Report from Dagstuhl Seminar 16221

Algorithms for Optimization Problems in Planar Graphs

Edited by

Jeff Erickson[1], Philip N. Klein[2], Dániel Marx[3], and Claire Mathieu[4]

1 **University of Illinois – Urbana-Champaign, US,** jeffe@illinois.edu
2 **Brown University – Providence, US,** klein@brown.edu
3 **Hungarian Academy of Sciences – Budapest, HU,** dmarx@cs.bme.hu
4 **ENS – Paris, FR,** cmathieu@di.ens.fr

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 16221 "Algorithms for Optimization Problems in Planar Graphs". The seminar was held from May 29 to June 3, 2016. This report contains abstracts for the recent developments in planar graph algorithms discussed during the seminar as well as summaries of open problems in this area of research.

Seminar May 29–June 3, 2016 – http://www.dagstuhl.de/16221
1998 ACM Subject Classification F.2 Analysis of Algorithms and Problem Complexity
Keywords and phrases Algorithms, planar graphs, theory, approximation, fixed-parameter tractable, network flow, network design, kernelization
Digital Object Identifier 10.4230/DagRep.6.5.94

1 Executive summary

Jeff Erickson
Philip N. Klein
Dániel Marx
Claire Mathieu

There is a long tradition of research in algorithms for optimization problems in graphs, including work on many classical problems, both polynomial-time solvable problems and NP-hard problems, e.g. shortest paths, maximum flow and minimum cut, matching, T-joins, disjoint paths, traveling salesman, Steiner tree, graph bisection, vehicle routing, facility location, k-center, and maximum cut. One theme of such research addresses the complexity of these problems when the input graph is required to be a planar graph or a graph embedded on a low-genus surface.

There are three reasons for this theme. First, optimization problems in planar graphs arise in diverse application areas. Second, researchers have discovered that, by exploiting the planarity of the input, much more effective algorithms can be developed – algorithms that are faster or more accurate than those that do not exploit graph structure. Third, the study of algorithms for surface-embedded graphs drives the development of interesting algorithmic techniques. One source of applications for planar-graph algorithms is geographic problems. Road maps are nearly planar, for example, so distances in planar graphs can model, e.g., travel times in road maps. Network design in planar graphs can be used to model scenarios in which cables must be run under roads. Planar graphs can also be used to model metrics on the earth's surface that reflect physical features such as terrain; this aspect of planar graphs has been used in studying wildlife corridors. Another source of applications is image processing. Some algorithms for problems such as image segmentation and stereo involve finding minimum cuts in a grid in which each vertex represents a pixel. Sometimes an aggregation technique (superpixels) coalesces regions into vertices, turning the grid into an arbitrary planar graph. A third example application is VLSI. Algorithmic exploitation of a planar embedding goes back at least to the introduction of maximum flow by Ford and Fulkerson in 1956. Current research can be divided in three parts. For polynomial-time-solvable problems, such as maximum flow, shortest paths, matching, and min-cost circulation, researchers seek planarity-exploiting algorithms whose running times beat those of general-graph algorithms, ideally algorithms whose running times are linear or nearly linear. For NP-hard problems, there are two strategies: fixed-parameter algorithms and approximation algorithms. In all three research subareas, there has recently been significant progress. However, many researchers are expert in only one or two subareas. This Dagstuhl Seminar brought together researchers from the different subareas, to introduce them to techniques from subareas that might be unfamiliar, and to foster collaboration across the subareas. The seminar will thus help to spur further advances in this active and growing area. The scientific program of the seminar consisted of twenty-two talks. Four of these talks were longer (60–90 minute) tutorials overviewing the three main areas of the seminar:

- *Polynomial-time algorithms:* "Tutorial on embedded graph algorithms" (Jeff Erickson) and "Monge property, dense distance graphs and speeding-up max-flow computations in planar graphs" (Piotr Sankowski)
- *Approximation schemes:* "Some techniques for approximation schemes on planar graphs" (Philip Klein)
- *Fixed-parameter tractability:* "The square-root phenomenon in planar graphs" (Dániel Marx)

One of the main goals of the seminar was to encourage collaboration between the three communities, and these well-received tutorials helped by introducing the basics of each of these topics.

The rest of the talks were 25-minute presentations on recent research of the participants. The time between lunch and the afternoon coffee break was left open for individual discussions and collaborations in small groups. An open-problem session was organized on Monday morning. Notes on the presented problems can be found in this report.

2 Table of Contents

3 Overview of Talks

3.1 A PTAS for Planar Group Steiner Tree via Spanner Bootstrapping and Prize Collecting

Mohammad Hossein Bateni (Google – New York, US)

Joint work of Mohammad Hossein Bateni, Erik D. Demaine, MohammadTaghi Hajiaghayi, Dániel Marx
Main reference M. H. Bateni, E. D. Demaine, M. Hajiaghayi, D. Marx, "A PTAS for Planar Group Steiner Tree via Spanner Bootstrapping and Prize Collecting," in Proc. of the 48th Annual ACM SIGACT Symp. on Theory of Computing, pp. 570–583, ACM, 2016; pre-print available from author's webpage.
URL http://dx.doi.org/10.1145/2897518.2897549
URL www.cs.bme.hu/~dmarx/papers/bateni-stoc2016-groupsteiner.pdf

We present the first polynomial-time approximation scheme (PTAS), i.e., $(1+\varepsilon)$-approximation algorithm for any constant $\varepsilon > 0$, for the planar group Steiner tree problem (in which each group lies on a boundary of a face). This result improves on the best previous approximation factor of $O(\log n(\log\log n)^{O(1)})$. We achieve this result via a novel and powerful technique called spanner bootstrapping, which allows one to bootstrap from a superconstant approximation factor (even superpolynomial in the input size) all the way down to a PTAS. This is in contrast with the popular existing approach for planar PTASs of constructing light-weight spanners in one iteration, which notably requires a constant-factor approximate solution to start from. Spanner bootstrapping removes one of the main barriers for designing PTASs for problems which have no known constant-factor approximation (even on planar graphs), and thus can be used to obtain PTASs for several difficult-to-approximate problems.

Our second major contribution required for the planar group Steiner tree PTAS is a spanner construction, which reduces the graph to have total weight within a factor of the optimal solution while approximately preserving the optimal solution. This is particularly challenging because group Steiner tree requires deciding which terminal in each group to connect by the tree, making it much harder than recent previous approaches to construct spanners for planar TSP by Klein (FOCS'05 & SICOMP'08), subset TSP by Klein (STOC'06), Steiner tree by Borradaile, Klein, and Mathieu (SODA'07 & TALG'09), and Steiner forest by Bateni, Hajiaghayi, and Marx (STOC'10 & JACM'11) (and its improvement to an efficient PTAS by Eisenstat, Klein, and Mathieu (SODA'12)). The main conceptual contribution here is realizing that selecting which terminals may be relevant is essentially a complicated prize-collecting process: we have to carefully weigh the cost and benefits of reaching or avoiding certain terminals in the spanner. Via a sequence of involved prize-collecting procedures, we can construct a spanner that reaches a set of terminals that is sufficient for an almost-optimal solution.

Our PTAS for planar group Steiner tree implies the first PTAS for geometric Euclidean group Steiner tree with obstacles, as well as a $(2+\varepsilon)$-approximation algorithm for group TSP with obstacles, improving over the best previous constant-factor approximation algorithms. By contrast, we show that planar group Steiner forest, a slight general- ization of planar group Steiner tree, is APX-hard on planar graphs of treewidth 3, even if the groups are pairwise disjoint and every group is a vertex or an edge.

3.2 Subgraph isomorphism on planar graphs, and related problems

Hans L. Bodlaender (Utrecht University, NL)

Joint work of Hans L. Bodlaender, Jesper Nederlof, Tom van der Zanden

In this talk, we show that the problem, given two planar graphs G and H, to decide if G is isomorphic with a subgraph of H can be solved in $2^{O(n/\log n)}$ time. We also show that this is optimal, assuming the exponential time hypothesis. A similar result holds for other embedding problems, including induced subgraph, minor and induced minor (and weighted variants), and for other graph classes, including graphs avoiding some fixed minor. This is joint work by Hans Bodlaender, Jesper Nederlof and Tom van der Zanden.

3.3 Approximating connectivity domination in weighted bounded-genus graphs

Vincent Cohen-Addad (CNRS / ENS – Paris, FR)

Joint work of Vincent Cohen-Addad, Éric Colin de Verdière, Philipp N. Klein, Claire Mathieu, David Meierfrankenfeld

We present a framework for addressing several problems on weighted planar graphs and graphs of bounded genus. With that framework, we derive polynomial-time approximation schemes for the following problems in planar graphs or graphs of bounded genus: edge-weighted tree cover and tour cover; vertex-weighted connected dominating set, maximum-weight-leaf spanning tree, and connected vertex cover. In addition, we obtain a polynomial-time approximation scheme for feedback vertex set in planar graphs. These are the first polynomial-time approximation schemes for all those problems in weighted embedded graphs. (For unweighted versions of some of these problems, polynomial-time approximation schemes were previously given using bidimensionality.)

3.4 Independent sets in planar graphs

Zdenek Dvorak (Charles University – Prague, CZ)

Joint work of Zdenek Dvorak, Matthias Mnich

It is a long-standing open problem in the algorithmic theory of planar graphs whether there exists a polynomial-time algorithm deciding if an n-vertex planar graph has an independent set of size greater than $n/4$. (An independent set of size at least $n/4$ is guaranteed by the Four Color Theorem.)

In joint work with Matthias Mnich, we investigate related (easier) questions of similar nature. For example we show that

- The problem can be solved under a variety of additional restrictions, e.g., when the considered graphs have maximum degree at most 4 or when they avoid 4- or 5-cycles; and,

- It is possible to decide whether an n-vertex planar triangle-free graph has an independent set of size at least $n/3+k$ in time $2^{O(\sqrt{k})}n^{O(1)}$, which is analogously related to Grotzsch's theorem.

3.5 Tutorial on embedded graph algorithms

Jeff Erickson (University of Illinois – Urbana-Champaign, US)

We consider several fundamental algorithmic tools for exact polynomial-time algorithms for graphs embedded on surfaces. Specific topics include combinatorial embeddings and duality, Euler's formula, the greedy tree-cotree decomposition, systems of loops and cycles, shortest noncontractible and nonseparating cycles, multiple-source shortest paths, homology and homology annotation, enforcing uniqueness of shortest paths, and covering spaces. As applications of these building blocks, we sketch recent algorithms to compute minimum (s,t)-cuts [Chambers, Erickson, Nayyeri 2009], Gomory-Hu trees [Borradaile, Eppstein, Nayyeri, and Wulff-Nilson 2016], and shortest cycle bases [Borradaile, Chambers, Fox, and Nayyeri 2016] in surface-embedded graphs.

3.6 On Temporal Graph Exploration

Thomas Erlebach (University of Leicester, GB)

Joint work of Thomas Erlebach, Michael Hoffmann, Frank Kammer

A temporal graph is a graph in which the edge set can change from step to step. The temporal graph exploration problem TEMPEX is the problem of computing a foremost exploration schedule for a temporal graph, i.e., a temporal walk that starts at a given start node, visits all nodes of the graph, and has the smallest arrival time. We consider only temporal graphs that are connected at each step. For such temporal graphs with n nodes, we show that it is NP-hard to approximate TEMPEX with ratio $O(n^{1-\epsilon})$, and that there are temporal graphs whose exploration requires $O(n^2)$ steps. The underlying graph (i.e. the graph that contains all edges that are present in the temporal graph in at least one step) used in these constructions is dense, which leads to the interesting question of studying TEMPEX for temporal graphs whose underlying graph is planar. We show that even such temporal graphs may require $\Omega(n \log n)$ steps, and that they can always be explored in $O(n^{1.8} \log n)$ steps. For the special case of $2 \times n$ grids, we show that $O(n \log^3 n)$ steps always suffice.

3.7 A Polynomial-time Bicriteria Approximation Scheme for Planar Bisection

Kyle Jordan Fox (Duke University – Durham, US)

Joint work of Kyle Jordan Fox, Philip N. Klein, Shay Mozes

Given an undirected graph with edge costs and node weights, the minimum bisection problem asks for a partition of the nodes into two parts of equal weight such that the sum of edge costs between the parts is minimized. We give a polynomial time bicriteria approximation scheme for bisection on planar graphs.

Specifically, let W be the total weight of all nodes in a planar graph G. For any constant $\varepsilon > 0$, our algorithm outputs a bipartition of the nodes such that each part weighs at most $W(1/2 + \varepsilon)$ and the total cost of edges crossing the partition is at most $(1 + \varepsilon)$ times the total cost of the optimal bisection. The previously best known approximation for planar minimum bisection, even with unit node weights, was $O(\log n)$. Our algorithm actually solves a more general problem where the input may include a target weight for the smaller side of the bipartition.

3.8 Turing Kernelization for Finding Long Paths and Cycles in Restricted Graph Classes

Bart Jansen (TU Eindhoven, NL)

The k-Path problem asks whether a given undirected graph has a (simple) path of length k. We prove that k-Path has polynomial-size Turing kernels when restricted to planar graphs, graphs of bounded degree, claw-free graphs, or to $K_{3,t}$-minor-free graphs. This means that there is an algorithm that, given a k-Path instance (G, k) belonging to one of these graph classes, computes its answer in polynomial time when given access to an oracle that solves k-Path instances of size polynomial in k in a single step. Our techniques also apply to k-Cycle, which asks for a cycle of length at least k.

3.9 Paradigms for obtaining approximation schemes for planar graphs

Philip N. Klein (Brown University – Providence, US)

In addressing an NP-hard problem in combinatorial optimization, one way to cope is to use an *approximation scheme*, an algorithm that, for any given $\epsilon > 0$, produces a solution whose value is within a $1 + \epsilon$ factor of optimal. For many problems on graphs, obtaining such accurate approximations is NP-hard if the input is allowed to be any graph but is tractable if the input graph is required to be planar.

Research on polynomial-time approximation schemes for optimization problems in planar graphs goes back to the pioneering work of Lipton and Tarjan (1977) and Baker (1983). Since then, however, the scope of problems amenable to approximation has broadened considerably. In this talk I will outline some of the approaches used, especially those that have led to recent results.

3.10 The Square Root Phenomenon in Planar Graphs – Survey and New Results

Dániel Marx (Hungarian Academy of Sciences – Budapest, HU)

Most of the classical NP-hard problems remain NP-hard when restricted to planar graphs, and only exponential-time algorithms are known for the exact solution of these planar problems. However, in many cases, the exponential-time algorithms on planar graphs are significantly faster than the algorithms for general graphs: for example, 3-Coloring can be solved in time $2^{O(\sqrt{n})}$ in an n-vertex planar graph, whereas only $2^{O(n)}$-time algorithms are known for general graphs. For various planar problems, we often see a square root appearing in the running time of the best algorithms, e.g., the running time is often of the form $2^{O(\sqrt{n})}$, $n^{O(\sqrt{k})}$, or $2^{O(\sqrt{k})} \cdot n$. By now, we have a good understanding of why this square root appears. On the algorithmic side, most of these algorithms rely on the notion of treewidth and its relation to grid minors in planar graphs (but sometimes this connection is not obvious and takes some work to exploit). On the lower bound side, under a complexity assumption called Exponential Time Hypothesis (ETH), we can show that these algorithms are essentially best possible, and therefore the square root has to appear in the running time.

In the talk, I will present a survey of the basic algorithmic and complexity results, and discuss some of the very recent developments in the area.

3.11 Local search yields an approximation scheme for uniform facility location in edge-weighted planar graphs

Claire Mathieu (ENS – Paris, FR)

Joint work of Vincent Cohen-Addad, Philip N. Klein, Claire Mathieu

We present a polynomial-time approximation scheme (PTAS) for uniform facility location in edge-weighted planar graphs. This is the easiest of several results showing the good performance of local search in Euclidean and minor-free metrics.

3.12 Computing the minimum cut of a weighted directed planar graph

Shay Mozes (Interdisciplinary Center Herzliya, IL)

Joint work of Shay Mozes, Cyril Nikolaev, Yahav Nussbaum, Oren Weimann

We give an $O(n \log \log n)$ time algorithm for computing the minimum cut (or equivalently, the shortest cycle) of a weighted directed planar graph. This improves the previous fastest $O(n \log^2 n)$ solution [SODA'04]. Interestingly, while in undirected planar graphs both min-cut and min st-cut have $O(n \log \log n)$-time solutions [ESA'11, STOC'11], in directed planar graphs our result makes min-cut faster than min *st*-cut, which currently requires $O(n \log n)$ [J. ACM'09].

3.13 Subexponential parameterized algorithms for planar and apex-minor-free graphs via low treewidth pattern covering

Marcin Pilipczuk (University of Warsaw, PL)

Joint work of Fedor Fomin, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, Saket Saurabh
Main reference F. V. Fomin, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, S. Saurabh, "Subexponential parameterized algorithms for planar and apex-minor-free graphs via low treewidth pattern covering," arXiv:1604.05999v1 [cs.DS], 2016.
URL http://arxiv.org/abs/1604.05999v1

We prove the following theorem. Given a planar graph G and an integer k, it is possible in polynomial time to randomly sample a subset A of vertices of G with the following properties: (i) A induces a subgraph of G of treewidth $\sqrt{k} \log k$, and (ii) for every connected subgraph H of G on at most k vertices, the probability that A covers the whole vertex set of H is at least $(2^{O(\sqrt{k} \log^2 k)} \cdot n^{O(1)})^{-1}$, where n is the number of vertices of G.

Together with standard dynamic programming techniques for graphs of bounded treewidth, this result gives a versatile technique for obtaining (randomized) subexponential parameterized algorithms for problems on planar graphs, usually with running time bound $2^{O(\sqrt{k} \log^2 k)} n^{O(1)}$. The technique can be applied to problems expressible as searching for a small, connected pattern with a prescribed property in a large host graph, examples of such problems include Directed k-Path, Weighted k-Path, Vertex Cover Local Search, and Subgraph Isomorphism, among others. Up to this point, it was open whether these problems can be solved in subexponential parameterized time on planar graphs, because they are not amenable to the classic technique of bidimensionality. Furthermore, all our results hold in fact on any class of graphs that exclude a fixed apex graph as a minor, in particular on graphs embeddable in any fixed surface.

Preprint available at http://arxiv.org/abs/1604.05999.

3.14 Optimal parameterized algorithms for planar facility location problems using Voronoi diagrams

Michal Pilipczuk (University of Warsaw, PL)

Joint work of Dániel Marx, Michal Pilipczuk

We study a general family of facility location problems defined on planar graphs and on the 2-dimensional plane. In these problems, a subset of k objects has to be selected, satisfying certain packing (disjointness) and covering constraints. We show that, for each of these problems, the $n^{O(k)}$ time brute force algorithm of selecting k objects can be improved to $n^{O(\sqrt{k})}$ time. The algorithm is based on an approach that was introduced recently in the design of geometric QPTASs, but we show that it can be applied also for exact and parameterized algorithms and for planar graphs. Namely, the idea is to focus on the Voronoi diagram of a hypothetical solution of k objects, guess a balanced separator cycle of this Voronoi diagram to obtain a set that separates the solution in a balanced way, and then recurse on the resulting subproblems. Finally, we also give evidence that the obtained algorithms are essentially optimal, under the Exponential Time Hypothesis.

The extended abstract of the paper appeared in the proceedings of ESA 2015.

3.15 Monge property, dense distance graphs and speeding up max-flow computations in planar graphs

Piotr Sankowski (University of Warsaw, PL)

In my talk, I will introduce the core technique that was used in a series of papers to speed-up max-flow computations in planar graphs. Min-cuts in planar graphs are related to shortest paths via duality. This allows to use simpler shortest path computations for finding minimum-cuts. Especially, it is possible to use a faster implementation of Dijkstra algorithm created by Fakcharoenphol and Rao in 2001. This implementation uses the fact that one do not need to search through shortest paths starting in the same source that would cross. In the algorithm one creates so called dense distance graphs, and needs to search only through square root of edges in such graphs. I will introduce the ideas behind the following three applications of this technique:

- computing all pairs min-cuts in undirected planar graphs in almost linear time by Borradaile, S. and Wulff-Nilsen '10,
- computing s-t max-flows in undirected planar graphs in $O(n \log \log n)$ time by Italiano, Nussbaum, S. and Wulff-Nilsen '11,
- computing single source-all sinks max flows in directed planar graphs by Łącki, Nussbaum, S. and Wulff-Nilsen '12.

3.16 Subexponential algorithms for rectilinear Steiner tree and arborescence problems

Saket Saurabh (The Institute of Mathematical Sciences, IN)

Joint work of Fedor Fomin, Sudeshna Kolay, Daniel Lokshtanov, Fahad Panolan, Saket Saurabh

A rectilinear Steiner tree for a set T of points in the plane is a tree which connects T using horizontal and vertical lines. In the Rectilinear Steiner Tree problem, input is a set T of n points in the Euclidean plane and the goal is to find an rectilinear Steiner tree for T of smallest possible total length. A rectilinear Steiner arborescence for a set T of points and root r in T is a rectilinear Steiner tree S for T such that the path in S from r to any point t in T is a shortest path. In the Rectilinear Steiner Arborecence problem the input is a set T of n points in the Euclidean plane, and a root r in T, the task is to find an rectilinear Steiner arborescence for T, rooted at r of smallest possible total length. In this talk, we give the first subexponential time algorithms for both problems. Our algorithms are deterministic and run in $2^{O(\sqrt{n}\log n)}$ time.

3.17 Embedding Planar Graphs into Low-Treewidth Graphs with Applications to Efficient Approximation Schemes for Metric Problems

Aaron Schild (Berkeley, US)

We give a stretch-$(1+\epsilon)$ embedding of edge-weighted planar graphs of bounded aspect ratio into bounded-treewidth graphs. We use this construction to obtain the first efficient bicriteria approximation schemes for weighted planar graphs addressing a metric generalization of dominating set, r-domination, and a metric generalization of independent set, r-independent set. The approximation schemes employ a metric generalization of Baker's framework based on our embedding result.

3.18 Match-And-Merge: A New Greedy Framework for Maximum Planar Subgraphs

Andreas Schmid (MPI für Informatik – Saarbrücken, DE)

In the maximum planar subgraph (MPS) problem, we are given a graph G, and our goal is to find a planar subgraph H with the maximum number of edges. Besides being a basic problem in graph theory, MPS has many applications including, for instance, circuit design, factory layout, and graph drawing, so it has received a lot of attention from both theoretical and empirical literature. Since the problem is NP-hard, past research has focused on approximation algorithms. The current best known approximation ratio is 4/9 obtained two decades ago based on, roughly speaking, computing as many edge-disjoint triangles in an input graph as possible. The factor 4/9 is also the limit of this "disjoint triangles" approach. We propose two new angles on MPS and provide some evidences that they might lead to improvements over this two-decade-old barrier.

Our first contribution is to initiate a systematic study of a class of greedy algorithms for MPS. Our class of algorithms is rich: All known greedy algorithms for MPS fit into our framework. We argue that these algorithms are unable to perform better than a 7/18-approximation and then show that a slight modification gives a 13/33-approximations, therefore being the first greedy algorithm that beats 7/18.

To facilitate an analytical task in our framework, we formulate a new optimization problem, that we call the Maximum Planar Triangles (MPT) problem. In MPT we are given an input graph and are interested in computing a subgraph that admits a planar embedding with as many triangular faces as possible. We show that MPT is NP-hard and quantify the connection between the two problems. This approach allows potentially up to a 1/2-approximation for MPS, provided the existence of a 1/4-approximation for MPT.

3.19 Face-rooted plane topological minors

Dimitrios M. Thilikos (University of Athens, GR)

Joint work of Petr Golovach, Spyridon Maniatis, Dimitrios M. Thilikos

Let G and H be a (not necessarily connected) plane graphs and let ϕ be a function mapping the faces of G to (some of) the faces of H. We consider the problem asking whether H a *plane* topological minor of G such that, for each face f of H, the pre-images, via ϕ, of f are all subsets of the realization of f in the plane embedding of H in G.

We prove that this problem is fixed parameter tractable when parameterized by the size of H. For this proof we introduce the notion of primal-dual graph and we extend the planar linkage theorem for this type of graphs. Subsequently, we reduce the initial problem to a question on primal-dual linkages that can be answered using suitable extensions of the irrelevant vertex technique for primal-dual graphs.

In our presentation, we stress the the particularities of this problem, mostly emerging from the fact that the graphs in the input of the problem are embedded (i.e., plain) and not planar.

On-going work with Petr Golovach and Spyridon Maniatis.

3.20 Independent set of convex polygons: from n^ϵ to $1 + \epsilon$ via shrinking

Andreas Wiese (MPI für Informatik – Saarbrücken, DE)

Suppose we are given a set of weighted convex polygons in the plane and we want to compute a maximum weight subset of non-overlapping polygons. This is a very natural and well-studied problem with applications in many different areas. Unfortunately, there is a very large gap between the known upper and lower bounds for this problem. The best polynomial time algorithm we know has an approximation ratio of n^ϵ and the best known lower bound shows only strong NP-hardness.

In this paper we close this gap, assuming that we are allowed to shrink the polygons a little bit, by a factor 1-delta for an arbitrarily small constant delta>0, while the compared optimal solution cannot do this (resource augmentation). In this setting, we improve the approximation ratio from n^ϵ to $1+\epsilon$ which matches the above lower bound that still holds if we can shrink the polygons.

3.21 Approximate Distance Oracles for Planar Graphs with Improved Query Time-Space Tradeoff

Christian Wulff-Nilsen (University of Copenhagen, DK)

We consider approximate distance oracles for edge-weighted n-vertex undirected planar graphs. Given fixed epsilon > 0, we present a (1+epsilon)-approximate distance oracle with $O(n(\log \log n)^2)$ space and $O((\log \log n)^3)$ query time. This improves the previous best product of query time and space of the oracles of Thorup (FOCS 2001, J.ACM 2004) and Klein (SODA 2002) from $O(n \log n)$ to $O(n(\log \log n)^5)$.

3.22 Correlation Clustering and Two-edge-connected Augmentation for Planar Graphs

Hang Zhou (MPI für Informatik – Saarbrücken, DE)

Joint work of Philip N. Klein, Claire Mathieu, Hang Zhou

We study two problems. In *correlation clustering*, the input is a weighted graph, where every edge is labelled either $\langle + \rangle$ or $\langle - \rangle$ according to whether its endpoints are in the same category or in different categories. The goal is to produce a partition of the vertices into categories that tries to respect the labels of the edges. In *two-edge-connected augmentation*, the input is a weighted graph and a subset R of edges of the graph. The goal is to produce a minimum weight subset S of edges of the graph, such that for every edge in R, its endpoints are two-edge-connected in $R \cup S$.

For *planar graphs*, we prove that correlation clustering reduces to two-edge-connected augmentation, and that both problems, although they are NP-hard, have a polynomial-time approximation scheme. We build on the *brick decomposition* technique developed recently for optimization problems in planar graphs.

4 Open problems

The following problems were posed at the open-problem session on May 30, 2016. The organizers would like to thank Eli Fox-Epstein for collecting these descriptions from the problem proposers.

4.1 Vertex-disjoint paths

Jeff Erickson (University of Illinois – Urbana-Champaign, US)

COUNTING VERTEX-DISJOINT PATHS

Instance: An undirected graph G embedded on a surface of genus g, a cardinality-k set S of source vertices, and a cardinality-l set T of target vertices.

Question: What is the maximum number of internally vertex-disjoint paths in G with one endpoint in S and one endpoint in T?

Open Problem: Is there a $O(n \operatorname{polylog} n)$-time algorithm for this problem?

Background: There is an $O(n)$ algorithm when $k = l = 1$ and $g = 0$ [17]; this is the only case where a near-linear-time algorithm is known. More generally, maximum flows in vertex-capacitated planar graphs with one source and one sink can be computed in $O(n \log n)$ time [15], but this algorithm breaks down in graphs with more terminals and/or positive genus.

4.2 PTASes for 2-edge-connectivity problems

Philip N. Klein (Brown University – Providence, US)

2-EDGE-CONNECTED SPANNING SUBGRAPH

Instance: undirected planar graph G with edge weights

Question: What is the minimum-cost 2-edge-connected spanning subgraph of G?

STEINER 2-EDGE-CONNECTED SUBGRAPH

Instance: undirected planar graph G with edge weights, subset $T \subseteq V(G)$

Question: What is the minimum cost 2-edge-connected subgraph spanning the terminals T?

2-EDGE-CONNECTED AUGMENTATION

Instance: undirected planar graph G with edge weights, edge subset $A \subseteq E(G)$

Question: What is the minimum cost subgraph where the endpoints of each edge of A are 2-edge-connected?

Open Problems: Are there efficient PTASes for the spanning and augmentation problems? Is there a PTAS for the Steiner version?

Background: There are inefficient PTASes for 2-EDGE-CONNECTED SPANNING SUBGRAPH and 2-EDGE-CONNECTED AUGMENTATION [2, 16].

4.3 Weighted Max Cut

Kyle Jordan Fox (Duke University – Durham, US)

WEIGHTED MAX CUT
Instance: edge-weighted graph G of genus g
Question: What is the maximum weight cut of G?

Open Problem: How quickly can this be solved? Can it be solved in polynomial time? The case $g = O(1)$ with integer weights is especially interesting. Is it FPT in the genus?

Background: The problem can be solved in polynomial time when $g = 0$ [13] and in $2^{O(g)}\operatorname{poly}(|G|)$ time when $g = O(1)$ and all edge weights are equal[10]. The problem is NP-hard for H-minor-free graphs even for unit weights[1]. On H-minor-free graphs where H has a single crossing has a polynomial time algorithm, even with weights [14]. However, if you need to remove two vertices to make H planar, the unweighted case may be NP-hard. For some related dichotomy theorems, see [14].

4.4 FPT Steiner Tree

Dániel Marx (Hungarian Academy of Sciences – Budapest, HU)

STEINER TREE
Instance: edge-weighted planar graph G, vertex subset $T \subseteq V(G)$
Question: What is the minimum-cost tree that includes each vertex of T?

Open Problem: Is there a $1.99^{O(k)}\operatorname{poly}(n)$ or even $2^{O(\sqrt{k}\operatorname{polylog}(k))}\operatorname{poly}(n)$ time FPT algorithm, parametrized by $k = |T|$, to answer this question?

Background: Originally $3^k\operatorname{poly}(n)$ in the general case [7], later improved to $2^k\operatorname{poly}(n)$ [3]. Standard lower bounds show that, assuming ETH, no $2^{o(\sqrt{k})}\operatorname{poly}(n)$ time algorithm is possible.

4.5 Immersion

Hans L. Bodlaender (Utrecht University, NL)

A graph $H = (V_H, E_H)$ is an *immersion* of a graph $G = (V_G, E_G)$ if we can map vertices in V_H to disjoint vertices in V_G such that edges are mapped to edge-disjoint paths between the images of their endpoints. Consider the following problem:

IMMERSION
Instance: graphs G and H
Question: Is G an immersion of H?

What is the running time of this problem when restricted to planar graphs or H-minor free graphs? Similar as for Subgraph Isomorphism, the problem has a lower bound of $2^{\Omega(n/\log n)}$ (with both G and H having $\Theta(n)$ vertices) for planar graphs of pathwidth two [12]; the algorithmic technique from [12] seems not applicable to immersion testing however.

4.6 Treewidth

Hans L. Bodlaender (Utrecht University, NL)

Treewidth
Instance: graph G
Question: What is G's treewidth?

Open Problem: Can we answer this question for planar graphs in polynomial time?

4.7 Independent Sets

Marcin Pilipczuk (University of Warsaw, PL)

Independent Set
Instance: planar graph G
Question: What is the biggest subset of pairwise non-adjacent vertices?

Every planar graph has an independent set of size $n/4$ by the Four Color Theorem. Every triangle-free planar graph has an independent set of size at least $(n+1)/3$; there is a $2^{O(\sqrt{k})}n$)-time algorithm to decide if such a graph has an independent set of size $(n+k)/3$ [8].

Open Problem: is there an FPT algorithm, parameterized by k, to find an independent set of size $n/4+k$ in a planar graph? Is there a polytime algorithm to find an independent set of size $n/4+1$?

This question arose in Dagstuhl Seminars 12241 and 13421 [4, 9].

4.8 Subgraph Isomorphism

Marcin Pilipczuk (University of Warsaw, PL)

Subgraph Isomorphism
Instance: planar graphs G and H
Question: Is H a subgraph of G?

Open Problem: Is this question FPT when parameterized by $|E(G)| - |E(H)|$ or $|V(G)| - |V(H)|$?

Background: Graph isomorphism is in P for planar graphs. It is $\#W[1]$-hard to count all matchings of a planar graph G where exactly k vertices are unmatched [5, 6].

This question arose in Dagstuhl Seminar 13421 [4].

4.9 Exact Distance Labeling

Oren Weimann (University of Haifa, IL)

A *labeling* is an assignment of a (short) value to each vertex in a graph such that the distance between two vertices can be determined from the labels alone.

EXACT DISTANCE LABELING
Instance: an undirected, unweighted planar graph G
Question: How many distinct labels are necessary in a labeling of G?

Open Problem: Can we tighten the bounds on the number of labels necessary?

Background: $O(\sqrt{n})$ and $\Omega(n^{1/3})$ labels are sufficient and necessary, respectively [11]. With edge lengths, the lower bound and upper bound are tight at $\Theta(\sqrt{n})$ labels [11].

4.10 Steiner Minimum Cost Perfect Matching

Sergio Cabello (University of Ljubljana, SI)

STEINER MINIMUM COST PERFECT MATCHING
Instance: planar graph G and vertex subset $S \subseteq V(G)$ of even cardinality
Question: What is the minimum sum of the costs in a perfect matching between vertices of S, where costs are determined by distances in G?

Open Problem: Is there a near-linear-time algorithm for the question? The bottleneck or min-max version is also interesting: minimize the maximum cost over the edges of a perfect matching.

References

1 F. Barahona. The max-cut problem on graphs not contractible to K_5. *Operations Research Letters*, 2(3):107–111, 1983.
2 A. Berger and M. Grigni. Minimum weight 2-edge-connected spanning subgraphs in planar graphs. In *Automata, Languages and Programming*, pages 90–101. Springer, 2007.
3 A. Björklund, T. Husfeldt, P. Kaski, and M. Koivisto. Fourier meets Möbius: Fast subset convolution. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pages 67–74. ACM, 2007.

4 G. Borradaile, P. Klein, D. Marx, and C. Mathieu. *Algorithms for Optimization Problems in Planar Graphs (Dagstuhl Seminar 13421)*. Dagstuhl Reports, 3(10):36–57, 2014.

5 R. Curticapean. The simple, little and slow things count: On parameterized counting complexity. 2015.

6 R. Curticapean and M. Xia. Parameterizing the permanent: Genus, apices, minors, evaluation mod 2k. In *Foundations of Computer Science (FOCS), 2015 IEEE 56th Annual Symposium on*, pages 994–1009. IEEE, 2015.

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

8 Z. Dvořák and M. Mnich. Large independent sets in triangle-free planar graphs. In *Algorithms–ESA 2014*, pages 346–357. Springer, 2014.

9 M. R. Fellows, J. Guo, D. Marx, and S. Saurabh. *Data Reduction and Problem Kernels (Dagstuhl Seminar 12241)*. Dagstuhl Reports, 2(6):26–50, 2012.

10 A. Galluccio, M. Loebl, and J. Vondrák. Optimization via enumeration: a new algorithm for the max cut problem. *Mathematical Programming*, 90(2):273–290, 2001.

11 C. Gavoille, D. Peleg, S. Prennes, and R. Raz. Distance labeling in graphs. *Journal of Algorithms*, 53(1):85–112, 2004.

12 J. Nederlof, H. L. Bodlaender and T. van der Zanden. Subexponential time algorithms for embedding H-minor free graphs. In *Proceedings ICALP 2016*, 2016. to appear.

13 F. Hadlock. Finding a maximum cut of a planar graph in polynomial time. *SIAM Journal on Computing*, 4(3):221–225, 1975.

14 M. Kamiński. Max-cut and containment relations in graphs. *Theoretical Computer Science*, 438:89–95, 2012.

15 H. Kaplan and Y. Nussbaum. Maximum flow in directed planar graphs with vertex capacities. *Algorithmica*, 61(1):174–189, 2011.

16 P. N. Klein, C. Mathieu, and H. Zhou. Correlation clustering and two-edge-connected augmentation for planar graphs. In *LIPIcs-Leibniz International Proceedings in Informatics*, volume 30. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2015.

17 D. Wagner and K. Weihe. A linear-time algorithm for edge-disjoint paths in planar graphs. Combina- torica, 15(1):135–150, 1995.

Participants

- Anna Adamaszek
University of Copenhagen, DK
- Mohammad Hossein Bateni
Google – New York, US
- Hans L. Bodlaender
Utrecht University, NL
- Sergio Cabello
University of Ljubljana, SI
- Vincent Cohen-Addad
CNRS / ENS – Paris, FR
- Zdenek Dvorak
Charles University – Prague, CZ
- Jeff Erickson
University of Illinois – Urbana-Champaign, US
- Thomas Erlebach
University of Leicester, GB
- Fedor V. Fomin
University of Bergen, NO
- Kyle Jordan Fox
Duke University – Durham, US
- Eli Fox-Epstein
Brown Univ. – Providence, US
- Michelangelo Grigni
Emory University, US
- Bart Jansen
TU Eindhoven, NL
- Philip N. Klein
Brown Univ. – Providence, US
- Lukasz Kowalik
University of Warsaw, PL
- Daniel Lokshtanov
University of Bergen, NO
- Dániel Marx
Hungarian Academy of Sciences – Budapest, HU
- Claire Mathieu
ENS – Paris, FR
- Bojan Mohar
Simon Fraser University – Burnaby, CA
- Shay Mozes
Interdisciplinary Center Herzliya, IL
- Marcin Pilipczuk
University of Warsaw, PL
- Michal Pilipczuk
University of Warsaw, PL
- Peter Rossmanith
RWTH Aachen, DE
- Piotr Sankowski
University of Warsaw, PL
- Ignasi Sau Valls
CNRS – Montpellier, FR
- Saket Saurabh
The Institute of Mathematical Sciences, IN
- Aaron Schild
Berkeley, US
- Andreas Schmid
MPI für Informatik – Saarbrücken, DE
- Anastasios Sidiropoulos
Ohio State University – Columbus, US
- Dimitrios M. Thilikos
University of Athens, GR
- Tom van der Zanden
Utrecht University, NL
- Erik Jan van Leeuwen
MPI für Informatik – Saarbrücken, DE
- Oren Weimann
University of Haifa, IL
- Andreas Wiese
MPI für Informatik – Saarbrücken, DE
- Christian Wulff-Nilsen
University of Copenhagen, DK
- Hang Zhou
MPI für Informatik – Saarbrücken, DE
- Anna Zych
University of Warsaw, PL