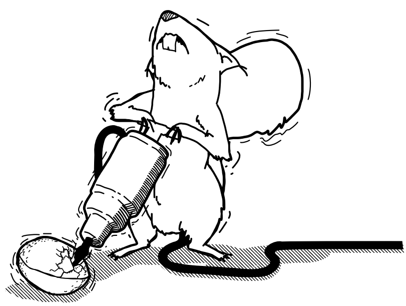
Worker 2015

Workshop on Kernelization

Nordfjordeid, Norway, June 1—June 4



2015
University of Bergen

Worker 2015

Workshop on Kernels, the biannual meeting of the kernelization community, took place June 1—4, 2015 at the Sophus Lie Conference Center and was organized by the algorithms group at the University of Bergen, Department of Informatics.

Worker 2015 is a continuation of the Worker series, with previous events being:

- Worker 2013, Warsaw, Poland
- Worker 2011, Vienna, Austria
- Worker 2010, Lorentz Center, The Netherlands
- Worker 2009, Bergen, Norway

The lectures and talks started on the morning of the 1st of June and ended on the 4th June after lunch.

Organizers

- Pål Grønås Drange
- Markus Dregi
- Fedor Fomin
- Daniel Lokshtanov
- Dániel Marx
- Saket Saurabh

Program

	Monday	Tuesday	Wednesday	Thursday
08:50—09:00	Welcome			
09:00—10:00	**Invited I**	**Invited III**	**Invited V**	**Invited VI**
10:00—10:30	Break	Break	Break	Break
10:30—12:00	Contributed	Contributed	Contributed	Contributed
12:00—14:00	Lunch	Lunch	Lunch	Lunch
14:00—15:00	**Invited II**	**Invited IV**	Excursion	Departure
15:00—16:00	Break	Break	Excursion	
16:00—17:30	What’s next?	O. P. Session	Excursion	
17:30—18:00	Break	Break	Excursion	
18:00—20:00	Dinner	Dinner	Excursion	
20:00—22:00			Banquette	

Monday morning

08:50—09:00 Worker welcome by Saket Saurabh

09:00—10:00 **Invited talk I:** Dániel Marx

Characterizing the Easy-to-Find Subgraphs from the Viewpoint of Polynomial-Time Algorithms, Kernels, and Turing Kernels

Chair: Saket Saurabh

10:00—10:30 Break

10:30—12:00

Fahad Panolan
Kernelization complexity of String Problems

Christophe Paul
FPT Algorithm and Polynomial Kernel for Linear Rankwidth One Vertex Deletion

Travis Gagie
Searching and Indexing Genomic Databases via Kernelization

Chair: Daniel Lokshtanov

12:00—14:00 Lunch

Monday afternoon

14:00—15:00 Invited talk II: Ankur Moitra

Vertex Sparsification: An Introduction, Connections and Applications

Chair: Daniel Lokshtanov

15:00—16:00 Break

16:00—17:30 Kernelization: What's next?

Saket Saurabh

17:30—18:00 Break

18:00—20:00 Dinner

Tuesday morning

09:00—10:00 **Invited talk III:** Marcin Pilipczuk

Sparsification for network design problems in planar graphs

Chair: Dániel Marx

10:00—10:30 Break

10:30—12:00

Ondřej Suchý
Extending the kernel for planar Steiner Tree to the number of Steiner vertices

M. S. Ramanujan
Kernels for Preserving Connectivity

Vincent Froese
Kernelization for Degree Sequence Completion Problems Using f-Factors

Chair: Dániel Marx

12:00—14:00 Lunch

Tuesday afternoon

14:00—15:00 **Invited talk IV:** Robert Krauthgamer

Vertex Sparsification of Cuts, Flows, and Distances

Chair: Michał Pilipczuk

15:00—16:00 Break

16:00—17:30 Open problem session

17:30—18:00 Break

18:00—20:00 Dinner

Wednesday morning

09:00—10:00 **Invited talk V:** Serge Gaspers

Kernelization in constraint satisfaction and reasoning

Chair: Michael R. Fellows

10:00—10:30 Break

10:30—12:00

Markus Dregi
Modulator based kernel for Threshold Editing

Erik Jan van Leeuwen
Polynomial kernel for removing induced claws and diamonds

Bart MP Jansen
Turing kernelization for finding long paths and cycles in restricted graph classes

Chair: Frances A. Rosamond

12:00—14:00 Lunch

Wednesday afternoon

14:00—20:00 Excursion

20:00—22:00 Banquette

Thursday

09:00—10:00 Invited talk VI: Michał Pilipczuk

Kernelization of Dominating Set on sparse graph classes

Chair: Fedor V. Fomin

10:00—10:30 Break

10:30—12:00

Bart M.P. Jansen
Constrained Bipartite Vertex Cover: The Easy Kernel is Essentially Tight

Holger Dell
AND-compression: Streamlined proof

Holger Dell
AND-compression: An application to counting problems

Chair: Markus Dregi

12:00—14:00 Lunch

14:00—15:00 Departure

Invited talks

Kernelization in constraint satisfaction and reasoning

Serge Gaspers

We present a study of kernelization algorithms for important combinatorial problems from various areas in AI. We consider problems from Constraint Satisfaction, Global Constraints, Satisfiability, Nonmonotonic and Bayesian Reasoning under structural restrictions. All these problems involve two tasks: *(i)* identifying the structure in the input as required by the restriction, and *(ii)* using the identified structure to solve the reasoning task efficiently. We show that for most of the considered problems, task *(i)* admits a polynomial-time preprocessing to a problem kernel whose size is polynomial in a structural problem parameter of the input, in contrast to task *(ii)* which does not admit such a reduction to a problem kernel of polynomial size, subject to a complexity theoretic assumption. As a notable exception we show that the consistency problem for the ATMOST-NVALUE constraint admits a polynomial kernel consisting of a quadratic number of variables and domain values. Our results provide a firm worst-case guarantees and theoretical boundaries for the performance of polynomial-time preprocessing algorithms for the considered problems.

The talk is based on joint work with Stefan Szeider.

Vertex Sparsification of Cuts, Flows, and Distances

Robert Krauthgamer

A key challenge in designing graph algorithms is to "compress" a graph G so as to preserve some of its basic properties, such as distances and cuts. Both spanners [Peleg and Schaffer, 1989] and cut sparsifiers [Benczur and Karger, 1996] fall into this category, as they reduce the number of edges in G without changing any distance or cut by more than a small factor.

I will discuss another flavor of this challenge, which asks instead to reduce the number of vertices. Specifically, given a graph G and k terminal vertices, we wish to construct a small graph G' that contains the terminals, such that between the terminals in G and G', all cuts/flows/distances are equal. Can we bound the size of G' by a function of k? And what if G' only needs to approximate G (say within $1 + \epsilon$)?

I plan to survey recent progress in this emerging area.

Characterizing the Easy-to-Find Subgraphs from the Viewpoint of Polynomial-Time Algorithms, Kernels, and Turing Kernels

Dániel Marx

We study two fundamental problems related to finding subgraphs: given graphs G and H, test whether H is isomorphic to a subgraph of G, or determine the maximum number of vertex-disjoint H-subgraphs that can be packed in G. We investigate these problems when the graph H belongs to a fixed hereditary family F. Our goal is to study which classes F make the two problems tractable in one of the following senses: *(a)* (randomized) polynomial-time solvable, *(b)* admits a polynomial (many-one) kernel, or *(c)* admits a polynomial Turing kernel.

Vertex Sparsification: An Introduction, Connections and Applications

Ankur Moitra

The notion of exactly (or approximately) representing certain combinatorial properties of a graph G on a simpler graph is ubiquitous in combinatorial optimization. In this talk, I will introduce the notion of vertex sparsification. Here we are given a graph $G = (V, E)$ and a set of terminals K and our goal is to find one single graph $H = (K, E_H)$ on just the terminal set so that H approximately preserves minimum cuts between every bipartition of the terminals, as well as minimum congestion routings of every multi-commodity flow problem.

I will prove that good vertex sparsifiers exist and in fact the approximation factor is independent of the size of the original graph (and is sub-logarithmic in the number of terminals). Moreover the approximation factor can be improved to a constant for any graph that excludes a fixed minor. I will give applications to approximation algorithms, including a master theorem for flow-cut gaps. Lastly, I will give lower bounds for vertex sparsification and open questions. This talk will be based on a number of connections to metric embeddings.

Sparsification for network design problems in planar graphs

Marcin Pilipczuk

A well-established technique for designing polynomial-time approximation schemes (PTASes) for connectivity or network design problems on planar graphs is to first establish a spanner—a subgraph or minor of the input graph that contains a near-optimal solution, while at the same time has total cost bounded by $f(\epsilon) \cdot \text{opt}$ (where ϵ is the accuracy parameter and opt the optimum cost)—and then use a variant of the Baker's shifting technique to solve the problem on the spanner only. This approach turned out to be very fruitful, yielding PTASes on planar graph for such problems as SUBSET TSP, STEINER TREE, STEINER FOREST. Moreover, recently, due to connections between cut problems in the primal graph and network design problems in the dual graph, we have seen a PTAS for MULTIWAY CUT and a bicriteria PTAS for BISECTION.

In our recent work with Michał Pilipczuk, Piotr Sankowski, and Erik Jan van Leeuwen, we used the insight from the aforementioned approximation schemes to design improved exact algorithms for (UNWEIGHTED) STEINER TREE in planar graphs: a subexponential algorithm and a polynomial kernel for the parameterization by the number of edges in the solution tree. Some results generalize to bounded-genus graphs, and also to the STEINER FOREST and MULTIWAY CUT problems. In the talk I will give an overview of both the spanner framework for designing PTASes on planar graphs and of recent results in parameterized world, focusing on the Steiner Tree problem.

Kernelization of Dominating Set on sparse graph classes

Michał Pilipczuk

During the talk we shall present a new approach to kernelization of DOMINATING SET on sparse graph classes, which enables us to show a linear kernel for the problem on every monotone graph class of bounded expansion, and an almost linear kernel for every monotone nowhere dense graph class. Since the class of H-topological-minor-free graphs has bounded expansion for every fixed H, these findings generalize all the previous results on linear kernelization for DOMINATING SET on sparse graph classes. However, the improvement is not only in the generality but also in the simplicity: the new approach is conceptually much easier and relies only on basic properties of graph classes of bounded expansion, and not on deep decomposition theorems, as was the case for H-(topological)-minor-free.

The talk will be based on a joint work with Pål Grønås Drange, Markus Dregi, Fedor V. Fomin, Stephan Kreutzer, Daniel Lokshtanov, Marcin Pilipczuk, Felix Reidl, Fernando Sánchez Villaamil, Saket Saurabh, and Somnath Sikdar.

Contributed talks

AND-compression: Streamlined proof and an application to counting problems

Holger Dell

We present Drucker's (FOCS 2012) proof in a streamlined fashion. His result implies that, unless the unlikely complexity-theoretic collapse coNP is in NP/poly occurs, there is no AND-compression for SAT. We observe that the proof even excludes AND-compressions that are only required to work when at most one instance is a no-instance. This observation turns out to be convenient when pondering over kernel lower bounds for counting problems: In ongoing joint work with Marx, we show that the permanent in bipartite graphs does not have polynomial kernels (the parameter is the size of the smaller side).

Modulator based kernel for Threshold Editing

Markus Dregi

We show that the problem of *editing to a threshold graph*, i.e., adding and deleting as few edges as possible to obtain a threshold graph is NP-complete, thereby solving a long-standing open problem in the field of graph modification problems. This problem has been repeatedly stated as open, and renewed interest appeared very recently in the field of social network theory (Brandes 2014), where it has been suggested as a good basis for an axiomatic centrality measure.

We show that the problems THRESHOLD EDITING, COMPLETION, and DELETION, as well as the corresponding problems for chain graphs, all admit polynomial kernels with $O(k^2)$ vertices. This answers a recent question by Liu, Wang and Guo, who asked whether the previously known kernel for THRESHOLD COMPLETION could be improved from $O(k^3)$ to $O(k^2)$. The main technique applied for obtaining polynomial kernels is that of a *vertex modulator* which allows for extracting structure.

Joint work. Pål Grønås Drange, Daniel Lokshtanov, Blair D. Sullivan.

Kernelization for Degree Sequence Completion Problems Using f-Factors

Vincent Froese

We answer an open question by Mathieson and Szeider [JCSS 2012] concerning the existence of polynomial-size kernels for the Degree Constraint Editing Problem. Given a graph G together with a list of admissible degrees for each vertex, the task is to modify G by applying at most k editing operations such that, for each vertex in the resulting graph, its degree is an element of its list.

We give an $O(kr^2)$-vertex kernel for the case of edge additions where r is the maximum admissable degree over all vertices. We further transfer this kernel into an $O(r^5)$-vertex kernel using a strategy based on previous work by Liu and Terzi [SIGMOD 2008] and Hartung et al. [ICALP 2013]. The method relies on transforming the graph completion problem into an efficiently solvable number problem. A solution for the number problem is then translated back into the graph setting using f-factor computations.

As an additional contribution, we generalize our methods to obtain a framework that is applicable to a wider range of degree sequence completion problems.

Joint work. Joint work with André Nichterlein and Rolf Niedermeier.

Searching and Indexing Genomic Databases via Kernelization

Travis Gagie

For twenty years researchers have been speeding up pattern-matching algorithms with the observation that if a pattern occurs in a text, then its first occurrence touches a boundary between LZ77 phrases. As long as no one was working with massive repetitive datasets this observation remained largely a theoretical curiosity, but bioinformaticians are now rediscovering it an applying it to search and index genomic databases. In this talk we argue that techniques based on this observation should be viewed as kernelizations, and discuss how this perspective may help us achieve new results.

Turing kernelization for finding long paths and cycles in restricted graph classes

Bart MP Jansen

We present a set of results concerning the existence of polynomial-size Turing kernels for the problems of finding long simple paths and cycles in restricted graph families such as planar graphs, bounded-degree graphs, and claw-free graphs. The adaptive Turing kernelization works on the Tutte decomposition of the graph into triconnected components. Existing graph-theoretical lower bounds on the circumference of triconnected graphs allow us to answer YES if the Tutte decomposition has a bag whose size exceeds some fixed polynomial in k. If this is not the case, we can identify a vertex that is irrelevant to the solution in polynomial time by querying an oracle for long paths or cycles in leaf components of the decomposition. The number of vertices involved in these queries is therefore polynomial in k, resulting in polynomial-size Turing kernels.

The employed method is called Decompose-Query-Reduce and is potentially useful for other problems. Our Turing kernel shows that on these restricted graph classes, the hard part of the computation for finding a length-k path or cycle can be restricted to graphs whose size is polynomial in k.

Constrained Bipartite Vertex Cover: The Easy Kernel is Essentially Tight

Bart MP Jansen

The Constrained Bipartite Vertex Cover problem asks, for a bipartite graph G with partite sets A and B, and integers k_A and k_B, whether there is a vertex cover for G containing at most k_A vertices from A and k_B vertices from B. The problem has an easy kernel with $2k_A \cdot k_B$ edges and $4k_A \cdot k_B$ vertices, based on the fact that every vertex in A of degree more than k_B has to be included in the solution, together with every vertex in B of degree more than k_A.

We prove that this kernel is asymptotically essentially optimal, both in terms of the number of vertices and the number of edges. We prove that if there is a polynomial-time algorithm that reduces any instance (G, A, B, k_A, k_B) to an equivalent instance (G', A', B', k'_A, k'_A) such that $k'_A \in (k_A)^{O(1)}$, $k'_B \in (k_B)^{O(1)}$, and $|V(G')| \in (k_A \cdot k_B)^{1-\epsilon}$, for any $\epsilon > 0$, then $\mathsf{NP} \subseteq \mathsf{coNP/poly}$ and the polynomial-time hierarchy collapses. Using a different construction, we prove that if there is a polynomial-time algorithm that reduces any instance to an equivalent instance with $O((k_A \cdot k_B)^{1-\epsilon})$ edges, then $\mathsf{NP} \subseteq \mathsf{coNP/poly}$.

Polynomial kernel for removing induced claws and diamonds

Erik Jan van Leeuwen

A graph is called {claw-diamond}-free if it contains neither a *claw* (a $K_{1,3}$) nor a *diamond* (a K_4 minus an edge) as an induced subgraph. Equivalently, {claw-diamond}-free graphs can be characterized as line graphs of triangle-free graphs, or as linear dominoes, i.e., graphs in which every vertex is in at most two maximal cliques and every edge is in exactly one maximal clique. In the {CLAW-DIAMOND}-FREE EDGE DELETION problem we are given a graph G and a parameter k, and the question is whether one can remove at most k edges from G to obtain a {claw-diamond}-free graph.

In joint work with Marek Cygan, Marcin Pilipczuk, Michal Pilipczuk, and Marcin Wrochna, we prove that {CLAW-DIAMOND}-FREE EDGE DELETION admits a polynomial kernel. We complement this by proving that, even on instances with maximum degree 6, the problem is NP-complete and cannot be solved in time $2^{o(k)}|V(G)|^{O(1)}$ unless the Exponential Time Hypothesis fails.

In the talk, we also provide an outlook on where {CLAW-DIAMOND}-FREE EDGE DELETION fits in the quest towards a polynomial kernel for CLAW-FREE EDGE DELETION.

Kernelization complexity of String Problems

Fahad Panolan

In Closest String problem we are given an alphabet Σ, a set of strings $S = \{s_1, s_2, \dots, s_k\}$ over Σ such that $|s_i| = n$ and an integer d. The objective is to check whether there exists a string s over Σ such that $d_H(s, s_i) \leqslant d$, $i \in \{1, \dots, k\}$, where $d_H(x, y)$ denotes the number of places strings x and y differ at. Closest String is a prototype string problem. This problem together with several of its variants such as Distinguishing String Selection and Closest Substring have been extensively studied from parameterized complexity perspective.

These problems have been studied with respect to parameters that are combinations of k, d, $|\Sigma|$ and n. However, surprisingly the kernelization question for these problems (for the versions when they admit fixed parameter tractable algorithms) is not studied at all. In this paper we fill this gap in the literature and do a comprehensive study of these problems from kernelization complexity perspective. We almost settle all the problems by either obtaining a polynomial kernel or showing that the problem does not admit a polynomial kernel assuming a complexity theoretic assumption.

FPT Algorithm and Polynomial Kernel for Linear Rank-width One Vertex Deletion

Christophe Paul

We consider the class of graphs having linear rank-width one, also known as thread graphs, and investigate a related graph modification problem called the Thread Vertex Deletion. In this problem, given an n vertex graph G and a positive integer k, we want to decide whether there is a set of at most k vertices whose removal turns G into a thread graph and if one exists, find such a vertex set. While the meta-theorem of Courcelle, Makowsky, Rotics implies that Thread Vertex Deletion can be computed in time $f(k) \cdot n^3$, it is not clear whether this problem allows a runtime with a modest exponential function. We establish that Thread Vertex Deletion can be solved in time $8^k \cdot n^{\mathcal{O}(1)}$. The major obstacle to this end is how to handle a long induced cycle as an obstruction. To fix this issue, we define a graph class called the necklace graphs and investigate its structural properties. We also show that the Thread Vertex Deletion has a polynomial kernel.

Joint work. M. Kanté, EJ Kim and O. Kwon

Kernels for Preserving Connectivity

M. S. Ramanujan

In connectivity augmentation problems, the input is a (multi) graph and the objective is to increase edge or vertex connectivity by adding the minimum number (weight) of additional edges, called links.

This problem was first studied by Eswaran and Tarjan (1976) who showed that increasing the edge connectivity of a given graph to 2 by adding minimum number of links (also called an augmenting set) is polynomial time solvable. Subsequent work by Watanabe (1987), and Frank (1992) showed that this problem is also polynomial time solvable for any given target value of edge connectivity to be achieved. However, if the set of links is restricted, that is, there are pairs of vertices in the graph which do not constitute a link, or if the links have (non-identical) weights on them, then the problem of computing the minimum size (or weight) augmenting set is NP-complete.

In this work, we study the kernelization complexity of the following problem: Given a $(p+1)$-(edge) connected graph H, a set L of edges of H (called links) such that $H - L$ is p-edge connected, and an integer k, can we delete at least k links such that the resulting graph is still $(p+1)$-edge connected?

We show that the problem admits a kernel with $12k$ vertices and $3k$ links when the graph H has even-connectivity and a kernel with $O(k^2)$ vertices and $O(k^2)$ links when H has odd-connectivity.

Joint work. This is joint work with Manu Basavaraju, Fedor Fomin, Petr Golovach, Pranabendu Misra and Saket Sarah.

Extending the kernel for planar Steiner Tree to the number of Steiner vertices

Ondřej Suchý

In the Steiner Tree problem one is given an undirected graph, a subset of its vertices T, and an integer k and the question is whether there is a connected subgraph of the given graph containing all the vertices of T and at most k other vertices. The vertices in the subset T are called terminals and the other vertices are called Steiner vertices. Recently, Pilipczuk, Pilipczuk, Sankowski, and van Leeuwen (FOCS 2014) gave a polynomial kernel for Steiner Tree in planar graphs, when parameterized by $|T| + k$, the total number of vertices in the constructed subgraph.

In this talk we present several polynomial time applicable reduction rules for planar Steiner Tree. We then show that in a yes-instance reduced with respect to the presented reduction rules, the number of terminals $|T|$ is at most cubic in the number of other vertices k in the subgraph. Hence, using and improving the result of Pilipczuk et al., we give a polynomial kernel for Steiner Tree in planar graphs for the parameterization by the number k of Steiner vertices in the solution.

Participants

- Akanksha Agrawal (University of Bergen)
- Amer Mouawad (University of Bergen)
- Ankur Moitra (Massachusetts Institute of Technology)
- Anthony Perez (LIFO – Université d’Orléans)
- Bart M. P. Jansen (Eindhoven University of Technology)
- Cristina Bazgan (Universite Paris-Dauphine)
- Christophe Paul (CNRS – LIRMM)
- Daniel Lokshtanov (University of Bergen)
- Dániel Marx (Hungarian Academy of Sciences)
- Edouard Bonnet (Hungarian Academy of Sciences (MTA SZTAKI))
- Erik Jan van Leeuwen (MPI Saarbrücken)
- Eunjung Kim (CNRS – Paris Dauphine)
- Fahad Panolan (Institute of Mathematical Sciences, India)
- Faisal N. Abu-Khzam (Lebanese American University)
- Fedor Fomin (University of Bergen)
- Felix Reidl (RWTH Aachen University)
- Fernando Sanchez Villaamil (RWTH Aachen)

- Florian Sikora (Lamsade, Paris-Dauphine)
- Frances Rosamond (Charles Darwin University)
- Hans-Joachim Böckenhauer (ETH Zürich, Department of Computer Science)
- Henning Fernau (Universität Trier)
- Holger Dell (Saarland University and Cluster of Excellence (MMCI))
- Ioan Todinca (Univ. Orléans, France)
- Ivan Bliznets (St. Petersburg Department of V.A. Steklov Institute of Mathematics of the RAS)
- Jakub Gajarský (Masaryk University, Brno)
- Jan Obdrzalek (Masaryk University, Brno)
- Jisu Jeong (KAIST)
- Manu Basavaraju (MPI Saarbrücken)
- Manuel Sorge (TU Berlin)
- Marcin Pilipczuk (University of Warwick)
- Marcin Wrochna (University of Warsaw)
- Mark Jones (Royal Holloway, University of London)

- Markus Dregi (University of Bergen)
- Michael Fellows (Charles Darwin University)
- Michał Pilipczuk (University of Warsaw)
- Mithilesh Kumar (University of Bergen)
- Ondra Suchý (Faculty of Information Technology, Czech Technical University Prague, Czech Republic)
- Pål Grønås Drange (University of Bergen)
- Petr Golovach (University of Bergen)
- Petr Hliněný (Masaryk University, Brno, CZ)
- Petteri Kaski (Aalto University, Helsinki)
- Prafullkumar P Tale (The Institute of Mathematical Sciences, Chennai. India)
- Pranabendu Misra (IMSc, Chennai)
- Ramanujan M. S. (University of Bergen)
- Reza Saei (UiB)
- Robert Ganian (Vienna University of Technology)
- Robert Krauthgamer (Weizmann Institute of Science)
- Rod Downey (Victoria University Wellington)

- Saket Saurabh (IMSc+UIB)
- Serge Gaspers (UNSW Australia and NICTA)
- Stefan Fafianie (University of Bonn)
- Sudeshna Kolay (Institute of Mathematical Sceinces)
- Till Fluschnik (TU Berlin)
- Torstein Strømme (UiB)
- Travis Gagie (University of Helsinki)
- Vincent Froese (TU Berlin)
- Vuong Anh Quyen (Bonn University, Germany)

UNIVERSITY OF BERGEN

Department of Informatics