ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645
Indian Journal of Science and Technology, *Vol 8(S7), 327–331, April 2015*

Hybrid Routing Algorithm on Mesh Network based on Traffic Records

Sunghyuck Hong*

Division of Information and Communication, Baekseok University, Korea; shong@bu.ac.kr

Abstract

This research proposes the determining shortest paths for hops, which are in the middle of the source and destination. The shortest path in this research means the fastest path between the source and destination. In Recent, dynamic routing algorithms are currently used and developed by researchers. This research algorithm suggests that the fastest paths between the source and the destination which is based on using the record of the network traffic history. Main subject on networking is a user, and the pattern of network usage is always corresponding to network history. Users are using the networking and the network traffic is always corresponding to how many users use the networking in specific time. Therefore, network traffic information can be predicted by the network condition which is referring to the history of network traffic record, and then the shortest path can be produced without using RIP too much. It will be helpful to improve the network traffic and contribute efficient packet transferring on network.

Keywords: Component, Efficient Communication, Hybrid Traffic Control, Network Traffic, Static Routing, The Shortest Paths

1. Introduction

The Internet began with ARPANET and just e-mail triggered rapid growth in ARPANET, so the Web has triggered explosive growth in the Internet. Today, there are 100 million hosts connected to Internet networks; many users are in the 100s of millions. Hundreds of countries access the Internet and the growth of Internet hosts and users are still increasing dramatically[11]. The growth of Internet hosts and users is much faster than the growth of network traffic speed because of too many users. When we travel on the Internet, the network routers have to find out the path from source to destination and give the shortest path to users for destination. If we find the shortest path between them, we could get the result faster, unless it takes somewhat longer than the previous case. Determining the shortest path (the fastest path) for hops is important to communicate with destination and source. According to *Roch Guerin*, and *Ariel Orda*[2], finding minimum hops for all possible paths is the most efficient in order to get the shortest path in a certain network. However, the path weight has to be considered additive and bottleneck weights. To get the shortest path routing, use Dijkstra's shortest path algorithm or Bellman–Ford algorithm, which is widely used for finding the shortest algorithm[5]. The shortest path algorithm is based on the graph theory. There are many algorithms to find the shortest path but they always pick the same path if their algorithms are correct. The only difference is the efficiency.

2. Hybrid Network Traffic Model

Here is a virtual mesh network topology in Figure 1. Suppose node 1 is the source and node 6 is the destination. Each node has a routing table and updates its routing table regularly. Also, they exchange RIP (Routing Information Protocol) every 60 seconds and after 180 seconds, if no response is detected, they regard as it is not connected[11].

*Author for correspondence

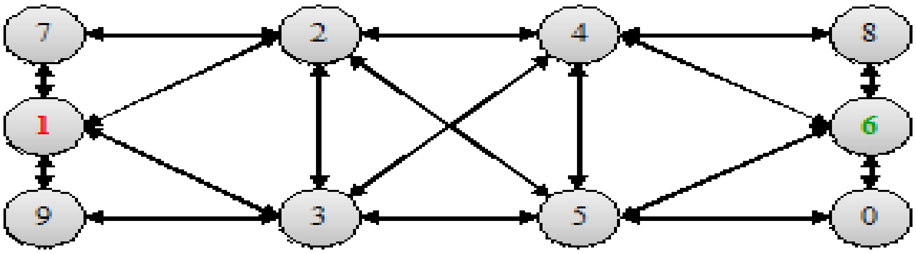

Figure 1. Virtual mesh network topology.

Table 1 show 1 hop network cost in 10:00 am, so calculating the shortest path between node 1 and node 6 depends on the network traffic table, and the values of inside parenthesisis actual network speed in Table 1.

This paper can simulate to determine the shortest path based on the history of network traffic records and also it will show the difference between the actual the shortest path and calculate the shortest path. Table 1 shows one hop network weight between each node at 12:00 pm. The weight value between two nodes is the prediction value of weight at 12:00 pm which means it collects records of weight values at every 12:00 pm in during the week. Table 2 shows how the prediction value has been calculated.

Therefore, according to Table 2, the shortest path between node 1 (in red) and node 6 (in green) is going to be 1-2-5-6. The total cost is 6.7ms. That is the expected cost, but it is can be a different current dynamic algorithm because the network condition always depends on how many users are using the network or how many packets are currently delivering on the network path. In fact, the actual shortest path may be different from this path, but it can be ignored because that is not going to be a big difference.

Table 1. Network cost on hop to hop

Date	Time	Node 1 to 2 weight value	Node 1 to 3	Node 2 to 3	…
Jan 21	12:00 pm	*1.8ms*	*1.9ms*	*1.8ms*	…
	12:10 pm	*1.9ms*	*1.8ms*	*1.0ms*	…
	12:20 pm	*2.0ms*	*2.1ms*	*2.0ms*	…
	12:30 pm	*2.1ms*	*2.1ms*	*2.0ms*	…
	12:40 pm	*2.2ms*	*2.0ms*	*2.2ms*	…
	12:50 pm	*2.2ms*	*2.2ms*	*2.5ms*	…
	13:00 pm	*2.3ms*	*2.5ms*	*2.5ms*	…

* Each weight value is an average of every day in specific time.

Suppose we send packets on the network like Figure 1, and we calculate the network cost such as when sending, receiving packets and exchanging RIP, updating routing table in every 60 seconds. New system model sends RIP information in every 10 minutes, and then updates the routing table and network traffic table when it gets RIP from its neighbor node.

Now, it is time to predict a certain time of network traffic information by using the mathematical method such as the linear least squares fitting technique because we have only a 10 minutes period between routing information. The linear least squares fitting technique[12-15] is the simplest and most commonly applied form of linear regression and provides a solution to the problem of finding the best fitting straight line through a set of points if the relation between time and network traffic data are a line. In this case, suppose time and network traffic are supposed to be a correlation from 12:00 pm to 13:00. RIP information can be exchanged every 10 minutes, thus the routing table also updates every 10 minutes like Table 1. Therefore, if the source sends a packet to destination at 12:05, the network traffic needs to be calculated by the linear least squares fitting technique. Because we knew

Table 2. Network cost on each node

12:00 pm	Node 1	Node 2	Node 3	Node 4	Node 5	Node 6
Node 1	-	*1.9 (1.8)*	*2.1(2.0)*	-	-	-
Node 2	*1.8(1.9)*	-	*3(2.9)*	*3.1(3.2)*	*2.8(2.7)*	-
Node 3	*1.5(1.4)*	*2.8(2.6)*	-	*2.9(3.0)*	*2.5(2.5)*	-
Node 4	-	*1.8(1.6)*	*1.6(1.5)*	-	*1.7(1.8)*	-
Node 5	-	*1.4(1.3)*	*1.1(1.2)*	*1.6(1.4)*	-	*1.9(2.0)*
Node 6	-	-	-	*2.3(2.1)*	*2.5(2.3)*	-

Table 3. The elapsed time for the shortest path

Time	The shortest path between node 1 and node 6	
	The shortest Path	Total time to reach
12:00 pm	*1 – 2 – 5 – 6*	*6.7ms*
12:10 pm	*1 – 2 – 5 – 6*	*6ms*
12:20 pm	*1 – 3 – 4 – 6*	*7ms*
…	…	…
14:00 pm	*1 – 3 – 4 – 6*	*10ms*
14:10 pm	*1 – 3 – 5 – 6*	*14ms*
…	…	…
20:00 pm	*1 – 3 – 4 – 6*	*11ms*
20:10 pm	*1 – 2 – 4 – 6*	*12ms*
…	…	…

the time and network traffic but were not certain about the time's network traffic information, the network traffic at 12:05 could be measured by using another data set. 12:05 is in range of Table 2, so it can be calculated. Let, x be the time variable and y be a network traffic weight value.

Suppose that the data points are(x_1, y_1), (x_2, y_2) ,…, (x_n, y_n),where, 'x' is the independent variable and y is the dependent variable. The fitting curve f(x) has the deviation (error) d from each data point, i.e., $d_1 = y_1 - f(x_1)$, $d_2 = y_2 - f(x)$,…, $d_n = y_n - f(x_n)$. According to the method of least squares, the best fitting curve has the property that:

$$\pi = d_1^2 + d_2^2 + \cdots + d_{ni}^2 = \sum_{i=1}^{n} d_i^2 = \sum_{i=1}^{n} [y_i - f(x_i)]^2 = a\ minimum$$

In this case, we suppose have the linear relationship between time and network traffic from 10:00 to 11:00, so we can apply this to the least-squares line uses a straight line.

$y = a + bx$ .... (the linear relationship of time and network traffic)

to approximate the given set of data, (x1, y1), (x2, y2) ,…, (xn, yn),

where, n>=2. The best fitting curve f(x) has the least square error, i.e.,

$$\pi = \sum_{i=1}^{n} [y_i - f(x_i)]^2 = \sum_{i=1}^{n} [y_i - (a + bx_i)]^2 = \min$$

Please note that a and b are unknown coefficients while all xi and yi are given. To obtain the least square error, the unknown coefficients a and b must yield zero first derivatives.

$$\begin{cases} \dfrac{\partial \pi}{\partial a} = 2\sum_{i=1}^{n} [y_i - (a + bx_j)] = 0 \end{cases}$$

$$\begin{cases} \dfrac{\partial \pi}{\partial b} = 2\sum_{i=1}^{n} x_i [y_i - (a + bx_j)] = 0 \end{cases}$$

Expanding the above equations, we have:

$$\sum_{i=1}^{n} y_i = a\sum_{i=1}^{n} 1 + b\sum_{i=1}^{n} x_i$$

$$\sum_{i=1}^{n} x_i y_i = a\sum_{i=1}^{n} x_i + b\sum_{i=1}^{n} x_i^2$$

The unknown coefficients a and b can therefore be obtained:

$$\begin{cases} a = \dfrac{\left(\sum y\right)\left(\sum x^2\right) - \left(\sum x\right)\left(\sum xy\right)}{n\sum x^2 - (\square)\sum x^2} \\ b = \dfrac{\left(\sum xy\right) - \left(\sum x\right)\left(\sum y\right)}{n\sum x^2 - (\square)\sum x^2} \end{cases}$$

where, $\sum \square \ldots$ stands for $\sum_{i=1}^{n} \ldots i$

Therefore, we get y line for time and network, so we can predict the network traffic information at 10:05 by using this method. I can get the results of x, y, x^2, y^2, and xy.

Table 4. Node 1 to node 2 only

n	*x*	*y*	x^2	y^2	*xy*
1	*1*	*1.8*	*1*	*3.24*	*1.8*
2	*2*	*1.9*	*4*	*3.61*	*3.8*
3	*3*	*2.0*	*9*	*4*	*6*
4	*4*	*2.1*	*16*	*4.41*	*8.4*
5	*5*	*2.2*	*25*	*4.84*	*11*
6	*6*	*2.2*	*36*	*4.84*	*13.2*
7	*7*	*2.3*	*49*	*5.29*	*16.1*
Sum	**28**	**14.5**	**140.0**	**30.2**	**60.3**
Average	**4**	**2.1**	**20.0**	**4.3**	**8.6**

Table 4 is based on Table 1 for node 1 to node 2 only.

a = (14.5x140–(28x60.3))/((7x140)–28x28) = 1.7429
b = (7x60.3–28x14.5)/(7x140–28x28) = 0.0821
Thus, y = 1.7429 + 0.0821 x

I would like to get network traffic at 12:05, so we assign x value as 1.5. 12:00 is assigned to 1, so 12:05 is between 12:00 and 12:10, so it will be 1.5. That means x is equal to 1.5, y is equal to 1.7429 + 0.0821x 1.5 = 1.8661. According to Table 1, this result sounds acceptable.

3. Conclusions

As we can see, the network traffic information can be predicted by the history of network traffic information by using mathematical prediction methods such as the linear least squares fitting technique if there is a linear relationship between two factors, time and network traffic information in a local area network without using too much routing information protocol every 60 seconds. Also, it reduced the routing information on the network to avoid router overhead that doesn't need to update routing information every 60 seconds. This paper focused on reducing unnecessary information and improvement of network traffic by using the linear least squares fitting technique, which is required to the linear relationship between factors, but fortunately in this case, we supposed that there is a linear relationship between time and network traffic, so it would be applied to this case. However, if the relationship is not a linear, we have to nth Degree Polynomials. That would be more complex and more errors will happen, so the predicted result wouldn't be trustable. I think that applying nth Degree Polynomials will be the future work in this paper.

4. Acknowledgement

This research is supported by 2015 Baekseok University research fund.

5. References

1. Guerin R, Orda A. Computing shortest paths for any number of hops. IEEE/ACM Transaction on Networking. 2002; 10(5):613-620 .
2. Bhide NM, Sivalingam KM, Fabry-Asztalos T. Routing mechanisms employing adaptive weight functions for shortest path routing in optical WDM networks. Tibor Source: Photonic Network Communication. 2001; 3(3):227–236.
3. William S. High-speed networks and Internets performance and quality of service. 2nd edition. Prentice Hall. P. 7–8.
4. Ran P, Sun MH, Zou YM, ZigBee Routing selection strategy based on data services and energy-balanced zigbee routing. Proceedings of the 2006 IEEE Asia-Pacific Conference on Services Computing, Guangzhou, China. 2006; 400–4.
5. Cai YN, Wang FB. Research on ZigBee routing strategy based on data service and energy control. Computer Measurement & Control. 2008; 16:42–3.
6. Chakeres I.D, Klein-verndt L. AODVjr, AODV Simplified. Mobile Computing and Communication Review. 2002; 6(3):100–1

7. Lin ZJ, Meng QH, Liang HW. A route discovery method based on limited flooding in ZigBee networks. IEEE International Conference on Automation and Logistics. 2008; 3039–3044.
8. Zhang Y, Sale H. The Study of ZigBee network routing algorithm based on adjacent relation. 2nd IEEE International Conference on Consumer Electronics, Communications and Networks. 2012; 1484–7.
9. Toh CK. Ad Hoc mobile wireless networks: protocols and systems. Prentice Hall. 2001.
10. Kuruvila J, Nayak A, Stojmenovic I. Hop count optimal position based packet routing algorithms for ad hoc wireless networks with a realistic physical layer. Journal IEEE Journal on Selected Areas in Communications archive. 2006; 23(6):1267–1275.
11. Srinivasan P, Kamalakkannan P. REAQ-AODV: Route stability and energy aware QoS routing in mobile Ad Hoc networks. Fourth International Conference on Advanced Computing (ICoAC); 2012.
12. Sasikumar P, Abraham Roy M. Analysis of power saving routing protocol for wireless. International Journal of Computer Applications; 2010.
13. Marie Feeney L. An energy consumption model for performance analysis of routing protocols for mobile ad hoc networks. Journal Mobile Networks and Applications. 2001; 6(3):239–249.
14. Kopekar AS, Dhakate AS, Kale A. Performance evaluation of routing algorithms for adhoc wireless sensor network and enhancing the parameters for good throughput. International journal of applied information systems (IJAIS). 2012; 3(6): 23–8.
15. Chlamtac I, Conti M,.Liu JJN. Mobile adhoc networking: imperatives and challenges. Ad Hoc Networks. 2003; 1(1):1364.
16. Wu Y, Fellow C, Zhang Q, Jain K, Zhu W, Yuan S. Network planning in wireless adhoc networks: a crosslayer approach. IEEE journal on selected areas in communications. 2005; 23(1):136–150.
17. Bansal N, Liu Z. Capacity, delay and mobility in wireless adhoc networks. INFOCOM Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies. 2003; 2:1553–1563 .
18. Albarazi K, Mohammad U, Al-holou N. Doppler shift impact on vehicular adhoc networks. Canadian journal on multimedia and wireless networks. 2011; 2(3).
19. Kanhere SK, Goudar M, Wadhai VM. End-to-end delay optimization in Wireless Sensor Network (WSN). International Journal of Smart Sensors and Ad Hoc Networks. 2012; 1(3):2248–2258.
20. Dyo V, Mascolo C. Efficient node discovery in mobile Wireless Sensor Network. DCOSS '08 Proceedings of the 4th IEEE International Conference on Distributed computing in Sensor Systems; 2008.
21. Cao Y, Zhong Z, Gu Y, He T. Safeguarding schedule updates in Wireless Sensor Networks. Mini-Conference at IEEE INFOCOM; 2011.
22. Dousse O, Mannersalo P, Thiran P. Latency of Wireless Sensor Networks with uncoordinated power saving mechanisms. MobiHoc '04 Proceedings of the 5th ACM International Symposium on Mobile ad hoc Networking and Computing. 2004; 109–120.
23. Lindsey S, Raghavendra C, Sivalingam KM. Data Gathering algorithms in sensor networks using energy metrics. IEEE Trans. On Parallel Distrib. Syst. 2002; 13(9): 924–935.
24. Wang Y, Qian Z, Feng L, Guo Y. ZigBee hybrid routing optimization algorithm based on energy balance. Advanced Materials Research. 2013; 765:1667–1670.
25. Ha JY, Park HS, Choi S, KwonWH. EHRP: enhanced hierarchical routing protocol for ZigBee mesh networks. 2007; 11(2):1028–1030.