Brief introduction to the Steiner tree problem with revenues, budget and hop constraints

Zhang-Hua Fu and Jin-Kao Hao

LERIA, Université d'Angers
2 Boulevard Lavoisier, 49045 Angers Cedex 01, France

1 Problem Definition

Let $G = (V, E)$ be an undirected graph with vertex set $V = \{1, \ldots, n\}$ and edge set $E = \{\{i, j\} \mid i, j \in V, i \neq j\}$. Each vertex $i \in V$ has an associated revenue $r_i \geq 0$ (a vertex with $r_i > 0$ is called a *profitable vertex*, otherwise, it is a *non-profitable vertex*), and each edge $\{i, j\} \in E$ has an associated cost $c_{ij} \geq 0$. Let B be the allowed budget and H be the hop limit. Given a root vertex denoted by *root*, the so-called Steiner Tree Problem with Revenues, Budget and Hop Constraints (STPRBH) is to determine a rooted subtree T that maximizes the collected revenue, while guaranteeing that the total cost of the tree does not exceed B, and the number of edges between *root* and any vertex $i \in v(T)$ never surpasses H [1,3]. According to this definition, only the root vertex is a terminal vertex that must be included in any feasible solution, while all the others are Steiner vertices.

As a generalization of both the Prize-Collecting Steiner Tree Problem (PC-STP) and the Steiner Tree Problem with Hop Constraints (STPH), the STPRBH can be used to model a number of relevant real-life systems where the objective is to maximize the collected revenue within a given budget, while guaranteeing the reliability of the system.

Email addresses: fu@info.univ-angers.fr (Zhang-Hua Fu), hao@info.univ-angers.fr (Jin-Kao Hao).

2 History

Since this problem was first proposed by A.M. Costa in 2006 [1], a number of exact or heuristic approaches have been proposed for the STPRBH. Exact algorithms like exemplified in [3–5] are applicable to solve small or mid-sized problem instances (with up to 500 vertices and 625 edges). For large instances, heuristics or meta-heuristics are naturally preferred, including the greedy, destroy-and-repair and tabu search (TS) algorithms described in [1,2], and our recently proposed breakout local search algorithm [6] and dynamic programming driven memetic search algorithm [7].

3 Benchmark Instances

3.1 Instances created by A.M. Costa

In order to evaluate the performances of various STPRBH algorithms, A.M. Costa created 384 STPRBH instances based on the classic STP graphs of the OR-Library (series B and C). At first, 58 STPRBH graphes are adapted from the original STP graphs in the following mode [1,2]: given a STP graph, consider each terminal vertex as a profitable vertex (with revenue randomly distributed within a given interval) and consider each Steiner vertex as a non-profitable vertex (attributed a zero revenue), while keeping the edge costs unchanged, to get an adapted SRPRBH graph. Apparently, different revenue intervals correspond to different STPRBH graph. According to A.M. Costa's method, each one of the 18 STP graphs of series B corresponds to one adapted STPRBH graph (with vertices revenues belonging to $[1, 100]$), and each one of the 20 STP graphs of series C corresponds to two different adapted STPRBH graphs (with vertices revenues belonging to $[1, 10]$ or $[1, 100]$), thus leading to a total number of $18 + 20 \times 2 = 58$ adapted STPRBH graphs.

Obviously, for each STPRBH graph, given a root vertex (the root vertex is chosen as the profitable vertex with the smallest index), a budget limitations (B) and a hop limitations (H), a STPRBH instance could be uniquely determined. Following this idea, A.M. Costa generated 384 STPRBH instances by varying the budget and hop limitations, as detailed in Table 1.

Table 1
384 STPRBH instances generated by A.M. Costa [1] based on the B and C series of OR-Library

Original STP Graph	Converted STPRBH instances			
	Revenue Interval	Budget Limitation ($S = \sum_{\{i,j\} \in E} c_{ij}$)	Hop Limitation	Number of Instances
B01.stp - B18.stp	[1,100]	$B = S/5, S/10$	$H = 3, 6, 9, 12$	$18 \times 1 \times 2 \times 4 = 144$
C01.stp - C05.stp	[1,10], [1,100]	$B = S/10, S/30$	$H = 5, 15, 25$	$5 \times 2 \times 2 \times 3 = 60$
C06.stp - C10.stp	[1,10], [1,100]	$B = S/20, S/50$	$H = 5, 15, 25$	$5 \times 2 \times 2 \times 3 = 60$
C11.stp - C15.stp	[1,10], [1,100]	$B = S/20, S/100$	$H = 5, 15, 25$	$5 \times 2 \times 2 \times 3 = 60$
C16.stp - C20.stp	[1,10], [1,100]	$B = S/100, S/200$	$H = 5, 15, 25$	$5 \times 2 \times 2 \times 3 = 60$

3.2 Our new instances

As we pointed out in [6,7], most of (328 out of 384) above instances have been solved to optimality by the existing algorithms, whereas only the 56 left instances remain unsolved. In order to generate more challenging benchmark instances, we generated a new group of 30 STPRBH instances [7] using a similar method like A.M. Costa, based the same adapted STPRBH graphs converted from C16.stp ... C20.stp (corresponding to 10 adapted STPRBH graphs), by restricting the budget limitation. More details are provided in Table 2, where the meaning of each column is the same as in Table 1. Note that, to get a new instance, one should only reset the value of parameter B, without changing the corresponding adapted STPRBH graph.

Table 2
30 new STPRBH instances generated by Z.H. Fu and J.K. Hao [7]

Original STP Graphs	Converted STPRBH instances			
	Revenue Interval	Budget Limitation ($S = \sum_{\{i,j\} \in E} c_{ij}$)	Hop Limitation	Number of Instances
C16.stp	[1,10], [1,100]	$B = S/10000$	$H = 5, 15, 25$	$1 \times 2 \times 1 \times 3 = 6$
C17.stp	[1,10], [1,100]	$B = S/5000$	$H = 5, 15, 25$	$1 \times 2 \times 1 \times 3 = 6$
C18.stp - C20.stp	[1,10], [1,100]	$B = S/1000$	$H = 5, 15, 25$	$3 \times 2 \times 1 \times 3 = 18$

More information could be found in [6,7]. Our best results are available online at `http://www.info.univ-angers.fr/pub/hao/stprbh.html`
and
`http://www.info.univ-angers.fr/pub/hao/stprbh-memetic.html`.

3.3 Possibly more challenging instances

In addition to above 414 instances based on the series B and C of OR-Library, if needed, we can also generate some more STPRBH instances based on the series D, E of the OR-Library (using similar method as described above). Since these graphs are of much larger size, we believe that the converted STPRBH instances would be even more challenging.

4 Research Groups

See the following references for the researchers who have directly contributed to the STPRBH. Since the STPRBH is a generalization of both the Prize-Collecting Steiner Tree Problem and the Steiner Tree Problem with Hop Constraints (which haven already been included in the 11th DIMACS challenge), perhaps some other groups working on these two related problems would also be interested in the STPRBH.

References

[1] A.M. Costa. Models and algorithms for two network design problems. *PhD thesis, Ecole des Hautes Etudes Commerciales*, Montreal, 2006.

[2] A.M. Costa, J.F. Cordeau, and G. Laporte. Fast heuristics for the Steiner tree problem with revenues, budget and hop constraints. *European Journal of Operational Research*, 190(1): 68–78, 2008.

[3] A.M. Costa, J.F. Cordeau, and G. Laporte. Models and branch-and-cut algorithms for the Steiner tree problem with revenues, budget and hop constraints. *Networks*, 53(2): 141–159, 2009.

[4] M. Sinnl. Branch-and-Price for the Steiner tree problem with revenues, budget and hop constraints. *Engineer Thesis*, Factulty of Computer Science, the Technical University of Vienna, 2011.

[5] S.B. Layeb, I. Hajri, and M. Haouari. Solving the Steiner tree problem with revenues, budget and hop constraints to optimality. *Proc. of The 5th International Conference on Modeling, Simulation and Applied Optimization*, pp. 1–4, 2013.

[6] Z.H. Fu, and J.K. Hao. Breakout local search for the Steiner tree problem with revenue, budget and hop constraints. *European Journal of Operational Research*, 232(1): 209–220, 2014.

[7] Z.H. Fu, and J.K. Hao. Dynamic programming driven memetic search for the Steiner tree problem with revenues, budget and hop constraints. Technical report.