Steiner Tree Heuristics for PACE 2018 Challenge Track C*

Radek Hušek[1], Tomáš Toufar[1], Dušan Knop[2], Tomáš Masařík[3], and Eduard Eiben[2]

[1]Computer Science Institute, Charles University, Prague, Czech Republic
[2]Department of Informatics, University of Bergen, Norway
[3]Department of Applied Mathematics, Charles University, Prague, Czech Republic

April 2018

This repository (`https://github.com/goderik01/PACE2018`) contains a program that computes a heuristic approximation for the Weighted Steiner Tree problem. It was created for the PACE Challenge 2018 Track C (heuristic track) `https://pacechallenge.wordpress.com/pace-2018/`, see also the report [2].

Compile instructions. The root folder contains a Makefile with all the neccessary targets. Our program uses the boost library `https://www.boost.org/` (it was tested with version 1.66.0, see also the documentation [1]). We recommend to dowload its sources (`make boost` does the job for you). Then, you can compile the sources by running `make` command which uses g++ compiler.

Input/Output description. To download public test data use `make data` which downloads test public data from PACE challenge into directory *data/*.

The input&output format is described on the PACE Challenge website.

How to run the program. The compiled binary (*bin/star_ contractions_ test*) expects input data on the standard input. The program run until it recieves the SIGTERM signal. After that it outputs a solution to the standard output within 30 seconds.

Concise description of our algorithm. Our algorithm proceeds with the following steps:

- Run simple 1-safe heuristics.
  - Handle small degree vertices.
    - Suppress Steiner vertices of degree two.
    - Remove Steiner vertices of degree one.
    - Buy edges incident to terminal of degree one.
  - Buy zero cost edges.
  - Run *Shortest path test*: Remove edges longer than the length of the shortest path between the endpoints.
  - Run *Terminal distance test* [5]: Buy the lightest edge on a cut if the second lightest edge in the cut is heavier than shortest path between any two terminals that are on different sides of the cut.

*The work was supported by the project GAUK 1514217 and the project SVV–2017–260452. R. Hušek, D. Knop, T. Masařík we also supported by the Center of Excellence – ITI, project P202/12/G061 of GA ČR. D. Knop was supported by the project NFR MULTIVAL of the Research Council of Norway. E. Eiben was supported by Pareto-Optimal Parameterized Algorithms (ERC Starting Grant 715744).

- Perform star contractions based on paper Parameterized Approximation Schemes for Steiner Trees with Small Number of Steiner Vertices [4] for at most 10 minutes.
    - Buy the star (center is any vertex and lists are Terminals) in the metric closure that has the best ratio (sum of the weights over number of terminals minus one).
    - After each step run following heuristics: handle small degree Steiner vertices and fast special case of shortest path test.
    - If time runs out, finish the partial solution using MST-approximation.
- Improve the solution by following methods:
    - *Local search using MST-approximation.* From the solution compute the MST-approximation on all terminals and all branching Steiner vertices plus a few randomly chosen promising Steiner vertices. We clean up the solution and repeat without additional random vertices.
    - *Local search using Dreyfus-Wagner partition.* Inspired by Dreyfus-Wagner FPT algorithm [3] we derive a heuristic that obtains the partition of the vertices given one of the promising solutions and then compute an optimal Steiner tree with this structure. This step is much more time consuming so we run it only sparsely.

Acknowledgments. We thank to Torstein Strømme for helpful discussions.

References

[1] *The Boost Graph Library: User Guide and Reference Manual.* Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2002.

[2] Édouard Bonnet and Florian Sikora. The PACE 2018 parameterized algorithms and computational experiments challenge: The third iteration. In *13th International Symposium on Parameterized and Exact Computation, IPEC 2018, August 20-24, 2018, Helsinki, Finland*, pages 26:1–26:15, 2018.

[3] S. E. Dreyfus and R. A. Wagner. The steiner problem in graphs. *Networks*, 1(3):195–207, 1971.

[4] Pavel Dvořák, Andreas Emil Feldmann, Dušan Knop, Tomás Masařík, Tomáš Toufar, and Pavel Veselý. Parameterized approximation schemes for steiner trees with small number of steiner vertices. In *35th Symposium on Theoretical Aspects of Computer Science, STACS 2018, February 28 to March 3, 2018, Caen, France*, pages 26:1–26:15, 2018.

[5] Eduardo Uchoa, Marcus Poggi de Aragão, and Celso C. Ribeiro. Preprocessing steiner problems from VLSI layout. *Networks*, 40(1):38–50, 2002.