Groverize Monotone Local Search. (Short Note)

David Ponarovsky

April 24, 2023

References

[1] A. Agrawal, D. Lokshtanov, S. Kolay, and S. Saurabh, *A faster FPT algorithm and a smaller kernel for block graph vertex deletion*, in Proceedings of the 12th Latin American Theoretical Informatics Symposium (LATIN 2016), Lecture Notes in Computer Science, Springer, to appear. Available as arXiv CoRR abs/1510.08154.

[2] N. Alon, L. Babai, and A. Itai, *A fast and simple randomized parallel algorithm for the maximal independent set problem*, J. Algorithms, 7 (1986), pp. 567–583.

[3] J. Baste, L. Faria, S. Klein, and I. Sau, *Parameterized complexity dichotomy for (r, ℓ)-vertex deletion*, Tech. Report abs/1504.05515, arXiv CoRR, 2015.

[4] I. Bliznets, F. V. Fomin, M. Pilipczuk, and Y. Villanger, *Largest chordal and interval subgraphs faster than 2^n*, in Proceedings of the 21st Annual European Symposium on Algorithms (ESA), vol. 8125 of Lecture Notes in Comput. Sci., Springer, 2013, pp. 193–204.

[5] A. Boral, M. Cygan, T. Kociumaka, and M. Pilipczuk, *A fast branching algorithm for cluster vertex deletion*, in Proceedings of the 9th International Computer Science Symposium in Russia (CSR), vol. 8476 of Lecture Notes in Computer Science, Springer, 2014, pp. 111–124.

[6] A. Brandstädt, V. B. Le, and J. P. Spinrad, *Graph Classes: A Survey*, Discrete Mathematics and Applications, SIAM, 1999.

[7] Y. Cao, *Unit interval editing is fixed-parameter tractable*, in Proceedings of the 42nd International Colloquium of Automata, Languages and Programming (ICALP), vol. 9134 of Lecture Notes in Comput. Sci., Springer, 2015, pp. 306–317.

[8] Y. Cao, *Linear recognition of almost interval graphs*, in Proceedings of the 27th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2016), to appear. Available as arXiv CoRR abs/1403.1515.

[9] Y. Cao, J. Chen, and Y. Liu, *On feedback vertex set: New measure and new structures*, Algorithmica, 73 (2015), pp. 63–86.

[10] J. Chen, F. V. Fomin, Y. Liu, S. Lu, and Y. Villanger, *Improved algorithms for feedback vertex set problems*, J. Computer and System Sciences, 74 (2008), pp. 1188–1198.

[11] M. Cochefert, J.-F. Couturier, S. Gaspers, and D. Kratsch, *Faster algorithms to enumerate hypergraph transversals*, in Proceedings of the 12th Latin American Theoretical Informatics Symposium (LATIN 2016), Lecture Notes in Computer Science, Springer, to appear. Available as arXiv CoRR abs/1510.05093.

[12] M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh, *Parameterized Algorithms*, Springer, 2015.

[13] M. Cygan, J. Nederlof, M. Pilipczuk, M. Pilipczuk, J. M. M. van Rooij, and J. O. Wojtaszczyk, *Solving connectivity problems parameterized by treewidth in single exponential time*, in Proceedings of the 52nd Annual Symposium on Foundations of Computer Science (FOCS), IEEE, 2011, pp. 150–159.

[14] E. Dantsin, A. Goerdt, E. A. Hirsch, R. Kannan, J. Kleinberg, C. Papadimitriou, P. Raghavan, and U. Schöning, *A deterministic* $(2-2/(k+1))^n$ *algorithm for k-SAT based on local search*, Theoretical Computer Science, 289 (2002), pp. 69–83.

[15] F. K. H. A. Dehne, M. R. Fellows, M. A. Langston, F. A. Rosamond, and K. Stevens, *An* $O(2^{O(k)}n^3)$ *FPT algorithm for the undirected feedback vertex set problem*, Theory of Computing Systems, 41 (2007), pp. 479–492.

[16] M. R. Fellows, F. V. Fomin, D. Lokshtanov, F. A. Rosamond, S. Saurabh, and Y. Villanger, *Local search: Is brute-force avoidable?*, J. Comput. Syst. Sci., 78 (2012), pp. 707–719.

[17] F. V. Fomin, S. Gaspers, D. Kratsch, M. Liedloff, and S. Saurabh, *Iterative compression and exact algorithms*, Theor. Comput. Sci., 411 (2010), pp. 1045–1053.

[18] F. V. Fomin, S. Gaspers, A. V. Pyatkin, and I. Razgon, *On the minimum feedback vertex set problem: Exact and enumeration algorithms*, Algorithmica, 52 (2008), pp. 293–307.

[19] F. V. Fomin, P. Heggernes, D. Kratsch, C. Papadopoulos, and Y. Villanger, *Enumerating minimal subset feedback vertex sets*, Algorithmica, 69 (2014), pp. 216–231.

[20] F. V. Fomin and D. Kratsch, *Exact Exponential Algorithms*, Springer, 2010. An EATCS Series: Texts in Theoretical Computer Science.

[21] F. V. Fomin, I. Todinca, and Y. Villanger, *Large induced subgraphs via triangulations and CMSO*, SIAM J. Comput., 44 (2015), pp. 54–87.

[22] S. Gaspers, E. J. Kim, S. Ordyniak, S. Saurabh, and S. Szeider, *Don't be strict in local search!*, in Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI), AAAI Press, 2012.

[23] S. Gaspers and M. Mnich, *Feedback vertex sets in tournaments*, Journal of Graph Theory, 72 (2013), pp. 72–89.

[24] P. A. Golovach, P. van 't Hof, and D. Paulusma, *Obtaining planarity by contracting few edges*, Theoretical Computer Science, 476 (2013), pp. 38–46.

[25] J. Guo, J. Gramm, F. Hüffner, R. Niedermeier, and S. Wernicke, *Compression-based fixed-parameter algorithms for feedback vertex set and edge bipartization*, J. Computer and System Sciences, 72 (2006), pp. 1386–1396.

[26] I. A. Kanj, G. Lin, T. Liu, W. Tong, G. Xia, J. Xu, B. Yang, F. Zhang, P. Zhang, and B. Zhu, *Algorithms for cut problems on trees*, in Proceedings of the 8th International Conference on Combinatorial Optimization and Applications (COCOA), vol. 8881 of Lecture Notes in Computer Science, Springer, 2014, pp. 283–298.

[27] M. M. Kanté, E. J. Kim, O. Kwon, and C. Paul, *An FPT algorithm and a polynomial kernel for linear rankwidth-1 vertex deletion*, in Proceedings of the 10th International Symposium on Parameterized and Exact Computation (IPEC 2015), vol. 43 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2015, pp. 138–150.

[28] T. Kociumaka and M. Pilipczuk, *Faster deterministic feedback vertex set*, Inf. Process. Lett., 114 (2014), pp. 556–560.

[29] S. Kolay and F. Panolan, *Parameterized algorithms for deletion to (r, ℓ)-graphs*, in Proceedings of FSTTCS 2015, to appear. Available as arXiv CoRR abs/1504.08120.

[30] M. Kumar and D. Lokshtanov, *Faster exact and parameterized algorithm for feedback vertex set in tournaments*, in Proceedings of the 33rd International Symposium on Theoretical Aspects of Computer Science (STACS 2016), to appear. Available as arXiv CoRR abs/1510.07676.

[31] K. Kutzkov and D. Scheder, *Using CSP to improve deterministic 3-SAT*, CoRR, abs/1007.1166 (2010).

[32] D. Marx and I. Schlotter, *Stable assignment with couples: Parameterized complexity and local search*, Discrete Optimization, 8 (2011), pp. 25–40.

[33] J. W. Moon, *On maximal transitive subtournaments*, Proc. Edinburgh Math. Soc. (2), 17 (1971), pp. 345–349.

[34] R. A. Moser and D. Scheder, *A full derandomization of Schöning's k-SAT algorithm*, in Proceedings of the 43rd Annual ACM Symposium on Theory of Computing (STOC), ACM, 2011, pp. 245–252.

[35] M. Naor, L. J. Schulman, and A. Srinivasan, *Splitters and near-optimal derandomization*, in Proceedings of the 36th Annual Symposium on Foundations of Computer Science (FOCS), IEEE, 1995, pp. 182–191.

[36] I. Razgon, *Exact computation of maximum induced forest*, in Proceedings of the 10th Scandinavian Workshop on Algorithm Theory (SWAT 2006), vol. 4059 of Lecture Notes in Comput. Sci., Berlin, 2006, Springer, pp. 160–171.

[37] ——, *Computing minimum directed feedback vertex set in $o(1.9977^n)$*, in Proceedings of the 10th Italian Conference on Theoretical Computer Science (ICTCS), 2007, pp. 70–81.

[38] U. Schöning, *A probabilistic algorithm for k-SAT and constraint satisfaction problems*, in 40th Annual Symposium on Foundations of Computer Science (New York, 1999), IEEE Computer Soc., Los Alamitos, CA, 1999, pp. 410–414.

[39] H. Shachnai and M. Zehavi, *A multivariate approach for weighted FPT algorithms*, in Proceedigs of the 23rd Annual European Symposium on Algorithms (ESA), vol. 9294 of Lecture Notes in Computer Science, Springer, 2015, pp. 965–976.

[40] P. van 't Hof and Y. Villanger, *Proper interval vertex deletion*, Algorithmica, 65 (2013), pp. 845–867.

[41] M. Wahlström, *Algorithms, measures and upper bounds for satisfiability and related problems*, PhD thesis, Linköping University, Sweden, 2007.

[42] ——, *Half-integrality, LP-branching and FPT algorithms*, in Proceedings of the 24th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), SIAM, 2014, pp. 1762–1781.

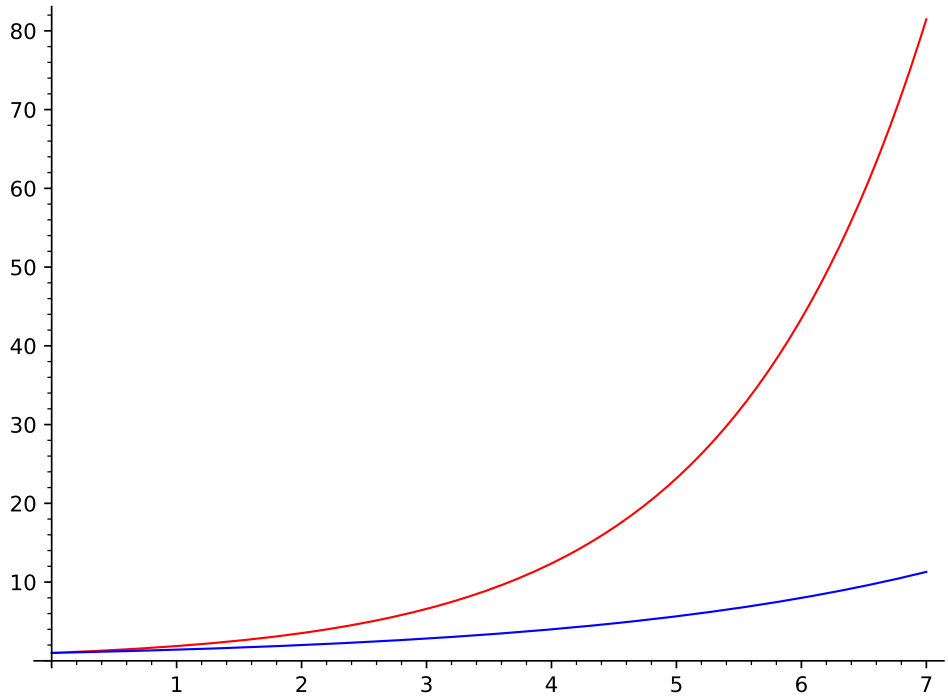
[43] M. Xiao and H. Nagamochi, *Exact algorithms for maximum independent set*, in Proceedings of the 24th International Symposium on Algorithms and Computation (ISAAC), vol. 8283 of Lecture Notes in Computer Science, Springer, 2013, pp. 328–338. See also arXiv CoRR abs/1312.6260.

[44] T. Yato and T. Seta, *Complexity and completeness of finding another solution and its application to puzzles*, IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, E86-A (2003), pp. 1052–1060.

1 Introduction.

We follow the study of [Fom+15], who relate between the parametrized complexity to the general average case complexity. Crudely put, they shown that for particular wide range of **NP** hard problems, a solution which run exponentially at some complexity parameter, for example the tree-width of a graph, can be used to derive a batter than bruteforce solution for the general problem. We continue their work by plugin the Grover search [Gro96] routine instead the original sampling process.

$$\sum_{k' \leq k} \frac{1}{\sqrt{p(k')}} \cdot c^{k'-t} N^{\mathcal{O}(1)} \leq \max_{k' \leq k} \left(\frac{\binom{n-|X|}{t}}{\binom{k'}{t}} \right)^{\frac{1}{2}} \cdot c^{k'-t} N^{\mathcal{O}(1)} =$$
$$\left(\max_{k' \leq k} \frac{\binom{n-|X|}{t}}{\binom{k'}{t}} \cdot c^{2\left(k'-t\right)} \right)^{\frac{1}{2}} N^{\mathcal{O}(1)} = \left(\max_{k \leq n-|X|} \frac{\binom{n-|X|}{t}}{\binom{k}{t}} \right)^{\frac{1}{2}} \cdot c^{2(k-t)} N^{\mathcal{O}(1)} \leq$$
$$\Rightarrow \left(2 - \frac{1}{c^2}\right)^{\frac{n-|X|}{2}} N^{\mathcal{O}(1)}$$



References

[Gro96] Lov K. Grover. *A fast quantum mechanical algorithm for database search*. 1996. arXiv: quant-ph/9605043 [quant-ph].

[Fom+15] Fedor V. Fomin et al. *Exact Algorithms via Monotone Local Search*. 2015. arXiv: 1512.01621 [cs.DS].

Problem Name	Parameterized		New bound	Previous Bound	
FEEDBACK VERTEX SET	3^k (r)	[**cut-and-count**]	1.6667^n (r)		
FEEDBACK VERTEX SET	3.592^k	[**KociumakaP13**]	1.7217^n	1.7347^n	
SUBSET FEEDBACK VERTEX SET	4^k	[**Wahlstrom14**]	1.7500^n	1.8638^n	
FEEDBACK VERTEX SET IN TOURNAMENTS	1.6181^k	[**KumarL16**]	1.3820^n	1.4656^n	
GROUP FEEDBACK VERTEX SET	4^k	[**Wahlstrom14**]	1.7500^n	NPR	
NODE UNIQUE LABEL COVER	$\lvert\Sigma\rvert^{2k}$	[**Wahlstrom14**]	$(2 - \frac{1}{\lvert\Sigma\rvert^2})^n$	NPR	
VERTEX (r, ℓ)-PARTIZATION $(r, \ell \leq 2)$	3.3146^k	[**BasteFKS15**; **KolayP15**]	1.6984^n	NPR	
INTERVAL VERTEX DELETION	8^k	[**Cao8kinterval**]	1.8750^n	$(2 - \varepsilon)^n$ for $\varepsilon < 10^{-20}$	
PROPER INTERVAL VERTEX DELETION	6^k	[**HofV13**; **Cao15**]	1.8334^n	$(2 - \varepsilon)^n$ for $\varepsilon < 10^{-20}$	
BLOCK GRAPH VERTEX DELETION	4^k	[**AgrawalLKS16**]	1.7500^n	$(2 - \varepsilon)^n$ for $\varepsilon < 10^{-20}$	
CLUSTER VERTEX DELETION	1.9102^k	[**BoralCKP14**]	1.4765^n	1.6181^n	
THREAD GRAPH VERTEX DELETION	8^k	[**Kante0KP15**]	1.8750^n	NPR	
MULTICUT ON TREES	1.5538^k	[**KanjLLTXXYZZZ14**]	1.3565^n	NPR	
3-HITTING SET	2.0755^k	[**MagnusPhD07**]	1.5182^n	1.6278^n	
4-HITTING SET	3.0755^k	[**FominGKLS10**]	1.6750^n	1.8704^n	
d-HITTING SET $(d \geq 3)$	$(d - 0.9245)^k$	[**FominGKLS10**]	$(2 - \frac{1}{(d-0.9245)})^n$	[**CochefertCGK16**	
MIN-ONES 3-SAT	2.562^k	[**abs-1007-1166**]	1.6097^n	NPR	
MIN-ONES d-SAT $(d \geq 4)$	d^k		$(2 - \frac{1}{d})^n$	NPR	
WEIGHTED d-SAT $(d \geq 3)$	d^k		$(2 - \frac{1}{d})^n$	NPR	
WEIGHTED FEEDBACK VERTEX SET	3.6181^k	[**AgrawalLKS16**]	1.7237^n	1.8638^n	[**F**
WEIGHTED 3-HITTING SET	2.168^k	[**ShachnaiZ15**]	1.5388^n	1.6755^n	[**C**
WEIGHTED d-HITTING SET $(d \geq 4)$	$(d - 0.832)^k$	[**FominGKLS10**; **ShachnaiZ15**]	$(2 - \frac{1}{d-0.932})^n$		[**C**

Table 1: Summary of known and new results for different optimization problems. NPR means that we are not aware of any previous algorithms faster than brute-force. All bounds suppress factors polynomial in the input size N. The algorithms in the first row are randomized (r).