A New Heuristic Method Improvement for Ring Topology Optimization: Proposal Algorithm

Syamsul Qamar, Sigit Haryadi, and Nana R. Syambas
School of Electrical Engineering and Informatics, Institut Teknologi Bandung (ITB)
Bandung 40132, Indonesia
Email: syamsul27@students.itb.ac.id; sigitharyadi59@gmail.com; nana@stei.itb.ac.id

Abstract —The issue of optimization is very important in designing computer network. Designing a ring topology is about connecting a set of point-shaped nodes which are each connected to two other points, thus forming a circular path forming a ring. This idea tries to formulate an algorithm to optimize the ring topology by using an approach that is on the Traveling Salesman Problem (TSP). There are two variables considered in TSP. Both are the strategy to find the node configuration which has minimum cost as well as the most minimum time consumption for its execution. To measure the extent to which the proposed algorithm can perform optimization, then made a comparison with some existing algorithms such as Brute force, Ant colony, and Bambang. Brute Force has the ability to find the absolute minimum node configuration cost but in terms of time to complete it increases exponentially. While it does not provide the minimum cost, the Ant Colony Optimization algorithm (ACO) has the minimal advantage needed to find the minimum node configuration. This paper proposes a heuristic algorithm to find the sub-minimum node configuration. The proposed algorithm has the shortest time of fewer than 50 seconds to 50 nodes, compared to other algorithms while the cost of node configuration is not always lower than Ant Colony.

Index Terms—TSP, heuristic, ring toplogy, network design, routing algorithm, ant colony, brute force

I. Introduction

Over the last few decades, we have witnessed a growth in data traffic. This growth, driven by Internet proliferation, has created an increasing demand for robust networks, with increased link and node capacity. In metropolitan networks ring network topology is the most popular topology which uses the link bandwidth efficiently and increases the capacity of the system. However, there is a need for an efficient solution for transporting and switching huge amounts of data at the boundaries of backbone networks, especially at metropolitan and local area networks [1].

Design ring topology itself can be modelled like the Traveling Salesman Problem (TSP), which is a problem of optimization to search the shortest route for peddler who wants to visit several cities [2]. The traveling salesman problem, TSP for short, has model character in many branches of Mathematics, Computer Science, and Operations Research. Heuristics, linear programming, and branch and bound, which are still the main components of today's most successful approaches to hard combinatorial optimization problems [3]. The complexity of the TSP problem algorithm is about continuously challenging issue until today, even after 50 years of searching. This makes TSP one of the most untapped issues in many mathematical optimization problems [4].

The proposed algorithm used a purely heuristic method, which is mean using only the available information at that time to make decision to added new node. In this study, we tested the algorithm we proposed to build a ring topology by taking the problems in the TSP and compares with some popular algorithms that are often used to solve this problem among others.

A. Brute Force

Brute force is a straightforward approach to solve a problem based on the problem's statement and definitions of the concepts involved. It is considered as one of the easiest approach to apply and is useful for solving small–size instances of a problem [5]. It is often easy to implement and almost certainly will find a solution. At this time, only the brute force algorithm is able to find the absolute minimum value of the TSP. Brute force algorithm is one of the easiest ways to find the shortest link, but Brute Force takes a very long time of execution. The complexity of the algorithm for TSP problems with the Brute Force algorithm is O (n!).

B. Ant Colony

Ant colony optimization (ACO) was one of the first techniques for approximate optimization that was introduced in the early 1990's. The inspiring source of ant colony optimization is the foraging behavior of real ant colonies [6]. The ant colony optimization (ACO) algorithm, a classical bionic algorithm for determining the optimal path, has several advantages: it is easy to integrate with other algorithms, is amenable to distributed parallel computing, includes an intelligent search, and has good global optimization and strong robustness when compared with other swarm intelligence algorithms [7]. This behavior is exploited in artificial ant colonies for the search of approximate solutions to discrete optimization problems, to continuous optimization problems, and to important problems in telecommunications, such as routing and load balancing.

Manuscript received April 16, 2018; revised September 12, 2018.
This work was supported by LPDP (Lembaga Pengelola Dana Pendidikan), Menistry of Finance, Republic of Indonesia
Corresponding author email: syamsul27@students.itb.ac.id
doi:10.12720/jcm.13.10.607-611

C. Bambang

We call it The Bambang algorithm, taking it from the author's own name because it does not mention the name of the algorithm [8]. Bambang algorithm uses semi brute force approach, which is not all possible combinations considered to be computed so that it can get a shorter time, but the cost is not absolute minimum than brute force because it does not take the entire information before making a complete ring.

II. Related Works

Many researchers to find the best heuristic method to solve TSP. The need to use heuristic method to solve TSP problem, is because the brute force, which acquired all information before choosing the minimum configuration nodes, needed an increasingly exponential computational time for every new node added. For brute force algorithm, the minimal expected time to obtain minimum cost is exponential for every new node added [9].

Some researchers who have been studying TSP propose their algorithm in order to solve the problem such as; "Comparison of proposed algorithm to ant colony optimization algorithm to form a ring topology with minimum path value" in which the proposed algorithm gave almost the same minimum value to brute force, and performed better than ACO, but as the number of node increase so is the time needed to calculate configuration nodes [8], and an Application of Ant Colony Optimization Algorithm in TSP [10].

Other papers make a comparison among several heuristic methods, like Pasquier J. L. *et al* in [11] made a comparison among Genetic Algorithm (GA), Ant Colony Optimization (ACO), and Simulated Annealing (SA). Various aspects are evaluated including processing time, development effort and quality of the solution. Overall, the paper concluded that, based on various aspects used for comparisons, ACO offers the best quality of results.

Based on these researches, it is still found many rooms to find better solutions for the TSP.

III. Proposed Algorithm

In this paper, the algorithm we propose is the development of our previous algorithm [12] which is used to find a sub optimal cost with respect to Brute Force algorithm but with a better time execution.

In the beginning we start by sorting the cost to connect each node to every other node, node i and node j, from the lowest to the highest. The pair of nodes with the lowest cost will serve as the base links. The second step is to find the second lowest pair of nodes that has either node of the base links, so that this new pair of nodes can be connected immediately to the base links without the need of additional node, which means adding other costs. After adding this pair of nodes, the search for the next pair of nodes will start again from the pair of nodes with the lowest cost. The reason for this is because the addition of the new pair of nodes open the possibility of adding other pair of nodes which previously impossible because the absent of the necessary node. These pattern of adding new pair of nodes to the base links is repeated continuously until the base links complete or become a full ring. One thing which need to be mention is, before adding the pair of nodes to the base link, the pair of nodes must be checked so that the nodes in the pair of nodes only contain exactly one same node to the base links and must be at the end or at the beginning of the base links. This is to make sure that the base links only have no repeating node. The flowchart can be seen in Fig. 1.

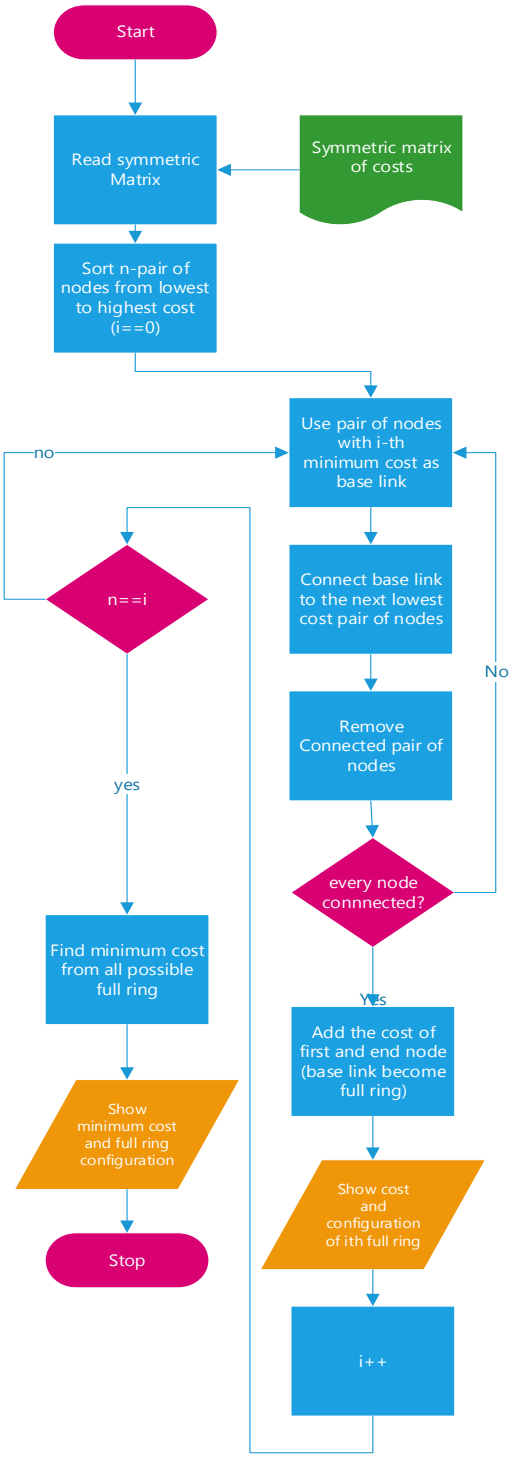

Fig. 1. Flowchart of the proposed algorithm.

This process is repeated again for the second lowest pair of nodes that will be used as the base links all the way to highest cost of pair of nodes. The reason for this is because using the initial lowest pair of nodes as the base links, doesn't necessarily resulted in the lowest cost for the full ring of base link, so for example if there are 4 nodes there will be 6 candidates; if there are 5 nodes, there will be 10 candidates. At the last step, we will compare these candidates and choose the one with the lowest cost.

To summarize the algorithm is as follows:

1. Sort each possible pair of nodes from the lowest cost to the highest cost.
2. Use the lowest cost pair of nodes as the base links
   a. Find the next lowest pair of nodes that has either node of the base links
   b. Start again from the pair of nodes with the lowest cost.
3. Repeat step 2 until base link become a full ring.
4. Repeat step 1-3 for each pair of nodes from the lowest cost to the highest cost.
5. Find the minimum cost from various full ring calculated in step 4.
6. The full ring with the lowest cost is the final full ring.

IV. Simulating and Results

This simulation we used computer with specifications as follows, Intel Core i5-7200U @2.5GHz and memory 8 Gigabyte and the operating system used is Ubuntu 16.04 64 bit. All the algorithm used in the simulation is implemented in python version 2.7.

In the simulation we tested the proposed algorithm performance, cost and the time needed to make a complete ring, for nodes from 4 to 50. The matrix that used for the simulation is the symmetric matrix used by [8]. The example below is the matrix from 11 nodes. (Table I)

Table I: Example Matrix of 11 Nodes

Nodes	0	1	2	3	4	5	6	7	8	9	10
0	0	930	993	1138	218	458	1194	637	234	577	529
1	930	0	1748	857	510	901	1022	785	659	421	130
2	993	1748	0	564	795	901	1076	749	795	612	441
3	1138	857	564	0	768	1460	481	930	49	948	381
4	218	510	795	768	0	1380	583	534	434	516	1133
5	458	901	901	1460	1380	0	946	552	549	658	1030
6	1194	1022	1076	481	583	946	0	754	1372	928	816
7	637	785	749	930	534	552	754	0	1257	856	769
8	234	659	795	49	434	549	1372	1257	0	390	757
9	577	421	612	948	516	658	928	856	390	0	681
10	529	130	441	381	1133	1030	816	769	757	681	0

Beside the proposed algorithm, we also simulated using Brute Force algorithm, Ant Colony Optimization algorithm from [13] and the algorithm from [8]. Just for clarity, in the chart, the proposed algorithm is named "Syam", while other is named as the such.

Based on the graph shown in Fig. 2, it shows the measurement done for several algorithms. Brute force is included in this comparison to describe the most minimum cost that should be reached by others algorithm. However, After node 12, brute force algorithm no longer show the minimum cost. it could not consider all the nodes due to the computation power for the reason later showed.

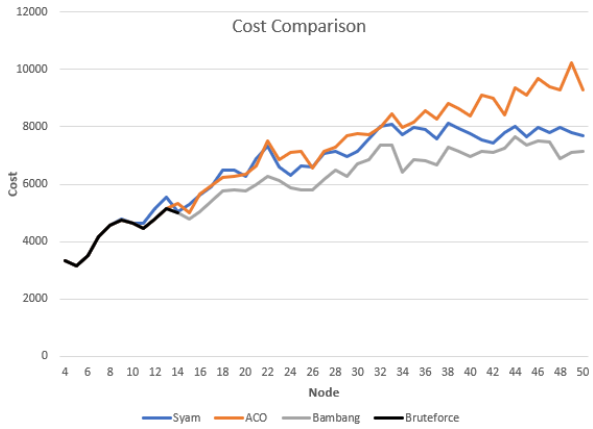

Fig. 2. Cost comparisons between 4 algorithms at various number of nodes.

An algorithm that proposed by Bambang is still lead to find the minimum cost compared to ACO as well as the proposed one. The proposed algorithm for several nodes at the under 10 nodes still shows the performance which is comparable to another algorithm including the Brute Force. It started becoming less stronger than ACO for several next nodes until 20. But for the all following number of nodes, it can lead to find the most minimum cost for forming the ring topology.

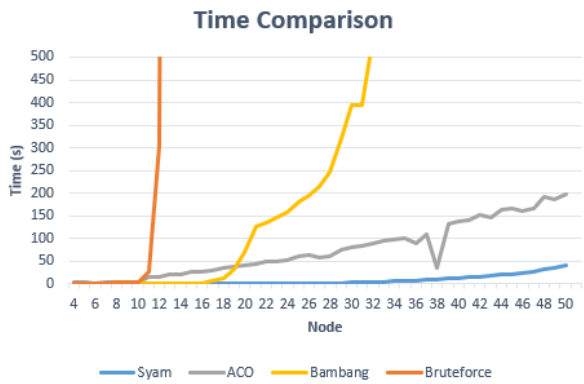

Fig. 3. Computation time between 4 algorithms at various number of nodes.

It can be seen in Fig. 3 that the time needed to create a ring-shaped topology varies between different algorithms. The brute-force algorithm shows very long computational time after the 12th node and when the node is increased, the time required is more exponential. Therefore, the Bruteforce costing test in Fig. 2 is only available until the 12th node. While the Bambang and ant colony algorithms have a gradual increase in computation time as the number of nodes increases.

With this time comparison, the proposed algorithm provides the lowest computation time among other algorithms. The reason is that the proposed algorithm uses a pure heuristic method, that is, it uses only the

information available at the time to make a decision to add a new node. Based on the graph we can see that the time required to complete computing up to 50 nodes is less than 50 seconds.

V. Conclusion

There are several methods which can be acknowledged to find the minimum cost to perform ring topology. The proposed algorithm which used heuristic algorithm can be an alternative where it shows performance to find minimum cost as well as the computation time. This proposed algorithm is better than Ant Colony for the bigger number of nodes, moreover it shows a good time computation for the bigger nodes of the measurement, the proposed algorithm performed better than all other comparised algorithm and stand at the order of 39.7 seconds for 50 nodes.

This algorithm still opened to be developed later to find its most proper performance. It quite effective for the number of nodes as the measurement scenario but of course it needs to prepare for the future need where thousands of them possibly utilized. Future works can be allocated to find the performance of thousand connected devices to simulate a condition such as big continet or archipelagos design of communication. However, it also can be developed more by comparising this algorithm with others that those are not mentioned at this paper.

Acknowledgment

We thank to our colleagues, Mr. Guntur P.K who provided insight and expertise that greatly assisted the research. It is a good harmony where he could help to translate the algorithm into code which later can be simulated to define the time execution.

Our thank also convey to LPDP which has supported and facilitated this research so that it can be presented.

References

[1] G. Kaur and M. S. Patterh, "Performance investigation of ring network topology in the presence of semiconductor optical amplifiers," *Opt. - Int. J. Light Electron Opt.*, vol. 124, no. 22, pp. 5366–5368, Nov. 2013.

[2] I. J. Rahman, A. R. Zain, and N. R. Syambas, "A new heuristic method for optical network topology optimization," in *Proc. 2nd International Conference on Wireless and Telematics (ICWT)*, 2016, pp. 73–77.

[3] M. Jünger, G. Reinelt, and G. Rinaldi, "Chapter 4 The traveling salesman problem," in *Handbooks in Operations Research and Management Science*, vol. 7, Elsevier, 1995, pp. 225–330.

[4] [Online]. Available: http://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/Makalah/

[5] CmSc360 CH3. [Online]. Available: http://faculty.simpson.edu/lydia.sinapova/www/cmsc250/LN250_Levitin/L05-BruteForce.htm.

[6] C. Blum, "Ant colony optimization: Introduction and recent trends," *Phys. Life Rev.*, vol. 2, no. 4, pp. 353–373, Dec. 2005.

[7] J. Ning, Q. Zhang, C. Zhang, and B. Zhang, "A best-path-updating information-guided ant colony optimization algorithm," *Inf. Sci.*, vol. 433–434, pp. 142–162, Apr. 2018.

[8] B. Supriadi, B. Budi, and N. R. Syambas, "Comparison of proposed algorithm to ant colony optimization algorithm to form a ring topology with minimum path value," in *Proc. 2nd International Conference on Wireless and Telematics (ICWT)*, 2016, pp. 67–72.

[9] M. L. Pilat and T. White, "Using Genetic Algorithms to Optimize ACS-TSP," in *Ant Algorithms*, 2002, pp. 282–287.

[10] L. Yuanyuan and Z. Jing, "An application of ant colony optimization algorithm in TSP," in *Proc. Fifth International Conference on Intelligent Networks and Intelligent Systems*, 2012, pp. 61–64.

[11] J. L. Pasquier, I. K. Balich, D. W. Carr, and C. López-Martín, "A comparative study of three metaheuristics applied to the traveling salesman problem," in *Proc. Sixth Mexican International Conference on Artificial Intelligence, Special Session (MICAI)*, 2007, pp. 243–254.

[12] B. K. G. Petrus, S. Qamar, and N. R. Syambas, "A new heuristic method for ring topology optimization: A proposal," in *Proc. 11th International Conference on Telecommunication Systems Services and Applications (TSSA)*, 2017, pp. 1–5.

[13] Trevlovett, *Python-Ant-Colony-TSP-Solver: Solves the TSP using Multithreaded ACO Method*, 2017.

Syamsul Qamar was born in Bakke, South Sulawesi, Indonesia in 1989. Now he is a second-year Telecommunication and Infromatics Master student at School of Electrical Engineering and Informatics, Institut Teknologi Bandung. he received a bachelor's degree in Computer Network from AMIKOM University Yogyakarta. His current research interests is in Computer Network, Routing Protocol and Telecommunication Network. He had experince working as trainer and auditor IT insfrasturcture. And He is currently as awardee of Indonesian Endowment Fund for Education (LPDP) from finance manistri of indonesia.

Sigit Haryadi (sigit@stei.itb.ac.id) born in Ende, Indonesia, on May 13, 1959, is currently working as an Associate Professor, at the School of Electrical Engineering and Informatics, Institute of Technology Bandung, Indonesia, a college founded by the Dutch government in 1920 under the name Bandung Technische Hooge School. He has worked in an Indonesian Telecommunication Industry after graduating in the Electrical - telecommunication Department, Institute of Technology Bandung, Indonesia in 1983 as an engineer. The

fields of the author are the Telecommunication Traffic Engineering, Network Performance, Quality of Service, and Law-Economics science.

Nana R. Syambas. He was graduated from his bachelor degree at Electrical Engineering Department, ITB in 1983. He got his Master by Research degree from Royal Melbourne Institute of Technology, Australia in 1990 and doctoral degree from School of Electrical Engineering and Informatics, ITB in 2011. He has been a lecturer at School of Electrical Engineering and Informatics, ITB since 1984. His research interest includes: Telecommunication Networks, Telematic Services, Content Centric Network (CCN), Software Define Network (SDN), Protocol engineering and Tele-traffic engineering. He has authored or coauthored over 50 published article.