International Review of Pure and Applied Mathematics
(January-June) 2017, Volume 13, No. 1, pp. 19–22

A Quick Survey on the Solutions of Degree Constrained Minimum Spanning Tree Problem

R. Anitha & K. Sangavai

Abstract: A spanning tree of a graph such that each node in the tree has degree at most d is called a degree constrained spanning tree. The problem of finding the degree constrained spanning tree of minimum cost in an edge weighted graph is *NP*-complete for every d, in the range $[2, n-2]$, where n denotes the number of nodes.

In this paper we give a brief summary on the solutions of this problem. This will serve as a complete reference for the people who are interested in doing research in this area.

Keywords: Degree-constrained minimum spanning trees, Lagrangean approach, Parallel algorithm, Genetic Algorithm, Evolutionary algorithm, TABU search, Ant Based algorithm.

1. Introduction

d-MST problem entails finding a spanning tree of minimum cost such that no node in the tree exceeds a given degree constraint. This concept is useful in designing networks for everything from computer and telephone communications to transportation and sewage system. For instance, switches in an actual communication network will each have a limited number of connections available. Transportation systems must place a limit on the number of roads meeting in one place. Also, limiting the degree of each node limits the potential impact if a node fails. A Degree constraint in a communication network also limits the liability in the case of node failure. In computer networks, the degree restrictions can be used to cater for the number of line interfaces available at a server or terminal.

2. Methods to Solve *d*-MST Problem

2.1. Lagrangean Approach

A. Volgenant in his paper [8] described an algorithm with a better performance, at least in the case of large random table problems, based on Lagrange multipliers as well as on edge exchanges. Here, the *d-MST* problem was formulated as a linear 0-1 integer programming problem. Weighting constraints by multipliers and adding them to the objective function, the Lagrangean was obtained. The branch and bound procedure has been applied together with an edge analysis procedure both as described by Savelsbergh and Volgenant [6]. The Lagrangean approach is more effective when the problems are more difficult to solve. They have also given computational results to illustrate the success of their approach in combinatorial optimization

2.2 Parallel Algorithms for The D-MST Problem

The Mao. L. J., *et al.*, [3] proposed two complementary approaches to solving the *d-MST* problem and presented a comparison of the parallel implementation of two approximate algorithms *IR* and *TC-RNN*.

2.2.1 *The Iterative – Refinement (IR) Algorithm*

The *IR* algorithm uses an iterative refinement process consisting of two phases: (1) Computing a *MST* (2) Penalizing the edges. First an *MST* using the initial weight matrix is computed. Then in the penalty phase they increase the weights of those tree edges that are incident to the nodes with the degree exceeding the constraint *d*. The *MST* of the graph with new weights is computed next. Alterations of the penalty phase followed by the *MST* computation are repeated until a spanning tree is produced in which every node satisfies the degree bound.

2.2.2 *The TC-RNN (Tree Construction with Reciprocal Nearest Neighbors) Algorithm*

The *TC-RNN* algorithm is based on Sollin's *MST* algorithm. While adapting, they assigned each node to a single processor, and these processors simultaneously find nearest neighboring trees to merge until a single tree remains. In each iteration of the parallel loop, two trees are assigned to two processors are merged if they are nearest neighbors of each other through a common shortest edge connecting the two tree assuming the edge does not cause degree constraint violation

2.3 Genetic Algorithms

2.3.1 *A Weighted Coding Algorithm*

The coding in which chromosomes represent candidate solutions is a fundamental design choice in a genetic algorithm. G. R. Raidl and B. A. Julstrom in their paper [5] described a novel coding of spanning trees in a genetic algorithm for the *d- MST* problem. In that coding chromosome is a string of weights associated with the target problem instance's nodes.

2.3.2 *A Bi-Objective Genetic Algorithm*

All the above algorithms/methods treated the *d-MST* problem as a single objective problem. L. Hanr and Y. Wang [2] presented a bi-objective genetic algorithm for *d- MST* problem with the violation degree as second objective. Based on this model, they have designed a new cross over operation, a local search scheme, a mutation operator and two objectives. The convergence of the proposed algorithm to globally optimal solution with probability one was proved. They have also proved that their algorithm was effective using simulation results.

2.4 Evolutionary Algorithm

The representation of candidate solutions and the variation operators are fundamental design choices in an evolutionary algorithm (EA). There are several EA's to solve *d-MST* problem. Among those, G. R. Raidl [1] proposed a novel representation technique and suitable variation operators. Within the EA, a candidate spanning tree is simply represented by its set of edges. Special initialization, crossover and mutation operators are used to generate new, always feasible candidate solution. In contrast to previous spanning tree representations, the proposed approach provides substantially higher locality and was nevertheless computationally efficient; and offspring was always created in $O(|V|)$ time. In addition, it was shown how problem dependent heuristics can be effectively incorporated into the initialization, crossover and mutation operators without increasing the time complexity. Empirical results were presented for hard problem instances with up to 500 nodes.

2.5 Tabu Search Method

Wamiliana [9] discussed about this TABU search method for the *d-MST* problem. The algorithm starts with finding the *MST*. This gives a lower bound and by using modified Kruskal algorithm a degree constrained spanning tree is obtained which provided an upper bound. The heuristic starts from the upper bound, which is feasible and work towards the optimality. The moves are the set of edges incident with the leaves in the G/T. Tabu tenure is set to be $0.1\ n$, where n is the number of vertices in the graph. The maximum number of iterations, is $0.2\ n$. The stopping criteria are maximum number of iterations and the tolerance, where tolerance = 1% of gap and gap = $UB - LB$. The aspiration criteria are applied if a degree violation is detected. All possible edge exchanges among the edges of T incident to the violated vertex I, are examined. If searching doesn't yield any better solution, we record the current best solution, put the currently used moves into Tabu status and restart.

2.6 An Ant Based Algorithm

In [7] Thang N. Bai and Catherine M. Zrncic described an Ant Based Algorithm for *d-MST* problem and this is the first ant algorithm for this problem. There are several different types of ant algorithms. The authors used Ant-Based algorithm. In this case, artificial ants maneuver based on local information and deposit pheromones as they travel. Then the algorithm uses cumulative pheromone levels to determine the candidate sets of edges from which degree-constrained spanning trees are built. They showed their Ant-Based algorithm finds results that are generally better than results produced by existing GA's through extensive experimental results.

They have explained their algorithm in five steps. They are (i) Initialization (ii) Exploration (iii) Tree construction (iv) Pheromone enhancement and (v) Stopping criteria. The parameters used are also listed. Algorithm was run on a set of 143 complete graphs ranging from 15 to 1000 vertices, with degree constraints of 2 through 5, giving a total of 572 problem instances. They compared their results with previous results and were summarized in tables.

3. Conclusion

In this paper we summarized the different kind of solutions exists for *d-MST* problem. And we have given brief explanations about those methods. It will help the researchers who are interested on spanning trees especially on *d-MST* problem for basic reference. In each approach further development can be made by including some local optimization. For example, Ant Based algorithm can be developed for a parallel implementation, since parallel algorithm will reduce the running time, making it possible to find solutions for larger graphs.

REFERENCES

[1] Gunther R. Raidl, An Efficient Evolutionary Algorithm for the Degree Constrained Spanning Tree Problem, *Proceedings of the 2000 Congress on Evolutionary Computation*, **1** (2000), 104-111.

[2] Hanr L., and Wang Y., A Novel Genetic Algorithm for Degree Constrained Minimum Spanning Tree Problem, *International Journal of Computer Science and Network Security*, **6** (7A) (2006).

[3] Mao L. J., Deo. N., Kumar N., and Lang S. D., *A Comparison of Two Parallel Approximate Algorithms for DCMST Problem*, Congressus Numerantium, **123** (1997), 15-32.

[4] Mao L. J, Deo N., and Lang S. D., A Parallel Algorithm for Degree Constrained *MST* Problem Using Nearest Neighbor Chains, *ISPAN*, (1999), 184-189.

[5] Raidl G. R., and Julstrom B. A., A Weighted Coding in a Genetic Algorithm for the Degree-Constrained Minimum Spanning Tree Problem, *Proceedings of 2000 ACM Symposium on Applied Computing*, **1** (2000), 440-445.

[6] Savelsbergh M., and Volgenant A., "Edge Exchanges in Degree Constrained Minimum Spanning Tree Problem, *Computers and Operations Research*, **12** (1985), 341-348.

[7] Thang N. Bai, and Catherine M. Zrncic, *An Ant Based Algorithm for Finding Degree-Constrained Minimum Spanning Tree*, GECCO'06, (2006), 11-18.

[8] Volgenant A, *A Lagrangean Approach to the Degree Constrained Minimum Spanning Tree Problem, European Journal of Operational Research*, **39** (1989), 325-331.

[9] Wamiliana, Solving the Degree Constrained Minimum Spanning Tree Problem using TABU and Modified Penalty Search Methods, *Jurnal Teknik Industri Th.*, 6 (1), (2004).

R.Anitha[1] & K. Sangavai[2]
Department of Mathematics and Computer Applications
PSG College of Technology, Coimbatore, Tamil Nadu, India.
E-mails: [1] anitha_nadarajan@mail.psgtech.ac.in & [2] sangavai_k@yahoo.com