Open Problems in Parameterized and Exact Computation — IWPEC 2006

Hans L. Bodlaender

Leizhen Cai

Jianer Chen

Michael R. Fellows

Jan Arne Telle

Dániel Marx

Department of Information and Computing Sciences,
Utrecht University

Technical Report UU-CS-2006-052

www.cs.uu.nl

ISSN: 0924-3275

Open Problems in Parameterized and Exact Computation — IWPEC 2006

Hans L. Bodlaender* Leizhen Cai Jianer Chen
Michael R. Fellows Jan Arne Telle Dániel Marx

Abstract

In September 2006, the Second International Workshop on Parameterized and Exact Computation was held in Zürich, Switzerland, as part of ALGO 2006. At the end of IWPEC 2006, a problem session was held. (Most of) the problems mentioned at this problem session, and some other problems, contributed by the participants of IWPEC 2006 are listed here.

1 Introduction

In September 2006, the Second International Workshop on Parameterized and Exact Computation was held in Zürich, Switzerland, as part of ALGO 2006. At the end of IWPEC 2006, a problem session was held. Below, you find (most of) the open problems, mentioned by the participants of IWPEC in this problem session, and a few other open problems in the field of parameterized and exact algorithms. The problems have sometimes been edited, to provide some additional backgrounds.

2 The problems

2.1 Polynomial Kernels

Let $Poly(k)$ be the class of problems that have a polynomial size kernel. For the precise definition of $Poly(k)$, and more discussion on kernelizability, see e.g., [14]. For the next three problems, it is open whether they belong to $Poly(k)$.

2.1.1 Clique Cover (contributed by Mike Fellows)

Clique Cover is the problem, given a graph G and a parameter k, to find at most k cliques in G, such that each edge belongs to a clique. This problem belongs to FPT [30]. Does this problem have a polynomial kernel?

*Correspondence to Hans Bodlaender, Department of Information and Computing Sciences, Utrecht University, P.O. Box 80.089, email: hansb@cs.uu.nl

2.1.2 Multicut in Trees (contributed by Mike Fellows)

In the MULTICUT IN TREES problem, we are given an undirected tree $T = (V, E)$¡ and a collection H of h pairs of vertices in V, and a parameter k. We ask for a set of at most k edges, whose removal separates each pair of vertices in H. This problem generalizes the VERTEX COVER problem. For a discussion, see [32].

Guo [32] gives a set of kernelization rules that give a kernel with $O(k^{3k})$ vertices. Does this problem have a polynomial kernel?

2.1.3 Deleting edges to obtain an H-free graph (contributed by Leizhen Cai)

Let H be a fixed graph with h vertices. Determine whether the following problem has a polynomial-size kernel: can we delete at most k edges from a graph G to make G an H-free graph?

We note that the problem is FPT by Cai [15], and that a similar problem of deleting k vertices to make G an H-free graph has a kernel of size $O(k^{h-1})$ as the problem is easily transformed into the h-HITTING SET problem, which has a kernel of size $O(k^{h-1})$ by Nishimura, Ragde and Thilikos [39].

2.2 Polynomial kernels for degree-bounded graphs (contributed by Leizhen Cai)

Recently, Cai, Chan and Chan [17] have put a wide range of problems on degree-bounded graphs into FPT by using their random separation method. Roughly speaking, the problem of finding k vertices (edges) S to optimize a value $\Phi(S)$ (satisfy a property $P(S)$) is FPT for degree-bounded graphs if for any two disjoint sets V_1 and V_2 of vertices, $\phi(V_1 \cup V_2) = \phi(V_1) + \phi(V_2)$ (respectively, $P(V_1 \cup V_2) = P(V_1) \wedge P(V_2)$) when V_1 and V_2 are a certain distance apart.

Is there a general method to construct polynomial-size kernels for such problems on degree-bounded graphs? In particular, find polynomial-size kernels for the problems of finding an induced H-subgraph, where H is a fixed graph with k vertices, in a degree-bounded graph.

2.3 Linear Kernels

For some problems, polynomial size kernels are known. Are there kernels of smaller, e.g., linear size.

2.3.1 Feedback Vertex Set (contributed by Mike Fellows)

In the FEEDBACK VERTEX SET problem, we are given an undirected graph, and a parameter k, and ask if there is a set S of at most k vertices, such that each cycle in the graph contains a vertex in S. A kernel with $O(k^{11})$ vertices was given in [14], which was improved to a kernel with $O(k^3)$ vertices in [9]. Does this problem have a linear size (or quadratic size) kernel?

2.3.2 Convex Recoloring of Trees (contributed by Mike Fellows)

In the Convex Recoloring of Trees problem, we are given a tree $T = (V, E)$, a *coloring* of the vertices $V \to \mathcal{C}$, and a parameter k. We must decide if we can change the colors of at most k vertices, such for each color c, the set of vertices with color c is connected. See [45, 10, 38, 7]. A kernel with $O(k^2)$ vertices was recently obtained by Bodlaender et al. Is there a linear size kernel?

2.3.3 Edge Dominating Set (contributed by Mike Fellows)

An *edge dominating set* in an undirected graph $G = (V, E)$ is a set of edges $D \subseteq E$ such that each edge in E belongs to D or has an endpoint in common with an edge in D. In the Edge Dominating Set problem, we are given a graph G and parameter k, and ask if there is an edge dominating set of size at most k.

The best known bound on a kernel for Edge Dominating Set is $O(k^2)$, which follows directly from the quadratic kernel for Minimum Maximal Matching by Prieto [41]. See e.g., also [26]. Is there a linear size kernel?

2.4 Membership in FPT

The following problems are not known to have an FPT algorithm, but are also not known to be $W[1]$-hard.

2.4.1 Cluster editing with don't care edges (contributed by Mike Fellows)

In the Cluster Editing problem, we are given a graph $G = (V, E)$ and a parameter k. We ask if we can obtain a disjoint union of cliques, by at most k editing operations: each editing operation adds or deletes an edge. This problem is known to be in FPT; see e.g., [29].

Consider the variant with *don't care* edges; i.e., for some edges, the cost of editing these is zero. Does this problem belong to FPT? Possibly, the technique of iterative compression can be applied here.

2.5 Exactly k-Edge Subgraph (contributed by Leizhen Cai)

Consider the following problem:
Exactly k-Edge Subgraph
Instance: Graph $G = (V, E)$ and parameter k.
Question: Is there $V' \subseteq V$ such that $G[V']$ contains exactly k edges?

Does this problem belong to FPT?

The problem is FPT if $k = \binom{t}{2}$ for some integer t or the clique number of G is bounded by a constant, but NP-complete if k is part of input. Furthermore, its parametric dual Exactly $(m-k)$-Edge Subgraph is equivalent to the problem of finding k vertices V' to cover exactly k edges, which is FPT by the random separation method of Cai, Chan and Chan [17] but is unknown whether it admits a polynomial-size kernel.

2.6 Maximum k-Edge Multicomponent Cut (contributed by Leizhen Cai)

Does the following problem belong to FPT?
Maximum k-Edge Multicomponent Cut
Instance: Graph $G = (V, E)$, integer l, and parameter k.
Question: Are there k edges E' in G such that $G - E'$ has at least l components?

We note that Cai [16] has shown that the parametric dual Maximum $(m-k)$-Edge Multicomponent Cut is W[1]-hard, and so are their corresponding vertex versions.

2.7 Computing Treewidth

2.7.1 Treewidth-k recognition (contributed by Jan Arne Telle)

Arnborg, Corneil, and Proskurowski gave a polynomial ($O(n^{k+2})$) time algorithm for the problem to determine if a given graph has treewidth at most k, for fixed k [6]. Using graph minor theory, Robertson and Seymour [43] showed (non-constructively) that the problem can be solved in $O(n^2)$ time (and thus belongs to FPT.) This was improved to $O(n \log^2 n)$ time by Lagergren [35], and to $O(n \log n)$ time by Reed [42]; and turned into a constructive result by Lagergren and Arnborg [36] and Bodlaender and Kloks [11]. A (constructive) linear time algorithm was given in [8]. As written at the end of Section 6 of [8], this linear time algorithm uses time $O(c^{k^3} n)$ for some (large) c. The version of this algorithm in [40] has the same type of running time. In both cases, this is because these algorithms use an algorithm from [11] as a subroutine, and that algorithm has this function of k in its running time.

Is there an algorithm for the fixed parameter case of treewidth whose running time as a function of k is better? E.g., is there an algorithm for Treewidth whose running time is $O(c^k p(n))$ for some constant c and a polynomial p?

2.7.2 Treewidth of planar graphs (contributed by Hans Bodlaender)

The famous ratcatcher algorithm by Seymour and Thomas [44], see also the implementation work by Hicks [33, 34] and the constructive version by Gu and Tamaki [31], computes in polynomial time the *branchwidth* of a planar graph. This gives an 1.5-approximation for the treewidth of planar graphs. However, it is a long outstanding and probably difficult open problem what the complexity is of computing the *treewidth* of *planar graphs.*

2.7.3 Approximation of treewidth (contributed by Hans Bodlaender)

A problem that is already open for a long time is whether there is a polynomial time approximation algorithm for treewidth with a constant performance ratio. The current best known result is that one can find in polynomial time for graphs of treewidth k a tree decomposition of width $O(k\sqrt{\log k})$, using an algorithm by Feige et al. [25]. Also, there are several algorithms that are exponential in k (but polynomial in n), that, given a graph, either decide that the treewidth is more than k, or give a tree decomposition of width at most ck for some constant c, e.g., [35, 5, 42].

2.7.4 Enumeration of potential maximal cliques (contributed by Hans Bodlaender)

In 2004, Fomin, Todinca, and Kratsch [27] gave an exact algorithm for treewidth that uses $O(1.9601^n)$ time. This algorithm was based upon the algorithm by Bouchitté and Todinca [13, 12] for computing the treewidth in time, polynomial in the number of minimal separators. The algorithm was improved by Villanger to $O(1.8899^n)$ time [46]. These algorithms list the collection of *potential maximal cliques* and then use time, linear in the number of *potential maximal cliques*. Villanger also has shown that the number of potential maximal cliques in a graph G is $O(1.8135^n)$.

A set of vertices $S \subseteq V$ is a *potential maximal clique* in a graph $G = (V, E)$, if there is a minimal triangulation $H = (V, F)$ of G where S is a maximal clique. (H is a minimal triangulation of G, if H is chordal, contains G as a subgraph, and there is no chordal graph K that contains G as a subgraph, and is a proper subgraph of K.) Potential maximal cliques can be seen as the building blocks of tree decompositions: there is always a tree decomposition of optimal width $(\{X_i \mid i \in I\}, T = (I, F))$ such that each X_i is a potential maximal clique.

Is there an algorithm that lists all the potential maximal cliques of a graph G in $O(p(n) \cdot r)$ time, if G has r potential maximal cliques?

Such an algorithm would imply a faster exponential time algorithm for Treewidth. Also, it might be of use for practical algorithms to compute the treewidth, as the listing of potential maximal cliques seem the practical bottleneck for the algorithms of [27, 46].

2.8 Subexponential time solutions for graph problems without topological constraints (contributed by Jianer Chen)

Several graph problems restricted to graphs with some topological constraint have subexponential time solutions. E.g., Alber et al. [1] have shown that Dominating Set can be solved in $O(c^{\sqrt{k}}p(n))$ time on planar graphs; many problems have $O(c^{\sqrt{n}}p(n))$ time algorithms on planar graphs [37]; see e.g., [3, 2, 28, 20, 24].

Algorithms with a similar type of running time have been obtained for problems on other classes with a *topological constraint*, e.g., for graphs on a fixed surface, or graphs avoiding a given graph as minor. See e.g., [22, 21, 23].

At the moment, all NP-hard graph problems that are known to have subexponential time algorithms have certain topological constraints on graphs. Topological constraints seem necessary for some graph problems to have subexponential time algorithms. For example, consider the problems Independent Set, Vertex Cover, Dominating Set (and other related problems) on graphs of genus $g(n)$. It is known that these problems have subexponential time algorithms IF AND ONLY IF $g(n) = o(n)$, see [19].

Are there examples of (natural) problems on graphs, that have not such a topological constraint, and also have subexponential running time, i.e., can be solved in $O(c^{o(n)}p(n))$ time?

2.9 The Exponential Time Hypothesis and Subgraph Problems (contributed by Dániel Marx)

Consider the SUBGRAPH problem: given are graphs $G = (V_G, E_G)$ and $H = (V_H, E_H)$, and the question is whether G is isomorphic to a subgraph of H. Consider the version of this problem, where the parameter is $|V_G|$. MAXIMUM CLIQUE is a special case of this problem, thus SUBGRAPH is clearly W[1]-hard. It is known that there is no $f(k)n^{o(k)}$ time algorithm for finding a k-clique, unless the Exponential Time Hypothesis (ETH) fails [18]; therefore, there is no $f(|V_G|)n^{o(|V_G|)}$ time algorithm for SUBGRAPH unless ETH fails.

Does the same result hold when we take as parameter $|E_G|$? I.e., prove that the following problem has no $f(|E_G|)n^{o(|E_G|)}$ time algorithm, unless ETH fails:

SUBGRAPH
Input: Graphs $G = (V_G, E_G)$, $H = (V_H, E_H)$.
Question: Is G isomorphic to a subgraph of H?
Parameter: $|E_G|$.

Note that if the treewidth of G is $o(|E_G|)$, then the problem can be solved in time $f(|E_G|)n^{o(|E_G|)}$ using color-coding [4]. Therefore, a positive answer to the question should rely on sparse graphs having treewidth linear in the number of vertices, e.g., expander graphs.

3 Conclusion

Many other interesting open problem can be found in current papers, of which we want to explicitly mention the overview of Woeginger on open problems in exact computation [47] from IWPEC 2004.

We hope to see answers to the problems, and more open problems in the literature in the coming years, perhaps in IWPEC 2008.

Acknowledgment

Help by Frances Rosamund with compiling this list is gratefully acknowledged.

References

[1] J. Alber, H. L. Bodlaender, H. Fernau, T. Kloks, and R. Niedermeier. Fixed parameter algorithms for dominating set and related problems on planar graphs. *Algorithmica*, 33:461–493, 2002.

[2] J. Alber, M. R. Fellows, and R. Niedermeier. Polynomial-time data reduction for dominating sets. *J. ACM*, 51:363–384, 2004.

[3] J. Alber, H. Fernau, and R. Niedermeier. Graph separators: A parameterized view. *J. Comp. Syst. Sc.*, 67:808–832, 2003.

[4] N. Alon, R. Yuster, and U. Zwick. Color-coding. *J. ACM*, 42:855–856, 1995.

[5] E. Amir. Efficient approximation for triangulation of minimum treewidth. In *Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence*, pages 7–15, 2001.

[6] S. Arnborg, D. G. Corneil, and A. Proskurowski. Complexity of finding embeddings in a k-tree. *SIAM J. Alg. Disc. Meth.*, 8:277–284, 1987.

[7] R. Bar-Yehuda, I. Feldman, and D. Rawitz. Improved approximation algorithm for convex recoloring of trees. In *Proceedings Third Workshop on Approximation and Online Algorithms WAOA 2005*, pages 55–68, 2005.

[8] H. L. Bodlaender. A linear time algorithm for finding tree-decompositions of small treewidth. *SIAM J. Comput.*, 25:1305–1317, 1996.

[9] H. L. Bodlaender. A cubic kernel for feedback vertex set. Technical Report UU-CS-2006-042, Department of Information and Computing Sciences, Utrecht University, Utrecht, The Netherlands, 2006.

[10] H. L. Bodlaender, M. R. Fellows, M. Langston, M. Ragan, F. Rosamond, and M. Weyer. Kernelization for convex recoloring of trees. Paper in preparation, 2006.

[11] H. L. Bodlaender and T. Kloks. Efficient and constructive algorithms for the pathwidth and treewidth of graphs. *J. Algorithms*, 21:358–402, 1996.

[12] V. Bouchitté and I. Todinca. Treewidth and minimum fill-in: Grouping the minimal separators. *SIAM J. Comput.*, 31:212–232, 2001.

[13] V. Bouchitté and I. Todinca. Listing all potential maximal cliques of a graph. *Theor. Comp. Sc.*, 276:17–32, 2002.

[14] K. Burrage, V. Estivill-Castro, M. R. Fellows, M. A. Langston, S. Mac, and F. A. Rosamond. The undirected feedback vertex set problem has a poly(k) kernel. In H. L. Bodlaender and M. A. Langston, editors, *Proceedings 2nd International Workshop on*

Parameterized and Exact Computation, IWPEC 2006, pages 192–202. Springer Verlag, Lecture Notes in Computer Science, vol. 4169, 2006.

[15] L. Cai. Fixed-parameter tractability of graph modification problems for heriditary properties. *Information Processing Letters*, 58:171–176, 1996.

[16] L. Cai. Parameterized complexity of cardinality constrained optimization problems. Manuscript, 2006.

[17] L. Cai, S. M. Chan, and S. O. Chan. Random separation: a new method for solving fixed-cardinality optimization problems. In H. L. Bodlaender and M. A. Langston, editors, *Proceedings 2nd International Workshop on Parameterized and Exact Computation, IWPEC 2006*, pages 239–250. Springer Verlag, Lecture Notes in Computer Science, vol. 4169, 2006.

[18] J. Chen, B. Chor, M. Fellows, X. Huang, D. Juedes, I. Kanj, and G. Xia. Tight lower bounds for certain parameterized NP-hard problems. In *Proceedings of 19th Annual IEEE Conference on Computational Complexity*, pages 150–160, 2004.

[19] J. Chen, I. A. Kanj, L. Perkovic, E. Sedgwick, and G. Xia. Genus characterizes the complexity of graph problems: Some tight results. In J. Baeten, J. K. Lenstra, J. Parrow, and G. J. Woeginger, editors, *Proceedings of the 30th International Colloquium on Automata, Languages and Programming, ICALP 2003*, pages 845–856. Springer Verlag, Lecture Notes in Computer Science, vol. 2719, 2003.

[20] 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. *J. ACM*, 52:866–893, 2005.

[21] E. D. Demaine and M. Hajiaghayi. Bidimensionality: New connections between FPT algorithms and PTASs. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms, SODA2005*, pages 590–601, 2005.

[22] E. D. Demaine, M. Hajiaghayi, and D. M. Thilikos. Exponential speedup of fixed parameter algorithms on $K_{3,3}$-minor-free or K_5-minor-free graphs. In *Proceeings of the 13th Annual Symposium on Algorithms and Computation SODA*, pages 262–273, 2002.

[23] E. D. Demaine, M. Hajiaghayi, and D. M. Thilikos. Exponential speedup of fixed-parameter algorithms for classes of graphs excluding single-crossing graphs as minors. *Algorithmica*, 41:245–267, 2005.

[24] F. Dorn, E. Penninkx, H. L. Bodlaender, and F. V. Fomin. Efficient exact algorithms on planar graphs: Exploiting sphere cut branch decompositions. In *Proceedings 13th Annual European Symposium on Algorithms ESA'2005*, pages 95–106. Springer Verlag, Lecture Notes in Computer Science, vol. 3669, 2005.

[25] U. Feige, M. Hajiaghayi, and J. R. Lee. Improved approximation algorithms for minimum-weight vertex separators. In *Proceedings of the 37th Annual Symposium on Theory of Computing, STOC 2005*, pages 563–572. ACM Press, 2005.

[26] H. Fernau. Edge dominating set: Efficient enumeration-based exact algorithms. In H. L. Bodlaender and M. A. Langston, editors, *Proceedings 2nd International Workshop on Parameterized and Exact Computation, IWPEC 2006*, pages 140–151. Springer Verlag, Lecture Notes in Computer Science, vol. 4169, 2006.

[27] F. V. Fomin, D. Kratsch, and I. Todinca. Exact (exponential) algorithms for treewidth and minimum fill-in. In *Proceedings of the 31st International Colloquium on Automata, Languages and Programming, ICALP 2004*, pages 568–580, 2004.

[28] F. V. Fomin and D. M. Thilikos. New upper bounds on the decomposability of planar graphs. *J. Graph Theory*, 51:53–81, 2006.

[29] J. Gramm, J. Guo, F. Hüffner, and R. Niedermeier. Automated generation of search tree algorithms for hard graph-modification problems. *Algorithmica*, 29:321–347, 2004.

[30] J. Gramm, J. Guo, F. Hüffner, and R. Niedermeier. Data reduction, exact, and heuristic algorithms for clique cover. In *Proceedings 8th Workshop on Algorithm Engineering and Experiments ALENEX'06*, pages 86–94. SIAM, 2006.

[31] Q.-P. Gu and H. Tamaki. Optimal branch-decomposition of planar graphs in $O(n^3)$ time. In *Proceedings of the 32nd International Colloquium on Automata, Languages and Programming, ICALP 2005*, pages 373–384. Springer Verlag, Lecture Notes in Computer Science, vol. 3580, 2005.

[32] J. Guo. *Algorithm Design Techniques for Parameterized Graph Modification Problems.* PhD thesis, Institut für Informatik, Friedrich-Schiller-Universität, Jena, Germany, 2006.

[33] I. V. Hicks. Planar branch decompositions I: The ratcatcher. *INFORMS J. on Computing*, 17:402–412, 2005.

[34] I. V. Hicks. Planar branch decompositions II: The cycle method. *INFORMS J. on Computing*, 17:413–421, 2005.

[35] J. Lagergren. Efficient parallel algorithms for graphs of bounded tree-width. *J. Algorithms*, 20:20–44, 1996.

[36] J. Lagergren and S. Arnborg. Finding minimal forbidden minors using a finite congruence. In *Proceedings of the 18th International Colloquium on Automata, Languages and Programming, ICALP'91*, pages 532–543. Springer Verlag, Lecture Notes in Computer Science, vol. 510, 1991.

[37] R. J. Lipton and R. E. Tarjan. Applications of a planar separator theorem. *SIAM J. Comput.*, 9:615–627, 1980.

[38] S. Moran and S. Snir. Efficient approximation of convex recolorings. In *Proceedings APPROX 2005: 8th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, published with Proceedings RANDOM 2005*, pages 192–208. Springer Verlag, Lecture Notes in Computer Science, vol. 3624, 2005.

[39] N. Nishimura, P. Ragde, and D. M. Thilikos. Smaller kernels for hitting set problems of constant arity. In R. G. Downey and M. R. Fellows, editors, *Proc. 1st Int. Workshop on Parameterized and Exact Computation, IWPEC 2004*, pages 239–250. Springer Verlag, Lecture Notes in Computer Science, vol. 3162, 2004.

[40] L. Perković and B. Reed. An improved algorithm for finding tree decompositions of small width. *Int. J. Found. Computer Science*, 11:365–371, 2000.

[41] E. Prieto. *Systematic Kernelization in FPT Algorithm Design.* PhD thesis, University of Newcastle, Australia, 2005.

[42] B. Reed. Finding approximate separators and computing tree-width quickly. In *Proceedings of the 24th Annual Symposium on Theory of Computing*, pages 221–228, New York, 1992. ACM Press.

[43] N. Robertson and P. D. Seymour. Graph minors. XIII. The disjoint paths problem. *J. Comb. Theory Series B*, 63:65–110, 1995.

[44] P. D. Seymour and R. Thomas. Call routing and the ratcatcher. *Combinatorica*, 14(2):217–241, 1994.

[45] S. Snir. *Computational Issues in Phylogenetic Reconstruction: Analytic Maximum Likelihood Solutions, and Convex Recoloring.* PhD thesis, Department of Computer Science, Technion, Haifa, Israel, 2004.

[46] Y. Villanger. Counting and listing all potential maximal cliques of a graph. Reports in Informatics 302, Department of Informatics, University of Bergen, Bergen, Norway, 2005.

[47] G. J. Woeginger. Space and time complexity of exact algorithms: some open problems (invited talk). In R. G. Downey and M. R. Fellows, editors, *Proceedings 1st International Workshop on Parameterized and Exact Computation, IWPEC 2004*, pages 281–290, Berlin, 2004. Springer Lecture Notes in Computer Science, vol. 3162.