COMPARATIVE ANALYSIS OF SOME MODIFIED PRIM'S ALGORITHMS FOR SOLVING THE MULTIPERIOD DEGREE CONSTRAINED MINIMUM SPANNING TREE PROBLEM

By Wamiliana Wamiliana

COMPARATIVE ANALYSIS OF SOME MODIFIED PRIM'S ALGORITHMS FOR SOLVING THE MULTIPERIOD DEGREE CONSTRAINED MINIMUM SPANNING TREE PROBLEM

Wamiliana[1], Asmiati[1], Mustofa Usman[1], Astria Hijriani[(2)] and Wibi Cahyo Hastono[(2)]
[1] Department of Mathematics, Faculty of Mathematics and Natural Science, Universitas Lampung, Indonesia
[2] Department of Computer Science, Faculty of Mathematics and Natural Science, Universitas Lampung, Indonesia email: wamiliana.1963@fmipa.unila.ac.id

ABSTRACT

Most in network design problems, The Minimum Spanning Tree (MST) is usually used as the backbone. If we add degree restriction on its vertices (can represent cities, stations, etc) of the graph (represents the network), the problem becomes the Degree Constrained Minimum Spanning Tree (DCMST) problem. Moreover, if we restrict and devide the stages or periods of the network's instalation, the problem emerges as The Multi Period Degree Constrained Minimum Spanning Tree (MPDCMST) problem. The later constraint occurs usually because of the fund limitation for installing (connecting) the network. In this research we propose three algorithms to solve the MPDCMST Problem, and implement those algorithms using 300 generated random table problems. Moreover, we compare our algorithms using those that already in the literature and show that the proposed algorithms perform better.

Keywords: multi periods, degree constrained, Prims' algorithms, comparative analysis

1. INTRODUCTION

There is no doubt that many network design problems usually use graph to represent the network. In network design problem we construct a network that satisfies certain requirements which is optimal according to some criterion. Graph is used to represent the network, where the vertices can represent cities/stations/computers etc. and the edges of the graph can represent roads/links etc. The requirements of the network can be reliability, efficiency, throughput, safety and so on; and the criterion can be cost, output, performance etc.

The minimum spanning tree as one of the fundamental structure in network design has many applications in real-life problem, especially when there are also some prescribed parameters under consideration. To solve a minimum spanning tree problem, Prim's algorithm is one of some of well-known algorithms. If, in addition to the MST we add degree restriction on every vertices, the problem becomes a Degree Constrained Minimum Spanning Tree (DCMST) problem. This problem typically arises in the design of telecommunication, transportation and energy networks. It is concerned with finding a minimum-weight (distance or cost) spanning tree that satisfies specified degree restrictions on its vertices (Wamiliana, 2002). Moreover, if in the network's installation process must be done in some stages or periods due to other restriction such as fund limitation, weather, and so on, the DCMST arises as Multi Period Degree Constrained Minimum Spanning Tree Problem. We will provide the brief review of the problem and the available methods for solving it in the literature review in Section 2. In Section 3 we discuss the proposed algorithm, followed by implementation, result and discussion in Section 4. In Section 5 we give the conclusion.

2. LITERATURE REVIEW

As already stated before, MST as one of fundamental structure has many applications. The MST structure usually is used as the backbone of the problem. There are two well-known and widely used algorithms to solve the MST, Kruskal's algorithms (Kruskal, 1956) and Prim's algorithm (Prim, 1957). Eventhough there are some other algorithms for solving the MST such as Sollin's algorithm or Boruvka, but the previous two are commonly used. The earliest algorithm for finding a minimum spanning tree according to Graham and Hell (1982) was suggested by Boruvka (1926a,b) who developed an algorithm for finding the most economical layout for a power-line network. Minimum spanning trees in general are used in many network optimization problems as the key structure. Since G.R. Kirchoff designed electrical circuits in the 19th century, spanning trees have been considered as one of the most used subgraphs in many network design applications (Wamiliana et al, 2010). Deo and Kumar (1997) reported that spanning trees can be computed in linear time for a given connected weighted graph. When the spanning tree is required to have minimum weight (minimum spanning tree), the computational time only slightly increases.

The Degree Constrained Minimum Spanning Tree Problem is a problem that concern of finding a Minimum Spanning Tree whilst also has degree restriction on the vertices. Garey and Johnson (1976) showed that the problem is NP complete by reducing the degree on every vertices exactly two in which the DCMST is exactly a famous and highly investigated problem : the travelling salesman problem. Eventhough not as highly investigated as the travelling salesman problem, the DCMST problem also investigated by many researchers by proposing exact and heuristics algorithms. Since this problem is NP complete, heuristic methods have dominated. Some of the heuristics that have been investigated include: a number of basic MST algorithms of Prim and Kruskal (Narula and Ho (1980); the genetic Algorithm by Zhou and Gen (1997); Simulated Annealing by Krishnamoorthy et al. (2001); and Iterative Refinement by Boldon et al. (1996) Deo and Kumar (1997); Tabu Search by Caccetta and Wamiliana (2001), Wamiliana and Caccetta (2003), Wamiliana (2012); and Modified Penalty by Wamiliana (2004).

Some of the exact algorithms for solving the DCMST problem already proposed such as the Branch and Bound by Narula and Ho (1980) in which the the branching procedure is an adaptation of the method due to Held and Karp (1970, 1971) for the traveling salesman problem; Savelbergh and Volgenant (1985) used the branch and bound algorithm based on an edge exchange analysis and utilized three heuristics (the primal method of Narula and Ho, a heuristic by Christofides (1976) and a heuristic based on edge exchange); Gavish (1982) and Volgenant (1989) using Lagrangian Relaxation method by introducing Lagrange multiplier π (introducing penalty), and adding them to the objective function; and Caccetta and Hill (2001) developed a branch and cut method for the DCMST problem by generating an upper bound using the heuristics of Savelsbergh and Volgenant (1985), the initial lower bounds are generated at the root node using the lagrangean procedure of Volgenant (1989), and the important features of their method are the used of depth first search procedure.

The Multi Period Degree Constrained Minimum Spanning Tree (MPDCMST) Problem was investigated by Kawatra (2002) using branch exchange technique as a hybrid to Lagrangean relaxation, and the method was implemented using vertices varying from 40 to 100; 10 year planning horizon; the time period for activating each terminal is uniformly distributed from 1 to 6; and set vertex 1 as central vertex. Wamiliana et al (2005) used one year planning horizon and divided the installation into three periods (four month each) and four periods (three month each). That modification of MPDCMST was made to mimic the real situation in Indonesia where the funding for every project usually divided into three terms or periods. By using the same data as Wamiliana et al (2005), Junaidi et al (2008) improved the method developed by Wamiliana et al (2005) and tested also the method using some problems taken from TSPLIB. Wamiliana et al (2010) developed WADR1 and WADR2 algorithms which are the modification of Kruskal's algorithm and in the searching they used DFS technique with k = 2, k is the length of the node path. In the algorithm they introduced a set HVT_i as a set of vertices that must be already in the networks after period i finished. The use of HVT_i is to tackle the problem that some facility (for example hospital, police station, or other public need facilities) must be in the network earlier to handle public needs. The difference between WADR1 and WADR2 lied on the process of installation HVT_i. Wamiliana et al (2013) improved the methods investigated by Wamiliana et al (2010) by setting $HVT_i = 3$, $k \leq 3$, $MaxVT_i = \lfloor \frac{n-1}{3} \rfloor$. Wamiliana et al (2015a) proposed three algorithms : two were based on Kruskal's and one were based on Prim's. The WADR3 and WADR4 were the two algorithm based on Kruskal's by modifying WADR1 and WADR2 by relaxing the HVT_i and introducing the best k-path, with k = 3. WADR5 is the algorithm developed based on Prim.
In the same year, Wamiliana et al (2015b) developed two algorithms by adopting the best 2-path, and gave the detail why the different solution occurs when the of algorithms developed in Wamiliana et al (2010, 2013) implemented with different HVT_i and provided the illustration.

3. THE ALGORITHMS AND DATA FOR IMPLEMENTATION

We propose two algorithms based on Prim's algorithm which shown on the following pseudocode:

```
Initiation:V={1},T=0,n=number_of_vertex,k=1, kMax=3
   begin
      while k < kMax
       do
       if |HVT_k| > MaxVT
        stop
        else
          T_k = 0
           while T_k < (|HVT_k|-1)
             do
                 find the shortest edge which connects with vertices in V
              store in T
              if the connecting vertex not include in HVT_k
```

```
go to the next edge
else
  if adding an edge constitute circuit
    choose the next edge
    else
      if adding an edge violate degree restriction
       choose the next edge
      else
      store the edge in T and the vertex incident to it in V
     Tk++
      endif
   endif
 endif
end
while Tk < (MaxVTk-1)
   do
   find the shortest edge which connects with vertices in V
   store in T
    if adding an edge constitute circuit
    choose the next edge
    else
      if adding an edge violate degree restriction
      choose the next edge
      else
      store the edge in T and the vertex incident to it in V
      Tk++
     endif
   endif
 end
k++;
endif
endwhile
end
```

HVT_k is the set of vertices that must be installed/connected on k^{th} period , $maxVT_k$ is the maximum number of vertices that can be installed/connected on k^{th} period, kMax is the number of period for installation, T_k = the number of edges that already installed /connected on k^{th} period , k is the current period.

The coding process is divided into four main stages as shown on the following flowchart :

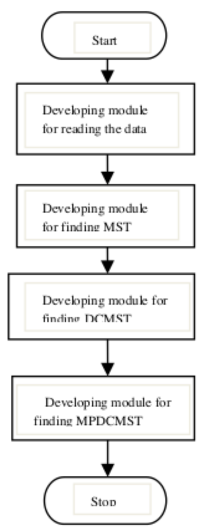


Fig. 1 Four main stages of the process

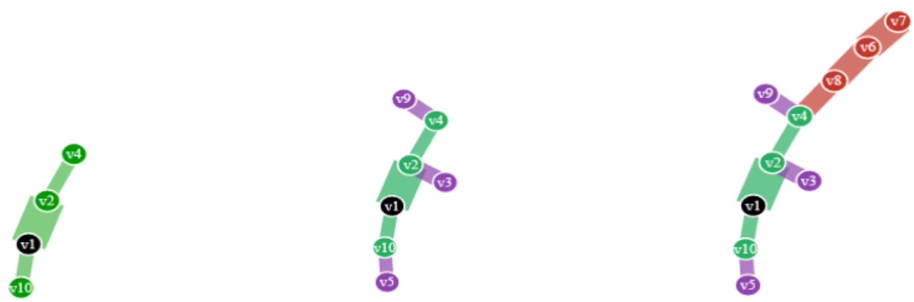
The module for reading the data is developed to read the data from the source. The data used is the same as the data used by Wamiliana et al (2015a, 2015b). One problem on the data represents a complete graph with specific order. The order of graphs are 10 to 100 with increment of 10 and for every vertex order there are 30 problems for simulation. Therefore, there are total 300 problems to be tested. The module for finding MST is designed to solve the MST of the problem using Prim's algorithm. The reason for applying Prim's algorithm instead of Kruskal'a algorithm because Prim's algorithm maintains the conectivity of the network during installation proccesses. The module for finding DCMST is the improvement of module for finding MST with the restriction on every vertices. Here, we add a degree rectriction on every vertex by setting $d_i \leq 3$, where d_i is the degree of vertex i. Finally, the module for finding MPDCMST is the improvement of the module for finding DCMST by adding number of periods for installing the network, and vertex priority to be installed on a certain period.

We developed three algorithms based on modified Prim's algorithm. The first algorithm (WAAC1) is the simplest one and exactly the same as WADR5 in Wamiliana et al (2015a). The algorithm just follows the original Prim's algorithm. The modification made by adding the degree restriction on edge insertion processes and checking the element on HVT_k on every period. In this algorithm the vertices on HVT_k are given priority to be connected/installed as early as possible. We give the illustration of the algorithm as follow:

Table 1. Data file 22.dat (10 vertices)

Edge	e_{12}	e_{13}	e_{14}	e_{15}	e_{16}	e_{17}	e_{18}	e_{19}	$e_{1,10}$	E_{23}	e_{24}	e_{25}	e_{26}	e_{27}	e_{28}
Weight	740	572	447	835	427	807	362	832	120	221	109	276	741	987	352
Edge	e_{29}	$E_{2,10}$	E_{34}	E_{35}	E_{36}	E_{37}	E_{38}	E_{39}	$E_{3,10}$	E_{45}	e_{46}	E_{47}	E_{48}	E_{49}	$E_{4,10}$
Weight	368	403	505	921	757	884	369	886	545	639	253	750	251	187	857
Edge	E_{56}	E_{57}	E_{58}	E_{59}	$E_{5,10}$	E_{67}	E_{68}	E_{69}	$E_{6,10}$	E_{78}	E_{79}	$E_{7,10}$	E_{89}	$E_{8,10}$	$E_{9,10}$
Weight	807	926	781	605	112	559	411	473	743	882	693	851	509	434	828

For n = 10, by setting v_1 as the root, $V=\{v_1\}$, and using k = 1,2,3 and $HVT_1 = \{2\}$, $HVT_2=\{3\}$, $HVT_3 = \{4\}$ we get the following figures for every period installation.



The first period The second period The third period

Fig.2 Stage of instalation for every period of WAAC1 Algorithm

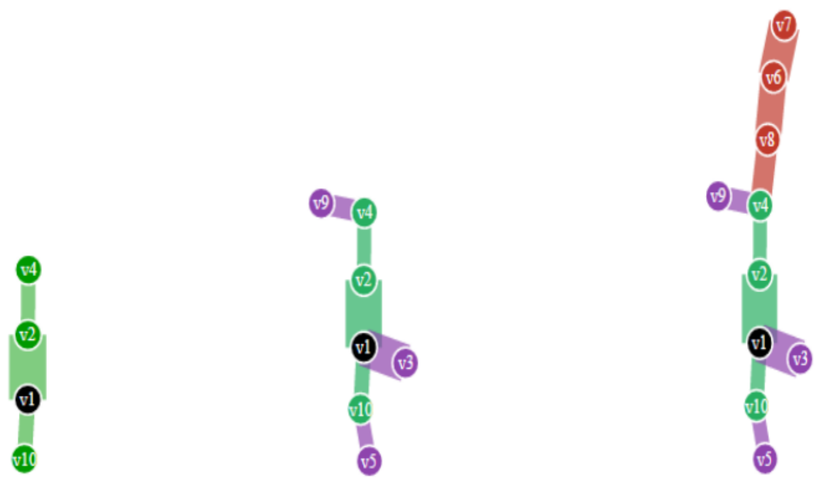
Node that the box between every pair of vertices represent the distance/cost/weight. For instance, the weight of edge e_{24} (weight from v2 to v4) is smaller than e_{12}.

For the second algorithm (WAAC2), the vertices on HVT_k are not given priority to be connected as soon as possible, but can be any time as long as the connection still on that certain period. The following figures give the illustration.

The first period The second period The third period

Fig.3 Stage of instalation for every period of WAAC2 Algorithm

For the third algorithm (WAAC3) we adopt the technique used by Wamiliana et al (2015b) by applying the smallest value for 2-path. The following figures show the illustration of the algorithm.



The first period The second period The third period

Fig.4 Stage of instalation for every period of WAAC3 Algorithm

4. IMPLEMENTATIONS, RESULTS AND DISCUSSION

We implemented our heuristic using the C++ programming language running on dual core computer, with 1.83 Ghz and 2 GB RAM. We used the same elements on HVT_k as used by Wamiliana et al (2015) as the following table:

Table 1. The list of vertices in HVT_i , i = 1,2,3

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}

We compare our algorithms with algorithms developed by Wamiliana et al (2015a, 2015b) and we get the following solutions

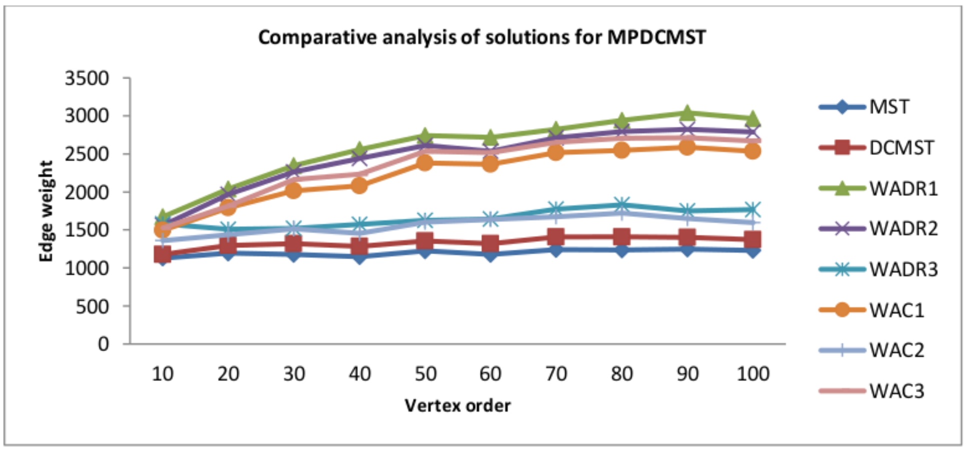


Fig. 5. Comparative analysis of some heuristics for the MPDCMST.

5. CONCLUSION

From the chart on Figure 4 we see that the average solutions of WAAC1 and WAAC3 heuristics are better than WADR1 and WADR 2 of Wamiliana et al (2010). The heuristics that performs better than WAAC1 and WAAC3 are WADR3 and WAAC2. But, WADR3 is an algorithm developed based on Kruskal's algorithm in which during installation process , disconnectivity of the networks is permissible, whilein WAAC1, WAAC2, and WAAC3 the connectivity is maintained. The result shows that on this comparison, the WAAC2 heuristics is the closest to the average solutions of DCMST which is the lower bound of MPDCMST, and therefore WAAC2 performs the best.

ACKNOWLEDGMENT

This research was supported by The Directorate General of Higher Education, Ministry of Research Technology and Higher Education of The Republic of Indonesia under contract # 382/UN26.21/KU/2017. The authors wish to thank for the support

REFERENCES

[1] Boldon, B., N. Deo and Nishit Kumar ,1996. ' Minimum Weight degree-constrained spanning tree problem: Heuristics and Implementation on an SIMD parallel machine', *Parallel Computing* vol. 22, pp.369 –382.

[2] Boruvka, O., 1926 a. ' O Jistém Problému Minimálním '_*Práce Mor. Prírodoved.Spol. v Brne (Acta Soc.Sci Nat. Moravicea)*, **3** , pp. 37-58.

[3] Boruvka, O. ,1926 b. '*Príspevek k. Resení Otázky Ekonomické Stavby Sítí', Electrotech . Obzor*, **15**, pp. 153-154.

[4] Caccetta,L. and Hill, S.P, 2001.' A Branch and Cut method for the Degree Constrained minimum Spanning Tree Problem', *Networks*, vol. 37, pp.74-83.

[5] Caccetta, L., and Wamiliana, Heuristics Algorithms for the Degree Constrained Minimum Spanning Tree Problems, 2001. Proceeding of the International Congress on Modelling and Simulation (MODSIM 2001), Canberra, Editors: F. Ghassemi et.al, pp. 2161-2166,.

[6] Christofides, N. ,1976. *Worst-case Analysis of a New Heuristic for the Travelling Salesman Problem*, Carnegy-Mellon University, Pittsburg, USA.

[7] Deo N. and Nishit Kumar, 1997. 'Computation of Constrained Spanning Trees: A Unified Approach', *Network Optimization* (Lecture Notes in Economics and Mathematical Systems, Editor : Panos M. Pardalos, et al , Springer-Verlag, Berlin, Germany, pp. 194 – 220.

[8] Garey, M.R. , and Johnson, D.S., 1979. *Computers and Intractibility, A Guide to the Theory of NP-Completeness*, Freemann, San Francisco.

[9] Gavish, B.,1982. 'Topological Design of Centralized Computer Networks Formulations and Algorithms ', *Networks*, vol. 12 , pp.355 -377.

[10] Graham, R.L., and Hell, P., 1982. ' On the history of the Minimum Spanning Tree Problem', Mimeographed , Bell Laboratories, Murray Hill, New Jersey.

[11] Held, M. and R.M. Karp, 1970. ' The travelling salesman problem and minimum spanning trees', *Operation Research*, vol. 18, pp.1138-1162

[12] Held, M. and R.M. Karp, 1971.' The traveling salesman problem and minimum spanning trees ', Part II, *Mathematical Programming*, vol. 1, pp.6-25

[13] Junaidi, A.., Wamiliana, Dwi Sakethi, and Edy Tri Baskoro, 2008. Computational Aspect of Greedy Algorithm for The Multi Period Degree Constrained Minimum Spanning Tree Problem, *Jurnal Sains MIPA*, Special Edition Vol. 14 No. 1. Pp 1-6

[14] Kawatra, 2002.A multi period degree constrained Minimum Spanning Tree Problem, *European Journal of Operational Research*, Vol **143**, pp. 53 – 63.

[15] Krishnamoorthy, M., A.T. Ernst and Yazid M Sharaila ,2001. 'Comparison of Algorithms for the Degree Constrained Minimum Spanning Tree', *Journal of Heuristics*, vol. 7, no. 6, pp. 587-611.

[16] Kruskal, J.B.,1956. 'On the Shortest Spanning Tree of a Graph and the Travelling Salesman Problem', *Proc. Amer Math .Soc*., vol. 7, pp. 48-50.

[17] Narula,S. C., and Cesar A.Ho.,1980 ' Degree-Constrained Minimum Spanning Tree', *Computer and Operation Research* , vol. 7,pp. 239-249.

[18] Prim, R.C. ,1957. ' Shortest Connection Networks and Some Generalizations', *Bell Syst. Tech Journal*, vol. 36, pp 1389-1401.

[19] Savelsbergh,M., and Ton Volgenant., 1985. 'Edge Exchange in The Degree-Constrained Minimum Spanning Tree', *Computer and Operation Research*, Vol. 12, pp. 341-348.

[20] Volgenant, A.,1989. 'A Lagrangean Approach to The Degree-Constrained Minimum Spanning Tree Problem', *European Journal of Operational Research,* vol. 39, pp.325- 331

[21] Wamiliana. 2002 "Combinatorial Methods for Degree Constrained Minimum Spanning Tree Problem", Doctoral Thesis, Department of Mathematics and Statistics, Curtin University and Technology, Australia.

[22] Wamiliana and Caccetta, 2003. Tabu search Based Heuristics for the Degree Constrained Minimum Spanning Tree Problem,Proceeding of South East Asia Mathematical Society, pp. 133-140

[23] Wamiliana, 2004. Solving the Degree Constrained Minimum Spanning Tree Using Tabu and Penalty Method', *Jurnal Teknik Industri,* pp.1-9.

[24] Wamiliana, Dwi Sakethi, Akmal J, and Edy Tri Baskoro, 2005. The Design of Greedy Algorithm for Solving The Multi Period degree Constrained Minimum Spanning Tree Problem, *Jurnal Sains dan Teknologi* **Vol 11** No. 2, pp. 93 – 96.

[25] Wamiliana, Dwi Sakethi, and Restu Yuniarti, 2010. Computational Aspect of WADR1 and WADR2 Algorithms for The Multi Period Degree Constrained Minimum Spanning Tree Problem, Proceeding SNMAP, Bandarlampung 8 – 9 December 2010, pp. 208 – 214.

[26] Wamiliana 2012. The Modified CW1 Algorithm for The Degree Restricted Minimum Spanning Tree Problem. Proceeding of International Conference on Engineering and Technology Development, Bandarlampupg 20-21 June, 2012, pp. 36-39.

[27] Wamiliana, Amanto, and Mustofa Usman. "Comparative Analysis for The Multi Period Degree Constrained Minimum Spanning Tree Problem, " in Proc. International Conference on Engineering and Technology Development (ICETD), 2013, pp. 39 – 43, 2013.

[28] Wamiliana, Faiz A.M. Elfaki, Mustofa Usman, and M. Azram. 2015a. Some Greedy Based Algorithms for Multi Periods Degree Constrained Minimum Spanning Tree Problem. *ARPN Journal of Engineering and Applied Sciences,* Vol 10 No. 21, pp.10147 – 10152.

[29] Wamiliana, Mustofa Usman, Dwi Sakethi, Restu Yuniarti, and Ahmad Cucus. 2015b.The Hybrid of Depth First Search Technique and Kruskal's Algorithm for Solving The Multiperiod Degree Constrained Minimum Spanning Tree Problem. 2015 4th International Conference on Interactive Digital Media (ICIDM). IEEE Explore, Dec 2015.
http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7516333&newsearch=true&queryText=wamiliana

[29] Zhou, G., and Mitsuo Gen.(1997), ' A Note on Genetics Algorithms for Degree- Constrained Spanning Tree Problems', *Networks,* vol. 30, pp.91 - 95.

COMPARATIVE ANALYSIS OF SOME MODIFIED PRIM'S ALGORITHMS FOR SOLVING THE MULTIPERIOD DEGREE CONSTRAINED MINIMUM SPANNING TREE PROBLEM

ORIGINALITY REPORT

19%

SIMILARITY INDEX

PRIMARY SOURCES

1	Wamiliana, Asmiati, Mustofa Usman, Astria Hijriani, Wibi Cahyo Hastono. "Comparative analysis of some modified prim�s algorithms to solve the multiperiod degree constrained minimum spanning tree problem", Indian Journal of Science and Technology, 2018 Crossref	78 words — 3%
2	ijamc.darbose.in Internet	59 words — 2%
3	L. Caccetta. "A branch and cut method for the degree-constrained minimum spanning tree problem", Networks, 03/2001 Crossref	37 words — 1%
4	Subhash C. Narula, Cesar A. Ho. "Degree-constrained minimum spanning tree", Computers & Operations Research, 1980 Crossref	25 words — 1%
5	www.ijicc.net Internet	22 words — 1%
6	algorithm-wiki.org Internet	22 words — 1%
7	Volgenant, A.. "A Lagrangean approach to the degree-constrained minimum spanning tree problem", European Journal of Operational Research, 19890414 Crossref	19 words — 1%

8	Wamiliana, A Nuryaman, Amanto, A Sutrisno, N A Prayoga. "Determining the Number of Connected Vertices Labelled Graph of Order Five with Maximum Number of Parallel Edges is Five and Containing No Loops", Journal of Physics: Conference Series, 2019 Crossref	17 words — 1%
9	repository.lppm.unila.ac.id Internet	16 words — 1%
10	Encyclopedia of Algorithms, 2016. Crossref	13 words — < 1%
11	journal.ubl.ac.id Internet	13 words — < 1%
12	L. Caccetta, S.P. Hill. "Branch and cut methods for network optimization", Mathematical and Computer Modelling, 2001 Crossref	13 words — < 1%
13	D. Silver. "Volume animation using the skeleton tree", IEEE Symposium on Volume Visualization (Cat No 989EX300) SVV-98, 1998 Crossref	12 words — < 1%
14	epdf.pub Internet	11 words — < 1%
15	Minhyeop Kim, In-Young Ko. "An Efficient Resource Allocation Approach Based on a Genetic Algorithm for Composite Services in IoT Environments", 2015 IEEE International Conference on Web Services, 2015 Crossref	11 words — < 1%
16	B. Srivastava, Wun-Hwa Chen. "Heuristic solutions for loading in flexible manufacturing systems", IEEE Transactions on Robotics and Automation, 1996 Crossref	11 words — < 1%
17	www.mdpi.com Internet	10 words — < 1%

18 www.journal.uhamka.ac.id
Internet
10 words — < 1%

19 Hsinghua Chou, G. Premkumar, Chao-Hsien Chu. "Genetic algorithms for communications network design - an empirical study of the factors that influence performance", IEEE Transactions on Evolutionary Computation, 2001
Crossref
10 words — < 1%

20 S. Bracchi, M. Rota, A. Penna, G. Magenes. "Consideration of modelling uncertainties in the seismic assessment of masonry buildings by equivalent-frame approach", Bulletin of Earthquake Engineering, 2015
Crossref
9 words — < 1%

21 calhoun.nps.edu
Internet
9 words — < 1%

22 Tiago L. Pereira, Eduardo G. Carrano, Ricardo H. C. Takahashi, Elizabeth F. Wanner, Oriane M. Neto. "Continuous-space embedding genetic algorithm applied to the Degree Constrained Minimum Spanning Tree Problem", 2009 IEEE Congress on Evolutionary Computation, 2009
Crossref
9 words — < 1%

23 Gengui Zhou, Zhiqing Meng, Zhenyu Cao, Jian Cao. "A New Tree Encoding for the Degree-Constrained Spanning Tree Problem", 2007 International Conference on Computational Intelligence and Security (CIS 2007), 2007
Crossref
9 words — < 1%

24 Salgueiro, Rui, Ana de Almeida, and Orlando Oliveira. "New genetic algorithm approach for the min-degree constrained minimum spanning tree", European Journal of Operational Research, 2016.
Crossref
8 words — < 1%

25 link.springer.com
Internet
8 words — < 1%

R. Kawatra, D. Bricker. "A multiperiod planning model for the

26	capacitated minimal spanning tree problem", European Journal of Operational Research, 2000 Crossref	8 words — < 1%
27	Andrade, R.. "Using Lagrangian dual information to generate degree constrained spanning trees", Discrete Applied Mathematics, 20060401 Crossref	8 words — < 1%
28	www.mpi-inf.mpg.de Internet	8 words — < 1%
29	mhbateni.com Internet	8 words — < 1%
30	Lecture Notes in Computer Science, 2011. Crossref	6 words — < 1%

EXCLUDE QUOTES	OFF	EXCLUDE MATCHES	OFF
EXCLUDE BIBLIOGRAPHY	ON		