Journal of Computers Vol. 29 No. 6, 2018, pp. 181-192
doi:10.3966/199115992018122906017

Research on Service-oriented Virtual Network Mapping Algorithm

Yonan Guo[1,2], Yichu Bai[1,2*], Hao She[2]

[1] College of Telecommunications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

[2] Jiangsu Key Lab of Wireless Communications, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
guo@njupt.edu.cn,bai.yc@foxmail.com, 896917450@qq.com

Received 9 January 2018; Revised 22 January 2018; Accepted 22 January 2018

Abstract. Studying service-oriented virtual network mapping algorithm aims for lowering the delay of important data transmission caused by bandwidth congestion which is possible to occur in the transmission of mass data. Service-oriented mapping algorithm can automatically and effectively utilize physical links with lesser delay in transmitting important data, making information transmission of network more reasonable and orderly. Through classifying virtual tasks into four precedence levels, different algorithms are used to map virtual links onto physical links, hence reaching the target of priority transmission of important information. In the end, Labview is adopted to simulate the algorithm proposed, by which it is concluded that the algorithm proposed shows good anti-congestion performance in transmitting important data.

Keywords: algorithm defined by service type introduction, network virtualization algorithm

1 Introduction

In the modern society with the all-around development of internet of things, the out-dated network framework has already failed to satisfy the requirements for large communication, enormous storage and low delay. The currently mainstream design scheme used to solve these problems is to develop a new set of network, designing the structure from substrate layer gradually up to top layer. By doing this, the current network environment will no doubt be refreshed and be able to satisfy various rapid transmissions of high-demanding tasks, but the cost is huge. Therefore, the research team led by Prof. Larry Peaterson at Princeton University proposed the approach of promoting network system framework reform with network virtualization technology in 2005 [1]. Users' demands upon physical network basically lie in network node and bandwidth link of processing tasks. Functions of bandwidth link for different demands vary. Hence, allowing operating multiple heterogeneous network virtualization upon planned physical network is highly mouldable. They are heterogeneous not only in structure but also in function, so it is a correct option to use network virtualization to realize network upgrading. In virtual network, different virtual nodes can be mapped onto the same physical node, and original network transmission can be mapped into the memory transmission of the same node. In this way, virtual network is much more flexible than traditional networks. Therefore, network virtualization remains an effective way to solve problems in future network development.

This paper studies service type based mapping algorithm, the core of which first lies in classifying tasks to be assigned into four types according to their requirements upon bandwidth capacity and link delay, and then marking the four types of tasks with different priorities before the virtual network maps different services. On the one hand, the high-priority tasks should be allocated to the link with rich recourses, and the low-priority tasks could leave the resource occupied to the tasks which need more

* Corresponding Author

resources. On the other hand, the tasks which need abundant bandwidth resource have no influence on the mappings that are high priority, hence making enough space for the links though calculation.

The remaining parts of this paper are arranged as follows: Part II presents related works, mainly analyzing the demerits of various virtual network mapping algorithms and comparing their advantages and disadvantages with the proposed algorithm; Part III is algorithm description, expounding the core part of the proposed algorithm; Part IV introduces the simulation results and analysis, in which labview software is used to simulate the proposed algorithm and obtain its advantages after comparison with others; Part V gives the conclusion of this paper.

2 Related works

Current network has been evolving from single physical mainframe and single virtual mainframe into virtual network, whose advantage lies in possessing both flexibility and effectiveness. Nowadays, research on virtual network mapping mainly focuses on two aspects, namely virtual network mapping algorithm, and re-modified mapping algorithm for the same task in dynamically changing network. This paper focuses on studying virtual network mapping algorithm. Mapping methods for virtual network mapping algorithm at present mainly include two types, static virtual network mapping and dynamic virtual network mapping. Static virtual network mapping refers to that all mapping tasks know information in advance and the information will not undergo any changes in operation. Dynamic virtual network mapping refers to that mapping tasks cannot get to know the information size and requirements upon various resources can change in operation.

In terms of static virtual network mapping, Li proposed the dynamic topological setting of network in 2015 [2]; Kareche discussed how to allocate substrate layer network resources to virtual network in 2014 [3]. Fischer summarized the then VNE conditions and analyzed current problems in 2013 [4] Minlan proposed that the split and migration of substrate layer links in virtual network could more effectively complete virtual network requests in 2008 [5]. Chowdhury proposed cluster mapping of virtual network nodes to lower the cost of node mapping in 2009 [6]. Khan proposed to complete node and link selection based on OSNs in 2016 [7]. Dietrich investigated the visibility of virtual network provider at substrate layer network resources and tested the applicability of topological structure in 2015 [8]. Haeri optimized Dijkstra's shortest path algorithm in 2016 [9]. Lira and Tavares proposed one type of metaheuristic based virtual network embedding method to lower the number of node and link to be activated and hence save resources [10]. The acceptance rate of virtual network requests by static virtual network mapping is relatively low with weak practicability, but the cost is relatively low.

In terms of dynamic virtual network mapping, Butt et al. proposed re-optimization of topological awareness in 2010 [11]. Shuopeng used cloud computing to apply network virtualization into more areas in 2016 [12]. Zangiabady et al. proposed VN migration method in 2016 to improve VN acceptance rate [13]. Guler et al. proposed one method named Virtual Multicast Tree Embedding based on dynamic Impact Factor (VMTE-IF) to lower resources requirement and path redundancy in 2017 [14]. Shahriar et al. proposed several solutions for failed node activation [15]. Purushothaman et al. simulated network virtualization with photoelectric data center [16]. Dynamic virtual network mapping is highly flexible and practical with a high acceptance rate of virtual network requests.

For example, the mapping algorithm proposed in [5] selected the node with the largest residual amount of CPU resources in physical nodes as mapping node every time, and when paths were not enough branching could be adopted to conduct mapping of one task with two paths. As Fig. 1 (lowercase letters are virtual nodes, uppercase letters are physical nodes), when mapping is conducted along the path of b-a-c, the residual bandwidth amount on D-E is only 20 due to the occupation of physical node D-E by path a-b, so the bandwidth amount of 30 required by d-e cannot be satisfied. Therefore, Yu et al. proposed to divide 1-b into upper and lower paths with bandwidth occupation of 10 to satisfy mapping of d-e. If nodes with largest residual amount of CPU resources are selected for mapping, then bandwidth resources around large capacity nodes must be enough, or otherwise bandwidth congestion will arise. And the branching method fails to classify tasks. Assume that the virtual task on the path a-b is remote operation of high-precision instrument while the task on d-b is browsing task for ordinary users, then the branching method as Fig. 1 will cause various mistakes for too much delay in operation. This is the result of failing to include task into precedence level.

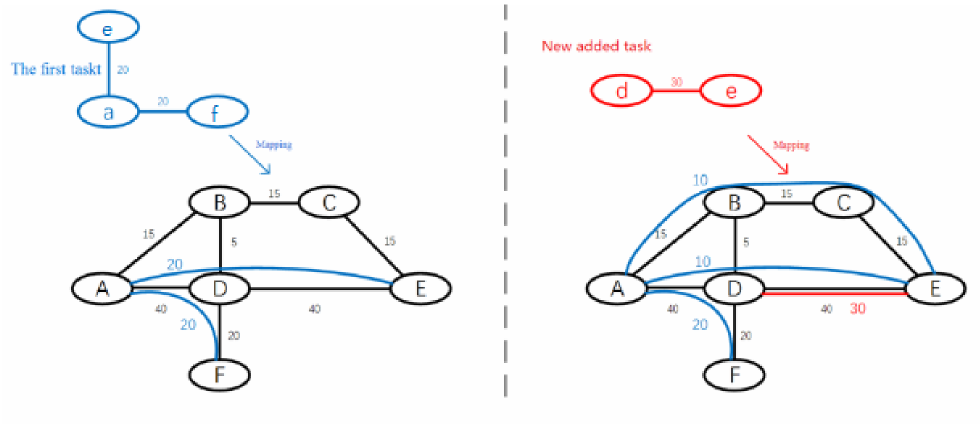

Fig. 1. Conduct mapping

[17] proposed one type of virtual topological pre-setting method to conduct mapping of virtual tasks as Fig. 2: its overall idea lied in using some algorithms in a large physical topology to merge the nodes with larger bandwidth amount between each pair into one node. As Fig. 2, the CPU amount of the first topological network is 3, 1, 2, and the three nodes turned into 6 in the second network. With their CPU amount added up, the bandwidth in between was regarded as the data cache for direct link, with the bandwidth amount for connecting other nodes unchanged. After transforming for 3 times with this method, a relatively complicated topological network would become a simple one. The merit lies in turning a complicated mapping into a simple one, lowering calculation amount, removing differences of tasks, and accelerating mapping. But the problems are that there can be data congestion on some key links and mistakes on merged nodes for insufficient path bandwidth. As Fig. 2 in the third block diagram, if CPU is fully occupied by the bandwidth 2 between the two nodes of 2 and 4, then node with a CPU of 6 must work as a transfer station if communication is to be carried out between 2 and 4. But this transfer becomes mutual transmission between two nodes with a CPU of 6 in the fourth diagram.

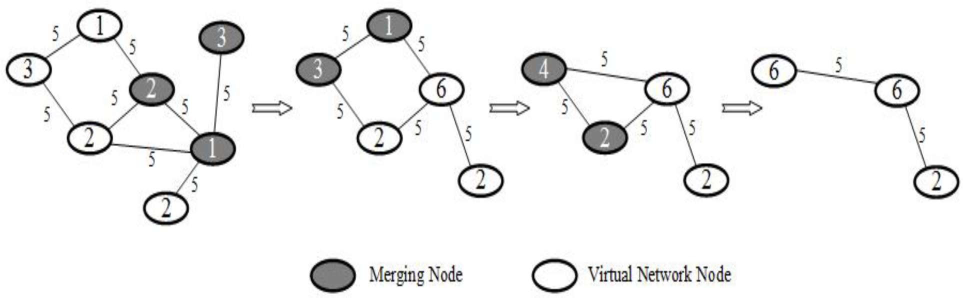

Fig. 2. Physical topology

Most of the current network service processes are static, that is, when the service is mapped, the service is no longer changed. If the current link resource is insufficient to process the later service, it will be discarded. This makes it difficult to utilize the resources of the physical network effectively. Therefore, some scholars put forward the establishment of service-oriented network. The core content of service oriented network is service, which classifies and integrates the logic modules with the same characteristics [18], Then perform the same operations on these integrated modules to solve the difference of network transmission. One of the research trends of service oriented network is that the network can combine services for users according to the needs of users [19]. Finally, the system maps different node links for different services according to the current state of the network, the main methods are Planning Domain Definition Language [20] and Hierarchical Task Network-based [21].

To sum up, network virtualization is a relatively easy way to achieve the network upgrade. However, although the service oriented network is very advanced and fully researched, it is still difficult to implement. Therefore, we use the service oriented network related algorithm to study the virtual network mapping algorithm, and propose the algorithm in this paper. At present, many virtual network mapping algorithms optimize the various tasks without distinction. Although this optimization can solve some of the current problems, increase efficiency, but these optimizations can not make all the tasks of users run smoothly without perception. Therefore, in the next part of the paper, we propose a service category based virtual network mapping algorithm and description of the algorithm.

(i.e., equal margins left and right and top and bottom). The format of the paper (A4, Letter, etc.) is irrelevant.

3 Algorithm Description

In this paper, we propose a service type based virtual network mapping algorithm, which, through dividing services into different types, realizes the largest and most scientific utilization of substrate layer network for mapping virtual network when using different virtual network algorithms. According to different service types, we divide tasks into four types according to precedence level, namely low bandwidth occupation with delay requirement, high bandwidth occupation with delay requirement, low bandwidth occupation without delay requirement, and high bandwidth occupation without delay requirement. If there is serious bandwidth occupation in executing tasks of low precedence level, then partial quality links will be reserved for tasks of high precedence level, or even tasks of low precedence level will be temporarily queued up so as to execute tasks of high precedence level first. For different types of tasks there are different mapping algorithms, through which users can be aware of the delayed tasks to utilize most quality resources in links on a precedence level. And for the resources without delay perceived by users, they can use links with high delay.

Based on different service types, virtual network services are divided into four types, namely low transmission amount with sensitivity to delay, large transmission amount with sensitivity to delay, low transmission amount without sensitivity to delay and large transmission amount without sensitivity to delay. For example, the first type of service corresponds to high-precision instrument operation or game battles, where the data amount is extremely small but slight delay will exert a huge influence upon users. The second type of service corresponds to social video networking software, where a large amount of data is used to transmit real-time video data, and delay will significantly affect the communication of two parties. The third type of service corresponds to word transmission service like chatting and forum commenting, where even the delay is calculated in hundreds of milliseconds or even seconds the influence upon users can be neglected. The fourth type of service corresponds to downloading and updating. As long as the bandwidth occupied is enough, users do not care about whether it is real-time or not but whether the transmission can be completed within a short time.

If a precedence level is to be arranged, then it is the first type of service > second type of service > third type of service > fourth type of service. Every type of service needs to map a virtual network. The substrate layer network is as shown in Fig. 3, where the big ellipse represents nodes, line represents the link between nodes, the black represents the size of node CPU, small number represents serial number, and the number in small ellipse represents bandwidth size of links. The red diagram shows the node and link size required by virtual network.

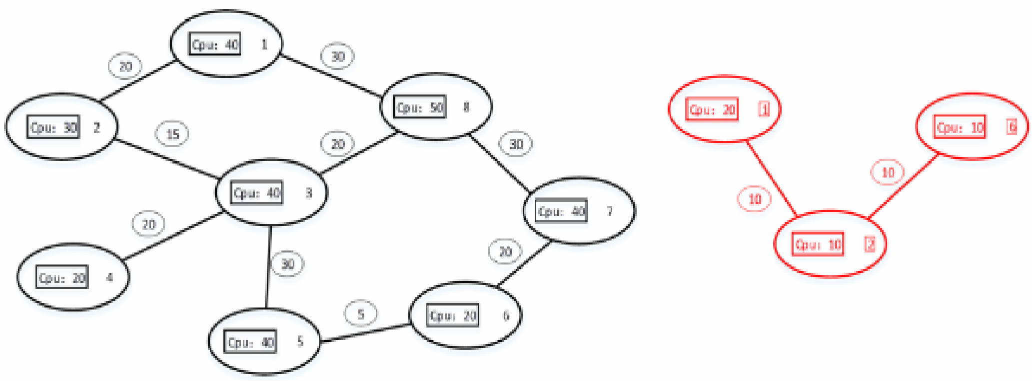

Fig. 3. Four services in substrate network

Network diagram after mapping is shown in Fig. 4.

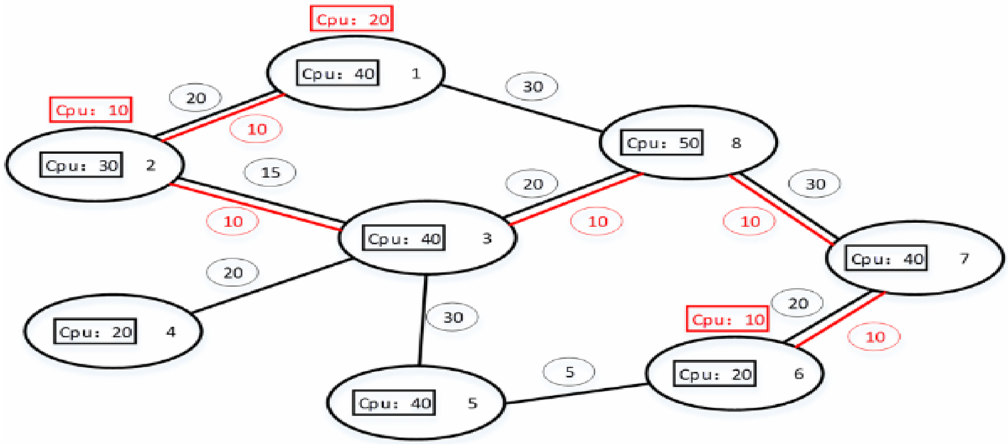

Fig. 4. Network diagram after mapping

Therefore, to enable the same hardware network to accommodate more virtual networks, we recommend four algorithms for the above four types of services as well as several additional ones.

3.1 First Type of Service: Low Transmission Amount with Sensitivity to Delay

The first type of service occupies few bandwidths, so its influence upon others can be neglected. But due to its strict requirement for bandwidth delay, its precedence level should be the highest. In mapping, the CPU and bandwidth occupied by the first type of service on original paths should be pruned. As Fig. 5, the blue represents the resources already occupied by the first type of service, so it needs to be pruned from hardware resources to simplify the topological structure. After pruning, the shortest path algorithm is used for path calculation of this service to judge whether the residual bandwidth is enough to operate this service; or otherwise, tasks on this path with lowest precedence level and highest bandwidth occupation should be terminated in order until it is able to map this task. Then the terminated tasks will be re-mapped according to the algorithm for the first type of service.

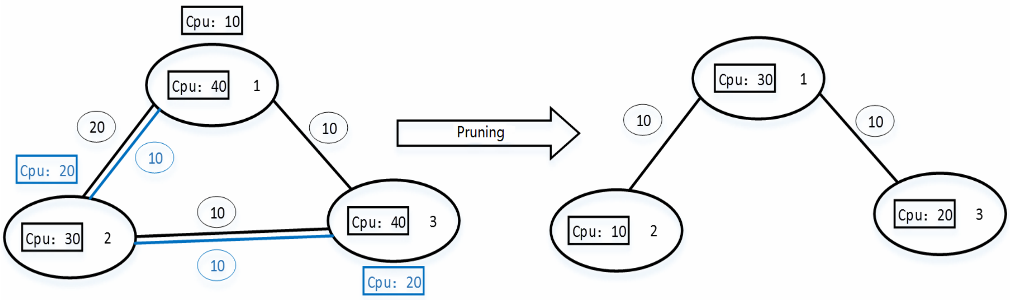

Fig. 5. Pruning example

Algorithm for the above service:

Input: current topological network structure VNR, service currently occupying network and service 1 n+1 to be mapped

Output: current topological network structure VNR, service occupying network

(1) Search for all tasks with X=1 in all services;

(2) Deduct CPU and bandwidth occupied by the above services from VNR, and obtain bandwidth now i;

(3) Adopt the shortest path algorithm to calculate route for service 1 n+1, and obtain route n+1;

(4) If route n+1 can complete mapping, then mapping terminates. If residue is not enough, all service X i passing on route n+1 are arranged according to the size of X in a descending sort, and terminate the first task;

(5) If route n+1 completes mapping, then redirect to the terminated task and re-map it. If it cannot be completed, repeat step (4).

(Notes: X in Service Xi represents precedence level, with 1 the highest and 4 the lowest, i is the serial number, bandwidth now i is the current residua amount of bandwidth, route n+1 represents the path data of new added tasks).

3.2 Second Type of Service: Large Transmission Amount with Sensitivity to Delay

The second type service applies to real-time services like online videoing, which occupies a lot of bandwidth and is very sensitive to delay, because slight delay will cause users dissatisfaction of the service quality. So the precedence level of the second service is lower than the first type of service for high precision but higher than other services without sensitivity to delay. As Fig. 6, when the second type of service needs bandwidth, first of all the bandwidth occupied by the first type of service (red) should be regarded as non-existent, and then the shortest path algorithm is used for route analysis upon remaining paths. If remaining space of current route satisfies the mapping of the tasks, then mapping completes; if not, a suitable number of tasks with low precedence level will be terminated for mapping, and then their algorithm will be used for mapping tasks with low precedence level.

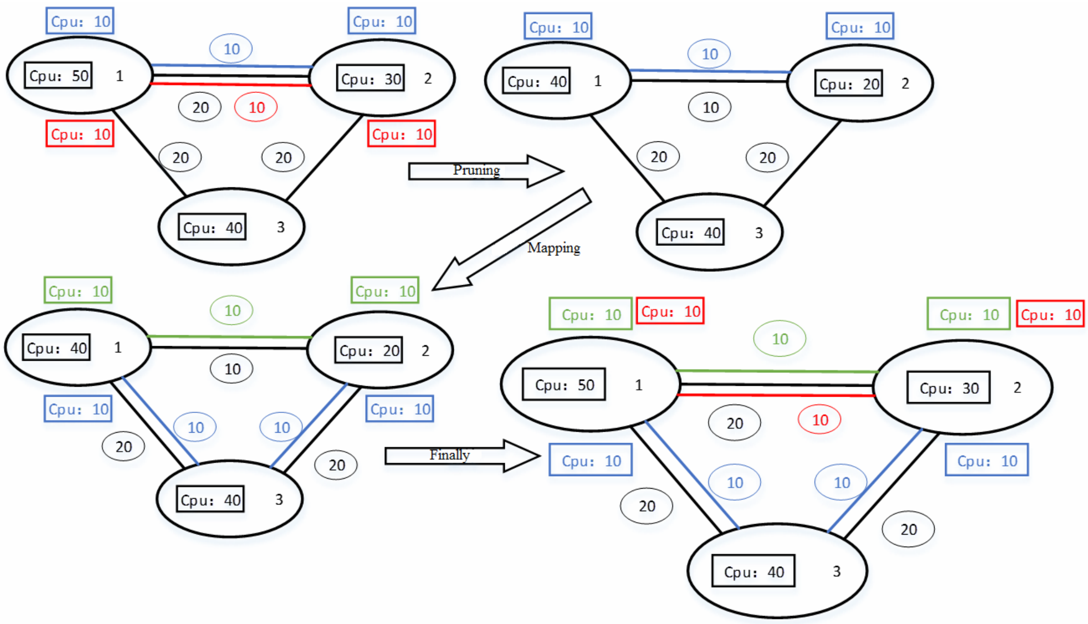

Fig. 6. Second Type of Service: Large Transmission Amount with Sensitivity to Delay

When the present bandwidth cannot satisfy the mapping requirement, we only neeed to terminate the smallest amount of the fourth task that occupies the bandwidth because the third task has extremely low footprint of bandwidth.

The algorithms for the second type of service:

Input: current topological network structure VNR, service currently occupying network $\sum_{X=1}^{4}\sum_{i=0}^{n}$ service X_i and service 1 n+1 to be mapped

Output: current topological network structure VNR, service occupying network

(1) Search for all tasks with X=1 in all services;

(2) Deduct CPU and bandwidth occupied by the above services from VNR, and obtain bandwidth now i;

(3) Adopt the shortest path algorithm to calculate route for third type of service n+1, and obtain route n+1;

(4) If route n+1 can complete mapping, then continue the next step. If residue is not enough, the task of X=4 on route n+1 is arranged in ascending order to terminate the smallest task.;

(5) If route n+1 completes mapping, then redirect to the terminated task and re-map it. If it cannot be completed, repeat step 4;

(6) If there is no task to be mapped at present, end the mapping.

3.3 Third Type of Service: Low Transmission Amount Without Sensitivity to Delay

The third type of service corresponds to word related tasks, where users sensitivity to word delay is much lower than the above two situations. So its precedence level is ranked third. In the same way, before mapping the third type of service, tasks of high precedence level should be pruned first followed by mapping it.

When the present bandwidth cannot satisfy the mapping requirement, we only need to terminate the smallest amount of the fourth task that occupies the bandwidth because the third task has extremely low footprint of bandwidth.

The algorithms for the third type of service:

Input: current topological network structure VNR, service currently occupying network $\sum_{X=1}^{4}\sum_{i=0}^{n}$service X_i and service 1 n+1 to be mapped

Output: current topological network structure VNR, service occupying network

(1) Search for all tasks with X=1 in all services;

(2) Deduct CPU and bandwidth occupied by the above services from VNR, and obtain bandwidth now i;

(3) Adopt the shortest path algorithm to calculate route for service 3 n+1, and obtain route n+1;

(4) If route n+1 can complete mapping, then continue the next step. If residue is not enough, the task of X=4 on route n+1 is arranged in ascending order to terminate the smallest task.;

(5) If route n+1 completes mapping, then redirect to the terminated task and re-map it. If it cannot be completed, repeat step 4;

(6) If there is no task to be mapped at present, end the mapping.

3.4 Fourth Type of Service: Large Transmission Amount without Sensitivity to Delay

The fourth type of service corresponds to various downloading and updating services, which are normally running backstage and have no requirement upon delay yet with serious occupation of bandwidth. So route split based dynamic network mapping method[3] can be adopted to accelerate transmission and lower occupation of single path.

As Fig. 7, The two links with the lowest residual bandwidth on the two paths from node 1 to node 4 have the residual bandwidth percentages of 40% and 80% respectively. After mapping a fourth class of services, the algorithm allocates 10 units of bandwidth to the links above and 20 units of bandwidth to the links below. Finally, the remaining percentages of the occupied bandwidth are as consistent as possible.

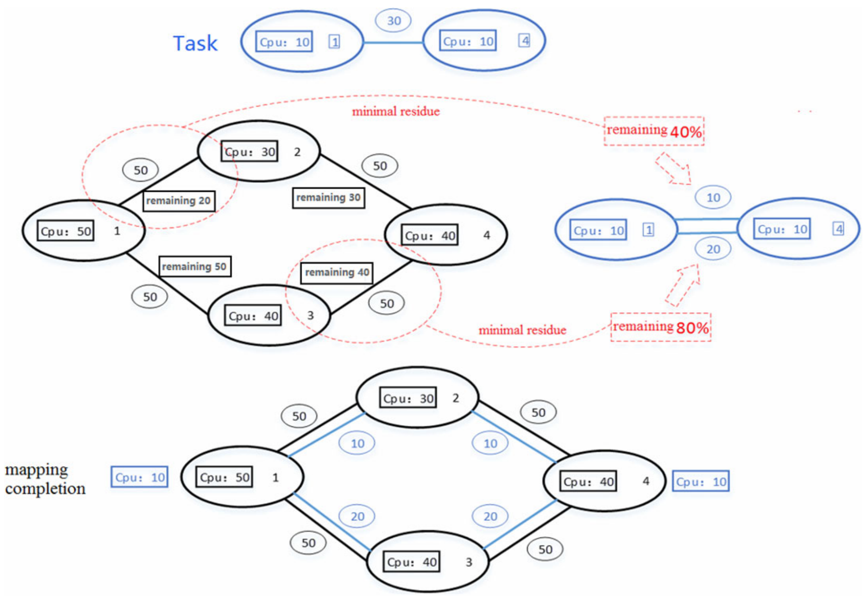

Fig 7. Fourth type of service: large transmission amount without sensitivity to delay

Based on the very serious nature of the fourth tasks for bandwidth occupancy, we must take a synthetic consideration of the bandwidth residual value, bandwidth residual ratio and the length of the occupied line to minimize the impact of the fourth service on bandwidth of other services. After the mapping is done, endeavor to the following aspects: 1. Occupying as little bandwidth as possible; 2. The residue after two bandwidth occupying is as large as possible; 3. The remaining ratio of the two bandwidth after occupying is as consistent as possible.

Hence, we propose the following algorithm to solve the fourth service:

Input: current topological network structure VNR, service currently occupying network $\sum_{X=1}^{4}\sum_{i=0}^{n}$service X_i and service 1 n+1 to be mapped

Output: current topological network structure VNR, service occupying network

(1) Adopt the shortest path algorithm to calculate route for service 4 n+1, and obtain route n+1;

(2) Adopt the shortest path algorithm to calculate route another shortest route n+2 other than the route n+1.

(3) Deduct CPU and bandwidth occupied by the above services from VNR, and obtain bandwidth now i;

(4) $A = \min[bandwidth\ now\ i(route\ n+1)] \times length(route\ n+2) \times \sqrt{\min(\frac{bandwidth\ now\ i(route\ n+1)}{bandwidth\ i(route\ n+1)})}$

(5) $B = \min[bandwidth\ now\ i(route\ n+2)] \times length(route\ n+1) \times \sqrt{\min(\frac{bandwidth\ now\ i(route\ n+2)}{bandwidth\ i(route\ n+2)})}$

(6) $bandwidth(route\ n+1) = bandwidth(n+1) \times \frac{A}{A+B}; bandwidth(route\ n+2) = bandwidth(n+1) \times \frac{B}{A+B}$

(7) Finally, the corresponding bandwidth is distributed according to the size of bandwidth (route n+1) and bandwidth (route n+2) and the mapping is completed.

(Annotation: min[bandwidth now i(route n+1)] represents the line with the minimum residual bandwidth value on route n+1. $\min(\frac{bandwidth\ now\ i(route\ n+1)}{bandwidth\ i(route\ n+1)})$ represents the line with the lowest residual bandwidth percentage on route n+1; length (routh n+1) represents the number of links passed by route n+1; bandwidth (n+1) represents the total bandwidth value of the task.)

The above four algorithms can accelerate the time effectiveness of various tasks under the condition of not lowering the effectiveness of physical links. Although the time effectiveness of some tasks can worsen, they are those having no influence upon users. Unlike remote operation, tasks like downloading, updating, word transmission do not have strong dependency upon time effectiveness. Hence, adopting service type based virtual network mapping algorithm in general can accelerate users' network.

4 Related Test

The above four algorithms can accelerate the time effectiveness of various tasks under the condition of not lowering the effectiveness of physical links. Although the time effectiveness of some tasks can worsen, they are those having no influence upon users. Unlike remote operation, tasks like downloading, updating, word transmission do not have strong dependency upon time effectiveness. Hence, adopting service type based virtual network mapping algorithm in general can accelerate users' network.

The test was completed on labview simulation software, with some key codes as Fig. 8.

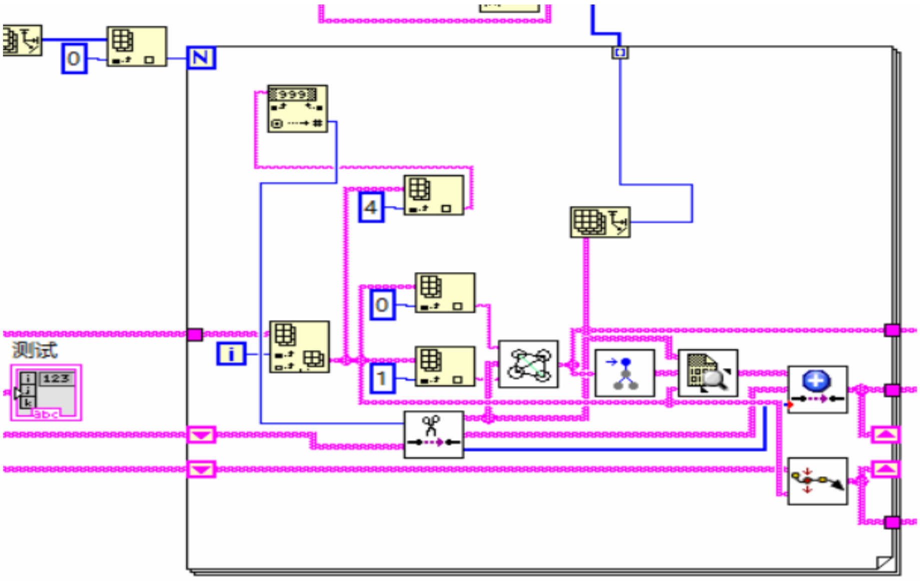

Fig. 8. Test using labview

Simulated physical paths are as Fig. 9.

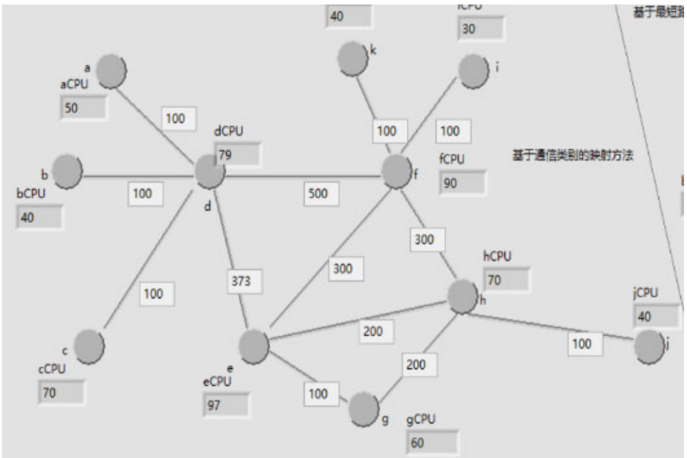

Fig. 9. Demonstrate of the test: simulated physical paths

The simulated four types of service random appear between any two nodes, and the calculation for path bandwidth occupied by task is $X^2/40$ (X is a random value between 0-40). Many tasks in practical use are small ones, and the tasks with a large occupation of bandwidth like downloading, updating and real-time videoing arise much less frequently than other tasks occupying almost no bandwidth. The calculation of delay utilizes the number of paths passed by each task, but the calculation methods for different types of service vary. Due to different influences of delay of different service types upon users, the delay calculation is 2Y for the first type of service (Y is the path passed by task), Y for the second type of service, Y/2 for the third type of service, and Y/4 for the fourth type of service. Then the added-up delay will be the total delay for these tasks. In the end, new tasks will be added constantly to obtain Fig. 10.

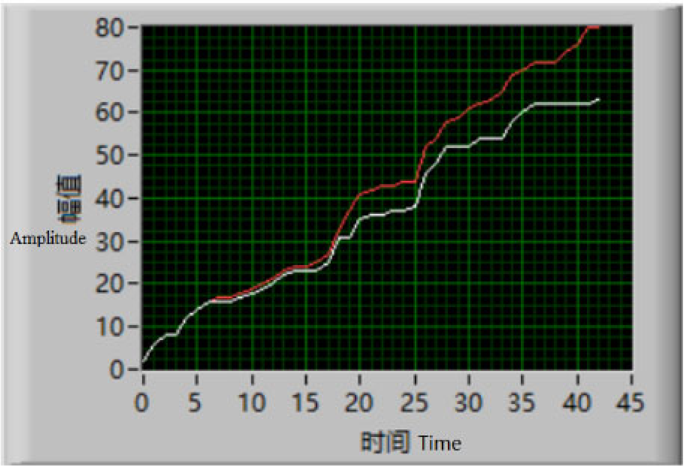

Fig. 10. In ordinary cases, the results occurred when the tasks were assigned randomly

Where, the red line is the oscillogram of the total delay for the shortest path algorithm, and the white line is the oscillogram for communication type based virtual network algorithm. It is clear that their delay difference is not large when there are few tasks. With increasing number of tasks, the total delay becomes larger. Therefore, communication type based virtual network mapping algorithm is more suitable for network with enormous communication amount and approaching physical saturation. If a large task with a bandwidth occupation of 240 is added into the e-f section at the beginning with no requirement for delay, and then tasks are randomly allocated according to previous algorithm, we can obtain Fig. 11.

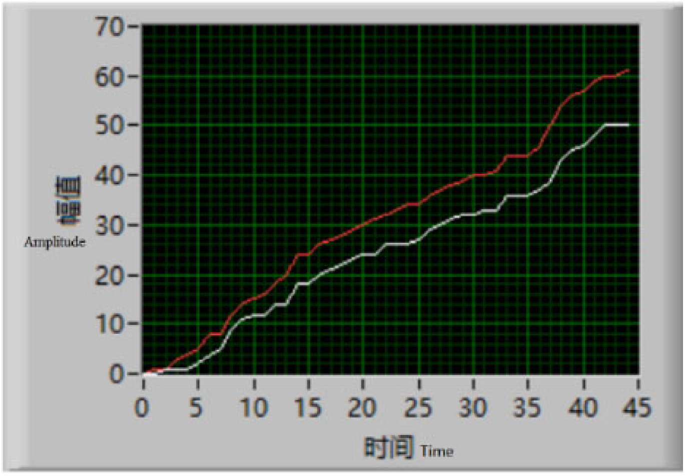

Fig. 11. If a high-bandwith needed task is set in advance, the results occurred when the tasks were assigned randomly

It can be seen that the path is split at the beginning. This is because common algorithms do not classify tasks, so at the beginning a large task is allocated to one shortest path according to optimal algorithm. However, communication type based virtual network mapping algorithm allocates this large task to other paths before the arrival of other more important tasks, and forces them to use the path with a shorter delay. Therefore, we can reach another conclusion that the network with irregular generation of enormous downloading tasks is more suitable for the communication type based mapping algorithm.

5 Conclusion

Future network development cannot be separated from upgrading and optimizing of fundamental network devices. This paper, after studying the mapping algorithm of using virtual network to optimize current network, proposes the service type based virtual network mapping algorithm. Unlike previous research directions, this paper starts from enhancing user experience and lowering special data transmission delay instead of enhancing link effectiveness and maximal number of containable tasks, and finally verifies with test that it boasts improved performance in the above two aspects.

At the same time, there are some problems that cannot be solved. The first is how to understand the delay requirement of each data for different data from different applications? It is not clear whether each task is added to the classified information or is identified by the network itself. Secondiy, although the algorithm has been greatly reduced for the important task of the transmission delay, but the task of how to optimize the different priority algorithm can make the overall link maximum effectiveness is still an important problem, which is always kept on the topic. The third is that we only consider the demand of the bandwidth on the link without adding the demand of link node CPU. Therefore, it is impossible to achieve simultaneous optimization of bandwidth and CPU in actual mapping. Therefore, our future research directions are mainly to identify different priority data and to further optimize the algorithm to improve the overall link validity.

Acknowledgments

The work is supported by the Natural Science Foundation of China under Grants 61372124, 61427801, the Natural Science Foundation of the Jiangsu Higher Education Institutions (Grant No. 13KJB520029), the Jiangsu Province colleges and universities graduate students scientific research and innovation program CXZZ13_0477, 2017WICOM06, NUPTSF (Grant No. NY214033, NY217033).

References

[1] T. Anderson, L. Peterson, S. Shenker, J. Turner, Overcoming the Internet impasse through virtualization, Computer 38(4)(2005) 34-41.

[2] H. Li, T. Zhou, Q. Wang, The study of dynamic topology remapping in virtual network embedding, in: Proc. International Conference on Information and Communication Technology Convergence, 2015.

[3] S. Kareche, S. Ductor, Z. Guessoum, M. Mezghiche, A new robust heuristic for assigning substrate network resources to virtual networks, in: Proc. International Conference on Advanced NETWORKING Distributed Systems and Applications, 2014.

[4] A. Fischer, J.F. Botero, M.T. Beck, H. de Meer, X. Hesselbach, Virtual network embedding: a survey, IEEE Communications Surveys & Tutorials 15(4)(2013) 1888-1906.

[5] M. Yu, Y. Yi, J. Rexford, M. Chiang, Rethinking virtual network embedding: substrate support for path splitting and migration, Acm Sigcomm Computer Communication Review 38(2)(2008) 17-29.

[6] N.M.M.K. Chowdhury, M.R. Rahman, R. Boutaba, Virtual network embedding with coordinated node and link mapping, in: Proc. INFOCOM DBLP, 2013.

[7] M.S. Khan, A.W.A. Wahab, T. Herawan, G. Mujtaba, S. Danjuma, M.A. Al-garadi, Virtual community detection through the association between prime nodes in online social networks and its application to ranking algorithms, IEEE Access 4(99)(2017) 9614-9624.

[8] D. Dietrich, A. Rizk, P. Papadimitriou, Multi-provider virtual network embedding with limited information disclosure, IEEE Transactions on Network & Service Management 12(2)(2015) 188-201.

[9] S. Haeri, Q. Ding, Z. Li, L. Trajkovic, Global resource capacity algorithm with path splitting for virtual network embedding, in: Proc. IEEE International Symposium on Circuits and Systems, 2016.

[10] V. Lira, E. Tavares, Energy-aware mapping for dependable virtual networks, in: Proc. International Workshop on Power and Timing Modeling, Optimization and Simulation, 2015.

[11] N.F. Butt, M. Chowdhury, R. Boutaba, Topology-awareness and reoptimization mechanism for virtual network embedding, in: Proc. Ifip Tc 6 International Conference on NETWORKING Springer-Verlag, 2010.

[12] S. Li, M.Y. Saidi, K. Chen, A cloud-oriented algorithm for virtual network embedding over multi-domain, in: Proc. Local Computer Networks Workshops, 2017.

[13] M. Zangiabady, C. Aguilar-Fuster, J. Rubio-Loyola, A virtual network migration approach and analysis for enhanced online virtual network embedding, in: Proc. International Conference on Network and Service Management, 2017.

[14] E. Guler, D. Zheng, G. Luo, L. Tian, X. Cao, Embedding virtual multicast trees in software-defined networks, in: Proc. ICC 2017 - 2017 IEEE International Conference on Communications, 2017.

[15] N. Shahriar, R. Ahmed, S.R. Chowdhury, A. Khan, R. Boutaba, J. Mitra, Generalized recovery from node failure in virtual network embedding, IEEE Transactions on Network & Service Management 14(2)(2017) 261-274.

[16] L. Purushothaman, T. Truong-Huu, M. Gurusamy, Time and bandwidth-aware virtual network embedding and migration in hybrid optical-electrical data centers, in: Proc. International Conference on Advanced Information NETWORKING and Applications, 2017.

[17] M. Yu, Y. Yi, J. Rexford, M. Chiang, Rethinking virtual network embedding: substrate support for path splitting and migration, Acm Sigcomm Computer Communication Review 38(2)(2008) 17-29.

[18] M.K. Haki, M.W. Forte, Service oriented enterprise architecture framework, in: Proc. Services IEEE, 2010.

[19] Y.M. Zhang, J. Wang, The modeling of service composition path selection problem: research on linear-structured composed service routing problem, in: Proc. International Symposium on Intelligence Information Processing and Trusted Computing, 2010.

[20] X. Wang, W. Zhang, An automatic Semantic Web Service composition approach based on PDDL, in: Proc. 2010 International Conference on Educational and Network Technology, 2010.

[21] A. Hristoskova, B. Volckaert, F.D. Turck, Dynamic composition of semantically annotated web services through QoS-aware HTN planning algorithms, in: Proc. International Conference on Internet and Web Applications and Services, 2009.