Jurnal CoreIT, Vol.9, No.2, 2023
ISSN 2460-738X (Print)
ISSN 2599-3321 (Online)

Network Routing Optimization Using Tabu Search Algorithm in Dynamic Routing

Iwan Iskandar
Dept. of Informatics Engineering, Universitas Islam Negeri Sultan Syarif Kasim, Indonesia
iwan.iskandar@uin-suska.ac.id

Abstract. Internet penetration is increasing along with the need for data packages for communication such as social media, chatting, video conferencing and others. On large-scale networks such as the Internet, dynamic routing is used to build routing protocol information in the routing table automatically. Currently, Djikstra's algorithm is used to solve the shortest path problem in dynamic routing. In this research, the optimization of the algorithm is carried out in determining the best path or trajectory. One of the optimization algorithms is the Tabu Search Algorithm which can guide heuristic local search procedures to explore the solution area outside the local optimum point. This optimization is assessed from the test parameters measured from the smallest cost. The data analyzed is in the form of bandwidth and topological flow. The result of this research is the path of data packets sent through 9 routers using the Tabu Search algorithm with parameters, namely the number of Neighbor Solutions = 50, Length of tabu list = 10, Maximum Number of Iterations = 100, resulting in a path matrix value of 180.9676. The taken path is through routers 0-2-4-8-9.

Keywords: Bandwidth; Cost; Internet; Network; Tabu Search Algorithm

Received November 2023 / **Revised** November 2023 / **Accepted** December 2023

INTRODUCTION

The use of the internet is increasing in line with the growing need for data packets for communication purposes such as social media, chatting, video conferencing, and more. According to the website wearesocial and Hootsuite, out of the world's population of 7.476 billion people, internet usage reached 3.773 billion users in 2017. Meanwhile, in Indonesia, it grew by 51% in early 2017 [1]. The advancement of the internet drives developments in network technology, especially in routing technology. Routing is used to direct the path of data packets to their recipients. Each router selects a path to reach the next hop through a Routing Protocol [2]. In large-scale networks like the internet, dynamic routing is employed to automatically build routing protocol information in the routing table. Dynamic Routing creates a dynamic (changing) routing table automatically if the topology used also changes. The values considered include bandwidth, network delay, hop count, path cost, load, reliability, and communication costs. In other words, routers must be able to find the path or route with the lowest cost [3].

Currently, the algorithm used to address the shortest path problem in dynamic routing is the Dijkstra algorithm [4]. The Dijkstra algorithm is used in dynamic routing utilizing the Open Shortest Path First (OSPF) routing protocol. OSPF protocol employs the link-state algorithm known as the Dijkstra algorithm or the Shortest Path First (SPF) algorithm. However, given the current increasingly large and dynamic network conditions, the Dijkstra algorithm itself is no longer efficient [5]. Hence, optimization of the algorithm in determining the best path or route is necessary. One of the optimization algorithms is the Tabu Search Algorithm, which guides a local heuristic search procedure to explore solution areas beyond the local optimum point. The Tabu Search Algorithm works by moving from one solution to another by choosing the best solution [6].

Several studies related to routing problems and tabu search have been conducted. First, a study by Christos D. Tarantilis in 2012 published in a journal titled "Adaptive Multi-restart Tabu Search Algorithm for the Vehicle Routing Problem with Cross-docking." According to Christos, utilizing tabu search can exploit new incentive-based mechanisms to guide its routing problem search. This approach has proven to be efficient and effective for most problems with reasonable computational requirements and proper parameter settings [7].

DOI: 10.24014/coreit.v9i2.26655

Another issue related to tabu search can be seen in a study conducted by Slim Belhaiza and colleagues in their journal titled "A Hybrid Variable Neighborhood Tabu Search Heuristic for the Vehicle Routing Problem with Multiple Time Windows," where they mentioned that the tabu search algorithm surpasses the Heuristic Ant Colony. The results of the study using the Hybrid Variable Neighborhood Tabu Search Heuristic (HVNTS) algorithm show the best solution value [8].

This research conducts simulations to optimize network path routes by applying the Tabu Search Algorithm. This optimization is evaluated based on test parameters measured by the smallest cost. Data obtained are from experiments using topologies created in the Cisco Packet Tracer simulator. Analyzed data include bandwidth and topology flow created in several cases with the measured result being the smallest cost value. The expected outcome of this research is to contribute knowledge to the optimization of routing in networks, especially considering the increasing demand for network connections.

METHODS

Tabu search was first introduced by Fred Glover in 1986. Later, in 1988, the Committee on the Next Decade of Operation Research (CONDOR) designated tabu search, along with Simulated Annealing and Genetic Algorithm, as highly promising methods for practical applications. Presently, tabu search is one of the algorithms categorized under heuristic algorithms and has become one of the widely used optimization techniques across various fields [9]. The fundamental concept of tabu search is an algorithm that guides each of its stages to achieve the most optimum goal function without revisiting the initial position during its progress [10]. The objective of this algorithm is to prevent repetition and rediscovery of the same solution within an iteration that might be reused. Additionally, tabu search efficiently conducts searches by utilizing information from the best solutions and exploring new search spaces.

The tabu search algorithm is one of many optimization algorithms used to find an optimal solution among several neighboring solutions, resulting in the best solution for a particular iteration. Each iteration might not always produce an optimal solution; in other words, the outcome may be considerably worse than the previous solution [11]. This depends on the objective and constraint functions . In the tabu search algorithm, a new solution is chosen if it is a member of the neighboring solution set. Neighbors of a solution refer to other solutions obtained by modifying the solution based on specific rules, usually known as the neighborhood function.

According to Fred Glover and Manuel Laguna in their book "Tabu Search" in 1997, the philosophy behind tabu search is to obtain and utilize a collection of intelligent problem-solving principles. This means that tabu search is grounded in selected concepts that unite the fields of artificial intelligence and optimization. An analogy that closely represents this concept is that of a mountain climber who must selectively remember the path taken and strategically choose the route to reach the mountain summit [12].

Betrianis and colleagues in 2003 cited research conducted by Chambers and Barnes. According to Chamber and Barnes, the application of tabu search utilizing dynamic approaches can be adjusted to changing search conditions, especially in highly flexible manufacturing systems [13].

TABU SEARCH FUNDAMENTAL CONCEPT

The basic concept of tabu search is similar to the simulated annealing method. The tabu search algorithm can also accept solutions that are worse than the current solution obtained [[14]. To prevent the loss of the best solution obtained, tabu search stores this best solution and continues searching based on the last solution found in the search process [14] This algorithm also remembers some solutions it has encountered and prohibits revisiting those solutions to avoid futile repetitions [15] . By utilizing this remembering and prohibiting technique, tabu search becomes one of the efficient algorithms in terms of effort and time [16].

According to Hillier (2005) as cited by Suyanto in his book, tabu search has five main parameters that must be carefully determined. These parameters include the local search procedure, neighborhood structure, tabu conditions, aspiration conditions, and termination criteria [15].

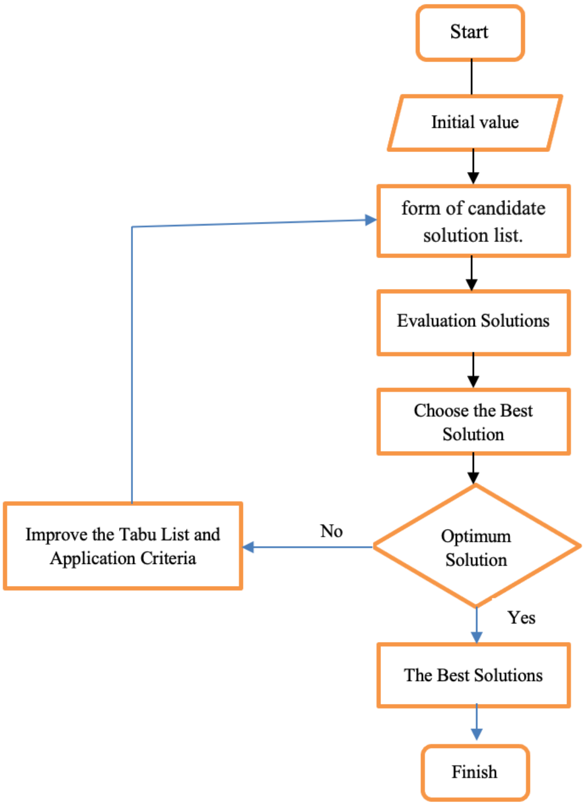

Figure 1. Stages of Tabu Search

The explanation for the standard tabu search flowchart above is as follows [17]:

1. The first step involves determining or defining candidate solutions.
2. Next, determining alternative solutions by exchanging (moving) between several points within those solutions.
3. The third step involves the tabu list, which evaluates the alternative solutions to see if the candidates from those solutions are already present in the tabu list. If the solution is already in the tabu list, it will not be evaluated. Conversely, if the solution is not present, it will be included in the tabu list for evaluation as the best alternative solution.
4. The subsequent step is to select the best solution from the multiple solutions to be designated as the latest optimum solution.

The final step, if the stopping criteria are met, the process will stop, and the optimum solution will be obtained. However, if not, the process will restart from the fourth step.

TABU SEARCH ALGORITHM

In general, the tabu search algorithm can be seen as follows:

```
1. Generate a feasible initial solution, for example, s, randomly or
   using a specific heuristic method.
2. OptimalCost = Cost(s)
3. S* = s (s* is the best solution obtained)
4. TabuList = null
5. Repeat
   a. V* = set of solutions that are neighbors of s and meet the
      aspiration criteria or are not in the tabu list
   b. Choose s' (s' is the solution with the minimum cost in V*)
   c. Store the reverse move into TabuList that changes s to s'
   d. s = s'
      if (Cost(s) < OptimalCost) then
      s* = s
      OptimalCost = Cost(s)
      End
   Until (StopCriteria = true)
6. Return (s*)
```

NETWORK DESAIN

The initial stage of this research involves creating a network topology design with predetermined nodes for each. Subsequently, each node or router is assigned a different bandwidth allocation [18]. From these bandwidth values, they are processed to derive a cost value. The cost values from each previously established scenario will serve as a benchmark for comparing the best cost value. To determine the cost value, it is obtained using the formula:

$$\text{Cost (bandwidth)} = \frac{10^8}{\text{bandwidth}} \qquad (1)$$

This formula will reveal all the cost values present for each network connection to its respective router.

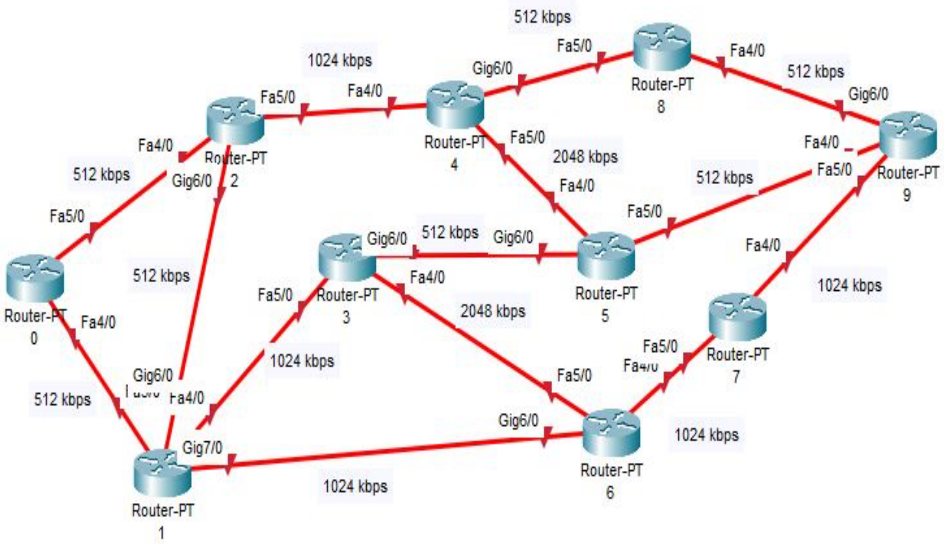

Figure 2. Network Desain

In Figure 2, there are 9 units of routers interconnected. In this scheme, it is assumed that the routers are interconnected and operating normally. Each router has been configured for routing access to all other routers along with the bandwidth used. The amount of bandwidth used varies between the router interfaces within each router. This is intended to generate different costs for each router interface.

APPLICATION OF TABU SEARCH ALGORITHM

The search for the best route using the tabu search algorithm begins with inputting matrix values into a pre-made table. Once the matrix is entered into the table, the tabu search will then examine neighbors from the initial node. If that node has been visited before, remove that neighbor from the tabu list; if it hasn't been visited, proceed directly to the neighboring node with the smallest value. Next, the tabu search will inspect other neighboring nodes within the tabu list: if a neighboring node is not on the tabu list, immediately move to the selected node; however, if the node is on the tabu list, select the node with the smallest value in that tabu list and reposition that node backward (prioritize the target node). The next step is to check the iteration; if the iteration is not complete, return to the process of node selection of neighbors. However, if the iteration is complete, calculate the total cost of each iteration; the iteration with the smallest cost is the chosen iteration. Below is a flowchart illustrating the sequence of the tabu search algorithm

Parameter input:

1. Number of Elements = 9
2. Number of Neighbor Solutions = 50
3. Length of tabu list = 10
4. Maximum Number of Iterations = 100

Stages in tabu search:
The first step is to initialize the initial value of the tabu search as follows:
Initialization
K = 1
Iteration = 100
Comparing each value
S_1 = B_1.1 ∶ B_2.1 ∶ B_3.1 ∶ B_4.1……..B_9.9
If K has reached the target, the search stops, and the optimal solution has been achieved

Set k = 1
S_1 = B_1.1 ∶ B_2.1 ∶ B_3.1 ∶ B_4.1……..B_9.9
S_0 = S_1
S_c1 = B_1.1 ∶ B_2.1 ∶ B_3.1 ∶ B_4.1……..B_9.9 and
S_(c2) = B_2.1: 〖B〗 1.1 ∶ B_3.1 ∶ B_4.1 ∶ B_5.1………..B_9.9

Is the move forbidden? No.

Then S_c1 = B_1.1 ∶ B_2.1 ∶ B_3.1 ∶ B_4.1……..B_9.9 and → Tabu List = {(B_1.1 ∶ B_2.1 ∶ B_3.1 ∶ B_4.1……..B_9.9)}
Check G(S(c1)) = 0 < G(S_(0))? G(S_(0)) = 0
G(S_c1) = 0 → G(S_best)
G(S_(c2)) = 0

K = 1 + 0 = 1;
K = 1? Yes, then stop because the target solution has been achieved with K = K+1

Table 1. The values of the matrix used.

Roi uter	0	1	2	3	4	5	6	7	8	9
0	0	36,055 51275	53,851 64807	41,231 05626	102,95 63014	94,339 81132	70,710 67812	106,30 14581	75	100

Roi uter	0	1	2	3	4	5	6	7	8	9
1	36,0555128	0	80	44,72135955	123,6931688	100	36,05551275	72,11102551	62,64982043	85,44003745
2	53,8516481	80	0	44,72135955	50	60	104,4030651	134,1640786	74,33034374	94,33981132
3	41,2310563	44,72135955	44,72135955	0	80,62257748	56,56854249	60,8276253	89,4427191	36,40054945	60,8276253
4	102,956301	123,6931688	50	80,62257748	0	50	140	162,788206	93,40770846	102,9563014
5	94,3398113	100	60	56,56854249	50	0	104,4030651	120	50,24937811	53,85164807
6	70,7106781	36,05551275	104,4030651	60,8276253	140	104,4030651	0	36,05551275	55,90169944	70,71067812
7	106,301458	72,11102551	134,1640786	89,4427191	162,788206	120	36,05551275	0	70,17834424	72,80109889
8	75	62,64982043	74,33034374	36,40054945	93,40770846	50,24937811	55,90169944	70,17834424	0	25
9	100	85,44003745	94,33981132	60,8276253	102,9563014	53,85164807	70,71067812	72,80109889	25	0

RESULT AND DISCUSSION

From the calculation results using the Tabu Search Algorithm, the route for transmitting data packets through the router can be obtained as follows:

Table 2. The Value of the Matrix

Source Router	Router Destination	Nilai
0	1	36,05551
1	6	36,05551
6	7	36,05551
7	9	72,8011

The total value of the matrix is 180.9676.
Below are the results of the data packet transmission route on the router.

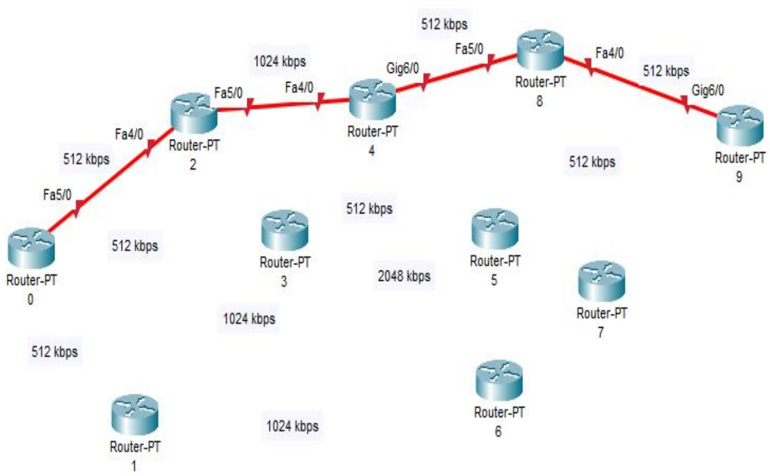

Figure 3. Tabu Search data packet routing

From the calculation results using the Tabu Search algorithm, the main result obtained is the route for sending data packets through routers 0-2-4-8-9.

CONCLUSION

The Tabu Search algorithm has optimized the packet route running on the router. In this research, it has produced the packet path sent from the source router to the destination router with predefined paths and bandwidth. From tracing the path of data packets sent through 9 routers using the Tabu Search algorithm with parameters: number of Neighbor Solutions = 50, Tabu List Length = 10, Maximum Iteration Number = 100, the result of the matrix route value obtained is 180.9676. The route taken is routers 0-2-4-8-9.

REFERENCES

[1] K. S, "Digital in 2017: Global Overview.," 2017. [Online]. Available: https://wearesocial.com/special-reports/digital-in-2017-global-overview.

[2] S. S, Jaringan Komputer, Politeknik Elektronika Negeri Surabaya, 2014.

[3] H. Nugroho, Matematika Diskrit dan Implementasinya dalam Dunia Teknologi Informasi, Deepublish, 2015.

[4] R. Munir, Matematika Diskrit, Bandung: Informatika Bandung, 2015.

[5] R. Kumar and M. Kumar, "Exploring Genetic Algorithm for Shortest Path Optimization in Data Networks," *Global Journal of Computer Science and Technology ,* vol. 10, no. 11, pp. 8-12, 2010.

[6] B. &. W. P. Santosa, Metoda Metaheuristik, Konsep dan Implementasi., 2011.

[7] C. D. Tarantilis, "Adaptive multi-restart Tabu Search algorithm for the vehicle routing problem with cross-docking Optimization Letters," vol. 7, p. 1583–1596, 2013.

[8] S. H. P. G. Belhaiza, " A hybrid variable neighborhood tabu search heuristic for the vehicle routing problem with multiple time windows Computers Operations Research 2014," vol. 52, pp. 269-281, 2014.

[9] I. Martina, "Penerapan Algoritma Genetika dengan Crossover Cut and Splice dalam Optimasi Routing Jaringan. Jurnal Telematika,," 2011. [Online]. Available: http://journal.ithb.ac.id/telematika/article/view/50.

[10] R. Novrianti, Analisa Algoritma Tabu Search untuk Penjadwalan Mata Pelajaran di SMPN 1 Cisolok Palabuhanratu, Sukabumi: 10104311, 2008.

[11] I. S. O. &. D. R. Varita, "Pencarian Jalur Tercepat Rute Perjalanan Wisata dengan Algoritma Tabu Search.," vol. 2, pp. 185-190, 2013.

[12] F. &. L. M. Glover, Tabu Search., 2007.

[13] E. D, *Kajian Algoritma Routing dalam Jaringan Komputer. ,* vol. 2(1), p. Edi D.

[14] B. L. Fox, " Integrating and Accelerating Tabu Search, Simulated Annealing and Genetic Algorithms. Annals of Operations Research,," *4,* Vols. 41(2), 47–67. , pp. 56 - 67, 1993.

[15] Suyanto, Algoritma Optimasi: Deterministik atau Probabilistik. Graha Ilmu., 2014.

[16] S. G. M. d. S. M. . Optimasi Penjadwalan Perkuliahan dengan Menggunakan Algoritma Genetika (Studi Kasus : AMIK MDP, "Witary, V., Rachmat, N., & Inayatullah. (2013)," 1–7..

[17] "Tarantilis," *Adaptive Multi-restart Tabu Search Algorithm for the Vehicle Routing Problem with Cross-docking. Optimization Letters,* no. Jurnal STMIK GI MDP, 1–7., pp. 1583 - 1596.

[18] A. I. &. H. B. Wijaya, "Manajemen Bandwidth dengan Metode HTB (Hierarchical Token Bucket) pada Sekolah Menengah Pertama Negeri 5 Semarang.," *Jurnal Teknik Informatika Udinus,* vol. 1, no. 1, p. 5–7., 2014.

[19] S. Saldi, "Implementasi Algoritma Tabu Search Dalam Pewarnaan Simpul Graf (STUDI KASUS : Penjadwalan mata kuliah jurusan matematika fakultas sains dan teknologi UIN Alauddin Makassar). Journal of Personality and Social Psychology,," *1(1), 1188–1197. ,* pp. http://doi.org/10.1111/j.1469-7610.2010.02280.x., 2017.