code using any alphabet $\mathbb{F}_q$ of size $q > N$.*

Lemma 4 *A randomly selected coding vector $c(\delta_k)$ at step k of the LIF preserves the multicast property with probability at least $1 - N/q$.*

Observation 5 *The LIF algorithm can be implemented to run in time $\mathcal{O}(|E|Nh^2)$.*

The alphabet size (the sizes of the underlying field size) is an important network code characteristic, as it corresponds to the packet size and thus computation time needed in each node. It is known that if we do require the smallest possible size we have to solve an NP-complete problem. We are going to discuss that large alphabet sizes are necessary for some graphs.

Definition 6 *Let f_i, f_j be functions mapping Σ^2 to Σ. We say f_i, f_j are* independent *iff there does not exist distinct points (α_1, β_1) and (α_2, β_2) in Σ^2 such that $f_i(\alpha_1, \beta_1) = f_i(\alpha_2, \beta_2)$ and $f_j(\alpha_1, \beta_1) = f_j(\alpha_2, \beta_2)$.*

Lemma 7 ([2]) *If $f_1, \ldots, f_n$ are pairwise independent functions of the form $f_i \colon \Sigma^2 \to \Sigma$, then $n \leq q$. Where q denotes the alphabet size $|\Sigma|$.*

Theorem 8 *There exist solvable multicast information flow problems that require an alphabet of size $\Omega(\sqrt{n})$, even if nonlinear network codes are permitted.*

Bibliography

[1] Christina Fragouli, and Emina Soljanin. *Network Coding Fundamentals.* The Essence of Knowledge, 2007. Foundations and Trends® in Networking 2.2 (2008): 135-269.

[2] April Rasala Lehman, Eric Lehman *Complexity Classification of Network Information Flow Problems* Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms. Society for Industrial and Applied Mathematics, 2004.

Standalone Talks

Onur Çağırıcı

onur@mail.muni.cz

Presented paper by Z. Yang, Y. Liu, and X-Y. Li

Beyond trilateration: On the localizability of wireless ad-hoc networks

(http://ieeexplore.ieee.org/abstract/document/5062166/?reload=true)

Introduction

he proliferation of wireless and mobile devices has fostered the demand of context aware applications, in which location is often viewed as one of the most significant contexts. Classically, trilateration is widely employed for testing network localizability; even in many cases it wrongly recognizes a localizable graph as non-localizable. In this study, we analyze the limitation of trilateration based approaches and propose a novel approach which inherits the simplicity and efficiency of trilateration, while at the same time improves the performance by identifying more localizable nodes. We prove the correctness and optimality of this design by showing that it is able to locally recognize all 1-hop localizable nodes. To validate this approach, a prototype system with 19 wireless sensors is deployed. Intensive and large-scale simulations are further conducted to evaluate the scalability and efficiency of our design.

Wheel Graph A wheel graph W_n is a graph with n vertices, formed by connecting a single vertex to all vertices of an $(n-1)$-cycle. The vertices in the cycle will be referred to as *rim vertices*, the central vertex as the hub, an edge between the hub and a rim vertex as a spoke, and an edge between two rim vertices as a rim edge.

Conditions for Node Localizability We define the distance graph G_N of a wireless ad-hoc network. Each wireless communication device (e.g., laptop, RFID, or sensor node) is modeled as a vertex of G_N and there is an un-weighted edge connecting two vertices if the distance between them can be measured or both of them are in known locations, e.g., beacon nodes. The closed neighborhood graph of a vertex v, denoted by $N[v]$, is a subgraph of G_N containing only v and its one-hop (direct) neighbors and edges between them in G_N. We also define the open neighborhood graph $N(v)$, obtained by removing v and all edges incident to v from $N[v]$.

Lemma 3 If a graph G is 2-connected, then G is globally rigid, where G' is obtained by adding a vertex v_0 and edges between v_0 to all vertices in G.

Theorem 1 In a neighborhood graph $N[v]$ with $k(k \geq 3)$, localizable vertices $v_i(i = 1, \ldots, k$ and $v = v_k)$, any vertex (other than v_i) belongs to a wheel structure with at least 3 localizable vertices if and only if it is included by the unique block in $N(v)$ containing $k-1$ localizable vertices.

Definition 2 In a network, a node is k-hop localizable if it can be localized by using only the information of at most k-hop neighbors.

Radovan Červený

cervera3@fit.cvut.cz

Presented paper by Paweł Obszarski, Andrzej Jastrzębski

Edge-coloring of 3-uniform hypergraphs

(http://dx.doi.org/10.1016/j.dam.2016.06.009)

Introduction

We will cover the edge-coloring problem for special classes of hypergraphs, where we will show a NP-completeness result together with a polynomial solution for some highly-structured classes of hypergraphs – hypertrees, hypercycles and hypercacti. We will conclude with an in-depth overview of the Orlin's algorithm for coloring proper circular arc graphs.

Definitions

Let $H = (V, E)$ be a *hypergraph*, $V(H)$ is a set of vertices, $E(H)$ is a multiset of non-empty subsets of $V(H)$ called *hyperedges(edges)*, an edge e and vertex v are *incident* if $v \in e$ and two edges e, e' are *adjacent* if they share a common vertex.

$\Psi(e) = |e|$ denotes the edge cardinality. $\Psi(H) = \max_{e \in E(H)} \Psi(e)$ denotes the maximum cardinality of an edge in H. For a vertex $v \in V$, *degree* $deg(v)$ is a number of edges to which v is incident. $\Delta(H) = \max_{v \in V(H)} deg(v)$ is a *degree of* H.

Hypergraph H is $d-$*uniform* if $\forall e \in E(H), \Psi(e) = d$.

A *proper edge-coloring* of a hypergraph H with k colors is a function $c : E(H) \to \{0, \ldots, k-1\}$ such that no two adjacent edges are assigned the same color.

The *chromatic index* $\chi'(H)$ of H is a number of colors in an optimal (minimal) edge-coloring of H.

A *line graph* $L(H)$ of hypergraph H is a simple graph where vertices represent hyperedges of H and two vertices in $L(H)$ are adjacent if and only if their corresponding hyperedges are adjacent.

A graph G is an *underlying (host) graph* of hypergraph H if $V(G) = V(E)$ and each edge $e \in E(H)$ induces a connected subgraph in G.

A hypergraph H is called a *hypertree/hypercycle/hypercactus* if there exists a *tree/cycle/cactus* which is an underlying graph for H.

Theorems

Fact 1 *For any 3-uniform hypergraph H the following holds:* $\Delta(H) \leq \chi'(H) \leq 3\Delta(H) - 2$.

The edge-coloring of hypergraph H is equivalent to vertex-coloring of $L(H)$, thus the above fact can be generalized to $\chi'(H) \leq \Delta(L(H)) + 1$, or due to Brooks' theorem to $\chi'(H) \leq \Delta(L(H))$ unless $L(H)$ is a complete graph or an odd cycle.

Fact 2 *Let H be hypertree. Then it can be edge-colored in polynomial time.*

Lemma 3 *Let H be hypercycle with m edges and $\Psi(H) = 3$. Then it can be edge-colored in $O(m^{3/2})$.*

It is done by transforming the graph into a proper circular arc graph which can be colored using an improved version of Teng and Tucker's approach in $O(n^{3/2})$ [1]. Teng and Tucker's work is a refinement of an original $O(n^2)$ algorithm by Orlin et al. [2].

Theorem 4 *Edge-coloring of a 3-uniform hypercactus with m edges can be done in time $O(m^{3/2})$.*

Using results for hypertrees and hypercycles, we devise an iterative procedure to color the whole hypercactus.

Theorem 5 *It is NP-complete to decide whether a 3-partite hypergraph of degree 3 is 3-edge-colorable.*

The reduction is done from the problem of edge precoloring extension to proper 3-edge-coloring for bipartite graphs of degree 3 precolored with at most 3 colors, which has been proven to be NP-complete by Fiala [3].

Orlin's algorithm — Definitions

A graph G is a *circular arc graph* if there is a 1:1 correspondence between vertices of G and the arcs of the circle such that two vertices are adjacent if and only if their corresponding arcs are intersecting.

Such graph G is called *proper* if no arc is wholly contained in another.

An *overlap clique* of G is a maximal set of vertices of G whose corresponding arcs intersect at a common point on the circle.

A set S is called *circularly consecutive* if $S = \{i, i+1, \ldots, j\}$ or $S = \{i, i+1, \ldots, n, 1, \ldots, j\}$ for some $i, j \in \{1, \ldots, n\}$.

A *last element* of a circularly consecutive set S is element $i \in S$ such that element $i+1 \notin S$.

We denote a circularly consecutive set $S = \langle i, j \rangle$ where j is the last element of S and i is the last element of $\{1, \ldots, n\} \setminus S$.

Orlin's algorithm — Theorems

Lemma 6 *Let G be a proper circular arc graph. Each overlap clique of G is circularly consecutive.*

Lemma 7 *Let G be a proper circular arc graph on n vertices and let k be an integer that divides n. G is k-colorable if and only if G has no overlap clique of size $k+1$.*

Lemma 8 *Let G be a proper circular arc graph on n vertices that is k-colorable. Then G can be colored in a way such that each color class contains $\lceil n/k \rceil$ or $\lfloor n/k \rfloor$ vertices.*

Theorem 9 *Let G be a proper circular arc graph on n vertices, let $k < n$ and $r = k \pmod{n}$. Then G is k-colorable if and only if there exists a subset $V' \subset V$ of size $r\lceil n/k \rceil$ together with a subset $V'' = V - V'$ of size $(k-r)\lfloor n/k \rfloor$ and it holds that V' does not contain a overlap clique of size $r+1$ and V'' does not contain a overlap clique of size $k-r+1$.*

Theorem 10 *The partition problem from the Theorem 9 can be formulated as an integer linear program such that it is a special case of a shortest path problem. Let G be a proper circular arc graph. G may be k-colored if there exists a feasible solution to the linear program and such solution can be found in $O(n^2)$.*

Bibliography

[1] Wei-Kuan Shih and Wen-Lian Hsu. *An $O(n^{1.5})$ algorithm to color proper circular arcs.* Discrete Applied Mathematics, 25 (1989), pp. 321–323, http://dx.doi.org/10.1016/0166-218X(89)90011-5.

[2] J. Orlin, M. Bonuccelli, D. Bovel. *An $O(n^2)$ algorithm for coloring proper circular arc graphs.* SIAM J. Algebraic Discrete Meths., 2 (1981), pp. 88–93, http://dx.doi.org/10.1137/0602012.

[3] J. Fiala. *NP completeness of the edge precoloring extension problem on bipartite graphs.* Journal of Graph Theory, 43, pp. 156–160, http://dx.doi.org/10.1002/jgt.10088.

Andrej Dedík
dedikandrej@gmail.com
Presented paper by A.Frieze, W.Pegden

The topology of competitively constructed graphs

(https://www.math.cmu.edu/∼af1p/Texfiles/reggame.pdf)

Introduction

In the paper authors consider a k-regular graph game, in which two players add one edge each turn into initially empty graph. The focal point of the paper shows, regardless of who starts, that a player can achieve resulting graph of a k-regular graph game to be planar, or contain a clique of arbitrary size as a minor for particular k.

Stuff I presume you are already familiar with

Unoriented graph, Planar graph, Graph minor, Clique, Spanning tree

Definitions

The **k-regular graph game** is a game for two players switching turns starting with an empty graph. Each turn player may add an edge between two vertices, if there isn't one already, and it doesn't increase the degree of any vertex above k. If such move is not possible, the game ends.

The **deficit of vertex** v denoted by $\mathrm{def}(v)$ is defined as $k - \deg(v)$. Basically the amount of edges containing v we can add to graph in the course of k-regular graph game.

The **deficit of subgraph** G' denoted by $\mathrm{def}(G')$ is defined as $\sum_{v \in G'} \mathrm{def}(v)$.

Theorems

Theorem 1 *Regardless of who has the first move, a player in the 3-regular graph game has a strategy to ensure that the resulting graph is planar.*

Theorem 2 *For any l and sufficiently large n, and regardless of who has the first move, a player in the 4-regular graph game on n vertices has a strategy to ensure that the resulting graph has a K_l minor.*

Lemma 3 *In the course of playing the 4-regular graph game, a player can force the appearance of components of arbitrarily large deficit.*

Lemma 4 *Suppose G is a connected labeled graph, with nonnegative vertex labels bounded some fixed number b. For any s, if the sum of the labels of G is sufficiently large relative to b and $\Delta(G)$, we can find k disjoint connected subgraphs of G each with label sums s.*

Anna Dresslerová
dresslerova@fmph.uniba.sk
Presented paper by Genghua Fan

Integer 4-flows and cycle covers

(http://link.springer.com/article/10.1007/s00493-016-3379-9)

Introduction

In the paper are improved upper bounds for shortest cycle cover problem for general graphs with loops and parallel edges.

Definition 1 *A* cycle *is a graph in which each vertex has even degree. A* circuit *is a minimal nonempty cycle. The* length *of a cycle is the number of edges it contains.*

Definition 2 *A collection of cycles of graph G* covers *G if each edge of G is in at least one of the cycles; such a collection is called a* cycle cover*. The* length *of a cycle cover is the sum of lengths of cycles in the cycle cover. Length of the shortest cycle cover is denoted by $cc(G)$.*

Alon and Tarsi ([1]) conjectured that:

Conjecture 3 *Every bridgeless graph G has cycle cover of length at most $\frac{7}{5}|E(G)|$ $(= 1.4|E(G)|)$.*

In this talk we prove this theorem.

Theorem 4 *Let G be a bridgeless graph in which each vertex has degree at least 3. Then $CC(G) < \frac{278}{171}|E(G)|$ $(\approx 1.6257|E(G)|)$, and if G is loopless, then $cc(G) < \frac{218}{135}|E(G)|$ $(\approx 1.6148|E(G)|)$.*

To prove this we need some auxiliary lemmas and theorems.

Auxiliary lemmas and theorems

In this talk by k-flow we mean Z_k-flow.

Definition 5 *The* support *of a k-flow f is $SP(f) = \{e \in E(G) : f(e) \neq 0\}$. If $SP(f) = E(G)$, than f is called to be* nowhere-zero *flow.*

Definition 6 *Let f be a k-flow in G and H a subgraph of G. Define*

$$E_i(f, H) = \{e \in E(H) : f(e) = i\}, \qquad O \leq i \leq k-1$$

When $H = G$ we use $E_i(f)$ instead of $E_i(f, G)$.

Definition 7 *Let f be a k-flow in G with a circuit C. A k-flow φ is (f, C)*-equivalent *if $SP(\varphi)\backslash E(C) = SP(f)\backslash E(C)$.*

Lemma 8 *Let f be a k-flow in a graph G with circuit C. Then there is an (f, C)-equivalent k-flow φ with $|E_0(\varphi, C)| \leq \frac{|C|}{k}$.*

Lemma 9 ([2]) *A graph G has a nowhere-zero 4-flow if and only if there are three cycles in G such that each edge of G is contained in exactly two of the three cycles.*

Definition 10 *Let f be a 4-flow in a graph G with a circuit C. A path P connecting two distinct vertices $x, y \in E(C)$ is called a* chord-path*, with respect to f and C, if $E(P) \cap E(C) = \emptyset$, $V(P) \cap V(C) = \{x, y\}$ and $E(P) \subseteq E_1(f) \cup E_3(f)$.*

Definition 11 *Let f be a 4-flow in a graph G with a circuit $C = x_0x_1 \ldots x_{n-1}$. A segment $S = x_ix_{i+1} \ldots x_{i+t}$ of C is E_0*-alternating *if edges of $E_0(f)$ appear alternatively on S and first and last* *edge of S are in $E_0(f)$ (indices are taken modulo n). Segment S is* maximal *if there is no (f, C)-equivalent 4-flow φ having an $E_0(f)$-alternating segment S' such that $E(S) \subseteq E(S')$ and $|S'| > |S|$.*

Lemma 12 *Let f be a 4-flow in a graph G with a circuit $C = x_0x_1 \dots x_{n-1}$. Suppose that for any (f, C)-equivalent 4-flow f', $E_\alpha(f', C)$ is a matching for each $\alpha \in Z_4$. Let $S = x_ix_{i+1} \dots x_{i+t}$ be a maximal $E_0(f)$-alternating segment. If $|E(S)| \leq |C| - 2$, then there is a 4-flow φ in G such that $SP(\varphi) = SP(f)$, $\varphi(x_{i+t}x_{i+t+1}) \in \{1, 3\}$ and a chordal-path P, with respect to φ and C, starts at x_{i+t+1} with the other end $x_j \in V(S) \backslash \{x_i\}$.*

Theorem 13 *Let f be a 4-flow in a graph G with a circuit C, where $|C| = 4m, m \geq 1$. Suppose that the orientation of G has been chosen such that C is a directed circuit. If for any $(f; C)$-equivalent 4-flow f', $E_\alpha(f', C)$ is matching for each $\alpha \in Z_4$, then there is an (f, C)-equivalent 4-flow φ such that $|E_0(\varphi, C)| \leq m - 1$.*

Theorem 14 *Let f be a 4-flow in a graph G with a circuit C. If $|C| \leq 19$, then there is an (f, C)-equivalent 4-flow φ such that $|E_0(\varphi; C)| < \frac{|C|}{4}$.*

Definition 15 *An edge is said to be* contracted *if it is deleted and its ends are identified. For a subgraph H in a graph G, the* contraction *of H, denoted by G/H, is the graph obtained by contracting all the edges of H.*

Lemma 16 *Let G be a bridgeless graph. If each vertex of G has degree at least 3, then G has a spanning cycle F such that $|F| \geq \frac{2}{3}|E(G)|$ and G/F has a nowhere-zero 4-flow.*

Lemma 17 *Let F be a cycle in a bridgeless graph G and d_m the number of components of m edges in F. If G/F has a nowhere-zero 4-flow, then*

$$cc(G) \leq 3(|V(G/F) - 1) + |E(G)| + \frac{1}{2}|F| - \frac{1}{2}\sum_{i \geq 0} d_{2i+1}.$$

Lemma 18 *Let F be a cycle in a bridgeless graph G. If f is a 4-flow in G with $E_0(f) \subseteq E(F)$, then*

$$cc(G) \leq 2|E(G)| - |F| + 2|E_0(f)|.$$

Bibliography

[1] N. Alon and M. Tarsi. *Covering multigraphs by simple circuits.* SIAM J. Algebraic Discrete Methods 6, 345-350, 1985

[2] P. D. Seymour. *On Tuttes extension of the four-color problem.* J. Combin. Theory Ser. B 30, 130-135, 1981

Pavel Dvořák
koblich@iuuk.mff.cuni.cz
Presented paper by Emanuele Viola

The communication complexity of addition

(http://www.ccs.neu.edu/home/viola/papers/ccsum.pdf)

Introduction

In this talk we will prove a lower bound of randomized communication complexity of function GT (greater-then). This lower bound meets an upper bound for this function.

Definition 1 *The function* $\mathsf{GT}_n : \{0,1\}^n \to \{0,1\}$ *is defined as* $\mathsf{GT}(x,y) = 1$ *if and only if* $x \geq y$ *(as binary numbers).*

In the *randomized communication model* there is Alice and Bob and they know some boolean function $f : X \times Y \to \{0,1\}$ and some random bits r. Alice gets some $x \in X$ and Bob gets $y \in Y$ and their task is to compute $f(x,y)$ with high probability. Let π be a protocol and $\pi(x,y)$ be an output of this protocol. The *randomized communication complexity* $R_\varepsilon(f)$ of the function f is the length of the optimal randomized protocol π such that for every $x \in X, y \in Y$ holds that $Pr_r[f(x,y) \neq \pi(x,y)] \leq \varepsilon$.

The *distributional communication model* is similar however there is a distribution μ on inputs $X \times Y$ and no random bits r. The error of the protocol is measured against the distribution μ. Thus, *distributional communication complexity* $D^\mu_\varepsilon(f)$ of the function f is the length of the optimal deterministic protocol π such that $Pr_{(x,y)\sim\mu}[f(x,y) \neq \pi(x,y)] \leq \varepsilon$.

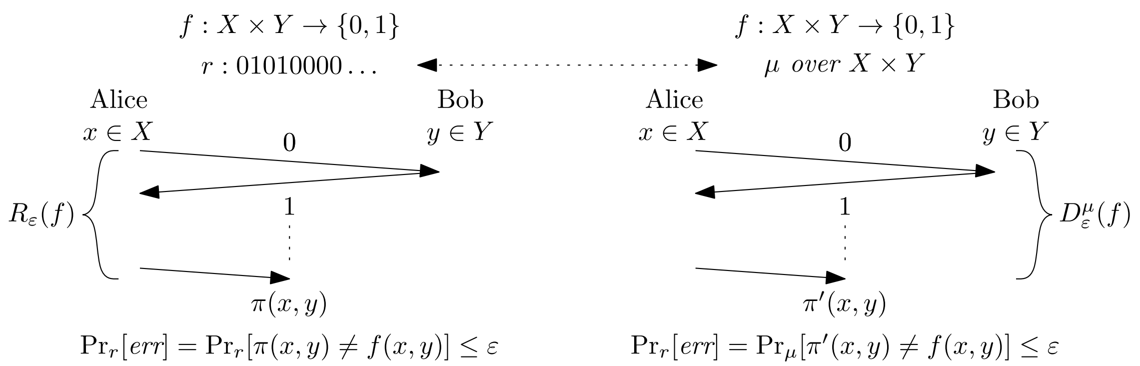

Figure 1: The difference between randomized and distributional communication models. The protocol π is randomized with random bits r and the protocol π' is deterministic.

Theorem 2 (Yao's Principle) *For every* f *holds that* $R_\varepsilon(f) = \max_\mu D^\mu_\varepsilon(f)$.

Lower Bound of GT

Theorem 3 (Main Theorem) $R_{1/100}(\mathsf{GT}_n) \geq \Omega(\log n)$.

The result is asymptotically optimal because there is a randomized protocol for GT of length $\mathcal{O}(\log n)$. The proof is done by contradiction – suppose we have a randomized protocol π for GT of length $\gamma \log n$ for a sufficiently small constant γ. We design two distributions G and B over

$\{0,1\}^n \times \{0,1\}^n$ such that

$$\Pr_{(x,y)\sim G}[\mathsf{GT}(x,y) = 1] = 1,$$
$$\Pr_{(x,y)\sim B}[\mathsf{GT}(x,y) = 1] = 1/2.$$

By Yao's principle we get a deterministic protocol π' which has a probability of error smaller then $1/100$ on the distribution $G/2 + B/2$. It follows the protocol π' has a probability of error smaller then $2/100$ on the distributions G and B. We prove that these two distributions are close. Thus, the protocol π' has similar outputs on the distributions G and B, which is the contradiction.

Technical Ingredients

We measure similarity of two distributions by the statistical distance.

Definition 4 *Let X and Y be distributions over S. The* statistical distance *of X and Y is defined as*

$$\Delta(X,Y) = \frac{1}{2}\sum_{s\in S}\Big|\Pr[X=s] - \Pr[Y=s]\Big| = \max_{T\subseteq S}\Big|\Pr[X\in T] - \Pr[Y\in T]\Big|.$$

We also use in the proof the notion of the entropy, which describe the uncertainty of result when we sample X.

Definition 5 *Let X and Y be random variables taking values in a set S. The* entropy *of X is defined as*

$$H(X) = \sum_{s\in S}\Pr[X=s]\log\frac{1}{\Pr[X=s]}.$$

The condition entropy *of X conditioned on Y is defined as*

$$H(X|Y) = \sum_{s\in S}\Pr[Y=s]H(X|Y=s).$$

For these notions we use the following results.

Proposition 6 *Let V be a random variable taking values in a set S. Let U be a uniform variable over S and $\pi : S \to S$ be a permutation. Then,*

1. $\Delta(V;\pi(V)) \le 2\Delta(V,U)$.
2. $\Delta(V;U) \le \sqrt{\log|S| - H(V)}$ *(Pinsker's inequality).*

Proposition 7 *Let V, W be two random variables and $E_1,\dots,E_t$ be mutually exclusive events. Then, $\Delta(V,W) \le \sum_{i\le t} Pr[E_i]\Delta(V|E_i; W|E_i)$.*

Proposition 8 (Chain Rule) *$H(X_1,\dots,X_n) = \sum H(X_i|X_{<i})$.*

Andreas Feldmann

andemil@kam.mff.cuni.cz

Parameterized Approximations of Symmetric and Planar Directed Steiner Networks

Joint work with Rajesh Chitnis.

Abstract

We study the Directed Steiner Network (DSN) problem, for which the input consists of a directed edge-weighted graph G on n vertices, together with a list of k vertex pairs $(s_1, t_1), \ldots, (s_k, t_k)$, the so called *terminals*. The aim is to compute the cheapest subgraph N of G such that there is an $s_i \to t_i$ path in N for each $i \in \{1, \ldots, k\}$. An important special case is the Strongly Connected Steiner Subgraph (SCSS) problem, in which every terminal needs to be connected to all other terminals in the solution N. Already this problem is notoriously hard both in terms of approximation, but also in terms of fixed-parameter tractability (FPT) for the well-studied parameter k: it is hard to approximate better than $\log^2 n$ [Halperin and Krauthgamer, *STOC* 2003], and it is W[1]-hard for k [Guo et al., *SIAM J. of Discrete Math.* 2011]. However, combining the two paradigms of approximation and FPT, it is known that for SCSS a 2-approximation can easily be computed in time $2^k \cdot n^{O(1)}$ [Chitnis et al., *IPEC* 2013]. We aim at obtaining better parameterized approximations for SCSS, but also for DSN, when using k as a parameter.

We present some preliminary results in this direction. In particular, we aim for parameterized approximation schemes, i.e. $(1+\varepsilon)$-approximations computed in *efficient* time $f(k, \varepsilon) \cdot n^{O(1)}$ or in time $f(k) \cdot n^{O(g(\varepsilon))}$ for some functions f and g independent of n. It follows from known results that there is no efficient $f(k, \varepsilon) \cdot n^{O(1)}$ time $(1+\varepsilon)$-approximation, unless P=W[1], even for SCSS on planar digraphs [Chitnis et al., *SODA* 2014]. We show that there are also no efficient parameterized approximation schemes for DSN on *symmetric* digraphs, i.e. where the weight of every edge uv is equal to the weight of its reverse edge vu. This is somewhat surprising as symmetric instances at first seem to closely resemble the undirected setting, in which DSN corresponds to the classical Steiner Forest problem, and the latter is known to be FPT for parameter k [Dreyfus and Wagner, *Networks* 1971]. On the positive side, we present a (non-efficient) $(1+\varepsilon)$-approximation algorithm running in time $2^k \cdot n^{2^{O(1/\varepsilon)}}$ for symmetric minor closed input graphs. It remains to generalize this algorithm to the purely planar (minor closed) and the purely symmetric settings, and/or show that the symmetric minor closed setting is hard as well.

Tomáš Gavenčiak

gavento@kam.mff.cuni.cz

Presented paper by I. Abraham, D. Delling, A. Fiat, A. Goldberg, R. Werneck

Highway Dimension and Provably Efficient Shortest Path Algorithms

(https://doi.org/10.1145/2985473)

The graph. Let $G = (V, E)$ be a simple undirected graph, $n = |V|$, $m = |E|$. Let $l(e) \geq 1$ be the edge-length, for paths let $l(P)$ denote the total length.

The metric. The shortest path length $d(u, v)$ is a metric. Let $D = \max_{u,v} d(u, v)$ be the diameter of G. Let $B_r(v) = \{u \mid d(u, v) \leq r\}$. Let $d(P, v) = \min_{u \in P} d(u, v)$, also saying that P is k-close to v for any $k \geq d(P, v)$. We assume all the shortest paths are unique (this can be ensured by length perturbation).

Significant paths and highway dimension

Significant paths. Let $r > 0$ be the A shortest path P is r-significant if there is a shortest $P' \supseteq P$ extending P by at most one vertex at each end and $l(P') > r$. Let $\mathcal{P}_r$ denote all r-significant paths. Let $S_r(v) \subseteq \mathcal{P}_r$ denote all r-significant paths $2r$-close to v.

Simplification: Imagine r-significant paths as paths of length at least r. Works for $l \equiv 1$.

Highway dimension. G has highway dimension HD at most h if for all $r > 0$ and all $v \in V$ there is $H \subseteq V$ such that $|H| \leq h$ and H hits all r-significant paths P that are $2r$-close to v (all $P \in S_r(v)$).

Sparse hitting sets

Sparse Shortest-path Hitting Set (h, r)-SPHS is a hitting set $C \subseteq V$ for $\mathcal{P}_r$ with $|B_{2r}(v) \cap C| \leq h$ for all $v \in V$ (locally sparse). Multiscale SPHS is a collection of $(h, 2^{i-1})$-SPHS for $i = 0, \ldots, \lceil \log_2 D \rceil$.

Theorem 4.2. A minimal hitting set for $\mathcal{P}_r$ on a graph with $HD(G) = h$ is (h, r)-SPHS.

Application: Hub labeling

Hub labeling $L : V \to 2^V$ such that $\forall u, v \in V$, $L(u) \cap L(v)$ contains a vertex on the shortest $u - v$ path.

Theorem 5.1. There is a hub labeling with $|L(v)| = \mathcal{O}(h \log D)$. After preprocessing, min-distance queries take $\mathcal{O}(h \log D)$ time.

Application: Transit node routing

Theorem 5.7. Choose M. Let q be smallest number with $|(h, 2^{i-1})\text{-SPHS}| \leq M$. Then after preprocessing, all distance-queries with $dist \geq 5 \cdot 2^{q-1}$ can be answered in time $\mathcal{O}(h^2)$ and using memory $\mathcal{O}(hn + M^2)$.

VC dimension and polynomial-time approximation

Vapnik-Chervonenkis dimension of a set system $(X, \mathcal{R} \subseteq 2^X)$ is the maximal size of a set $Y \subseteq X$ that is *shattered* by R. Y is shattered by R if every $Y' \subseteq Y$ can be obtained as $Y' = Y \cap R$ with $R \in \mathcal{R}$.

Theorem 7.1. There is an algorithm that finds a hitting set of size $\mathcal{O}(hd \log(hd))$ where d is the VC dimension and h the optimal hitting set size.

Theorem 7.2. A unique shortest path system has VC dimension 2.

Theorem 8.2. There is an algorithm that finds $(\mathcal{O}(h \log h), r)$-SPHS in polynomial time for any $r > 0$.

Jaroslav Hančl
jarda.hancl@gmail.cz
Presented paper by P. Aboulker, G. Lagarde, D. Malec, A. Methuku, C. Tompkins

De-Bruijn-Erdős-type theorems for graphs and posets

(https://arxiv.org/pdf/1501.06681.pdf)

Introduction

It all started with the classical theorem of De Bruijn and Erdős:

Theorem 1 *Every non-collinear set of n points in the plane determines at least n lines. Moreover, equality occurs if and only if the configuration is a near-pencil (that is, exactly $n-1$ of the points are collinear).*

There were many attempts to generalize that theorem. The paper studies the variations of this theorem in posets with the natural betweenness relation given by the poset relation. Second part is devoted to the triangle relation in a given graph which sounds less intuitive, but has a close connection to comparability graphs.

Lines in Posets

Let $P = (X, \prec)$ be a poset and $h(P)$ be the height of P given by the maximum size of a chain in P. We define the natural betweenness relation $[abc]$, that is, b is between a and c, in the following way

$$[abc] \Leftrightarrow a \prec b \prec c \text{ or } c \prec b \prec a.$$

Then we define the analog of a line

$$\overline{ab} = \{a, b\} \cup \{z : [zab] \text{ or } [azb] \text{ or } [abz]\}.$$

Theorem 2 *Let P be a poset on n points with no universal line and with $h(P) \geq 2$. Then P induces at least*

$$h(P)\binom{\lfloor n/h(P)\rfloor}{2} + \lfloor n/h(P)\rfloor(n \mod h(P)) + h(P)$$

distinct lines. Moreover, P induces exactly n lines if and only if it consists of a chain of size $n-1$ and a point which is comparable to at most one point of this chain.

Lines in Graphs

Let G be a graph. We define a line given by two vertices $a, b, \in V(G)$ by

$$\overline{ab} = \{a, b\} \cup \{x \in V(G) : abx \text{ is a triangle in } G\}.$$

Theorem 3 *If a graph G on $n \geq 4$ vertices does not contain a universal line, then it induces at least n distinct lines, and equality occurs if G consists of a clique of size $n-1$ and a vertex that has at most one neighbor in the clique.*

This concept generalizes the poset case from the previous section. Indeed, if we construct the comparability graph G_P of the poset $P = (X, \prec)$, that is the graph with X as a vertex set and ab as an edge if and only if $a \prec b$ or $b \prec a$, then xyz is a triangle in G_P if and only if $[xyz]$ or $[zxy]$ or $[yzx]$.

Petr Hliněný
hlineny@fi.muni.cz

Euler's polyhedral formula

Abstract

"$V - E + F = 2$", the famous Euler's polyhedral formula, has a natural generalization to convex polytopes in every finite dimension, also known as the Euler–Poincaré Formula. We provide a short new inductive combinatorial proof of this general formula. Our proof is self-contained and it does not use shellability of polytopes.

Dušan Knop
knop@kam.mff.cuni.cz
Presented paper by Walter Kern, Bodo Manthey

Approximating bounded-degree spanning trees and connected factors with leaves

(http://www.sciencedirect.com/science/article/pii/S0167637717300263)

Abstract

We present constant factor approximation algorithms for the following two problems: First, given a connected graph $G = (V, E)$ with non-negative edge weights, find a minimum weight spanning tree that respects prescribed upper bounds on the vertex degrees. Second, given prescribed (exact) vertex degrees $d = (d_v)_{v \in V}$, find a minimum weight connected d-factor. Constant factor approximation algorithms for these problems were known only for the case that $d_v \geq 2$ for all $v \in V$.

Problems

We consider two different optimization problems. In each case, an instance consists of a simple undirected complete graph $G = (V, E)$ with edge weights $w\colon E \to \mathbb{N}$ that satisfy the triangle inequality and given $(d_v)_{v \in V}$ to be interpreted as either prescribed vertex degrees or upper bounds thereof. For $F \subseteq E$, let $\deg_F(v)$ be the degree of node $v \in V$ in the graph (V, F). Furthermore, $w(F) = \sum_{e \in F} w(e)$ is the total weight of the edge set F.

In the Bounded-Degree Minimum Spanning Tree problem (denoted by BMST), we are to compute a tree $T \subseteq E$ of minimum weight with the additional condition that $\deg_T(v) \leq d_v$ for all $v \in V$. We call such a tree a d-bounded tree.

In the Connected Factor problem (denoted by ConnFact), our goal is to compute a connected d-factor F of minimum weight. This means that (V, F) must be connected and $\deg_F(v) = d_v$ for all vertices $v \in V$.

Lemmata and Algorithms

Lemma 1 *Given an undirected, complete graph $G = (V, E)$ with edge weights w that satisfy the triangle inequality and an edge $f = \{u, v\}$, we can compute in polynomial time a Hamiltonian path P with endpoints u and v such that $w(P) \leq 2w(T)$, where $T \subseteq E$ is a spanning tree that contains f and has minimum weight among all such trees.*

We define two sets $V_{=1} = \{v \in V : d_v = 1\}$, $V_{\geq 2} = \{v \in V : d_v \geq 2\}$.

We proceed in two steps.

- In the first step, we compute a forest that spans all of $V_{=1}$ and a subset of $V_{\geq 2}$ without violating the degree constraints. This forest is computed by solving an appropriate minimum-cost flow problem.
- In the second step, we connect the components of this forest along a Hamiltonian Path through a subset of the $V_{\geq 2}$ nodes. In this way, we construct a tree whose leaves are a subset of $V_{=1}$. Note that an optimal tree can also have leaves from $V_{\geq 2}$.

The Flow Problem Consider the following flow problem MCF_f: The underlying graph has vertex set $V \cup \{r\}$, where r is a new node, and edge set $(E \setminus f) \cup \Big\{\{u,r\} \colon v \in V_{\geq 2}\Big\}$. All edges $e \in E \setminus f$ have a capacity of 1 in both directions and costs of $w(e)$ per unit of flow. Each node $v \in V_{\geq 2}$ has a node capacity of $d_v - 1$. The edges $\{v,r\}$ for $v \in V_{\geq 2}$ are *overflow edges* and have cost 0. For $v \in V_{\geq 2} \setminus f$, edge $\{v,r\}$ has a capacity of $d_v - 2$. For $v \in f$, edge $\{v,r\}$ has a capacity of $d_v - 1$. The task is to find a minimum-cost flow from the $V_{=1}$ nodes, each having a supply of 1, to the new root node r, which has a demand of $|V_{=1}|$. Such a minimum-cost flow can be computed in polynomial time [1].

Lemma 2 *Let $f = \{u,v\}$ be an edge. Let $\mathcal{F}$ be an integral optimum solution of MCF_f with minimum support S. Then we have the following properties:*

1. *$w(S) \leq w(Tree_d)$, where $Tree_d$ is an optimal solution to BMST,*
2. *S is a forest,*
3. *$deg_S(u) \leq d_u - 1$ and $deg_S(v) \leq d_v - 1$, and*
4. *each connected component of S contains u or v or a vertex $x \in V_{\geq 2}$ with $deg_S(x) \leq d_x - 2$.*

Now given S, the support of a flow as in Lemma 2, we connect the connected components via a Hamilton Path P with endpoints u and v as in Lemma 1: In each component of S that contains neither u nor v, we pick a root r of degree at most $d_r - 2$ in S. Such a root exists by Lemma 2. Then we connect the components of S by following P, starting in u, ending in V and skipping all other vertices except the root nodes chosen. This yields a d-bounded tree T of weight

$$w(T) \leq w(S) + w(P) \leq w(Tree_d) + 2w(Tree_d) \leq 3w(Tree_d).$$

Theorem 3 *There is a 3-approximation algorithm for BMST.*

Lemma 4 *Let T be a d-bounded tree, and let F be a d-factor. If F is not connected, then we can find an edge $f = \{u,v\} \in F \setminus T$ and vertices u' and v' with the following properties:*

1. *f connects two components of F,*
2. *$\{u,u'\}, \{v,v'\} \in F \setminus T$, and*
3. *$\{u',v'\} \in E \setminus F$.*

Theorem 5 *There is a 7-approximation algorithm for* ConnFact.

Bibliography

[1] Ravindra K. Ahuja, Thomas L. Magnanti, James B. Orlin. *Network Flows: Theory, Algorithms, and Applications.* Prentice-Hall, 1993.

Tomáš Masařík

maso@kam.mff.cuni.cz

On hardness of some planar SAT problems

In this talk we will discuss several complexity results about variants of the planar SAT problem.

Definitions

SAT is a classical problem. Given a formula in CNF the task is to find an assignment to variables such that the formula is true. *Planar SAT* means that the underlying incidence graph of variables and clauses is planar. *Strongly planar SAT* is a modification such that the incidence graph of literals and clauses is planar, even when the edge between the two literals of the same variable is added.

In *3 SAT* are only 3 literals in a clause. *Not-all-equal* variant is satisfied if each clause has at least one literal set to true and one set to false. In contrary to *1-in-3* SAT that is satisfied if exactly one literal is set to true.

Theorems

Theorem 1 ([1]) *Planar not all equal 3SAT is polynomial time solvable.*

Theorem 2 ([2]) *Planar 1-in-3SAT is* NP*-complete.*

Theorem 3 ([3]) *Strongly planar 3SAT is* NP*-complete.*

Theorem 4 ([4]) *Strongly planar preassigned not all equal 3SAT is* NP*-complete.*

Bibliography

[1] Bernard M. E. Moret. *Planar* NAE3SAT is in P. SIGACT News 19, 2, 51–54, 1988.

[2] Dyer, Martin E., and Alan M. Frieze. *Planar 3DM is NP-complete.* Journal of Algorithms 7.2, 174–184, 1986.

[3] Wu, Lidong. *On strongly planar 3SAT.* Journal of Combinatorial Optimization 32.1, 293–298, 2016.

[4] Ali Deghan. *On strongly planar not-all-equal 3SAT.* Journal of Combinatorial Optimization 32.3, 721–724, 2016.

Ondřej Mička

mitch.ondra@gmail.com

Presented paper by Anthony Bonato, Przemyslaw Gordinowicz, Geňa Hahn

Cops and Robbers ordinals of cop-win trees

(https://arxiv.org/abs/1603.04266)

Introduction

The Cops and Robbers is a simple combinatorial game played by two players (cop and robber) on a graph. First, cop chooses his starting vertex, then robber chooses his starting vertex. Then both players take turns and move cop, resp. robber, along one edge (or they may choose not to move). For the cop the goal is to catch robber, that is, to occupy same vertex as the robber. Robber has to outrun the cop and he wins only if he can move in such way, that cop can never catch him.

This article explores game of Cops and Robbers on infinite cop-win trees and it characterizes CR-ordinals for cop-win (infinite) trees.

Ordinal numbers

Definition 1 *Ordinal number is set α such that it is strictly well-ordered with respect to relation $\in$ and for every β it holds $\beta \in \alpha \Rightarrow \beta \subset \alpha$. We will denote proper class of all ordinals as* ON*. For two ordinals α, β we also define relation $<$ as $\alpha < \beta \Leftrightarrow \alpha \in \beta$.*

Finite ordinals are precisely natural numbers, that is $0 = \emptyset$, $1 = \{0\}$, $2 = \{0, 1\}$, etc. Smallest infinite ordinal is set of all natural numbers $\mathbb{N} = \omega$. Ordinals can be seen as transfinite extension of natural numbers.

Definition 2 *Let $\alpha, \beta \in$ ON. We define $\alpha + \beta$ to be the unique ordinal isomorphic to set $\{0\} \times \alpha \cup \{1\} \times \beta$ with lexicographical ordering. Successor of α is ordinal $\alpha + 1$.*

Note that while ordinal addition is associative, it is *not* commutative. For example $1+\omega = \omega \neq \omega+1$.

Definition 3 *Nonzero $\alpha \in$ ON is successor ordinal if there exists $\beta \in$ ON such that $\alpha = \beta + 1$. If there exists no such β, than α is limit ordinal.*

Every $n \in \omega$ except zero is successor ordinal, while ω itself is smallest limit ordinal. Now we may introduce *transfinite induction.* It is an extension of mathematical induction to ordinals. The only difference is that we have to consider limit ordinals. So, apart from successor case $\alpha \to \alpha + 1$ we have limit case $(\forall \beta < \alpha) \to \alpha$.

Capture relation and capture-time ordinals

Definition 4 *Let $G = (V, E)$ is (infinite) graph. We define set of relations $\{\leq_\alpha \mid \alpha \in \text{ON}\}$ on V as follows:*

- *$u \leq_0 v$ iff $u = v$.*
- *$u \leq_\alpha v$ iff for every $x \in N[u]$ there exists $y \in N[v]$ and $\beta < \alpha$ such that $x \leq_\beta y$.*

We call these relations capture relation. Smallest ordinal $\rho(G)$ such that $\leq_{\rho(G)} = \leq_{\rho(G)+1}$ (i.e. the relation stabilizes) is called *CR-ordinal.*

It can be seen that capture relation stabilizes and $\rho(G)$ is well defined. Also, it holds that given graph G is cop-win iff $\leq_{\rho(G)}= V \times V$ and capture relation can be used to obtain best moves for cop.

Definition 5 *Let $G = (V, E)$ is cop-win graph and $u, v \in V$. We define:*

- $\eta(u, v) = \min\{\alpha |\, u \leq_\alpha v\}$
- $\eta(v) = \sup \{\eta(u, v)|u \in V\}$
- $\eta(G) = \min\{\eta(v)|v \in V\}$
- $\theta(G) = \{v \in V|\eta(v) = \eta(G)\}$

For finite graph $\eta(G)$ is capture time – maximum number of moves the cop needs to capture the robber, minimized over all starting positions of cop (i.e. the cop chooses his starting point). Similarly the CR-ordinal $\rho(G)$ is maximum number of moves the cop needs to capture robber, but maximized over all starting positions (i.e. robber chooses cop's starting point). Clearly it holds that $\rho(G) = \sup \eta(v)|v \in V$.

Classification of CR-ordinals for trees

Theorem 6 *Let T is a cop-win tree. Than $\rho(T)$ is either finite or in form $\alpha + \omega$, where α is limit ordinal. Moreover, for every $n \in \omega$ there is a cop-win tree with $\rho(T) = n$ and for every limit ordinal α there is a cop-win tree with $\rho(T) = \alpha + \omega$.*

This is the main result of the article. Finite case is trivial. So first we proof following lemma which gives the first part of the theorem.

Lemma 7 *If T is a cop-win tree with infinite radius, than $\eta(T)$ is infinite and $\rho(T) = \eta(T) + \omega$.*

For the second part, we will recursively construct family of trees with desired CR-ordinal.

Definition 8 *For every $\alpha \in \mathrm{ON}$ we will define tree S_α with root r_α as follows:*

- *(S_0, r_0) is a tree containing only vertex v_0.*
- *(S_α, r_α) for $\alpha > 0$ is created by attaching trees S_β for every $\beta < \alpha$ to the new vertex r_α with an edge from their root.*

Lemma 9 *For every $\alpha \in \mathrm{ON}$ it holds $\eta(S_{\alpha+1}) = \alpha$.*

Jana Novotná

janka.novot@seznam.cz

Presented paper by François Dross

Fractional triangle decompositions in graphs with large minimum degree

(https://arxiv.org/abs/1503.08191)

Introduction

A triangle decomposition of a graph is one of decomposition problems, classical problems in combinatorics. The question is whether a graph can be decomposed into triangles. There are known results that ensure a triangle decomposition, these results are based on determining the lower bound on minimum degree and on the number of vertices. The paper focus on fractional variant of the problem where non-negative weights are assigned to the triangles of the graph such that for each edge the sum of the weights of the triangles containing this edge is one. The lower bound $(\frac{9}{10}+\varepsilon)n$ on minimum degree is proven and furthermore this result is used to improved the previously known bound for the non-fractional variant.

More formally

Definition 1 *Let H be a graph. An* H-decomposition *of a graph G is a set of subgraphs of G isomorphic to H that are edge disjoint such that each edge of G is contained in one of them. A K_3-decomposition is also called a* triangle decomposition.

Definition 2 *A* fractional H-decomposition *of a graph G is an assignment of non-negative weights to the copies of H in G such that for an edge e, the sum of the weights of the copies of H that contain e is equal to one. A fractional K_3-decomposition is also called a* fractional triangle decomposition.

Definition 3 *A graph G is H-divisible if gcd(G) is a multiple of gcd(H), and the number of edges of G is a multiple of the number of edges of H.*

Theorem 4 *(Barber et al.). There exists an n_0 such that every K_3-divisible graph G on $n \geq n_0$ vertices with minimum degree at least $0.956n$ is K_3-decomposable.*

Theorem 5 *(Garaschuk). Let G be a graph with n vertices and minimum degree at least $0.956n$. The graph G admits a fractional triangle decomposition.*

Theorem 6 *(the paper). Let $\varepsilon > 0$. There exists an n_0 such that every graph with $n \geq n_0$ vertices and minimum degree at least $(\frac{9}{10}+\varepsilon)n$ admits a fractional triangle decomposition.*

Theorem 7 *(Barber et al.). Suppose there exist n_0 and δ such that every graph on $n \geq n_0$ vertices with minimum degree at least δn is fractionally K_3-decomposable. For all $\varepsilon > 0$, there exist n_1 such that every K_3-divisible graph on $n \geq n_1$ vertices with minimum degree at least $\max(\delta, \frac{3}{4}+\varepsilon)n$ vertices is K_3-decomposable.*

Theorem 8 *(the paper). Let $\varepsilon > 0$. There exists an n_0 such that every K_3-divisible graph on $n \geq n_0$ vertices with minimum degree at least $(\frac{9}{10}+\varepsilon)n$ is K_3-decomposable.*

Jitka Novotná
jitka.novotna@fel.cvut.cz

Contraction Hierarchies & Hub Labeling

The graph. Let $G = (V, E)$ be a simple undirected graph, $n = |V|$, $m = |E|$. Let $l(e) \geq 1$ be the edge-length, for paths let $l(P)$ denote the total length.

The metric. The shortest path length $d(u, v)$ is a metric. Let $D = \max_{u,v} d(u, v)$ be the diameter of G. Let $B_r(v) = \{u \mid d(u, v) \leq r\}$. Let $d(P, v) = \min_{u \in P} d(u, v)$, also saying that P is k-close to v for any $k \geq d(P, v)$. We assume all the shortest paths are unique (this can be ensured by length perturbation).

Sparse Shortest-path Hitting Set (h, r)-SPHS is a hitting set $C \subseteq V$ for $\mathcal{P}_r$ with $|B_{2r}(v) \cap C| \leq h$ for all $v \in V$ (locally sparse). Multiscale SPHS is a collection of $(h, 2^{i1})$-SPHS for $i = 0, \ldots, \lceil \log_2 D \rceil$.

Theorem 4.2. [2] A minimal hitting set for $\mathcal{P}_r$ on a graph with $HD(G) = h$ is (h, r)-SPHS.

Algorithm: Contraction Hierarchies [5]

Contraction. A shortcut is a new edge e = (u,w) with length dist(u,w). The contraction operation deletes a vertex v from the graph and adds edges between its neighbors to maintain the shortest path information.

Preprocessing We contract all vertices from least important to most important. The output of preprocessing is the set $E+$ of shortcut edges and the vertex order. We denote the position of a vertex v in the ordering by $rank(v)$.

Query An $s - t$ CH query runs a pruned bidirectional Dijkstra search on the graph $G+ = (V, E \cap E+)$. When scanning v, only the edges (v, w) with $rank(v) < rank(w)$ are examined.

Lemma 6.1. [8] For fixed v and j, the number of edges $(v, w) \in E+$ with $w \in Q_j$ is at most h.

Theorem 6.2. If (G, l) has highway dimension h, then preprocessing based on multiscale SPHS produces a set of shortcuts $E+$ such that degree of every vertex in $G(V, E \cap E+)$ is at most $h + h\log(D)$ and $|E+| \leq nh \log D$.

Algorithm: Bidirectional A* [7]

Priority function	
$f(n) = g(n)$	Dijkstra
$f(n) = g(n) + h(n)$	The A^* algorithm
$f(n) = max(g(n) + h(n), 2g(n))$	MM search

Lemma 1 *MM never expands nodes with $g(n) > C^*/2$.*

Algorithm: Hub labeling [3]

Hub labeling $L : V \to 2^V$ such that $\forall u, v \in V$, $L(u) \cap L(v)$ contains a vertex on the shortest $u - v$ path.

Bibliography

[1] Berger, Annabell and Grimmer, Martin and Müller-Hannemann, Matthias. *Fully dynamic speed-up techniques for multi-criteria shortest path searches in time-dependent networks.* International Symposium on Experimental Algorithms,2010.

[2] Abraham, Ittai and Delling, Daniel and Fiat, Amos and Goldberg, Andrew V and Werneck, Renato F. *Highway dimension and provably efficient shortest path algorithms.* Journal of the ACM, 2016.

[3] Abraham, Ittai and Delling, Daniel and Goldberg, Andrew V and Werneck, Renato F. *Hierarchical hub labelings for shortest paths.* European Symposium on Algorithms, 2012.

[4] Ma, Hui and Liang, Ruishi. *Using bidirectional search to compute optimal shortest paths over multi-weight graphs.* Information Science and Cloud Computing Companion (ISCC-C), 2013 International Conference on.

[5] Geisberger, Robert and Sanders, Peter and Schultes, Dominik and Delling, Daniel. *Contraction hierarchies: Faster and simpler hierarchical routing in road networks.* International Workshop on Experimental and Efficient Algorithms, 2008.

[6] Bast, Holger and Funke, Stefan and Matijevic, Domagoj. *Ultrafast Shortest-Path Queries via Transit Nodes.* The Shortest Path Problem, 2006.

[7] Holte, Robert C and Felner, Ariel and Sharon, Guni and Sturtevant, Nathan R. *Bidirectional Search That Is Guaranteed to Meet in the Middle.*. AAAI, 2016.

[8] Abraham, Ittai and Fiat, Amos and Goldberg, Andrew V and Werneck, Renato F. *Highway dimension, shortest paths, and provably efficient algorithms.* Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms, 2010.

[9] Köhler, Ekkehard and Möhring, Rolf H and Schilling, Heiko. *Acceleration of shortest path and constrained shortest path computation.* International Workshop on Experimental and Efficient Algorithms,2005.

Petra Pelikánová
ppelikanova@gmail.com
Presented paper by Andrew D. King, Bruce A. Reed

A Short Proof That χ Can be Bounded ε Away from $\Delta + 1$ toward ω

(https://arxiv.org/pdf/1211.1410.pdf)

Introduction

Conjecture 1 (Reed's) *Every graph satisfies* $\chi \leq \left\lceil \frac{(\Delta+1)\ +\ \omega}{2} \right\rceil$.

This conjecture has been proven for some restricted classes of graphs.

Notation

Let $G = (V, E)$ be a graph. Chromatic number χ is the smallest number of colors needed to color the vertices of G such that no two adjacent vertices share the same color.

Clique number ω denotes number of vertices in a maximum clique.

Maximum degree of G is denoted Δ.

Let $N(v)$ be a neighborhood of a vertex $v \in V$, an induced subgraph of G consisting of all vertices adjacent to v. Size of the maximum closed neighborhood (neighborhood which contains the vertex v) is $\Delta + 1$.

Independent set (stable set) in G is a set $S \subseteq V$ such that no two vertices are adjacent.

Antimatching in G is a matching in the complement of G.

Theorems

Theorem 2 *There exists an* $\varepsilon > 0$ *such that every graph satisfies*

$$\chi \leq \lceil (1-\varepsilon)(\Delta+1) + \varepsilon\omega \rceil.$$

Theorem 3 *Every graph satisfying* $\omega > \frac{2}{3}(\Delta + 1)$ *contains an independent set hitting every maximum clique.*

Theorem 4 *There is a* Δ_0 *such that for any graph with maximum degree* $\Delta > \Delta_0$ *and for any* $B > \Delta(\log\Delta)^3$, *if no N(v) contains more than* $\binom{\Delta}{2} - B$ *edges then* $\chi(G) \leq (\Delta + 1) - \frac{B}{e^6\Delta}$.

Corollary 5 *There is* Δ_0 *such that for any graph with maximum degree at most* $\Delta > \Delta_0$ *and for any* $\alpha > \frac{2(log\Delta)^3}{(\Delta-1)}$, *if no* $N(v)$ *contains more than* $(1-\alpha)\binom{\Delta}{2}$ *edges then*

$$\chi(G) \leq (1 - \frac{\alpha}{2e^6})(\Delta + 1) + \frac{\alpha}{2e^6}\omega.$$

Theorem 6 *Let* α *be any positive constant and let* ε *be any constant satisfying* $0 < \varepsilon < \frac{1}{6} - 2\sqrt{\alpha}$. *Let* G *be a graph with* $\omega \leq \frac{2}{3}(\Delta + 1)$ *and let* v *be a vertex whose neighborhood contains more than* $(1-\alpha)\binom{\Delta}{2}$. *Then*

$$\chi(G) \leq \max\{\chi(G - v), (1-\varepsilon)(\Delta + 1)\}.$$

Corollary 7 *Let* ρ *be a positive constant satisfying* $\rho \leq \frac{1}{160}$, *let* G *be a graph with maximum degree at most* Δ, $\omega \leq \frac{2}{3}(\Delta + 1)$, *and let* v *be a vertex whose neighborhood contains at least* $(1-\rho)\binom{\Delta}{2}$ *edges. Then*

$$\chi(G) \max\{\chi(G - v), (1-\rho)(\Delta + 1)\}.$$

Jakub Svoboda

3.14159.jakub@gmail.com

Presented paper by Mark Wildon

Searching for knights and spies: A majority/minority game

(https://arxiv.org/pdf/1412.4247.pdf)

Introduction

In a room there are n people, numbered from 1 up to n. Each person is either a knight or a spy, and will answer any question of the form 'Person x, is Person y a knight?' Knights always answer truthfully. We consider two types of spy: **liars**, who always lie, and **moles**, who lie or tell the truth as they see fit.

In this talk we determine the minimum number of questions that are necessary and sufficient to find a spy, or to find at least one person's identity, or to find an identity of a specific person, nominated in advance.

$T_L(n,k)$ is the minimum number of questions that are necessary and sufficient either to identify a liar, or to make a correct claim that everyone in the room is a knight if we know that there in the room with n people are at least k ($\frac{n}{2} < k < n$) knights and others are liars (not moles).

$T_L^*(n,k)$ is defined like $T_L(n,k)$, but we know, that at least one spy is present.

Theorem 1 *Let $n = q(n-k+1) + r$ where $0 \le r \le n-k$. Then*

$$T_L(n,k) = \begin{cases} n-q+1 & \textit{if } r = 0 \\ n-q & \textit{if } r = 1 \\ n-q & \textit{if } r \ge 2 \end{cases}$$

and

$$T_L^*(n,k) = \begin{cases} n-q & \textit{if } r = 0 \\ n-q & \textit{if } r = 1 \\ n-q-1 & \textit{if } r \ge 2 \end{cases}$$

with the single exception that $T_L^*(5,3) = 4$.

Let $T_S^*(n,k)$ and $T_S(n,k)$ be the analogously defined numbers if all spies are moles.

Theorem 2 *We have $T_S^*(n,k) = n-1$ and $T_S(n,k) = n$.*

If all spies are liars, then let minimum number of questions that are necessary and sufficient to find a knight be $K_L(n,k)$, to find at least one person's identity be $E_L(n,k)$ and to identify Person 1 be $N_L(n,k)$.

Let $K_S(n,k)$, $E_S(n,k)$ and $N_S(n,k)$ be the analogously defined numbers when all spies are moles. Let $N_L^*(n,k)$ and $N_S^*(n,k)$ be the analogously defined numbers to $N_L(n,k)$ and $N_S(n,k)$ when it is known that a spy is present. Let $B(s)$ be the number of 1s in the binary expansion of $s \in \mathbb{N}$.

Theorem 3 *We have*

$$K_S(n,k) = K_L(n,k) = E_S(n,k) = E_L(n,k) = 2(n-k) - B(n-k)$$

and the same holds for the analogous numbers defined on the assumption that a spy is present. Moreover

$$N_S(n,k) = N_L(n,k) = N_S^*(n,k) = N_L^*(n,k) = 2(n-k) - B(n-k) + 1$$

with the exception that $N_L^*(n,k) = 2(n-k) - B(n-k) = n-2$ *when* $n = 2^{e+1}+1$ *and* $k = 2^e+1$ *for some* $e \in \mathbb{N}$.

Theorem 4 *Suppose that spies always lie. There is a questioning strategy that will find a knight by question* $K(n,k)$ *, find Person* 1*'s identity by question* $K(n,k)+1$ *and by question* $T_L(n,k)$ *either find a spy or prove that everyone in the room is a knight. Moreover if a spy is known to be present then a spy will be found by question* $T_L^*(n,k)$.

Theorem 5 *Suppose that all spies are moles. There is a questioning strategy that will find a knight by question* $K(n,k)+1$*, find Person* 1*'s identity by question* $K(n,k)+2$*, and by question* $T_S(n,k) = n$ *either find a spy, or prove that everyone in the room is a knight. Moreover, if a spy is known to be present, then a spy will be found by question* $T_S^*(n,k) = n-1$*. When* $n = 7$ *and* $k = 4$ *and a spy is known to be present there is no questioning strategy that will both find a knight by question* $K_S(7,4) = 4$ *and find a spy by question* $T_S^*(7,4) = 6$.

Josef Svoboda
josefsvobod@gmail.com
Presented paper by Andy Hardt, Pete McNeely, Tung Phan and Justin M. Troyka

Combinatorial species and graph enumeration

(https://arxiv.org/abs/1312.0542)

Basic definitions

Definition 1 *Species $\mathcal{F}$ is a procedure which assigns*

- *to every finite set U a set of structures $\mathcal{F}(U)$.*
- *to every bijection $\sigma : U \to V$ a transport function or relabeling $\mathcal{F}(\sigma) : \mathcal{F}(U) \to \mathcal{F}(V)$*

such that for $\sigma : U \to V$ and $\varphi : V \to W$:

- $\mathcal{F}(\varphi \circ \sigma) = \mathcal{F}(\varphi) \circ \mathcal{F}(\sigma)$
- $\mathcal{F}(id_U) = id_{\mathcal{F}(U)}$

In other words, a species is a functor from the category of finite set and bijections into category of sets.

Definition 2 *Two species $\mathcal{F}$ and $\mathcal{G}$ are combinatorially equivalent if there is a collection $\alpha_U : \mathcal{F}(U) \to \mathcal{G}(U)$ such that for every transport $\sigma : U \to V$ the following diagram commutes:*

$$\begin{tikzcd} \mathcal{F}(U) \arrow[r, "\alpha_U"] \arrow[d, "\mathcal{F}(\sigma)"'] & \mathcal{G}(U) \arrow[d, "\mathcal{G}(\sigma)"'] \\ \mathcal{F}(V) \arrow[r, "\alpha_V"] & \mathcal{G}(V) \end{tikzcd}$$

Species generating functions

Definition 3 *The exponenential generating function of a species $\mathcal{F}$ is*

$$\mathcal{F}(x) = \sum_{n=0}^{\infty} f(n) \frac{x^n}{n!}$$

where $f(n) = |F\{1, 2, \dots, n\}|$.

Definition 4 *The type generating function of a species $\mathcal{F}$ is*

$$\tilde{\mathcal{F}}(x) = \sum_{n=0}^{\infty} \tilde{f}(n) x^n$$

where $\tilde{f}(n)$ is the number of isomorphism classes of $\mathcal{F}$-structures of order n.

Proposition 5 *Two combinatorial equivalent species have the same exponential (resp. type) generating function.*

Operations on species

Definition 6 *Let $\mathcal{F}$ and $\mathcal{G}$ be species. Then their sum $\mathcal{F}+\mathcal{G}$ is the species where*

$$\mathcal{F}+\mathcal{G}(U)=\mathcal{F}(U)\dot{\cup}\mathcal{G}(U).$$

Definition 7 *Let $\mathcal{F}$ and $\mathcal{G}$ be species. Then their product $\mathcal{F}\cdot\mathcal{G}$ is the species where*

$$\mathcal{F}\cdot\mathcal{G}(U)=\bigcup_{S\subseteq U}\mathcal{F}(S)\cdot\mathcal{G}(U\setminus S).$$

Definition 8 *Let $\mathcal{F}$ and $\mathcal{G}$ be species and $\mathcal{G}\emptyset=\emptyset$. Then their composition $\mathcal{F}\circ\mathcal{G}$ is the species where $\mathcal{F}\circ\mathcal{G}(U)$ is given by: $\{(s,T): s\in\mathcal{F}(T)$ and T is a set of $\mathcal{G}$-structures whose label sets form a partition of $U\}$.*

Note 9 *All operations are equipped with obvious transport functions.*

Most of applications are based on the following theorems:

Theorem 10 *Exponential (resp. type) gererating functions of the sum of two species $\mathcal{F}$ and $\mathcal{G}$ is the sum of corresponding exponential (resp. type) generating functions. Similarly for the product and the composition of species.*

Bibliography

[1] Andy Hardt, Pete McNeely, Tung Phan and Justin M. Troyka, *Combinatorial species and graph enumeration*, https://arxiv.org/abs/1312.0542, (2013).

[2] André Joyal, *Une théorie combinatoire des séries formelles*, Advances in mathematics 42, l-82, (1981).

[3] F. Bergeron, G. Labelle, P. Leroux, *Combinatorial species and tree-like structures.* Enc. of Math. and its Appl., Camb. Univ. Press, (1997).

Jana Syrovátková
syrovatkova@kam.mff.cuni.cz
Presented paper by Andreas Darmann, Ulrich Pferschy, Joachim Schauer

On the Shortest Path Game

(https://arxiv.org/abs/1506.00462)

Introduction

There is a directed graph $G(V, E)$ with positive costs $c(u, v)$ for each edge $(u, v) \in E$ and two vertices $s, t \in V$. The Shortest Path Game is played by two players who have full knowledge of the graph. They start in s and always move together along edges of the graph. In each vertex the player selects the next vertex among all neighboring vertices of the current vertex with the oponent taking decision in his previous turn. So player A starts in s, select the edge from s to some x, then B selects the edge from x to the next vertex. The player deciding in the current vertex also has to pay the cost of the chosen edge. Each player wants to minimize the total arc costs it has to pay. The game continues until the players reach the destination vertex t.

The similar problem could be defined on undirected graph or on some specific types of graphs.

Possible restrictions of the game:

- No player can select an arc which does not permit a path to vertex t.
- Possibility to restrict the game to simple paths - each vertex may be visited at most once.
- The players cannot select an arc which implies necessarily a cycle of even length.

The goal of paper is to study the complexity status of finding this spe-path.

Definitions

Definition 1 Backward induction - *algorithm, where is each node in the game tree, whose child nodes are all leaves, the associated player can reach a decision by simply choosing the best of all child nodes.*

Definition 2 Spe-path *is subgame perfect equilibrium (SPE).*

Definition 3 Quantified 3-SAT *can be interpreted as the following game: There are two players (the existential- and the universal-player) moving alternately, starting with the existential-player. The ith move consists of assigning a truth value to variable x_i. After n moves, the existential-player wins if and only if the produced assignment makes φ true.*

Definition 4 Cactus graph *is graph where each edge is contained in at most one simple cycle.*

Theorems

Theorem 5 *Shortest Path Game is PSPACE-complete for bipartite directed graphs.*

Theorem 6 *The spe-path of Shortest Path Game on acyclic directed graphs can be computed in $O(|A|)$ time.*

Theorem 7 *Shortest Path Game on undirected graphs is PSPACE-complete for bipartite graphs.*

Theorem 8 *Q is a "yes"-instance of Quantified 3-SAT $\Leftrightarrow$ s is a "yes"-instance of Shortest Path Game.*

Theorem 9 *The spe-path of Shortest Path Game on undirected cactus graphs can be computed in $O(n^2)$ time.*

Theorem 10 *The spe-path of Shortest Path Game on directed cactus graphs can be computed in $O(n)$ time.*

Jakub Tětek
j.tetek@gmail.com
Presented paper by Ervin Győri, Tamás Róbert Mezei, Gábor Mészáros
Note on terminal-pairability in complete grid graphs
(https://arxiv.org/abs/1605.05857)

Introduction

Given graph G and multigraph D on the same vertex set, we say that G realizes D if there exist edge disjoint paths $P_1, P_2, \ldots, P_n$ such that P_i is connecting the endpoints of the edge e_i for all i. The graph D is called a demand graph. Given a graph G and a family of demand graphs $\mathcal{F}$ we say that G is terminal-pairable with respect to $\mathcal{F}$ if and only if all demand graphs $D \in \mathcal{F}$ are realizable in G. We call G path-pairable if it is terminal pairable with respect to $\mathcal{M}$, the set of all perfect matchings on K_n.

This paper deals with upper bound on the minimal value of the maximal degree in a path-pairable graph. The best known lower bound is $\frac{\log n}{\log \log n}$. The best upper bound proved before this paper is $\sqrt{n}$. Upper bound of roughly $5.2 \log n$ is proved in this paper.

Cartesian product of graphs

Vertex set of the Cartesian product $G = G_1 \square G_2 \square \cdots \square G_n$ is the Cartesian product $V(G) = V(G_1) \times V(G_2) \times \cdots \times V(G_n)$. Edge between two vertices $(v_1, v_2, \ldots, v_n)$ and $(w_1, w_2, \ldots, w_n)$ is present if and only if there exists i such that v_i and w_i are adjacent in graph G_i.

Terminal pairability of complete grid graphs

Let $K_t^n = \prod_{i=1}^{n} K_t$ be defined as the Cartesian product of n copies of K_t.

Theorem 1 *Let D be a demand multigraph with $V(D) = K_t^n$ and $\Delta(D) \leq \lfloor \frac{t}{6} \rfloor - 2$ even. Then G is terminal-pairable with respect to D for all D.*

Corollary 2 *K_t^n is path-pairable for $t \geq 24$. When we set $t = 24$, we get an upper bound on minimal value of $\Delta(G) = \log N \frac{t}{\log t} \approx 5.2 \log N$ where N is the number of vertices.*

In the proof of the theorem stated above, we will use the following two theorems.

Theorem 3 *Let $K_t(q)$ be a q-regular demand multigraph of the complete graph K_t. If $q \leq 2\lfloor \frac{6}{t} \rfloor - 4$, then K_t is terminal-pairable with respect to $K_t(q)$.*

Theorem 4 *[1] Let G be a $2k$-regular multigraph. Then $E(G)$ can be decomposed into the union of k edge-disjoint 2-factors.*

The following notation is used in the proof. Let L_i be the subgraph of K_t^n induced by

$$\{(a_1, a_2, \ldots, a_{n-1}, i) \text{ for } 1 \leq a_j \leq t \text{ for } 1 \leq j \leq n-1\}.$$

We call $L_1, L_2, \ldots, L_n$ layers of K_t^n. Similarly we will define columns $l_1, l_2, \ldots, l_{t^{n-1}}$ as the induced K_t that we get by fixing the first $n-1$ coordinates.

Bibliography

[1] Petersen, Julius. Die Theorie der regulären graphs. Acta Math. 15 (1891), 193–220. doi:10.1007/BF02392606. http://projecteuclid.org/euclid.acta/1485881825

List of participants

Deniz Ağaoğlu

Milad Ahanjideh

Onur Çağrıcı

Radovan Červený

Andrej Dedík

Anna Dresslerová

Pavel Dvořák

Zdeněk Dvořák

Andreas Feldmann

Jiří Fiala

Tomáš Gavenčiak

Petr Glivický

Jaroslav Hančl

Milan Hladík

Petr Hliněný

Tereza Klimošová

Dušan Knop

Karel Král

Robert Lukoťka

Leila Parsaei Majd

Tomáš Masařík

Ondřej Mička

Jana Novotná

Jitka Novotná

Ondřej Pangrác

Petra Pelikánová

Robert Šámal

Veronika Slívová

Jakub Svoboda

Josef Svoboda

Jana Syrovátková

Martin Tancer

Jakub Tětek

Tomáš Toufar

Tomáš Valla