TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 06/08 pp374-386
Volume 19, Number 4, August 2014

Social Choice Meets Graph Drawing: How to Get Subexponential Time Algorithms for Ranking and Drawing Problems

Henning Fernau*, Fedor V. Fomin, Daniel Lokshtanov,
Matthias Mnich, Geevarghese Philip, and Saket Saurabh

Abstract: We analyze a common feature of p-Kemeny AGGregation (p-KAGG) and p-One-Sided Crossing Minimization (p-OSCM) to provide new insights and findings of interest to both the graph drawing community and the social choice community. We obtain parameterized subexponential-time algorithms for p-KAGG—a problem in social choice theory—and for p-OSCM—a problem in graph drawing. These algorithms run in time $\mathcal{O}^*(2^{\mathcal{O}(\sqrt{k}\log k)})$, where k is the parameter, and significantly improve the previous best algorithms with running times $\mathcal{O}^*(1.403^k)$ and $\mathcal{O}^*(1.4656^k)$, respectively. We also study natural "above-guarantee" versions of these problems and show them to be fixed parameter tractable. In fact, we show that the above-guarantee versions of these problems are equivalent to a weighted variant of p-directed feedback arc set. Our results for the above-guarantee version of p-KAGG reveal an interesting contrast. We show that when the number of "votes" in the input to p-KAGG is *odd* the above guarantee version can still be solved in time $\mathcal{O}^*(2^{\mathcal{O}(\sqrt{k}\log k)})$, while if it is *even* then the problem cannot have a subexponential time algorithm unless the exponential time hypothesis fails (equivalently, unless FPT = M[1]).

Key words: Kemeny aggregation; one-sided crossing minimization; parameterized complexity; subexponential-time algorithms; social choice theory; graph drawing; directed feedback arc set

1 Introduction

In this paper, we link two seemingly different areas of algorithmics: computational social choice and graph drawing. In both areas, ordering of permutations plays some role. This was already observed by Biedl et al.[1], but not as fully exploited as we do it here. We study such problems in the realm of parameterized complexity, yielding subexponential algorithms, as well as in that of approximability, obtaining Polynomial Time Approximation Scheme (PTAS) results. We also provide an extensive list of references, allowing the reader to follow down the different areas of computer science that we touch.

1.1 Computational social choice

Many voting schemes are known to be computationally hard, as already shown by Bartholdi et al.[2] Betzler et al.[3] and Simjour[4] considered Kemeny scores under the point of view of parameterized algorithms. We show here that the standard parameterization of this minimization problems allows for better parameterized algorithms. This problem offers a variety

• Henning Fernau is with FB 4—Abteilung Informatikwissenschaften, Universität Trier, D-54286 Trier, Germany. E-mail: fernau@uni-trier.de.
• Fedor Fomin, Daniel Lokshtanov, and Saket Saurabh are with the Department of Informatics, University of Bergen, N-5020 Bergen, Norway. E-mail: fomin@ii.uib.no; daniello@ii.uib.no.
• Matthias Mnich and Geevarghese Philip are with Max-Planck-Institut für Informatik, D-66123 Saarbrücken, Germany. E-mail: gphilip@mpi-inf.mpg.de; mmnich@mpi-inf.mpg.de.
• Saket Saurabh is also with the Institute of Mathematical Sciences, Chennai - 600113, India. E-mail: saket@imsc.res.in.
∗ To whom correspondence should be addressed.
Manuscript received: 2014-06-22; accepted: 2014-07-04

of applications.

The Dodgon's Score Problem (DSP) only differs from the Kemeny score problem the way two votes are compared: Within DSP, the distance between two votes, interpreted as permutations π_1 and π_2 is given by the minimum number of transpositions whose composition τ satisfies $\tau \circ \pi_1 = \pi_2$. The Dodgson voting scheme was first considered by Bartholdi et al.[2] under complexity-theoretic viewpoints, whilst first considerations from the viewpoint of parameterized algorithms are derived by Betzler et al.[5] However, our approach (and the problem statement) here is different from that in Ref. [5], since we will only consider aggregation-type problems. Further issues regarding parameterized complexity of voting schemes can be found in Refs. [6, 7]. Computational experiments and complexity results, in particular concerning approximability, are reported by McCabe-Dansted in Ref. [8].

Kemeny aggregation

Preference lists are extensively used in social science surveys and voting systems to capture information about choice. In many such scenarios there arises the need to combine the data represented by many lists into a single list which reflects the opinion of the surveyed group as much as possible. The p-Kemeny AGGregation (p-KAGG) problem was introduced by Kemeny and Snell[9, 10] to abstract out the problem of combining many preference lists into one. This problem appears in a variety of applications, from rank aggregation methods for the web[11] up to breeding problems in agronomy[12]. In p-KAGG we are given a set of permutations (also called votes) over a set of alternatives (also called candidates), and a positive integer k, and are asked for a permutation π of the set of candidates, called an *optimal aggregation*, such that the sum of the Kendall-Tau distances (KT-distances) of π from all the votes is at most k. The KT-distance between two permutations π_1 and π_2 is the number of pairs of candidates that are ordered differently in the two permutations and is denoted by KT-dist(π_1, π_2). The problem is known to be NP-complete[2] and admits PTASs[13]. Betzler et al.[3] considered this problem from the point of view of parameterized algorithms and obtained an algorithm that runs in time $\mathcal{O}^*(1.53^k)$ which suppresses polynomial terms in the expression. Simjour[4] obtained an algorithm for the problem that runs in time $\mathcal{O}^*(1.403^k)$. Independently from our exposition in the conference version of this paper[14], Karpinski and Schudy[15] obtained an algorithm for p-KAGG that runs in $\mathcal{O}^*(2^{\mathcal{O}(\sqrt{k})})$ time.

1.2 Graph drawing

The graph drawing problem that we are mainly interested in is the p-One-Sided Crossing Minimization (p-OSCM) problem, which is a key ingredient of the well-known "Sugiyama approach" to layered graph drawing[16]. After the first phase (the assignment of the vertices to layers), the order of the vertices within the layers has to be fixed such that the number of the corresponding crossings of the edges between two adjacent layers is minimized. Finally, the concrete position of the vertices within the layers is determined according the previously computed order. The crossing minimization step alone, although most essential in the Sugiyama approach, comprises an NP-complete problem. The most commonly used method is the layer-by-layer sweep heuristics where, starting from $i = 1$, the order for L_i is fixed and the order for L_{i+1} that minimizes the number of crossings amongst the edges between layers L_i and L_{i+1} is determined. After increasing index i to the maximum layer index, we turn around and repeat this process from the back with decreasing indices. In each step, a p-OSCM problem has to be solved. For an introduction into this specific area of graph drawing, we refer the interested reader to the exposition of Bastert and Matuszewski[17].

One-sided crossing minimization

An input to this problem consists of a bipartite graph $G = (V_1, V_2, E)$, a permutation π of V_1, and a positive integer k. The vertices of V_1 are placed on a line, also called a *layer*, in the order induced by π. The objective is to check whether there is a permutation π_m of V_2 such that, when the vertices of V_2 are placed on a second layer parallel to the first one in the order induced by π_m, then drawing a straight-line segment for each edge in E will introduce no more than k pairwise edge crossings. As shown by Eades and Wormald, this seemingly simple problem is NP-complete[18], even on sparse graphs[19].

The study of the parameterized algorithmics of graph drawing problems was initiated by Dujmović et al.[20], and several new generic results were later obtained by Dujmović and others[21]. Dujmović and Whitesides[22] investigated the p-OSCM problem and obtained an algorithm for this problem which runs in time $\mathcal{O}^*(1.6182^k)$. This was later improved to $\mathcal{O}^*(1.4656^k)$ by Dujmović et al.[23] We also refer to the survey article[24] on parameterized approaches to graph drawing. There has been a similar race to obtain better approximation algorithms for the problem. To the best of our knowledge, the current best approximation factor for p-OSCM is 1.4664, due to Nagamochi[25].

Discussing related problems

Already in the paper by Sugiyama et al.[16], the concept of a *penalty graph* was introduced that links this graph-drawing problem with the problem of finding a lightest set of arcs whose removal turns a given arc-weighted digraph into an acyclic digraph. This type of relation will be exploited in the following. After the conference version[14] appeared, another approach to subexponential algorithms for p-OSCM was presented, based on interval graphs[26]. In some sense, also this idea is already present in earlier papers on this problem. All this might also give a reason why heuristics or also translations to other areas (mathematical optimization and SAT solving are prominent examples) work very well on this particular problem and also some of its generalizations and variants (see Refs. [27-33] and the references cited therein for some examples). This also proves the practical interest in this kind of graph drawing problems.

Many variants of these problems have been considered in the literature. We only mention some of them which we will also treat in this paper, complementing what we did in the conference version[14].

- Çakiroglu et al.[34] considered drawing graphs as in the OSCM setting, but with edge weights. If two edges cross, then the crossing receives as a weight the product of both according edge weights. The overall weight of a crossing is then the sum of all respective crossing weights, and the goal is to minimize this weight. They call this problem Weighted One-Layer-Free Problem (WOLF).
- Forster[35,36] considered the so-called *constrained variant* where the ordering of some of the vertices of the free layer is already fixed (as part of the input).
- In Refs. [37,38], another generalization of OSCM, called Positive weighted Completion of an Ordering (PCO), was studied. An Fixed-Parameter Tractability (FPT) result was obtained both by kernelization and by a search tree approach, leading to a running time of $\mathcal{O}^*(1.52^k)$.
- In radial drawings of graphs, also the restricted (NP-complete) variant called Radial One-Sided Two-Level Crossing Minimization (ROSCM) has been considered[39,40]. ROSCM (OSCM for radial drawings) tries to map the two vertex sets on two concentric cycles (as opposed to two parallel lines); otherwise, the game is the same.

Not only for p-OSCM, but also for many of its variants related to the Sugiyama approach, parameterized algorithms have been published. We only mention Refs. [20,41-45] to give the reader some impression.

1.3 Subexponential algorithms and polynomial-time approximation schemes

By now, it is a classical result that many NP-hard problems admit parameterized subexponential algorithms with running times of the type $\mathcal{O}(c^{\sqrt{k}})$ when restricted to planar graphs, or, more generally, to graphs of bounded genus (for instance)[46-50]. However, such type of algorithms are rarely observed in other contexts except for very few examples[51,52].

1.4 Our results

We obtain $\mathcal{O}^*(2^{\mathcal{O}(\sqrt{k}\log k)})$-time algorithms for both p-KAGG and p-OSCM. These significantly improve the previous best algorithms with running times $\mathcal{O}^*(1.403^k)$ and $\mathcal{O}^*(1.4656^k)$, respectively. Both of our algorithms are based on modeling these problems as the p-Weighted Directed Feedback Arc Set (p-WDFAS) problem. In p-WDFAS we are given a directed (multi)graph $D = (V, A)$, a weight function $w : A \to \mathbb{R}^+$ and a positive integer k, and the objective is to find a set of arcs $F \subseteq A$ of total weight at most k such that deleting F from D turns D into a directed acyclic graph; such an F is called a *feedback arc set* of D. Both p-KAGG and p-OSCM have been modeled as p-WDFAS in earlier work as well[4,16,53]; the novelty in our modeling is that it allows us to work with p-WDFAS on "tournament-like" structures.

A *tournament* is a digraph in which between every two vertices there is exactly one arc. A *semi-complete digraph* is a directed (multi)graph on n vertices that contains a tournament on n vertices as a subgraph. We study a problem that we call parameterized Feedback Arc Set on Semi-Complete digraphs (p-FASSC). Our modeling allows us to use the chromatic-coding technique recently developed by Alon et al.[54], which they used to obtain the first subexponential time algorithm for p-WDFAS on tournaments.

Independently, Karpinski and Schudy[15] have obtained a faster algorithm for a special case of p-WDFAS restricted to complete digraphs where, for every two vertices u, v in the digraph, $w(uv) + w(vu) = 1$ (the *probability constraint*). This algorithm runs in $\mathcal{O}^*(2^{\mathcal{O}(\sqrt{k})})$ time. Using essentially the same modeling as we use for p-KAGG, they show that p-KAGG can be solved in $\mathcal{O}^*(2^{\mathcal{O}(\sqrt{k})})$ time. Let us also mention algorithms of Feige[55] and Fomin and Pilipczuk[56] for the unweighted case of the problem.

We also study natural "above-guarantee" versions of these problems and show them to be fixed parameter tractable. We show that the above-guarantee versions of p-KAGG (A-p-KAGG) and p-OSCM(A-p-OSCM) are both equivalent to p-WDFAS and hence both have algorithms that run in time $\mathcal{O}^*(2^{\mathcal{O}(k\log k)})$[57]. A finer analysis of A-p-KAGG reveals an interesting contrast in its running time: If the number of votes in the input to p-KAGG is *odd*, then A-p-KAGG can still be solved in time $O^*(2^{\mathcal{O}(\sqrt{k}\log k)})$, while if it is *even*, then the problem cannot have any subexponential-time algorithm unless the Exponential Time Hypothesis (ETH) is false[58], or equivalently[59], unless FPT = M[1].

It is also worth mentioning that our reduction from p-OSCM to p-WDFAS on tournaments implies a PTAS for the graph drawing problem, as well as for the related ones. To summarize, we analyze a common feature of p-KAGG and p-OSCM to provide new insights and findings of interest to both the graph drawing community and the social choice community.

2 Preliminaries

A *parameterized problem* Π is a subset of $\Gamma^* \times \mathbb{N}$, where Γ is a finite alphabet. An instance of a parameterized problem is a tuple (x, k), where k is called the parameter. A central notion in parameterized complexity is FPT which means, for a given instance (x, k), decidability in time $\mathcal{O}(f(k) \cdot p(|x|))$, where f is an arbitrary function of k and p is a polynomial in the input size.

Let Π_1 and Π_2 be two parameterized problems. A *parameterized reduction* from Π_1 to Π_2 is an algorithm that takes an instance (x, k) of Π_1 as input, runs in time $\mathcal{O}(f(k) \cdot p(|x|))$, and outputs an instance (y, ℓ) of Π_2 such that ℓ is some function of k alone and (x, k) is a YES instance of Π_1 if and only if (y, ℓ) is a YES instance of Π_2.

We (partially) repeat and summarize some notions from the theory of directed graphs that we need in the following. A *tournament* is a directed graph in which there is exactly one directed arc between every two vertices. A *feedback arc set* in a tournament can be also characterized a set of arcs whose *reversal* results in a Directed Acyclic Graph (DAG). A *semi-complete digraph* is a directed (multi)graph on n vertices, for some $n \in \mathbb{N}$, which contains a tournament on n vertices as a subgraph.

3 FPT Algorithms for *p*-KAGG

Let S be a finite set, and let π_1 and π_2 be two permutations of S. For $u, v \in S$, we define

$$d_{\pi_2}^{\pi_1}(u, v) = \begin{cases} 0, & \text{if } \pi_1 \text{ and } \pi_2 \text{ rank} \\ & u \text{ and } v \text{ in the same order;} \\ 1, & \text{otherwise.} \end{cases}$$

The KT-distance of π_1 and π_2 is defined as: KT-dist$(\pi_1, \pi_2) = \sum_{\{u,v\}\subseteq S} d_{\pi_2}^{\pi_1}(u, v)$.

Let C be a set of candidates and V a set of votes over C. For any permutation r of C, the *Kemeny score of r with respect to V* is defined as:

$$\text{KS}(r, V) = \sum_{\pi\in V} \text{KT-dist}(r, \pi).$$

Observe that

$$\text{KS}(r, V) = \sum_{\pi\in V} \text{KT-dist}(r, \pi) = \sum_{\pi\in V} \sum_{\{u,v\}\subseteq C} d_{\pi}^{r}(u, v) = \sum_{\{u,v\}\subseteq C} \sum_{\pi\in V} d_{\pi}^{r}(u, v) \quad (1)$$

These preparations allow us now to formally state the problem Kemeny score problem p-KAGG that we are going to consider:

Given: A set C of candidates, a set V of votes over C, and an integer $k \geqslant 0$ Parameter: k Question: Is there a permutation r of C such that $\text{KS}(r, C) \leqslant k$?

A slight generalization is to weight the votes $\pi \in V$ with positive integers $\omega(\pi)$ (see Ref. [3]). The question is then to see if the *weighted Kemeny score*

$$w - \text{KS}(r, V) = \sum_{\pi\in V} \omega(\pi) \cdot \sum_{\{u,v\}\subseteq C} d_{\pi}^{r}(u, v)$$

is at most k. Both the weighted and the unweighted variant can be easily seen as minimization problems; the task is now to find an ordering of the candidates whose Kemeny score is minimum.

3.1 Parameterized reduction from *p*-KAGG to *p*-WDFAS

We now describe a parameterized reduction from *p*-KAGG to *p*-WDFAS, briefly mentioned by Betzler et al.[3], which runs in *polynomial* time and takes the parameter from k to k. Let (C, V, k) be an instance of *p*-KAGG. In what follows, we assume without loss of generality that $|V| \geqslant 1$. We construct a digraph G such that (C, V, k) is a YES instance of *p*-KAGG if and only if G has a feedback arc set of weight at most k. We set the vertex set of G to be the set C of candidates. For each vote $\pi_i \in V$ and for each pair of vertices (u, v) of G, we add a new arc with weight 1 from u to v in G if and only if u appears before v in π_i (equivalently, when u is preferred over v by π_i). In the weighted case, this arcs gets the weight $\omega(\pi_i)$. This completes the construction; the parameter is k.

Fix a vote $\pi_i \in V$. For each pair of candidates $u, v \in C$, π_i prefers exactly one of these candidates over the other. Thus, for any two vertices u, v of G, each vote contributes exactly one arc between u and v in G. As a consequence, we have:

Observation 1 Let G be the digraph constructed from an instance (C, V, k) of *p*-KAGG as described above. For any two vertices u, v of G, let i be the number of arcs in G from u to v, and j the number of arcs from v to u. Then $i + j = |V|$.

The next two claims show that the reduction is sound.

Claim 1 Let (C, V, k) be a YES instance of the problem *p*-KAGG; let G be the digraph constructed from (C, V, k) as described above. Then G has a feedback arc set of weight at most k.

Proof Since (C, V, k) is a YES instance of *p*-KAGG, there exists a permutation r of the set C such that

$$\sum_{\pi \in V} \text{KT-dist}(r, \pi) \leqslant k.$$

For $u, v \in V(G)$, let $\overline{r_{uv}}$ be the set of arcs in G between u and v that are oriented *contrary* to the direction implied by r. That is, if u appears before v in r, then $\overline{r_{uv}}$ consists of all arcs from v to u in G; if u appears after v in r, then $\overline{r_{uv}}$ consists of all arcs from u to v in G. Using Eq. (1), we get $\sum_{\{u,v\} \subseteq C} \sum_{\pi \in V} d^r_\pi(u, v) \leqslant k$. By construction, this implies $\sum_{\{u,v\} \subseteq V(G)} |\overline{r_{uv}}| \leqslant k$.

That is, there are at most k arcs in G, each of weight exactly 1, that are oriented contrary to the directions implied by r. Reversing these arcs, we get a digraph G' in which every arc is oriented according to the direction implied by r. Since r is a permutation of $V(G) = V(G')$, it follows that G' is acyclic. ■

Claim 2 Let G be the digraph constructed from an instance (C, V, k) of *p*-KAGG as described above. If G has a feedback arc set S of weight at most k, then (C, V, k) is a YES instance of *p*-KAGG.

Proof Note that, as each arc in G has weight exactly 1, S contains exactly k arcs. Consider the DAG G' obtained from G by reversing the arcs in S. Note that this operation preserves the number of arcs between any pair of vertices. From Observation 1, and since G' is a DAG, between each pair u, v of vertices of G' there are exactly $|V|$ arcs, all of which are in the same direction. The arcs of G' thus define a permutation r of C, where for any $u, v \in C$, u appears before v in r if and only if there is an arc (in fact, $|V|$ arcs) from u to v in G'. For $u, v \in V(G)$, let $\overline{r_{uv}}$ be the set of arcs between u and v in G that are oriented *contrary* to the direction implied by r. Then $\bigcup_{\{u,v\} \subseteq V(G)} \overline{r_{uv}} = S$, $\sum_{\{u,v\} \subseteq V(G)} |\overline{r_{uv}}| = |S| \leqslant k$, and from this and the construction we get $\sum_{\{u,v\} \subseteq C} \sum_{\pi \in V} d^r_\pi(u, v) \leqslant k$. From Eq. (1) it follows that $\text{KS}(r, V) \leqslant k$, and so (C, V, k) is a YES instance of *p*-KAGG. ■

The reduction above can clearly be done in polynomial time, and the number of vertices in the reduced instance (G, k) is equal to the number of candidates $|C|$ in the input instance (C, V, k). Further, the reduced instance has at least one arc (in fact, exactly $|V|$ arcs) between every pair of vertices. Let H be the edge-weighted digraph obtained from G by replacing parallel arcs with single weighted arcs in the natural way. That is, if there are $i > 0$ arcs from u to v in G, then H contains a single arc of weight i from u to v. It is easy to verify that H has a feedback arc set of weight at most k if and only if G has a feedback arc set of weight at most k. Hence from Claims 1 and 2, we obtain:

Lemma 1 Given an instance (C, V, k) of *p*-KAGG, we can construct an equivalent instance (G, k) of *p*-WDFAS in polynomial time, where G is a semi-complete digraph such that $|V(G)| = |C|$.

Notice that the previous claims and also the lemma is also valid for the weighted version of the problem.

3.2 A subexponential FPT algorithm for *p*-KAGG

Our algorithm is based on the observation that the algorithm of Alon et al.[54] for *p*-WDFAS on tournaments also works for semi-complete digraphs. The algorithm presented in Ref. [54] starts by preprocessing the instance and then obtains an equivalent instance with at most $\mathcal{O}(k^2)$ vertices in polynomial time. That is, given a tournament T and a positive integer k, using only polynomial time, the preprocessing algorithm either concludes that T does not have a feedback arc set of weight at most k or finds a new tournament T' with $\mathcal{O}(k^2)$ vertices and $k' \leqslant k$ such that the original tournament T has a feedback arc set of weight at most k, if and only if T' has a feedback arc set of weight at most k'. This preprocessing allows them to assume that the instance where they actually apply the subexponential time algorithm is of size $\mathcal{O}(k^2)$ only. Their preprocessing can also be applied to semi-complete digraphs by allowing both directed cycles of length two and triangles in the reduction rules proposed in Ref. [54].

So we always first apply these preprocessing rules and obtain a semi-complete digraph on $\mathcal{O}(k^2)$ vertices. Let the preprocessed semi-complete digraph be $T = (V, A)$.

To obtain our algorithm we also use *universal coloring families* introduced by Alon et al. in Ref. [54]. For integers m, k, and r, a family $\mathcal{F}$ of functions from $[m]$ to $[r]$ is called a universal (m, k, r)-coloring family if for any graph G on the set of vertices $[m]$ with at most k edges, there exists an $f \in \mathcal{F}$ which is a proper vertex coloring of G. The following result gives a bound on the size of universal coloring families.

Proposition 1[54] For any $n > 10k^2$ there exists an explicit universal $(n, k, \mathcal{O}(\sqrt{k}))$-coloring family $\mathcal{F}$ of size $|\mathcal{F}| \leqslant 2^{\mathcal{O}(\sqrt{k}\log k)} \log n$.

We enumerate each function in the universal coloring family and then color the vertices of T with the help of these functions. Observe that since the number of arcs possible in the solution is at most k, there exists a function $f \in \mathcal{F}$ such that no end-points of the arc in the solution are colored with same color, that is, no arc of the solution is mono-chromatic. By making use of the dynamic programming algorithm proposed in Ref. [54], we can find a feedback arc set of weight at most k of T, if there exists one, in time $\mathcal{O}(2^{\mathcal{O}(\sqrt{k}\log k)})$. This yields the following theorem.

Theorem 1 Every p-KAGG instance with n candidates can be solved in $2^{\mathcal{O}(\sqrt{k}\log k)} + n^{\mathcal{O}(1)}$ time.

By slightly adapting the dynamic programming, the weighted version of the problem can be captured as well.

3.3 FPT algorithms for *A*-*p*-KAGG

Discussing parameterizations above guaranteed value has become an important topic in the development of parameterized algorithms since Mahajan and Raman had introduced this concept in the context of MAX SAT[60]. We will discuss this idea now for p-KAGG in the unweighted case.

Consider an instance of the p-KAGG problem. Let π be *any* permutation of the candidate set C, let V be the set of all votes, and let $\mathrm{KS}(\pi, V)$ denote the sum of the KT-distances of π from all the votes in the set V. Suppose A and B are two candidates in the input, and let i votes prefer A over B and j votes prefer B over A. Clearly, the pair $\{A, B\}$ contributes at least $\min(i, j)$ to $\mathrm{KS}(\pi, V)$. For $\{u, v\} \subseteq C$, let $I(u, v)$ (respectively $J(u, v)$) be the number of votes that rank u before v (respectively v before u), and let

$$g = \sum_{\{u,v\}\subseteq C} \min\{I(u, v), J(u, v)\}.$$

Then $\mathrm{KS}(\pi, V) \geqslant g$, and so in the natural above-guarantee version of p-KAGG, which we are going to call A-p-KAGG, we ask for a permutation π of C such that $\mathrm{KS}(\pi, V) \leqslant g + k$.

We now describe a reduction from A-p-KAGG to p-WDFAS, originally due to Dwork et al.[61]

- When the number of votes in the input instance is odd (A-p-KAGG(odd)), the reduced instance is a tournament with positive integral edge weights.
- When the number of votes in the input instance is even (A-p-KAGG($even$)), the reduced instance is not necessarily a tournament.

In both cases, the parameter goes from k to k. That is, the reduction takes A-p-KAGG(odd) to p-WDFAS on tournaments, and A-p-KAGG($even$) to p-WDFAS in general digraphs, in both cases preserving the parameter. Together with the subexponential FPT algorithm of Alon et al.[54] for p-WDFAS on tournaments, this implies a subexponential FPT algorithm for A-p-KAGG(odd).

In the next subsection, we describe a parameterized reduction from p-WDFAS to A-p-KAGG($even$) in which the parameter goes from k to $2k$. This implies that A-p-KAGG($even$) does not have a subexponential FPT algorithm unless the exponential time hypothesis is false.

Let (C, V, k) be an instance of A-p-KAGG. We are going to construct an instance (H, k) of p-WDFAS in two stages, as follows.

Stage 1 We construct a digraph G exactly as in the previous reduction. We set the vertex set of G to be the set C of candidates. For each vote $\pi_i \in V$ and for each pair of vertices (u, v) of G, we add a new arc of weight 1 from u to v in G if and only if u appears before v in π_i (equivalently, when u is preferred over v by π_i).

Stage 2 We now prune the "above-guarantee" arcs of G. We process every two-vertex subset $\{u, v\}$ of G as follows: Let there be a total of i arcs from u to v and j arcs from v to u in H. Assume without loss of generality that $i \geqslant j$. We replace all the arcs between u and v by a single arc of weight $i - j$ from u to v. If $i - j = 0$, then we just remove all the arcs between u and v, and do not add any arc to replace them. We repeat this for every 2-subset of vertices of G to obtain a digraph H with integer-weighted arcs. (H, k) is the desired instance of p-WDFAS.

Suppose the number $|V|$ of votes in the input instance (C, V, k) is odd. Then, with the same notation as above, $i + j = |V|$ is odd for each 2-subset $\{u, v\}$ of G (Observation 1), and so $i > j$. Thus there is exactly one arc between every two vertices of H, and so H is a tournament. If $|V|$ is even, then it is possible that $i = j$ for some $\{u, v\} \subseteq V(G)$, and so in H there will not be any arc between u and v. Hence when $|V|$ is even, H is not necessarily a tournament or a semi-complete digraph.

Dwork et al.[61] showed that the above reduction is sound†; see also Mahajan et al.[62]:

Lemma 2[61,62] Let (H, k) be the instance of p-WDFAS obtained from an instance (C, V, k) of A-p-KAGG as described above. Then (H, k) is a YES instance of p-WDFAS if and only if (C, V, k) is a YES instance of A-p-KAGG.

The fastest known FPT algorithm for p-WDFAS runs in time $\mathcal{O}^*(2^{\mathcal{O}(k \log k)})$[57]; and the fastest known FPT algorithm for p-WDFAS on tournaments runs in time $2^{\mathcal{O}(\sqrt{k} \log k)} + n^{\mathcal{O}(1)}$[54]. Hence from Lemma 2 we get:

Theorem 2 Any A-p-KAGG problem instance with n candidates can be solved

- in time $2^{\mathcal{O}(\sqrt{k} \log k)} + n^{\mathcal{O}(1)}$ when the number of votes is odd,
- and in time $\mathcal{O}^*(2^{\mathcal{O}(k \log k)})$ when the number of votes is even.

† The quoted paper is nearly identical with Ref. [11], but the reduction we are interested in is contained in the appendix that is not part of the conference paper.

3.4 A lower bound for *A*-*p*-KAGG(*even*)

We now argue that the A-p-KAGG($even$) problem does not have a subexponential FPT algorithm, unless the ETH is false. ETH is a well-known complexity hypothesis formulated by Impagliazzo et al.[58]

Exponential Time Hypothesis: There is a positive real s such that 3-CNF-SAT with n variables and m clauses cannot be solved in time $2^{sn}(n + m)^{\mathcal{O}(1)}$.

To see this, consider the following sequence of two reductions:

verterx cover →directed feedback arc set →
A-p-KAGG.

The first reduction is due to Karp[63], and the second is due to Dwork et al.[61] This sequence of reductions takes an input instance (G, k) of vertex cover where G is a graph on n vertices and m edges and $k \leqslant n$ is a positive integer, and outputs an instance $(C, V, 2k)$ of A-p-KAGG($even$) where $|C| = 3n + 2m$, $|V| = 4$, and the guarantee is

$$g = 2\left(\binom{2n}{2} + \binom{n+2m}{2} + n + 2m\right);$$

see Refs. [61,63] for details. Suppose A-p-KAGG($even$) has an algorithm that runs in time $\mathcal{O}^*(2^{o(k)})$. Since $k = \mathcal{O}(n)$ throughout the reduction, we can then use this algorithm to solve vertex cover in $\mathcal{O}^*(2^{o(n)})$ time: We first apply the above sequence of reductions and then apply the supposed subexponential FPT algorithm for A-p-KAGG($even$) to the resulting instance. This would in turn imply that ETH is false[58], and so we have Theorem 3.

Theorem 3 The A-p-KAGG problem with an even number of votes cannot be solved in $\mathcal{O}^*(2^{o(k)})$ time, unless ETH is false.

4 FPT Algorithms for *p*-OSCM

Let us now turn our attention to graph drawing. We first formally state the problem p-OSCM that we are going to consider:

> Given: A bipartite graph $G = (V_1, V_2, E)$, a bijection $\pi\colon \{1, \cdots, |V_1|\} \to V_1$ and an integer $k \geqslant 0$
> Parameter: k
> Question: Is there a bijection $\pi'\colon\{1, \cdots, |V_2|\} \to V_2$ such that, if G is drawn according to (π, π'), no more than k crossings are incurred?

We say that a bipartite graph $G = (V_1, V_2, E)$ is drawn according to (π, π') (in the plane) if the vertex $\pi(\ell), \ell \in \{1, \cdots, |V_1|\}$, of V_1 gets the position $(\pi(\ell), 0)$ and the vertex $\pi'(\ell)$, $\ell \in \{1, \cdots, |V_2|\}$, of V_2 gets the position $(\pi'(\ell), 1)$ and edges are drawn as straight lines. Whenever there are two edges $e_1 = u_1 v_1$ and $e_2 = u_2 v_2$ such that $\pi^{-1}(u_1) < \pi^{-1}(u_2)$ but $\pi'^{-1}(v_1) > \pi'^{-1}(v_2)$, we say that e_1 and e_2 *incur a crossing*. The lines $\{(x, 0) \mid x \in \mathbb{R}\}$ and $\{(x, 1) \mid x \in \mathbb{R}\}$ are also referred to as the first and second layers of the drawing.

Let (G, π, k) with $G = (V_1, V_2, E)$ be an instance of p-OSCM. In what follows, we assume without loss of generality that in G, every vertex in V_2 has at least one neighbor in V_1.

4.1 Parameterized reduction from *p*-OSCM to *p*-WDFAS

We now describe a parameterized reduction from p-OSCM to p-WDFAS which runs in *polynomial* time and takes the parameter from k to k. Let (G, π, k) with $G = (V_1, V_2, E)$ be an instance of p-OSCM. For two vertices $u, v \in V_2$, let C_{uv} denote the number of crossings of edges incident to u with edges incident to v, when u appears before v in the second layer. It is known[22] that for a given graph G and a fixed ordering π of the vertices of V_1, C_{uv} is a constant and can be computed in polynomial time. Moreover, if we are given a bijection $\pi' : \{1, \cdots, |V_2|\} \to V_2$, the number of crossings incurred by (π, π') can be computed as:

$$\sum_{u,v \in V_2, \pi'^{-1}(u) < \pi'^{-1}(v)} C_{uv} \qquad (2)$$

We construct a digraph H as follows: H has one vertex for each vertex of V_2. For $\{u, v\} \subseteq V_2$, we draw the arc uv with weight C_{uv} if $C_{uv} > 0$.

Claim 3 Let (G, π, k) with $G = (V_1, V_2, E)$ be an instance of p-OSCM, and let H be the digraph obtained from this instance as described above. (G, π, k) is a YES instance of p-OSCM if and only if H has a feedback arc set of weight at most k.

Proof Suppose (G, π, k) with $G = (V_1, V_2, E)$ is a YES instance of p-OSCM, and let π_m be a permutation of V_2 that witnesses this fact. Place the vertices of H on a line in the order induced by π_m : u is to the left of v if and only if u comes before v in π_m. From the construction it is clear that the sum of the weights of the arcs in H that go from left to right is at most k, and so these arcs together form a feedback arc set of H of weight at most k.

Now suppose S is a *minimal* feedback arc set of H of weight at most k. Let π' be the unique permutation of V_2 such that if we place the vertices of H on a line in the order specified by π', then the arcs that go from left to right are exactly the arcs in S. It is easily verified that if the vertices of V_2 are placed on the second layer in the order specified by π', then the number of crossings will be at most k. ■

The reduction above can clearly be done in polynomial time, and the graph H in the reduced instance (H, k) has $|V_2|$ vertices, where the p-OSCM instance is (G, π, k), with $G = (V_1, V_2, E)$. Further, it is not difficult to see that the reduced instance has at least one arc between every pair of vertices. Hence, from Claim 3 we can deduce the following result:

Lemma 3 Given an instance (G, π, k) of p-OSCM, where $G = (V_1, V_2, E)$, we can construct, in polynomial time, an equivalent instance (H, k) of p-WDFAS where H is a semi-complete digraph and $|V(H)| = |V_2|$.

4.2 A subexponential FPT algorithm for *p*-OSCM

From Lemma 3, and using the same argument as in Section 3.2, we get:

Theorem 4 Any p-OSCM instance can be solved in $2^{\mathcal{O}(\sqrt{k} \log k)} + n^{\mathcal{O}(1)}$ time, where n is the number of vertices in the layer that is not fixed.

4.3 Lower and upper bounds for *A*-*p*-OSCM

Let (G, π, k) with $G = (V_1, V_2, E)$ be an instance of p-OSCM. For two vertices $u, v \in V_2$, let C_{uv} be defined as in Section 4.1. It is known that the minimum possible number of crossings is $g = \sum_{\{u,v\} \subseteq V_2} \min(C_{uv}, C_{vu})$[22]. So in the natural above-guarantee version of p-OSCM, which we call A-p-OSCM, we ask for a permutation π of V_2 such that the number of crossings induced by π is at most $g + k$.

Given an instance (G, π, k) of p-OSCM, with $G = (V_1, V_2, E)$, the well-known *penalty graph* construction of Sugiyama et al.[16] constructs an arc-weighted digraph H with V_2 as the vertex set, and there is an arc in H from u to v with weight $C_{vu} - C_{uv}$ if $C_{uv} < C_{vu}$. It is easy to verify that there is a permutation π_m of V_2 such that the number of crossings induced by π_m is at most $g + k$ if and only if H has a feedback arc set of weight at most k. Thus, using the algorithm in Ref. [57], we have:

Theorem 5 The A-p-OSCM problem can be solved in $\mathcal{O}^*(2^{\mathcal{O}(k \log k)})$ time.

Muñoz et al.[19] described a reduction from directed feedback arc set to p-OSCM that, in fact, is a parameterized reduction from directed feedback arc set (where the parameter k is the solution size) to A-p-OSCM which takes the parameter from k to $2k$. Hence by a similar argument as in Section 3.4, we can conclude:

Theorem 6 The A-p-OSCM problem cannot be solved in $\mathcal{O}^*(2^{o(k)})$ time, unless ETH is false.

4.4 Problem variants

Equation (2) can be easily modified to host edge weights as required by the WOLF variant as introduced by Çakiroglu et al[34]. This immediately implies:

Theorem 7 The edge-weighted p-OSCM problem can be solved in $2^{\mathcal{O}(\sqrt{k}\log k)} + n^{\mathcal{O}(1)}$ time, where n is the number of vertices in the layer that is not fixed.

We now define a generalization of p-OSCM that has been introduced and studied in Ref. [38]; PCO is the following problem:

> Given: A partial order $R \subseteq V \times V$ on a finite set V, a cost function $c : V \times V \setminus R \to (0, \infty)$ and an integer $k \geqslant 0$
> Parameter: k
> Question: Is there a linear ordering $L \supseteq R$ such that $c(R \setminus L) \leqslant k$?

There is an easy translation of PCO to Feedback Arc Set in Tournaments (FAST): Interpret V as the vertices of a digraph. We have one or two arcs between two vertices.

- If $(u, v) \in R$ and $u \neq v$, then the arc (u, v) is assigned ∞ as its weight.
- If neither $(u, v) \in R$ nor $(v, u) \in R$, then the arc (u, v) is assigned the weight $c((u, v))$.

Hence, we can immediately conclude a subexponential algorithm for PCO:

Theorem 8 PCO can be solved in $2^{\mathcal{O}(\sqrt{k}\log k)} + |V|^{\mathcal{O}(1)}$ time.

Notice that the additive term results from the possible kernelization for PCO as explained in Ref. [38]. We now explain how to model p-OSCM by PCO:

- $V = V_2$, the set of vertices whose ordering is not yet fixed.
- We assume that some arbitrary initial linear ordering $\leqslant_{\text{init}}$ of V_2 is given.
- Given two vertices $u, v \in V_2$, set $(u, v) \in R$ if $C_{uv} = 0$ and $C_{vu} > 0$.
- Given two vertices $u, v \in V_2$, set $(u, v) \in R$ if $C_{uv} = 0$ and $C_{vu} = 0$ and $u \leqslant_{\text{init}} v$.

Using this translation, Theorem 8 implies Theorem 4. Moreover, Foster's constrained OSCM variant can be treated in the same way. The parts of V_2 whose ordering is already prescribed can be incorporated within R (in the translation to the PCO instance). We can hence conclude:

Theorem 9 The constrained p-OSCM problem can be solved in $2^{\mathcal{O}(\sqrt{k}\log k)} + n^{\mathcal{O}(1)}$ time, where n is the number of vertices in the layer that is not fixed.

It is not so clear how (or even whether) we could obtain subexponential algorithms for ROSCM.

5 Polynomial-Time Approximation Schemes

The following result largely improves on its precursors, the best one being[25]. It is an immediate consequence of our reductions and the approximation schemes known for directed feedback arc set.

Theorem 10 There is a PTAS for approximating OSCM, given a bipartite graph $G = (V_1, V_2, E)$ and a strict linear ordering $<$ on V_1.

Observing Ref. [64], we can conclude the next result by making use of Theorem 10:

Theorem 11 There is a PTAS for approximating ROSCM, given a bipartite graph $G = (V_1, V_2, E)$ of minimum degree two and a strict linear ordering $<$ on V_1.

It is however unclear how to overcome the seemingly technical degree condition in the previous theorem.

Notice that, due to the tight connection between the problems directed feedback arc set and PCO, the following results are again immediate:

Corollary 1 There is a PTAS for approximating PCO.

Corollary 2 There is a PTAS for approximating constrained OSCM.

6 Conclusion and Future Work

The general conclusion that we can draw from this problem can be the advice to look over the rim of one's teacup. There are thousands of algorithmic problems in different areas of applications, but these problems appear often to be unrelated, as the people working in these different domains use their own vocabulary and express their thoughts in different ways. Hence, possible interconnection between different areas remains mostly hidden and unobserved. It would be most inspiring and fruitful for computer science if we could overcome such barriers in language and style. A steady flow of ideas could result in quite an amount of progress in different domains.

In this paper we modeled two basic problems, from two different domains, as the weighted feedback arc set problem on "tournament-like" structures. This allowed us to utilize the earlier developed technique of chromatic-coding[54] to obtain subexponential-time algorithms, that is, algorithms that run in time $\mathcal{O}^*(c^{\sqrt{k}\log k})$, most notably for p-KAGG and for p-OSCM. The running time of these algorithms is a significant improvement over the hitherto best published algorithms, which had running times of the form roughly $\mathcal{O}^*(1.5^k)$. The same idea allowed us to conclude PTAS results.

Our approach also allowed us to show that the above-guarantee versions of these problems are fixed parameter tractable with algorithms having running times of the form $\mathcal{O}^*(c^{k\log k})$. We also show that the above-guarantee versions of these problems cannot have algorithms that run in $\mathcal{O}^*(2^{\mathcal{O}(k)})$ time, unless ETH fails.

It also might be interesting to consider the crossing minimization variant of these problems that attempts to minimize the maximum number of crossings per edge as proposed by Biedl et al.[1] from the viewpoints both of fixed parameter tractability and of approximability.

Acknowledgements

The whole research was supported by a German-Norwegian PPP grant. Geevarghese Philip was also supported by the Indo-German Max Planck Center for Computer Science (IMPECS).

References

[1] T. C. Biedl, F.-J. Brandenburg, and X. Deng, Crossings and permutations, in *Proceedings of GD 2005* (P. Healy and N. S. Nikolov, eds.), vol. 3843 of *LNCS*, Springer, 2006, pp. 1-12.

[2] J. Bartholdi III, C. A. Tovey, and M. A. Trick, Voting schemes for which it can be difficult to tell who won the election, *Social Choice and Welfare*, vol. 6, pp. 157-165, 1989.

[3] N. Betzler, M. R. Fellows, J. Guo, R. Niedermeier, and F. A. Rosamond, Fixed-parameter algorithms for Kemeny rankings, *Theoretical Computer Science*, vol. 410, no. 45, pp. 4554-4570, 2009.

[4] N. Simjour, Improved parameterized algorithms for the Kemeny aggregation problem, in *Proceedings of IWPEC 2009* (J. Chen and F. V. Fomin, eds.), vol. 5917 of *LNCS*, Springer, 2009, pp. 312-323.

[5] N. Betzler, J. Guo, and R. Niedermeier, Parameterized computational complexity of Dodgson and Young elections, *Information and Computation*, vol. 208, no. 2, pp. 165-177, 2010.

[6] N. Betzler and J. Uhlmann, Parameterized complexity of candidate control in elections and related digraph problems, *Theoretical Computer Science*, vol. 410, pp. 5425-5442, 2009.

[7] R. Christian, M. R. Fellows, F. Rosamond, and A. Slinko, On complexity of lobbying in multiple referenda, *Review of Economic Design*, vol. 11, pp. 217-224, 2007.

[8] J. McCabe-Dansted, Approximability and computational feasibility of Dodgson's rule, Master degree dissertation, The University of Auckland, NZ, 2006.

[9] J. Kemeny, Mathematics without numbers, *Daedalus*, vol. 88, pp. 571-591, 1959.

[10] J. Kemeny and J. Snell, *Mathematical Models in the Social Sciences*. Waltham: Blaisdell Publishing Company, 1962.

[11] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar, Rank aggregation methods for the web, in *Proceedings of the 10th International Conference on World Wide Web, WWW 10*, ACM, 2001, pp. 613-622.

[12] A. Guénoche, B. Vandeputte-Riboud, and J. B. Denis, Selecting varieties using a series of trials and a combinatorial ordering method, *Agronomie*, vol. 14, pp. 363-375, 1994.

[13] C. Kenyon-Mathieu and W. Schudy, How to rank with few errors, in *Proceedings of STOC 2007*, ACM, 2007, pp. 95-103.

[14] H. Fernau, F. V. Fomin, D. Lokshtanov, M. Mnich, G. Philip, and S. Saurabh, Ranking and drawing in subexponential time, in *Combinatorial Algorithms—21st International Workshop, IWOCA 2010*, (C. S. Iliopoulos and W. F. Smyth, eds.), vol. 6460 of *LNCS*, Springer, 2011, pp. 337-348.

[15] M. Karpinski and W. Schudy, Faster algorithms for feedback arc set tournament, Kemeny rank aggregation and betweenness tournament, in *Algorithms and Computation—21st International Symposium, ISAAC (1)*, (O. Cheong, K.-Y. Chwa, and K. Park, eds.), vol. 6506 of *LNCS*, Springer, 2010, pp. 3-14

[16] K. Sugiyama, S. Tagawa, and M. Toda, Methods for visual understanding of hierarchical system structures, *IEEE Trans. Systems Man Cybernet.*, vol. 11, no. 2, pp. 109-125, 1981.

[17] O. Bastert and C. Matuszewski, Layered drawings of digraphs, in *Drawing Graphs: Methods and Models*, M. Kaufmann and D. Wagner, eds. Springer, 2001, pp. 87-120.

[18] P. Eades and N. C. Wormald, Edge crossings in drawings of bipartite graphs, *Algorithmica*, vol. 11, pp. 379-403, 1994.

[19] X. Muñoz, W. Unger, and I. Vrt'o, One sided crossing minimization is NP-hard for sparse graphs, in *Proceedings of GD 2001* (P. Mutzel, M. Jünger, and S. Leipert, eds.), vol. 2265 of *LNCS*, Springer, 2002, pp. 115-123.

[20] V. Dujmović, M. R. Fellows, M. Hallett, M. Kitching, G. Liotta, C. McCartin, N. Nishimura, P. Ragde, F. A. Rosamond, M. Suderman, S. Whitesides, and D. R. Wood, A fixed-parameter approach to 2-layer planarization, *Algorithmica*, vol. 45, pp. 159-182, 2006.

[21] V. Dujmović, M. R. Fellows, M. Kitching, G. Liotta, C. McCartin, N. Nishimura, P. Ragde, F. A. Rosamond, S. Whitesides, and D. R. Wood, On the parameterized

complexity of layered graph drawing, *Algorithmica*, vol. 52, no. 2, pp. 267-292, 2008.

[22] V. Dujmović and S. Whitesides, An efficient fixed parameter tractable algorithm for 1-sided crossing minimization, *Algorithmica*, vol. 40, pp. 15-32, 2004.

[23] V. Dujmović, H. Fernau, and M. Kaufmann, Fixed parameter algorithms for one-sided crossing minimization revisited, *Journal of Discrete Algorithms*, vol. 6, pp. 313-323, 2008.

[24] H. Fernau, Parameterized algorithms for drawing graphs, in *Encyclopedia of Algorithms*, M.-Y. Kao, ed. Springer, 2008, pp. 631-635.

[25] H. Nagamochi, An improved bound on the one-sided minimum crossing number in two-layered drawings, *Discrete and Computational Geometry*, vol. 33, pp. 569-591, 2005.

[26] Y. Kobayashi and H. Tamaki, A fast and simple subexponential fixed parameter algorithm for one-sided crossing minimization, in *Algorithms—ESA 2012—20th Annual European Symposium* (L. Epstein and P. Ferragina, eds.), vol. 7501 of *LNCS*, Springer, 2012, pp. 683-694.

[27] F. Baumann, C. Buchheim, and F. Liers, Exact bipartite crossing minimization under tree constraints, in *9th International Symposium on Experimental Algorithms, SEA* (P. Festa, ed.), vol. 6049 of *LNCS*, Springer, 2010, pp. 118-128.

[28] C. Buchheim and L. Zheng, Fixed linear crossing minimization by reduction to the maximum cut problem, in *Computing and Combinatorics, COCOON* (D. Z. Chen and D. T. Lee, eds.), vol. 4112 of *LNCS*, Springer, 2006, pp. 507-516.

[29] M. Chimani, C. Gutwenger, and P. Mutzel, Experiments on exact crossing minimization using column generation, in *Workshop on Experimental Algorithms, WEA* (C. Alvarez and M. Serna, eds.), vol. 4007 of *LNCS*, Springer, 2006, pp. 303-315.

[30] M. Chimani and R. Zeranski, Upward planarity testing via SAT, in *Graph Drawing20th International Symposium, GD 2012* (W. Didimo and M. Patrignani, eds.), vol. 7704 of *LNCS*, Springer, 2013, pp. 248-259.

[31] G. Gange, P. J. Stuckey, and K. Marriott, Optimal k-level planarization and crossing minimization, in *Graph Drawing—18th International Symposium, GD 2010* (U. Brandes and S. Cornelsen, eds.), vol. 6502 of *LNCS*, Springer, 2011, pp. 238-249.

[32] M. Jünger and P. Mutzel, 2-layer straightline crossing minimization: Performance of exact and heuristic algorithms, *Journal of Graph Algorithms and Applications*, vol. 1, pp. 1-25, 1997.

[33] L. Zheng and C. Buchheim, A new exact algorithm for the two-sided crossing minimization problem, in *Combinatorial Optimization and Applications, First International Conference, COCOA* (A. W. M. Dress, Y. Xu, and B. Zhu, eds.), vol. 4616 of *LNCS*, Springer, 2007, pp. 301-310.

[34] O. A. Çakiroglu, C. Erten, Ö. Karatas, and M. Sözdinler, Crossing minimization in weighted bipartite graphs, in *Proceedings of Workshop on Experimental Algorithms, WEA 2007* (C. Demetrescu, ed.), vol. 4525 of *LNCS*, Springer, 2007, pp. 122-135.

[35] M. Forster, A fast and simple heuristic for constrained two-level crossing reduction, in *Graph Drawing, GD* (J. Pach, ed.), vol. 3383 of *LNCS*, 2004, pp. 206-216.

[36] M. Forster, Crossings in clustered level graphs, PhD dissertation, Universität Passau, Germany, 2004.

[37] V. Dujmović, H. Fernau, and M. Kaufmann, Fixed parameter algorithms for one-sided crossing minimization revisited, in *Proceedings of GD 2003* (G. Liotta, ed.), vol. 2912 of *LNCS*, Springer, 2004, pp. 332-344.

[38] H. Fernau, Parameterized algorithmics: A graph-theoretic approach. Habilitationsschrift, Universität Tübingen, Germany, 2005.

[39] C. Bachmaier, A radial adaptation of the Sugiyama framework for visualizing hierarchical information, *IEEE Transactions on Visualization and Computer Graphics*, vol. 13, no. 3, pp. 583-594, 2007.

[40] C. Bachmaier, H. Buchner, M. Forster, and S.-H. Hong, Crossing minimization in extended level drawings of graphs, *Discrete Applied Mathematics*, vol. 158, pp. 159-179, 2010.

[41] S. Böcker, F. Hüffner, A. Truß, and M. Wahlström, A faster fixed-parameter approach to drawing binary tanglegrams, in *Parameterized and Exact Computation, 4th International Workshop, IWPEC* (J. Chen and F. V. Fomin, eds.), vol. 5917 of *LNCS*, Springer, 2009, pp. 38-49.

[42] W. Didimo, F. Giordano, and G. Liotta, Upward spirality and upward planarity testing, *SIAM J. Discrete Math.*, vol. 23, no. 4, pp. 1842-1899, 2009.

[43] H. Fernau, Two-layer planarization: Improving on parameterized algorithmics, *Journal of Graph Algorithms and Applications*, vol. 9, pp. 205-238, 2005.

[44] H. Fernau, M. Kaufmann, and M. Poths, Comparing trees via crossing minimization, *Journal of Computer and System Sciences*, vol. 76, pp. 593-608, 2010.

[45] J. Uhlmann and M. Weller, Two-layer planarization parameterized by feedback edge set, *Theoretical Computer Science*, vol. 494, pp. 99-111, 2013.

[46] J. Alber, H. Fernau, and R. Niedermeier, Parameterized complexity: Exponential speedup for planar graph problems, *Journal of Algorithms*, vol. 52, pp. 26-56, 2004.

[47] E. D. Demaine, F. V. Fomin, M. Hajiaghayi, and D. M. Thilikos, Subexponential parameterized algorithms on graphs of bounded genus and H-minor-free graphs, *Journal of the ACM*, vol. 52, no. 6, pp. 866-893, 2005.

[48] E. D. Demaine and M. Hajiaghayi, The bidimensionality theory and its algorithmic applications, *The Computer Journal*, vol. 51, no. 3, pp. 292-302, 2008.

[49] F. Dorn, F. V. Fomin, and D. M. Thilikos, Subexponential parameterized algorithms, *Computer Science Review*, vol. 2, no. 1, pp. 29-39, 2008.

[50] F. Dorn, F. V. Fomin, D. Lokshtanov, V. Raman, and S. Saurabh, Beyond bidimensionality: Parameterized subexponential algorithms on directed graphs, in *27th International Symposium on Theoretical Aspects*

of Computer Science, STACS (J.-Y. Marion and T. Schwentick, eds.), vol. 5 of *Leibniz International Proceedings in Informatics (LIPIcs)*, Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2010, pp. 251-262.

[51] F. V. Fomin, S. Kratsch, M. Pilipczuk, M. Pilipczuk, and Y. Villanger, Tight bounds for parameterized complexity of cluster editing, in *30th International Symposium on Theoretical Aspects of Computer Science, STACS* (N. Portier and T. Wilke, eds.), vol. 20 of *LIPIcs*, Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2013, pp. 32-43.

[52] F. V. Fomin and Y. Villanger, Subexponential parameterized algorithm for minimum fill-in, *SIAM Journal on Computing*, vol. 42, no. 6, pp. 2197-2216, 2013.

[53] N. Ailon, M. Charikar, and A. Newman, Aggregating inconsistent information: Ranking and clustering, *Journal of the ACM*, vol. 55, no. 5, pp. 23:1-23:27, 2008.

[54] N. Alon, D. Lokshtanov, and S. Saurabh, Fast FAST, in *Proceedings of ICALP 2009, Part I* (S. Albers, A. Marchetti-Spaccamela, Y. Matias, S. E. Nikoletseas, and W. Thomas, eds.), vol. 5555 of *LNCS*, Springer, 2009, pp. 49-58.

[55] U. Feige, Faster fast (feedback arc set in tournaments), Tech. Rep. 0911.5094, ArXiv/CoRR, 2009.

[56] F. V. Fomin and M. Pilipczuk, Subexponential parameterized algorithm for computing the cutwidth of a semi-complete digraph, in *Algorithms—ESA 2013—21st Annual European Symposium* (H. L. Bodlaender and G. F. Italiano, eds.), vol. 8125 of *LNCS*, Springer, 2013, pp. 505-516.

[57] J. Chen, Y. Liu, S. Lu, B. O'Sullivan, and I. Razgon, A fixed-parameter algorithm for the directed feedback vertex set problem, *Journal of the ACM*, vol. 55, no. 5, pp. 21:1-21:19, 2008.

[58] R. Impagliazzo, R. Paturi, and F. Zane, Which problems have strongly exponential complexity?, *Journal of Computer and System Sciences*, vol. 63, no. 4, pp. 512-530, 2001.

[59] J. Flum and M. Grohe, *Parameterized Complexity Theory*. Springer-Verlag, 2006.

[60] M. Mahajan and V. Raman, Parameterizing above guaranteed values: MaxSat and MaxCut, *Journal of Algorithms*, vol. 31, no. 2, pp. 335-354, 1999.

[61] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar, Rank aggregation revisited, http://www.cs.msu.edu/∼cse960/papers/games/rank.pdf, 2014.

[62] M. Mahajan, V. Raman, and S. Sikdar, Parameterizing above or below guaranteed values, *Journal of Computer and System Sciences*, vol. 75, no. 2, pp. 137-153, 2009.

[63] R. M. Karp, Reducibility among combinatorial problems, in *Complexity of Computer Communications*, 1972, pp. 85-103.

[64] S.-H. Hong and H. Nagamochi, New approximation to the one-sided radial crossing minimization, *Journal of Graph Algorithms and Applications*, vol. 13, no. 2, pp. 179-196, 2009.

Henning Fernau received his PhD degree from the University of Karlsruhe in 1993, and his habilitation at the University of Tübingen in 2005. Since 2005, he has been full professor for theoretical computer science at the University of Trier, Germany. H. Fernau has over 90 journal publications and over 100 conference publications. His research interests stretch from formal languages to fixed parameter algorithms, touching as different areas as fractal geometry, graph theory, and grammatical inference.

Fedor V. Fomin received his PhD degree in 1997 from the Faculty of Mathematics and Mechanics, St. Petersburg State University, Russia. Since 2002, he is a professor in algorithms, at the Department of Informatics, University of Bergen, Norway. F. Fomin has over 100 journal publications and over 120 conference publications. His research focuses on graph algorithms, moderately exponential-time algorithms, fixed parameter algorithms, kernelization, and parameterized complexity.

Daniel Lokshtanov received his PhD degree in 2009 from the University of Bergen, Norway. After a year in San Diego, US, he returned to Norway and is currently project leader for BeHard, a four-year research project on kernelization funded by Bergen Research Foundation and University of Bergen. D. Lokshtanov has over 30 journal publications and over 70 conference publications. His research interests focus on algorithmic graph theory in a broad sense.

Matthias Mnich obtained his PhD degree from Eindhoven University of Technology, in the Netherlands. After research assignments in Berkeley and Saarbrücken, and a visiting professorship at TU Darmstadt, he now works as a researcher in algorithms and complexity at Bonn University, Germany.

Geevarghese Philip obtained his PhD degree with Venkatesh Raman and Saket Saurabh at the Institute of Mathematical Sciences, Chennai, India. He is currently a post-doctoral fellow at the Max Planck Institute for Informatics in Saarbrücken, Germany. His research interests are in fixed parameter algorithms and parameterized complexity.

Saket Saurabh received his PhD degree in 2008 from the Institute of Mathematical Sciences (IMSc), Chennai, India. Since 2009, he is a faculty member at IMSc. From 2013 onwards, S. Saurabh also holds a professor position at University of Bergen, Norway. S. Saurabh has over 60 journal publications and over 100 conference publications. His research focuses on graph algorithms, moderately exponential-time algorithms, fixed parameter algorithms, kernelization, and parameterized complexity.