Makespan Scheduling With Neighborly Help

Fabian Frei [fabian.frei@inf.ethz.ch] and
Rodrigo R. Gumucio Escobar [grodrigo@student.ethz.ch]
Department of Computer Science, ETH Zürich

1 Handling Hard Optimization Problems

Confronted with the notorious difficulty of addressing the P versus NP question, it is natural to wonder how far and in what way NP-hard optimization problems may differ from each other. A good way to do this is by examining these problems under alternative models.

1.1 Established Models

Approximation The approximation approach is very good at revealing distinctions among NP-hard problems. Some of them become polynomial-time solvable as soon as we are willing to settle for a solution that is worse than the optimum just by an arbitarily small factor, while others remain NP-hard even for very coarse approximations. For example, Knapsack admits an FPTAS, implying that an optimal solution can be approximated to an arbitrarily small factor in polynomial time, whereas Max-Clique lies at the other end of the spectrum, having no polynomial-time algorithm that computes even an $n^{1-\varepsilon}$-approximation for any $\varepsilon > 0$.

Reoptimization Another approach to simplify NP-hard problems is (classical) reoptimization. Here, we help the algorithm by supplying it with an optimal solution to a neighboring instance. We speak of a *neighbor instance* if it can be reached from the given instance by a single *local modification*, which needs to be specified of course. Considering graph problems, for example, the most natural choices of local modifications are arguably the deletion or addition of an edge or vertex. We can describe the task that is set by the reoptimization model for an optimization problem more precisely as follows:

Given an instance, an optimal solution to it, and a local modification of this instance, compute an optimal solution to the modified instance.

We could hope that the considerable advice provided to us in the form of an optimal solution to a neighbor instance might help us to solve an otherwise NP-hard problem in polynomial time, as it is the case for approximation. However, most problems examined so far remain NP-hard under the reoptimization model; see Böckenhauer et al. [5]. Only for some of them could at least an improvement of the approximation ratio be achieved after extensive studies, the first discovered examples being TSP under edge-weight changes [4] and addition or deletion of vertices [1].

1.2 New Model: Neighborly Help

Burjons et al. [2, 3] recently introduced a new model under the moniker "Neighborly Help."

Definition of the model The neighborly-help model asks us to find an optimal solution with the support of an oracle that provides optimal solutions to neighboring instances. This models natural situations such as that of a newcomer who may ask experienced peers for advice on how to solve a difficult problem, for instance finding an optimal service tour.

Formally, we consider the following oracle model: An algorithm may, on any given input, repeatedly select an arbitrary instance neighboring the given one and query the oracle for an optimal solution to it. Occasionally, it will be interesting to limit the number of queries that we grant the algorithm. In general, we do not impose such a restriction, however.

Relation to Reoptimization The neighborly-help model is quite close to that of classical reoptimization. In order to highlight both the proximity and differences between them, we reformulate the reoptimization task from above by reversing the roles of the given and modified instance:

> *Given an instance, a local modification of it, and an optimal solution to the modified instance, compute an optimal solution to the original instance.*

Note that this change of perspective flips the definition of local modification; for example, edge deletion turns into edge addition. Aside from this, the task now reads almost identical to the one demanded in our model. The sole but crucial difference is the following. In reoptimization, the neighboring instance and the optimal solution to it are given as part of the input, whereas in our model, the algorithm may select any number of neighboring instances and query the oracle for optimal solutions to them. Even if we limit the number of queries to just one, our model is still more generous since the algorithm is choosing (instead of being given) the neighboring instance to which the oracle will supply an optimal solution. Thus, hardness in our model always implies hardness for reoptimization, but not vice versa. Indeed, there are problems that become polynomial-time solvable with this extended advice, even though they remain NP-hard in the classical reoptimization model as the following results show.

Known Results The paper introducing the neighborly-help model [2, 3] analyzes two problems, COLORABILITY and VERTEX COVER, under the standard modifications. It is shown that both problems, depending on the local modification and the number of queries, either remain NP-hard or become polynomial-time solvable. The results are summarized in Table 1.

1.3 New Contribution: Makespan Scheduling

We examine MAKESPAN in the neighborly-help model and show it remains strongly NP-hard for the four most natural modifications: adding a machine, deleting a machine, adding a job of arbitrary size, and deleting a job. The proofs for the addition and deletion of a job require rather involved constructions that yield a number of interesting corollaries; among these are nontrivial results on related problems such us BINPACKING and the strong NP-hardness of transposing a makespan solution—for example, given an arrangement of n jobs on m machines with makespan h, re-arrange them on h machines with makespan m.

This constitutes an interesting addition to the known results on Neighborly Help in two ways. On the one hand, it is the first analysis of an NP-hard problem that is not related to graphs. On the other hand, Makespan is the first problem shown to remain NP-hard (and strongly NP-hard, in fact) for the most natural local modifications, even with an unrestricted number queries to the oracle.

Table 1. An overview of the results regarding the hardness of Colorability and Vertex Cover in the neighborly-help model. The v stands for a vertex and the e stands for an edge. The question mark indicates an interesting open problem.

fbur-fre

No. of Queries	Colorability				Vertex Cover			
	Add v	Delete v	Add e	Delete e	Add v	Delete v	Add e	Delete e
1	P	NP-hard	NP-hard	NP-hard	P	NP-hard	NP-hard	NP-hard
2 or more	P	NP-hard	P	NP-hard	P	P	P	?

References

[1] Giorgio Ausiello, Bruno Escoffier, Jérôme Monnot, and Vangelis Paschos. Reoptimization of minimum and maximum traveling salesman's tours. In: *Proceedings of the 10th Scandinavian Workshop on Algorithm Theory (SWAT 2006)*, volume 4059 of *Lecture Notes in Computer Science*, pages 196–207. Springer-Verlag, 2006.

[2] Elisabet Burjons, Fabian Frei, Edith Hemaspaandra, Dennis Komm, and David Wehner. Finding optimal solutions with neighborly help. *CoRR*, abs/1906.10078, 2019. URL: http://arxiv.org/abs/1906.10078, arXiv:1906.10078.

[3] Elisabet Burjons, Fabian Frei, Edith Hemaspaandra, Dennis Komm, and David Wehner. Finding optimal solutions with neighborly help. In: *Proceedings of the 34th International Symposium on Mathematical Foundations of Computer Science (MFCS 2019)*. Springer-Verlag *Lecture Notes in Computer Science*, 2019. To appear.

[4] Hans-Joachim Böckenhauer, Luca Forlizzi, Juraj Hromkovič, Joachim Kneis, Joachim Kupke, Guido Proietti, and Peter Widmayer. Reusing optimal TSP solutions for locally modified input instances. In: *Proceedings of the 4th IFIP International Conference on Theoretical Computer Science (IFIP TCS 2006)*, pages 251–270. Springer, 2006.

[5] Hans-Joachim Böckenhauer, Juraj Hromkovič, and Dennis Komm. Reoptimization of hard optimization problems. In: Teofilo F. Gonzalez, editor, *AAM Handbook of Approximation Algorithms and Metaheuristics*, volume 1, chapter 25, pages 427–454. CRC Press 2018, 2nd edition, 2018.