Journal of Physics: Conference Series

PAPER • OPEN ACCESS

WAC4 algorithm to solve the multiperiod degree constrained minimum spanning tree problem

To cite this article: Wamiliana *et al* 2020 *J. Phys.: Conf. Ser.* **1524** 012046

WAC4 algorithm to solve the multiperiod degree constrained minimum spanning tree problem

Wamiliana, A Junaidi, Amanto, M Usman, and Warsono

Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Lampung

Corresponding author: akmal.junaidi@fmipa.unila.ac.id

Abstract. The Multiperiod Degree Constrained Minimum Spanning Tree (MPDCMST) is a problem of finding the smallest weight spanning tree while also maintaining the degree restriction in every vertex and satisfying the vertex installation requirement in every period. This problem arises in the networks installation problem where the degree restriction represents the reliability of each vertex and the vertex installation requirement represents the priority vertices that must be installed in the network on a certain period. The installation is divided into some periods because some conditions occur such as harsh weather, fund limitation, etc. In this paper, we propose a WAC4 Algorithm to solve the MPDCMST problem. The performance of the algorithm will be compared to the WAC1 algorithm already in the literature.

1. Introduction

The Multiperiod Degree Constrained Minimum Spanning Tree (MPDCMST) is a problem that arises from the Degree Constrained Minimum Spanning Tree (DCMST) by introducing a period for vertex installation/connection. Some examples of this problem such as the installation process of electrical networks, telecommunication networks, computer networks, freshwater pipe networks, and so on. However, the process of the installation itself should be performed in some periods due to some restrictions, for example, fund limitation, harsh weather, etc. Therefore, the arrangement of periods is a must.

In 2002, Kawatra investigated this problem and solved it using a hybrid of Lagrangean Relaxation and Branch Exchange and implemented the method using problems with orders ranging from 40 to 100. In the implementation, Kawatra uses 10 years planning horizon and directed graphs [1]. In [2-8] some algorithms had been developed and implemented to solve the problem where the graphs are undirected graphs, using a one-year planning horizon and three periods. The undirected graph used is of order 10 to 100 with an increment of 10. By enhancing the algorithm developed by [1], [9] implemented using three benchmark problems taken from TSPLIB.

In this research, we propose a WAC4 Algorithm based on Prim's algorithm by modifying the WAC1 Algorithm so that it can handle the degree restriction, priority and period requirements on every vertex. Moreover, the probability factor is assigned for every vertex in the set of vertices that must be installed in a certain period. The later condition applied is the novelty of this research. We organize the paper as follows: after Introduction is given in Section 1, in Section 2 a brief review of the literature will be provided. The Algorithm proposed and data implementation will be described in Section 3. In Section 4 the Results and Analysis will be discussed, followed by Conclusion in Section 5.

Journal of Physics: Conference Series **1524** (2020) 012046 doi:10.1088/1742-6596/1524/1/012046

2. Literature review

Many network design problems are used Minimum Spanning Tree as the backbone of the problem, including The Multi-period Degree Constrained Minimum Spanning Tree (MPDCMST). This problem arises due to some constraints that occur such as the limitation of the fund, the geographical or local obstacles, the prediction of harsh weather, and so on so that the installation process must be arranged into some periods. This problem is an enhanced problem of the famous Degree Constrained Minimum Spanning Tree Problem. The DCMST problem already highly investigated, and some methods including exact and heuristics algorithms had been proposed, some of those are named such as Branch and Cut [10], Simulated Annealing [11], Iterative Refinement [12], Tabu Search [13-16] , Genetic Algorithm [17].

For solving MPDCMST many algorithms proposed, some to be named are: the algorithm that use the hybrid of Lagrange Relaxation with Branch Exchange [1], Greedy based algorithm [2, 9], algorithms based on Kruskal algorithm and Prim algorithm [5], algorithms based on hybrid of Kruskal algorithm and Depth First Search (DFS) Technique [3,4,6]. Analysis comparative of some algorithms is given in [5] and the detail of the hybrid of modified Kruskal and DFS is given in [6]. The WAC1, WAC2, and WAC3 algorithms which were based on the Modified Prim algorithm are proposed in [7], and in [8] the details of why the performance of the algorithm is influenced or affected by period and flexibility of vertex installation is given.

3. The Algorithm and data for implementation

The WAC4 Algorithm is an algorithm developed based on the WAC1 Algorithm. The main difference between WAC1 and WAC4 algorithms lies in the installation of the set of vertices in $\mathrm{HVT_i}$. In the WAC1 Algorithm, the priority vertices are installed based on the list order, while in the WAC4 Algorithm the installation of vertices in the set of $\mathrm{HVT_i}$ is determined by p and q factors. p is the ratio of the remaining vertices in the set $\mathrm{HVT_i}$ with the different of the remaining maximum vertices that can be installed/connected on that period ($\mathrm{MAXVT_i}$) with the remaining vertices in $\mathrm{HVT_i}$, in another word,

$$\mathrm{p} = \frac{|\mathrm{HVT_i}|_{Temp}}{[\mathrm{MAXVT_i}]_{Temp} - |\mathrm{HVT_i}|_{Temp}}.$$

The algorithm starts by setting vertex 1 as the central vertex and put it in T. Then, the algorithm counts the p-value for every vertex in $\mathrm{HVT_i}$ for i=1. Next, the random number q , $0 < q < 1$ is generated. If $q > p$, then the algorithm starts searching the nearest edges (maybe not in the set $\mathrm{HVT_i}$) that connected with the vertices in T as long as the connection of the edges neither violates the degree constrained nor constitutes a cycle. If $q \leq p$ then the algorithm starts searching the nearest edges that connect the vertices in $\mathrm{HVT_i}$ as long as the connection of the edges neither violates the degree constrained nor constitutes a cycle. Continuing this procedure until the maximum number of installed/connected vertices in that period (i=1) is reached, and the next period starts. The value of p will guarantee that all vertices in the set of $\mathrm{HVT_i}$ will be connected in the i^{th} period or before, because if the value of denominator of p is equal with numerator then the vertices must be connected/installed. The process in the second and third periods is similar to the process in the first period (except setting vertex 1 as the central vertex) until all vertices in the network have been connected.

The WAC4 Algorithm is implemented using the same data as in [5, 7]. The data consists of 300 problems of complete graphs with orders ranging from 10 to 100 with an increment of 10. For every vertex order, 30 problems are generated. We set $\mathrm{MAXVT_i} = \left\lceil \frac{n-1}{3} \right\rceil$, and the number of periods i = 3. For the set of $\mathrm{HVT_i}$, we use the same set as used by [5-7] as the following table 1:

Table 1. Elements of HVT_i for every period.

n	HVT_1	HVT_2	HVT_3
10	2	3	4
20	2	3	4
30	2,3	4,5	6,7
40	2,3,4	5,6,7	8,9,10
50	2,3,4,5	6,7,8,9	10,11,12,13
60	2,3,4,5,6	7,8,9,10,11	12,13,14,15
70	2,3,4,5,6,7	8,9,10,11,12,13	14,15,16,17,18,19
80	2,3,4,5,6,7,8	9,10,11,12,13,14,15	16,17,18,19,20,21,22
90	2,3,4,5,6,7,8	9,10,11,12,13,14,15	16,17,18,19,20,21,22
100	2,3,4,5,6,7,8,9	10,11,12,13,14,15,16,17	18,19,20,21,22,23,24,25

4. Results and analysis.

Table 2 below provides the solutions for every vertex order.

Table 2. The Solutions of WAC1 and WAC4 algorithms for different vertex order

No	Vertex order	MST	DCMST	WAC1	WAC4	$\frac{\mathbf{WAC1-DCMST}}{\mathbf{DCMST}}$	$\frac{\mathbf{WAC4-DCMST}}{\mathbf{DCMST}}$
1	10	1129.433	1178.8	1495.1	1462.084	0.268323719	0.24031595
2	20	1196.1	1299	1790.367	1663.552	0.378265332	0.28064067
3	30	1177.433	1319.533	2018.9	1819.442	0.53001061	0.378852788
4	40	1151.233	1286.3	2079.733	1666.271	0.616833813	0.295398509
5	50	1223.433	1356.467	2381	1792.12	0.755295621	0.321167745
6	60	1175.567	1320.733	2364.4	1752.181	0.790217556	0.326672889
7	70	1242.1	1410.033	2520.2	1784.9	0.787333633	0.265856599
8	80	1236.833	1410.233	2547.8	1789.909	0.806651382	0.269228892
9	90	1248	1404.933	2588.067	1712.158	0.84212774	0.21867546
10	100	1234.1	1370.8	2535.2	1686.173	0.849430989	0.230065174

We compare the WAC4 algorithm to the WAC1 Algorithm. As the order of the graph increases, the gap between WAC1 Algorithms and it's lower bound monotonically increases, from 26,78% for the graph of order 10 to around 85% for the graph of order 100, while for the WAC4 algorithm the gap stays between 23% to around 38%. For the WAC4 Algorithm, the smallest gap occurs when the order of the graph is 100 and the largest gap occurs when the order of the graph is 30. For all orders, WAC4 performs better than WAC1, and this result also can be seen graphically from Figure 1 which provides the comparative analysis of the solution for WAC1 and WAC4 algorithms. Figure 2 describes the percentage of the ratio between the algorithm and the lower bound (DCMST), where the percentage of $\frac{\mathrm{WAC1-DCMST}}{\mathrm{DCMST}}$ reaches its largest value at the maximal order, while the percentage of $\frac{\mathrm{WAC4-DCMST}}{\mathrm{DCMST}}$ reaches its largest value at order 30.

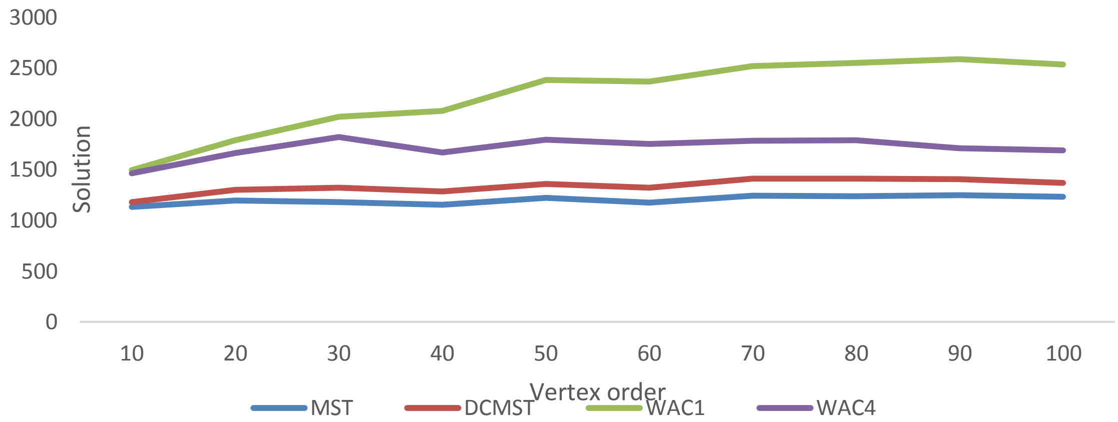

Figure 1. Comparative solutions of WAC1 and WAC4 Algorithms for solving the MPDCMST

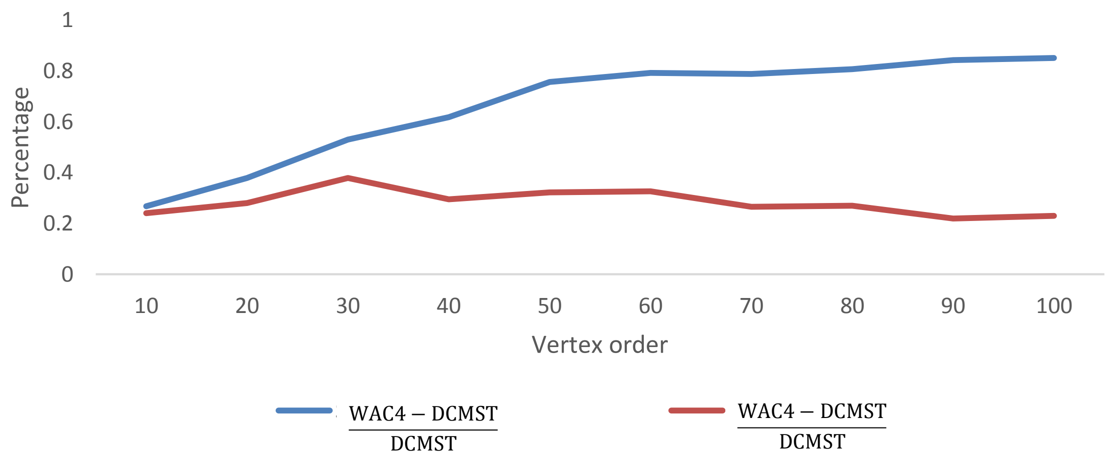

Figure 2. The percentage of the ratio for the WAC1 and WC4 algorithms with the lower bound

5. Conclusion

Based on the result above we can conclude that introducing the probability factor for the vertices in the set HVT_i will improve the quality of the solution. Therefore, if a certain network requires an arrangement of periods for installation of all its components, postponing the process of connection/installation of the priority components until the end of the period probably will ensure the better solution (lower cost).

Acknowledgment

This research is fund by Universitas Lampung under contract number 2843/UN26.21/PN/2019, and the authors want to thanks for the contribution of the research.

References

[1] Kawatra T A 2002 *European Journal of Operational Research* Vol **143** pp. 53–63
[2] Wamiliana, D Sakethi, A Junaidi and E T Baskoro 2005 *J. Sains Tek* Vol. 11 No.2 pp. 93 -96
[3] Wamiliana, D Sakethi and R Yuniarti 2010 *Proceeding SNMAP* p. 208–214

[4] Wamiliana, Amanto and M Usman 2013 *Proceeding The International Conference on Engineering and Technology Development* (ICETD) pp. 39–43
[5] Wamiliana, Elfaki F A M, Usman M and Azram M 2015 *ARPN Journal of Engineering and Applied Sciences* Vol. **10** (**21**) pp.10147–10152
[6] Wamiliana, Usman M, Sakethi D, Yuniarti R and Cucus A 2015 *The 4th International Conference on Interactive Digital Media (ICIDM)* IEEE Explore
[7] Wamiliana, Asmiati, M Usman, A Hijriani and W C Hastono 2018a *Indian Journal of Science and Technology* Vol.**11** No. 11
[8] Wamiliana, Warsono, Asmiati, A Hijriani and Hastono W C 2018b *Far East Journal of Electronics and Communications* Vol. **18** No. 2 pp. 291-300
[9] Junaidi A, Wamiliana, Sakethi D and Baskoro E T 2008 *Jurnal Sains MIPA* Special Edition pp. 1- 5
[10] Caccetta L and S P Hill 2001 *Networks* vol. **37** pp.74-83
[11] Krishnamoorthy M, Ernst A T and Yazid M Sharaila 2001 *Journal of Heuristics* vol. **7** no. 6 pp. 587-611
[12] Deo N and Nishit Kumar 1997 *Computation of constrained spanning trees: a unified approach, Network Optimization, Lecture Notes in Economics and Mathematical Systems*, Panos M. Pardalos, Donald W. Hearn and William W. Hager, eds.,Springer-Verlag, Berlin, Germany, pp. 194-220
[13] Wamiliana and L Caccetta 2003 *Proceeding of South East Asia Mathematical Society* p. 133-140
[14] Wamiliana 2004 *Jurnal Teknik Industri* Universitas Petra Surabaya Vol. 6 No. 1 pp. 1-9
[15] Wamiliana and L Caccetta 2012 *Proceeding of International Conference on Engineering and Technology Development* p.36-39
[16] Caccetta L and Wamiliana 2001 *Proceeding of the International Congress on Modelling and Simulation (MODSIM)*,Canberra, Editors : F. Ghassemi, D. Post, M. Sivapalan and R. Vertessy p. 2161 2166
[17] G Zhou and M Gen 1997 *Networks* Vol. **30** p.91– 95