Recent Trends in Algorithms

February 7-10, 2019

National Institute of Science Education and Research

Coordinators: Venkatesh Raman, Saket Saurabh and Aritra Banik

School of Computer Sciences, NISER, Bhubaneswar

Urmila Mahadev's work on classical verification of quantum computations

Jaikumar Radhakrishnan

The paper shows how a classical verifier can interact with a single quantum prover and verify the result of an efficient quantum computation. The soundness of the protocol depends on an assumption that learning from errors is hard even for efficient quantum machines. We hope to introduce the main components of this work as outlined in Mahadev's FOCS 2018 paper[1].

References

[1] Urmila Mahadev. Classical verification of quantum computations. In Mikkel Thorup, editor, *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 259–267. IEEE Computer Society, 2018.

Locality Sensitive Orderings

Anil Maheshwari

I will discuss a recent result of Chan, Har-Peled, and Jones that appeared in 10th Innovations in Theoretical Computer Science (ITCS 2019). The main result is that a certain number of linear orderings of a point set can be computed with the following property: for any two points there is an ordering such that all the points between these two specific points in the ordering are very close to one of these two points. This has interesting applications in the computation of approximate nearest neighbors, geometric spanners, approximate MST, etc. For details, see the article in ITCS 2019.

Algorithms with Uncertainty

Ravishankar Krishnaswamy

In this talk, we will survey the design and analysis of online algorithms for two fundamental problems, namely matching and set cover.

An Online Metric Matching instance is given by a bipartite graph of n clients and servers, one or both sides of which are revealed online. A client must be matched to one free server at the time a client is revealed with the goal of minimizing the total cost of the final matching where the costs of (client, server) pairs are assumed to obey the triangle inequality. We will review the rich history of this problem, and present our results on the recourse version of the problem where the online algorithm is allowed to rematch some clients online.

Similarly, the online set cover problem is where the elements arrive online, and we must include sets to cover them irrevocably. We will again go over the classical results, and also present our results on the recourse version of this problem where the algorithm is allowed to make a few changes to the sets it maintains over time.

Based on joint works with: Anupam Gupta, Varun Gupta, Janardhan Kulkarni, Amit Kumar, Debmalya Panigrahi, and Sai Sandeep.

A panorama of scaling algorithms and applications

Ankit Garg

The story starts with Sinkhorn in 1964 who studied the problem of matrix scaling, where the goal is to scale, i.e. multiply rows and columns by scalars, a non-negative matrix to a doubly stochastic one. Since Sinkhorn's work, matrix scaling has been extensively studied and has found applications in numerous areas such as statistics, numerical linear algebra as well as theoretical computer science. Next Gurvits, in early 2000's, generalized these ideas in several directions. One generalization was operator scaling which recently resulted in a deterministic polynomial time algorithm for the non-commutative rational identity testing problem (one other generalization was in the direction of real stable polynomials which resulted in a "proof from the book" of the Van der Waerden conjecture). It turns out these scaling problems arise naturally in the context of group actions and invariant theory which provide a further plethora of scaling problems. Some additional problems that fall under this umbrella are the so called non-uniform scaling problems which include Brascamp-Lieb inequalities, Horn's problem, quantum marginal problem etc. Algorithms have been recently designed for many of these scaling problems as well. We will catch a glimpse of this exciting story.

Spanning tree congestion problem

L. Sunil Chandran [1]

Given a connected graph $G = (V, E)$, let T be a spanning tree. For an edge $e = (u, v) \in E$, its detour with respect to T is the unique path from u to v in T; let $\mathsf{DT}(e, T)$ denote the set of edges in this detour. The stretch of e with respect to T is $|\mathsf{DT}(e, T)|$, the length of its detour. The dilation of T is $\max_{e \in E} |\mathsf{DT}(e, T)|$. The edge-congestion of an edge $e \in T$ is $\mathsf{ec}(e, T) := |\{f \in E : e \in \mathsf{DT}(f, T)\}|$, i.e., the number of edges in E whose detours contain e. The congestion of T is $\mathsf{cong}(T) := \max_{e \in T} \mathsf{ec}(e, T)$. The spanning tree congestion (STC) of the graph G is $\mathsf{STC}(G) := \min_T \mathsf{cong}(T)$, where the minimization is among all spanning trees of G.

We note that there is an equivalent cut-based definition for edge-congestion, which we will use in our proofs. For each tree-edge in $e \in T$, its removal from T results in two connected components; let U_e denote one of the components. The edge-congestion of the edge e is $\mathsf{ec}(e, T) := |E(U_e, V \setminus U_e)|$.

The most general results regarding STC of general graphs is an $\mathcal{O}(n\sqrt{n})$ upper bound by Löwenstein, Rautenbach and Regen in 2009 [1], and a matching lower bound by Ostrovskii in 2004 [2]. Note that the above upper bound is not interesting when the graph is sparse, since there is also a trivial upper bound of m. In our paper we come up with a strong improvement to these bounds after 8 years:

Theorem: For a connected graph G with n vertices and m edges, its spanning tree congestion is at most $\mathcal{O}(\sqrt{mn})$. In terms of average degree $d_{avg} = 2m/n$, we can state this upper bound as $\mathcal{O}(n\sqrt{d_{avg}})$. There is a matching lower bound.

Though in the general case we do not have any polynomial time algorithm for the STC problem, we are able to provide efficient constant factor approximation algorithms for two important cases. In both cases we prove that the spanning tree congestion is $\Theta(n)$ and provide efficient polynomial time algorithms to find spanning trees with congestion $\mathcal{O}(n)$.

- For random graphs $\mathcal{G}(n, p)$ with $1 \geq p \geq \frac{c \log n}{n}$ for some small constant $c > 1$. It should be noted that the STC problem is relevant only for connected graphs and since the threshold function for graph connectivity is $\frac{\log n}{n}$, we are providing the polynomial time algorithm for almost all of the relevant range of values of p.

[1] My presentation will be based on the recent work done in collaboration with Yun Kuen Cheung and Davis Issac, both researchers from Max Planck Institute für Informatik, Saarbrücken, Germany.

- The other important case where we can give a constant factor approximation algorithm is for the class of graphs with minimum degree $(1/2 + f)n$ for any fixed positive constant f.

As a crucial ingredient for the above results, we prove the following lemma:

Lemma 1. *Let G be a k-connected graph with m edges. Then its spanning tree congestion is $\mathcal{O}(m/k)$.*

References

[1] Christian Löwenstein, Dieter Rautenbach, and Friedrich Regen. On spanning tree congestion. *Discrete Mathematics*, 309(13):4653 – 4655, 2009.

[2] M.I Ostrovskii. Minimal congestion trees. *Discrete Mathematics*, 285(1):219 – 226, 2004.

Classified Matchings with one sided preferences

Meghna Nasre*

In this talk, we consider the problem of computing an optimal matching in a bipartite graph where elements of one side of the bipartition specify preferences over the other side, and the other side can have capacities and classifications. The input instance is a bipartite graph G=(A U P,E), where A is a set of applicants, P is a set of posts, and each applicant ranks its neighbors in an order of preference, possibly involving ties. Moreover, each vertex p in P has a quota denoting the maximum number of applicants that can be assigned to p in an allocation of applicants to posts - referred to as a matching. A classification for a post p is a collection of subsets of the neighbors of p. Each subset (class) has a quota denoting the maximum number of applicants from the class that can be matched to p. The goal is to find a matching that is optimal amongst all the feasible matchings, which are matchings that respect quotas of all the posts and classes.

We consider the well-studied notion of popularity and show that a popular amongst feasible matchings (if one exists) can be computed efficiently if the classifications satisfy a certain property. To solve the classified popular matchings problem, we present a framework that involves computing max-flows iteratively in multiple flow networks. We use the fact that, in any flow network, w.r.t. any max-flow the vertices can be decomposed into three disjoint sets and this decomposition is invariant of the flow. This simple fact turns out to be surprisingly useful in the design of our combinatorial algorithm. Besides giving polynomial-time algorithms for classified popular matching problem, our framework unifies prior algorithms from literature on popular matchings without classifications, matching the respective time complexities

*This is joint work with Prajakta Nimbhorkar (CMI, Chennai) and Nada Pulath (IIT Madras).

Group Steiner Problems on Low Treewidth Graphs

Syamantak Das

The *Group Steiner Tree* (GST) problem is a classical problem in network design optimization. In the Edge (Node)-Weighted Group Steiner Tree (EW(NW)-GST) problem, we are given an undirected graph $G = (V, E)$ on n vertices with edge costs $c : E \to \mathbb{R}_{\geq 0}$, a source vertex s and a collection of subsets of vertices, called *groups*, $S_1, \ldots, S_k \subseteq V$. The goal is to find a minimum-cost tree $H \subseteq G$ that connects s to some vertex from each group S_i, for all $i = 1, 2, \ldots, k$. The goal is to find a minimum-cost node set $X \subseteq V$ such that $G[X]$ connects every group to the source.

When G is a tree, both EW-GST and NW-GST admit a polynomial-time $O(\log n \log k)$ approximation algorithm due to the seminal result of [Garg et al. SODA'98 / J.Algorithm]. The matching hardness of

$\log^{2-\epsilon} n$ is known even for tree instances of EW-GST and NW-GST [Halperin and Krauthgamer STOC'03]. In general graphs, all polynomial-time approximation algorithms reduce the problem to a tree instance using the metric-tree embedding, incurring a loss of $O(\log n)$ on the approximation factor [Bartal, FOCS'96; Fakcharoenphol et al., FOCS'03 / JCSS]. This yields an approximation ratio of $O(\log^2 n \log k)$ for EW-GST. Using metric-tree embedding, this factor cannot be improved: The loss of $\Omega(\log n)$ is necessary on some input graphs (e.g. grids and expanders). There are alternative approaches that avoid metric-tree embedding, e.g., the algorithm of [Chekuri and Pal, FOCS'05], which gives a tight approximation ratio, but none of which achieves polylogarithmic approximation in polynomial-time. This state of the art shows a clear lack of understanding of GST in general graphs beyond the metric-tree embedding technique. For NW-GST, for which the metric-tree embedding does not apply, not even a polynomial-time polylogarithmic approximation algorithm is known.

In this talk, based on joint works with Parinya Chalermsook, Guy Even, Bundit Laekhanukit and Daniel Vaz, I shall present a $O(\log n \log k)$ approximation algorithms that run in time $n^{O(tw(G)^2)}$ for both NW-GST and EW-GST, where $tw(G)$ denotes the treewidth of graph G. The key to both results is a different type of "tree-embedding" that produces a tree of much bigger size, but *does not cause any loss on the approximation factor*. Our embedding is inspired by dynamic programming, a technique which is typically not applied to group connectivity problems. We also show that this technique is applicable to Group Steiner problems that require higher connectivity between the source and the groups.

Integer Multicommodity Flow in Series-Parallel Graphs

Nikhil Kumar

We consider the problem of routing multicommodity flows in series-parallel graphs. When each commodity has an associated demand and the cut condition is met, flow can be routed fractionally with congestion 2 and integrally with congestion 5. It is conjectured that integral flow can be routed with congestion 2. In this paper, we make some progress towards it. We show that if G is a parallel composition of paths and every cut has capacity at least twice the demand across it, then all demands can be routed integrally. For a series-parallel graph G if the set of demands has a certain "$levelled$" property and every cut has capacity at least thrice the demand across it, we can route flow integrally. Both these results require that $G + H$ (G is the supply and H the demand graph) be Eulerian. Note that above results are stronger than just congestion 2 routing.

Our algorithm is purely combinatorial, introduces new ideas on how to prove integral flow cut gaps and doesn't require any previous knowledge other than some basic matching theory (algorithm for congestion 5 uses the fact that fractional flow can be routed with congestion 2). The algorithm proceeds by first finding a partial routing and then iteratively improves upon it. If we are not able to route all the flow, we show a violated cut. The improvement step is non-trivial and key to our algorithm.

On Graphs with Minimal Eternal Vertex Cover Number

Veena Prabhakaran

The eternal vertex cover problem is a vertex reconfiguration problem in which each configuration is a vertex cover. This variant of the classical vertex cover problem is modelled as an attacker-defender game where a set of guards on the vertices have to be dynamically reconfigured from one vertex cover to another in every round. The minimum number of guards required to protect a graph from an infinite sequence of attacks is the eternal vertex cover number (evc) of the graph. For a given a graph G and an integer k, it is NP-Hard to check whether $evc(G) \leq k$. However, for any graph G, $mvc(G) \leq evc(G) \leq 2mvc(G)$, where $mvc(G)$ is the minimum vertex cover number of G. Here, we achieve a characterization for a class of graphs that includes chordal graphs and internally triangulated planar graphs for which $evc(G) = mvc(G)$. For some graph classes including biconnected chordal graphs, our characterization leads to a polynomial time algorithm to precisely determine $evc(G)$ and to determine a safe strategy of guard movement in each round of the game with $evc(G)$ guards.

Concentration bounds for randomized incremental construction

Sandeep Sen

By combining several interesting applications of random sampling in geometric algorithms like point location, linear programming, segment intersections, binary space partitioning, Clarkson and Shor [DCG 1989] developed a general framework of randomized incremental construction (RIC). The basic idea is to add objects in a random order and show that this approach yields efficient/optimal bounds on **expected** running time. Even quicksort can be viewed as a special case of this paradigm. However, unlike quicksort, for most of these problems, sharper tail estimates on their running times are not known. Barring some promising attempts in [MSW93,CMS92,Seidel91] the general question remains unresolved.

In this talk, we present a general technique to obtain tail estimates for RIC and and provide applications to some fundamental problems like Delaunay triangulations and construction of Visibility maps of intersecting line segments. The main result is derived from a new and careful application of Freedman's [Fre75] inequality for Martingale concentration that overcomes the bottleneck of the better known Azuma-Hoeffding inequality. Further, we explore instances, where an RIC based algorithm may not have inverse polynomial tail estimates. In particular, we show that the RIC time bounds for trapezoidal map can encounter a running time of $\Omega(n \log n \log\log n)$ with probability exceeding $\frac{1}{\sqrt{n}}$. This rules out inverse polynomial concentration bounds within a constant factor of the $O(n \log n)$ expected running time.

Local Search in geometric optimization

Rajiv Raman

Given a hypergraph $\mathcal{H} = (V, E)$, a support graph is a graph $G = (V, F)$ on the vertex set V, such that the sub-graph induced by each hyperedge $e \in E$ is connected. If the graph G is planar, then this is called a planar support. We prove the existence of a planar support for a class of geometric hypergraphs, namely intersection hypergraphs of non-piercing regions in the plane. These hypergraphs are defined as follows: A region is a connected subset of the plane, whose boundary is defined by a set of disjoint, simple jordan curves. Two regions A and B are said to be non-piercing if $A \setminus B$, and $B \setminus A$ are both connected. Given two collections of non-piercing regions: R, B, the intersection hypergraph is a hypergraph with vertex set R, and hyperedges consisting of a set of regions in R intersecting each region $b \in B$. As a consequence, we derive a unified PTAS for several covering and packing problems on these hypergraphs. The proof also yields a polynomial time algorithm to construct such hypergraphs.

Distance-d Independent Set Problem on Unit Disk Graphs

Gautam K. Das

In this talk, we study the maximum distance-d independent set problem, a variant of the maximum independent set problem, on unit disk graphs. We first show that the problem is NP-hard. Next, we propose a polynomial-time constant-factor approximation algorithm and a PTAS for the problem.

Approximation Schemes for Geometric Coverage Problems

Minati De

Mustafa and Ray [?] showed that a wide class of geometric *set cover (SC)* problems admit a PTAS via local search – this is one of the most general approaches known for such problems. Their result applies if a naturally defined "exchange graph" for two feasible solutions is planar and is based on subdividing this graph via a planar separator theorem due to Frederickson [?]. Obtaining similar results for the related *maximum coverage problem (MC)* seems non-trivial due to the hard cardinality constraint.

In this talk, we provide a way to address the above-mentioned issue. First, we propose a *color-balanced* version of the planar separator theorem. The resulting subdivision approximates locally in each part the global distribution of the colors. Second, we show how this roughly balanced subdivision can be employed in a more careful analysis to strictly obey the hard cardinality constraint. More specifically, we obtain a PTAS for any "planarizable" instance of MC and thus essentially for all cases where the corresponding SC instance can be tackled via the approach of Mustafa and Ray.

This is based on a joint work with Steven Chaplick, Alexander Ravsky, and Joachim Spoerhase.

References

[1] Amit Deshpande, Anand Louis, and Apoorv Vikram Singh. Clustering perturbation resilient instances. *CoRR*, abs/1804.10827, 2018.

Clustering and stability

Amit Deshpande

The k-means clustering problem is NP-hard but often solved efficiently by simple heuristics in practice. An implicit assumption behind optimizing any clustering objective is that the optimal solution would recover the underlying ground-truth clusters. In most real-world datasets, the underlying ground-truth clusters are unambiguous and stable under small input perturbations. As a consequence, the ground-truth clusters satisfy center-proximity, i.e., every point is closer to the center of its own cluster than the center of any other cluster by some multiplicative factor greater than 1. We study the Euclidean k-means objective only over α-center-proximal solutions, for $\alpha > 1$. We show that this problem remains NP-hard even when the optimal clusters are balanced. We complement it with exact algorithms in time exponential in k and the center-proximity parameter but linear in the number of points and the dimension. This talk is based on joint work with Anand Louis and Apoorv Vikram Singh [1].

References

[1] Greg N. Frederickson. Fast algorithms for shortest paths in planar graphs, with applications. *SIAM J. Comput.*, 16(6):1004–1022, 1987.

[2] Nabil H. Mustafa and Saurabh Ray. Improved results on geometric hitting set problems. *Discrete & Computational Geometry*, 44(4):883–895, 2010.

D^2-sampling and k-means clustering

Ragesh Jaiswal

The k-means/median problem is to find good representative points (called centers) for a given set of point in $\mathbb{R}^d$. The D^2-sampling technique is to iteratively pick points (from among the given point set) where the probability of picking a point as a centre in the i^{th} iteration is proportional to the (squared) Euclidean distance of the point from the closest previously chosen centres. In this talk we will discuss the effectiveness of the D^2-sampling technique in the context of k-means/median problem.

Ensuring your favorite player wins: Tournament rigging and bribery

Sushmita Gupta

A knockout tournament is a standard format of competition, ubiquitous in sports, elections and decision making. Such a competition consists of several rounds. In each round, all players that have not yet been eliminated are paired up into matches. Losers are eliminated, and winners proceed to the next round, until only one winner exists. Given that we can correctly predict the outcome of each potential match (modeled by a digraph D), a seeding of the tournament deterministically determines its winner. The inherently competitive nature of these competitions makes it very attractive for bettors.

There are two common types of manipulations, where the common goal is to decide if we can ensure that our favorite player wins by (1) controlling how the various players are paired up via the seeding and/or (2) bribing certain players to alter the outcome of some matches. In this talk we will see new results about both these forms of manipulations and along the way discuss some structural properties and algorithmic techniques.

FPT Algorithms for Corridor Guarding Problems

R Subashini

Given an orthogonal connected arrangement of line-segments, Minimum Corridor Guarding(MCG) problem asks for an optimal tree/closed walk such that, if a guard moves through the tree/closed walk, all the line-segments are visited by the guard. This problem is referred to as Corridor-MST/Corridor-TSP (CMST/CTSP) for the cases when the guarding walk is a tree/closed walk, respectively. The corresponding decision version of MCG is shown to be NP-Complete[2]. The parameterized version of CMST/CTSP referred to as k-CMST/k-CTSP, asks for an optimal tree/closed walk on at most k vertices, that visits all the line-segments. Here, vertices correspond to the endpoints and intersection points of the input line-segments. We show that k-CMST/k-CTSP is fixed-parameter tractable (FPT) with respect to the parameter k. Next, we propose a variant of CTSP referred to as Minimum Link CTSP(MLC), in which the link-distance of the closed walk has to be minimized. Here, link-distance refers to the number of links or connected line-segments with same orientation in the walk. We prove that the decision version of MLC is NP-Complete, and show that the parameterized version, namely b-MLC, is FPT with respect to the parameter b, where b corresponds to the link-distance. We also consider another related problem, the Minimum Corridor Connection (MCC). Given a rectilinear polygon partitioned into rectilinear components or rooms, MCC asks for a minimum length tree along the edges of the partitions, such that every room is incident to at least one vertex of the tree. The decision version of MCC is shown to be NP-Complete[1]. We prove the fixed parameter tractability of the parameterized version of MCC, namely k-MCC with respect to the parameter k, where k is the number of rooms.

References

[1] Hans L Bodlaender, Corinne Feremans, Alexander Grigoriev, Eelko Penninkx, René Sitters, and Thomas Wolle. On the minimum corridor connection problem and other generalized geometric problems. *Computational Geometry*, 42(9):939–951, 2009.

[2] Ning Xu. Complexity of minimum corridor guarding problems. *Information Processing Letters*, 112(17-18):691–696, 2012.

A constant factor approximation for asymmetric TSP

NAVEEN GARG

I will present the paper by Svensson, Tarnawski and Vegh [1] which settles a long standing conjecture by providing a constant factor approximation for asymmetric TSP. It also establishes a constant upper bound on the integrality gap of the natural asymmetric TSP LP.

References

[1] Ola Svensson, Jakub Tarnawski, and Lászl´o A. Végh. A constant-factor approximation algorithm for the asymmetric traveling salesman problem. *CoRR*, abs/1708.04215, 2017.

A 2-Approximation Algorithm for Weighted Feedback Vertex Set in Tournaments

GEEVARGHESE PHILIP

A tournament is a directed graph T such that every pair of vertices in T is connected by an arc. A feedback vertex set is a set S of vertices in T such that deleting S from T gives a directed graph with no directed cycles. We consider the Feedback Vertex Set problem in tournaments. Here the input consists of a tournament T and a weight function w which assigns integer weights to vertices of T. The task is to find a feedback vertex set S in T with the minimum total weight. We give the first polynomial-time factor-2 approximation algorithm for this problem. Assuming the Unique Games conjecture, this is the best possible approximation ratio achievable in polynomial time.

This is joint work with Daniel Lokshtanov, Pranabendu Misra, Joydeep Mukherjee, Fahad Panolan, and Saket Saurabh.

Chamberlin-Courant on Restricted Domains

NEELDHARA MISRA

This talk will be a general introduction to popular choices of restricted domains used in the context of voting, such as single-peaked and single-crossing domains. As a case in point, we will consider the Chamberlin-Courant multiwinner rule, for which the problem of winner determination is well-known to be NP-hard on general profiles. We will see how the structure of single-peaked and single-crossing profiles can be exploited to obtain efficient algorithms for this problem. We will also see how these can be extended to profiles that are "close to" being structured for certain simple notions of distances from a domain. Finally, we will explore issues of manipulative behavior and robustness of voting rules, also in the context of restricted domains.

Recent Trends in Computational Social Choice

Palash Dey

In this talk, we will survey about some of the exciting new research directions in Computational Social Choice. This will include preference elicitation, sampling based election prediction, and various campaign related problems. We will also discuss interesting trending concepts like liquid democracy, participatory budgeting, fairness in voting, distortion of voting rules, etc.

Succinct Data Structures and FM index

Venkatesh Raman

In the area of succinct data structures, one is interested in representing the given data structure using *information theoretically optimal* (plus some lower order term) bits of space and still not compromise on the time taken to support queries and updates.

After a brief overview of succinct data structures for

- maintaining a bit vector to support rank and select queries,
- maintaining a binary tree to support left child, right child and parent queries
- maintaining a subset of a finite universe to support membership and select (find the i-th smallest element) queries

we will describe a practical success story of succinct data structures: Burrows-Wheeler Transform along with an (FM) index, on which efficient string matching queries are supported.

Matroid Secretary Problem

Sourav chakraborty

The classical secretary problem is the following: there are n candidates for a secretarial post. Only one of them has to be selected for the post. All the candidates are interviewed sequentially in a random order. After each interview a score is given to the candidate just interviewed. The goal is to select the candidate with a maximum score. But there is a catch. As soon as a candidate is interviewed, the decision whether to hire or not, has to be taken before the next candidate arrives. If that candidate is hired then the interview process stops. If the candidate is not hired then the next candidate is interviewed, and the candidate that was not hired cannot be hired in the future. Because of this catch, the best that can be done, is to maximize the probability of picking the best candidate.

The Matroid Secretary Problem, introduced by Babaioff et al. (2007) [1], is a generalization of the Classical Secretary Problem. In this problem, elements from a matroid are presented to an on-line algorithm in a random order. Each element has a weight associated with it, which is revealed to the algorithm along with the element. After each element is revealed the algorithm must make an irrevocable decision on whether or not to select it. The goal is to pick an independent set with the sum of the weights of the selected elements as large as possible. Babaio et al gave an algorithm for the Matroid Secretary Problem with a competitive ratio of O(log d), where d is the rank of the matroid. It has been conjectured that a constant competitive ratio is achievable for this problem.

In 2012, in a joint work with Oded Lachish [2], we give an algorithm that has a competitive ratio of $O(\sqrt{logd})$. In 2014, the idea in our previous paper was improved by Lachish to obtain a competitive ratio of O(loglog d) [4]. In 2015 Moran et al [3] used different tools to obtain the same bound of O(loglog d). While the believe is that the best competitive ratio is constant for this problem no progress has been made since 2015. We will look at the different approaches and see how one can expect to get an improved algorithm.

References

[1] Moshe Babaioff, Nicole Immorlica, and Robert Kleinberg. Matroids, secretary problems, and online mechanisms. In Nikhil Bansal, Kirk Pruhs, and Clifford Stein, editors, *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2007, New Orleans, Louisiana, USA, January 7-9, 2007*, pages 434–443. SIAM, 2007.

[2] Sourav Chakraborty and Oded Lachish. Improved competitive ratio for the matroid secretary problem. In Yuval Rabani, editor, *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2012, Kyoto, Japan, January 17-19, 2012*, pages 1702–1712. SIAM, 2012.

[3] Moran Feldman, Ola Svensson, and Rico Zenklusen. A simple $O(\log\log(\text{rank}))$-competitive algorithm for the matroid secretary problem. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4-6, 2015*, pages 1189–1201, 2015.

[4] Oded Lachish. O(log log rank) competitive ratio for the matroid secretary problem. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pages 326–335. IEEE Computer Society, 2014.

Dynamic and Fault-Tolerant Algorithms

Manoj Gupta

In this talk, we will give an introduction to Dynamic and Fault Tolerant Graph Algorithms. For Dynamic Graph Algorithm, we will describe a simple algorithm for approximate maximum matching [1]. For the fault tolerant graph algorithm, we will describe a simple result of single fault tolerant exact distance oracle [2]. This will also help the audience in grasping two talks that will follow ("Dynamic Algorithms for Matching" by N.S. Narayanaswamy and "Fault Tolerant Reachability" by Keerti Chaudhary)

References

[1] Manoj Gupta and Richard Peng. Fully dynamic $(1+\epsilon)$-approximate matchings. In *54th Annual IEEE Symposium on Foundations of Computer Science*, 2013.

[2] Manoj Gupta and Aditi Singh. Generic single edge fault tolerant exact distance oracle. In *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic*, pages 72:1–72:15, 2018.

Fully dynamic maximal matching without 3 length augmenting paths in O($\sqrt{n}$) update time

N.S. Narayanaswamy

We present a randomized algorithm to maintain a maximal matching without 3 length augmenting paths in the fully dynamic setting. Consequently, we maintain a $3/2$ approximate maximum cardinality matching. Our algorithm takes expected amortized $O(\sqrt{n})$ time where n is the number of vertices in the graph when the update sequence is generated by an oblivious adversary. Over any sequence of t edge insertions and deletions presented by an oblivious adversary, the total update time of our algorithm is $O(t\sqrt{n})$ in expectation and $O(t\sqrt{n} + n \log n)$ with high probability. To the best of our knowledge, our algorithm is the first one to maintain an approximate matching in which all augmenting paths are of length at least 5 in $o(\sqrt{m})$ update time. We achieve our results by extending the simple algorithm of Baswana, Gupta, and Sen to maintain a maximal matching in a fully dynamic setting. The relevant papers are at [1, 2].

References

[1] Surender Baswana, Manoj Gupta, and Sandeep Sen. Fully dynamic maximal matching in o(log n) update time. *CoRR*, abs/1103.1109, 2011.

[2] Manas Jyoti Kashyop and N. S. Narayanaswamy. Fully dynamic 3/2 approximate maximum cardinality matching in o(sqrt{n}) update time. *CoRR*, abs/1810.01073, 2018.

Fault-Tolerant Algorithm for Single Source Reachability

Keerti Choudhary

In this talk, we look at the problem of single-source-reachability (SSR) in presence of failures of vertices and edges. In the static setting, SSR can be solved in $O(m+n)$ time, for any directed graph G on n vertices and m edges. To model the scenario of a faulty network, we associate a parameter k with the network such that there are at most k vertices (or edges) that are failed at any stage in the network. The goal is to preprocess the graph G in polynomial time, and compute a compact data structure, that, given any set F of at most k vertices (or edges), efficiently computes vertices reachable from source in the graph $G \setminus F$. We show that for any set F of size k, our algorithm takes $O(2^k n)$ time. Previously the only known construction was for single fault. Our data structure can also be extended to obtain a fault tolerant algorithm for computing SSCs (strongly connected components) after k failures.

Parameterized Algorithms for Longest paths and cycles Above Some Natural Lower Bounds

Saket Saurabh

Recall that the Longest Path problem asks, given an undirected vertex graph G and an integer k, whether G has a path with at least k vertices. Similarly, the Longest Cycle problem asks about a cycle with at least k vertices. We study these problems from the perspective of an "above guarantee" parameterization. In this talk we will see some natural above guarantee parameterizations for Longest Path and Cycle and newly designed parameterized algorithms for it.

FPT algorithms for problems with conflict-free constraints

Roohani Sharma

Consider a class of Vertex Subset Problems where one is given as input a graph G and an integer k (together with more input in some cases), and the goal is to find a set of k vertices, S, such that $G - S$ has some desired property. A few examples of Vertex Subset Problems are Vertex Cover, Feedback Vertex Set, Odd Cycle Transversal, Multicut etc. Additionally, consider the class of Conflict-free versions of these problems, where additionally, the vertex set S is also required to induce an independent set in G. We call these the Conflict-free Vertex Subset Problems.

In this talk, we study some Conflict-free Vertex Subset Problems with the aim of designing Fixed-Parameter Tractable (FPT) algorithms for them. Typically, the design of algorithms for the Conflict-free Vertex Subset Problems requires working on the problem from scratch in the sense that the algorithms for the standard version of the problem (without the conflict-free condition) are not really extensible to encompass the conflict-free constraint. In this talk, we will see a combinatorial tool that helps one to design FPT algorithms for many Conflict-free Vertex Subset Problems in a more robust way, using the FPT algorithms for their standard counterparts. Amongst a plethora of results that this combinatorial tool implies, the most notable contribution is the design of an FPT algorithm for Conflict-free Multicut (also called Stable Multicut in literature) and the design of single-exponential FPT algorithms for Conflict-free s-t Separator (Stable s-t Separator) and Conflict-free Odd Cycle Transversal (Stable Odd Cycle Transversal), all of which were posed as open problems by Marx et al. in TALG 2013. These results appeared at SODA 2018.

Parameterized Distributed Algorithms

Krithika Ramaswamy

In this talk, a recent study of parameterized graph optimization problems in the distributed setting will be discussed. Specifically, parameterized upper and lower bounds for classical graph-theoretic problems will be presented. This talk is based on the manuscript titled *Parameterized Distributed Algorithms* by Ran Ben-Basat, Ken-ichi Kawarabayashi and Gregory Schwartzman [1].

References

[1] Ran Ben-Basat, Ken-ichi Kawarabayashi, and Gregory Schwartzman. Parameterized distributed algorithms. *CoRR*, abs/1807.04900, 2018.

Simultaneous Consecutive Ones Submatrix and Editing Problems

Rani M. R.

A binary matrix M has the consecutive ones property ($C1P$) for rows (resp. columns) if there is a permutation of its columns (resp. rows) that arranges the ones consecutively in all the rows (resp. columns). If M has the $C1P$ for rows and the $C1P$ for columns, then M is said to have the simultaneous consecutive ones property ($SC1P$). Binary matrices having the $SC1P$ plays an important role in theoretical as well as practical applications.

We focus on the classical complexity and fixed parameter tractability of (a) Simultaneous Consecutive Ones Submatrix ($SC1S$) and (b) Simultaneous Consecutive Ones Editing ($SC1E$) [3] problems here. $SC1S$ problems focus on deleting a minimum number of rows, columns and rows as well as columns to establish the $SC1P$, whereas $SC1E$ problems deal with flipping a minimum number of 0-entries, 1-entries and 0-entries as well as 1-entries to obtain the $SC1P$. We show that the decision versions of the $SC1S$ and $SC1E$ problems are NP-complete.

We consider the parameterized versions of the $SC1S$ and $SC1E$ problems with d, being the solution size, as the parameter and are defined as follows. Given a binary matrix M and a positive integer d, d-$SC1S$-R, d-$SC1S$-C and d-$SC1S$-RC problems decide whether there exists a set of rows, columns, and rows as well as columns of size at most d, whose deletion results in a matrix with the $SC1P$. The d-$SC1P$-$0E$, d-$SC1P$-$1E$ and d-$SC1P$-$01E$ problems decide whether there exists a set of 0-entries, 1-entries and 0-entries as well as 1-entries of size at most d , whose flipping results in a matrix with the $SC1P$.

Our main results include:

1. The decision versions of $SC1S$ and $SC1E$ problems are NP-complete.
2. Using bounded search tree technique, certain reductions and related results from the literature [1, 2], we show that d-$SC1S$-R, d-$SC1S$-C, d-$SC1S$-RC and d-$SC1P$-$0E$ are fixed-parameter tractable on binary matrices with run-times $O^*(8^d)$, $O^*(8^d)$, $O^*(2^{O(dlogd)})$ and $O^*(18^d)$ respectively.

We also give improved FPT algorithms for $SC1S$ and $SC1E$ problems on certain restricted binary matrices.

References

[1] Yixin Cao and Dániel Marx. Chordal editing is fixed-parameter tractable. *Algorithmica*, 75(1):118–137, 2016.

[2] Haim Kaplan, Ron Shamir, and Robert E Tarjan. Tractability of parameterized completion problems on chordal, strongly chordal, and proper interval graphs. *SIAM Journal on Computing*, 28(5):1906–1922, 1999.

[3] Marcus Oswald and Gerhard Reinelt. The simultaneous consecutive ones problem. *Theoretical Computer Science*, 410(21-23):1986–1992, 2009.