

On the Dynamic Replacement of Virtual Service Resources for Mobile Users in Virtual Networks

Jean Frédéric Myoupo^{1*}, Yannick Florian Yankam², Vianney Kengne Tchendji²

¹ Computer Science Lab-MIS, University of Picardie Jules Verne, 33 rue St. Leu 80039 Amiens, France.

² Department of Mathematics and Computer Science, University of Dschang, Dschang, Cameroon.

* Corresponding author. Tel.:+33 322825915; email: jean-frederic.myoupo@u-picardie.fr

Manuscript submitted November 10, 2019; accepted January 8, 2020.

doi: 10.17706/jcp.15.1.10-21

Abstract: In virtual networks, network traffic is managed through the resources of a substrate network. When users move from one access point to another, the traffic that is generated increases in variation, and the virtual service resources within each virtual network must be maintained at an acceptable quality of service (QoS). However, the resources that are offered by this virtual service resource may become insufficient to manage this traffic, harming the QoS. In this paper, we propose a virtual service resource replacement method that consists of migrating a service from one virtual node to another, offering sufficient resources to provide a good QoS when the traffic changes. Our method improves those of the literature by dealing with the cases of equal traffic weight at the node level and the service migration strategy. Our approach reduces the service migration time and the number of virtual service resource replacements compared to those of the literature.

Key words: Virtual network, virtual service resource replacement, traffic change, mobile user, service migration.

1. Introduction

Virtual networks are networks that are built on the resources of a physical network and are used to provide services. These virtual networks have special properties such as architectural flexibility and isolation ([1], [2]) that they provide to the users, and various services that are hosted in servers are called virtual service resources. A virtual service resource in a virtual machine supports the incoming and outgoing traffic that is generated by mobile users relative to the requested service. Consequently, the quality of service (QoS) depends on the amount of traffic involved.

However, the traffic that is associated with a given service is not constant; rather, it increases and decreases depending on the number of users and the network structure mutation. Sometimes, the resources of the virtual server become insufficient to handle these changes wisely, harming the QoS. Therefore, there is a need for policies that allow the virtual network to continuously ensure the QoS to end users and deal with these traffic fluctuations and topology changes. These topology mutations come from the addition and removal of virtual machines in the network. Several possible solutions can be explored to deal with such a problem:

- Increase the resources of the virtual service resource so that it can support the amount of generated traffic. This solution is related to the well-known resource allocation and Virtual Network Embedding (VNE) problems in the virtual networks. These problems have been widely studied in the literature [3]-

[5]. The common disadvantage of the methods proposed in these works is that they progressively reduce the physical resources that are available for the benefit of virtual networks;

- move the virtual service from a virtual service resource to another one with more available resources: this solution has been very little explored so far [6]. This approach does not require additional resources from the substrate network. Our contribution in this paper is based on this approach.

With regards to the virtual service resource replacement, the most important challenges to be overcome in this scenario are the following:

- Find the new node that will receive the virtual service resource;
- migrate the service to the new node;
- restructure the virtual network after a topology change. When the network topology change is from a tree to a graph structure, a tree topology needs to be rebuilt if we want to use method [6] as the virtual service resource replacement approach;
- design a multiple virtual services resources migration approach in a VN.

1.1. Related Works

Revisiting the dynamic replacement of virtual service resources in the literature, we found that this approach has been explored very little to the best of our knowledge.

In the field of virtual networks with a tree topology, [7] proposes a dynamic replacement of virtual network resources to satisfy the quality of service of the provided service when the traffic changing comes from a new node that is added in the network. The idea is to migrate the virtual service resource successively by one hop from the overloaded node to another one that has enough resources, until the new destination node is reached. At each migration to a node, it is determined whether the quality of service is satisfied before proceeding to the next node. The new node is the node at one hop of the new added node. Fig. 1 shows this replacement method. When the new node n_k (Fig. 1.a) is added in the topology, the virtual service resource that is initially located in node n_2 is replaced in node n_6 at one hop of node n_k through the shortest path (Fig. 1.b). The main drawback of this method is that it takes a lot of time to replace a virtual network service, which harms the QoS.

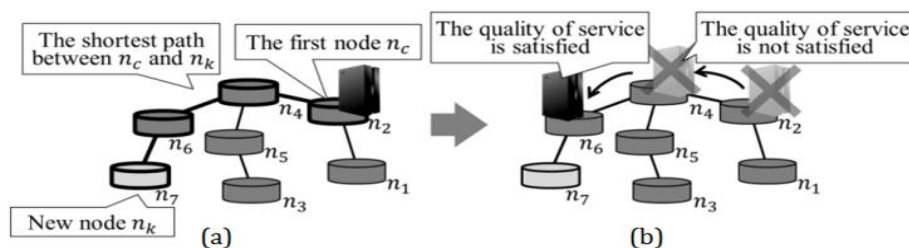


Fig. 1. A virtual service resource replacement method by moving the virtual network service by one hop.

To overcome the drawback of [7], [6] proposes a dynamic virtual service resource replacement method that does not check if the quality of service is satisfied at each node between n_c and n_k . This results in a reduced virtual service resource replacement time. The method proposed in [6] deals with three key challenges:

- a virtual service resource replacement from one virtual node to another;
- a tree topology redesign when the network topology change after a node is added into or removed from a virtual network with tree topology;
- the replacement of 2^n virtual service resources.

However, the solutions proposed in [6] have the following limitations:

- no replacement solution when all of the leaf nodes of the tree topology have the same traffic weight;
- too much virtual service replacement depending on traffic variations;
- the service is not replaced by the initial node when traffic decreases or returns to normal. The replacement had been done to face a traffic change in the network. The virtual service should be replaced by its initial virtual node to satisfy the QoS when the network traffic returns to normal;
- the migration strategy (pre-copy, post-copy, hybridization) of the virtual service from one node to another is not studied or discussed. The service migration time is significant and greatly influences the QoS ([8], [9]) of the virtual networks when we are dealing with virtual network resource replacement.

1.2. Our Contribution

The objective of this paper is to improve the QoS of virtual networks when the traffic changes by providing solutions to the limits cited above concerning the approach proposed by Horiuchi and Tachibana [6]. Our contribution is summarized in four points:

- a selection strategy of a QoS-satisfying node when all of the leaf nodes in the tree topology have the same traffic weight. This selection is based on the amount of traffic of the parent node. The selected leaf node is one whose parent node has the largest traffic weight;
- avoid automatic virtual resource replacement at the first sign of QoS dissatisfaction. A bad QoS state can be temporary. In this case, a virtual service replacement is not necessarily required. Therefore, we define a QoS dissatisfaction delay as one that is beyond the virtual service resource replacement that is performed;
- automatic replacement of the virtual service to its initial host node after solving the problem that is caused by traffic variation. This allows a better use of the virtual network resources;
- a virtual service migration strategy from the source node to the destination node. We propose a hybrid migration approach because it has the advantage of minimal downtime and data copying as well as the retention of the replaced service data consistency ([8], [10]).

The remainder of this paper is organized as follows: In Section 2, we describe in detail the approach of [6] with its drawbacks. Section 3 presents our contribution in more detail. The simulation results and discussions are presented in Section 4. Finally, Section 5 concludes the paper.

2. Presentation of Former Work

In this section, we describe the Horiuchi and Tachibana approach [6] with its drawbacks.

2.1. The Virtual Service Resource Replacement Approach of Horiuchi and Tachibana and its Drawbacks

Consider a virtual network $G = (N, L)$ as a tree, where $|N|$ denotes the number of nodes and $|L|$ the number of links. Each node n_i has a quantity of resources m_i , and each link l_{ij} between the nodes n_i and n_j has a traffic weight m_{ij} . The variables γ_i and γ_{ij} are the amount of traffic passing through the node n_i and link l_{ij} , respectively. The idea is to select in the tree, the node of which allows the management of the greatest amount of traffic in the topology. The search algorithm starts by selecting the leaf node with the minimum amount of traffic; then, the weight of this traffic is added to that of its parent node in the tree. After that, this leaf node is logically removed from the research process. This process is repeated until the tree contains two nodes; in this case, the new node to select for the virtual service replacement is the one with the greatest traffic weight. An illustration of this selection approach is presented in Fig. 2. At the end of the search process for the case of Fig. 2, the remaining nodes are n_1 and n_2 . The node n_1 is selected as the new node for the virtual service since its amount of traffic is 40 compared to n_2 which has an amount of traffic of 35.

The first limit of this approach is that there is no replacement solution when all of the leaf nodes of the tree

have the same traffic weight. In this case, the algorithm fails from the beginning, and no result is produced at the end of the search process. Fig. 3 illustrates this limit. All of the leaf nodes F, E and C have a current traffic weight that is equal to 12. Which one should be selected? The approach of Horiuchi and Tachibana ([6]) does not answer this question.

The second limit is that there are too many virtual service replacements, depending on the traffic variations. Each time the QoS is not satisfied because the amount of traffic changes, a virtual service replacement is performed. These multiple replacements create instability in the offered network service as well as a load imbalance in the nodes of the virtual network.

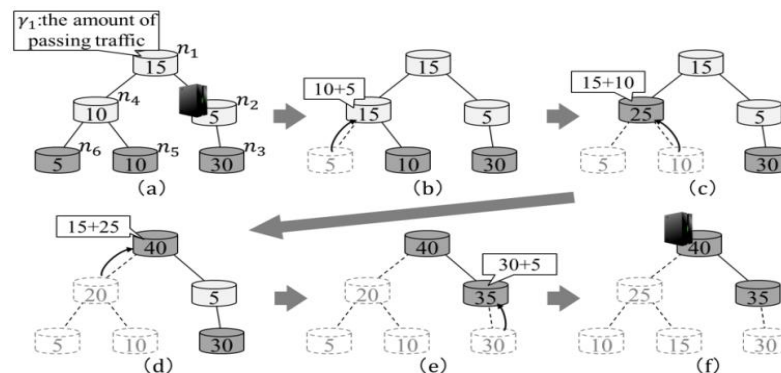


Fig. 2. The virtual service resource replacement method of Horiuchi and Tachibana.

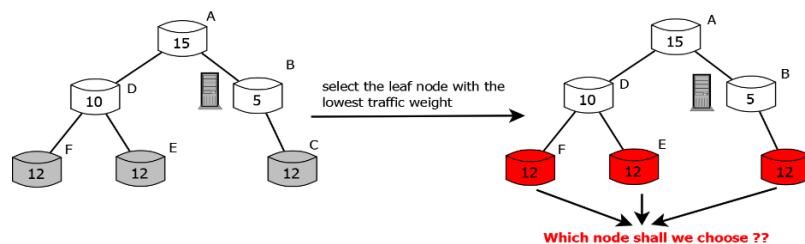


Fig. 3. Limit 1: The equal traffic weight challenge.

Third, the virtual service resource should be replaced with its original virtual node once the amount of traffic has returned to normal. The virtual service replacement in another node leads to the use of the resources of this node that are planned in principle for a specific service. The exploitation of these resources in an unexpected way can create some disturbances in the service proposed by the node; thus, it is necessary to temporarily use the resources to manage the traffic changing and then replace the service in its starting node. This would avoid overusing the resources of some nodes. This returning aspect of the virtual service to its original location after replacing the service is not considered by Horiuchi and Tachibana.

Finally, a service migration strategy is not discussed. A virtual service resource replacement involves the transfer of a data set from one virtual machine to another; these data are characteristic of the state of the migrated service so that it is restored more efficiently in the new physical or virtual node that will receive it. These characteristics are: the amount of volatile memory required, the non-volatile memory, the status of the virtual CPUs, the status of the connected devices and the status of the active connections [9]. It takes time for these elements that characterize the service to be routed from a virtual node to another, and this process requires the implementation of an adapted migration approach to maintain the consistency of the service provided to end users. Depending on the case, it can be considered as a migration of the service files or a migration of the operating system hosting the virtual service. In both cases, the migration time affects the service availability. Therefore, this migration time should be considered and minimized.

2.2. Dynamic Replacement of 2^n Virtual Service Resources of Horiuchi and Tachibana and its Drawbacks

The objective is to replace multiple virtual service resources in a virtual network. This is an extension of the previous approach that was proposed for one virtual service resource replacement.

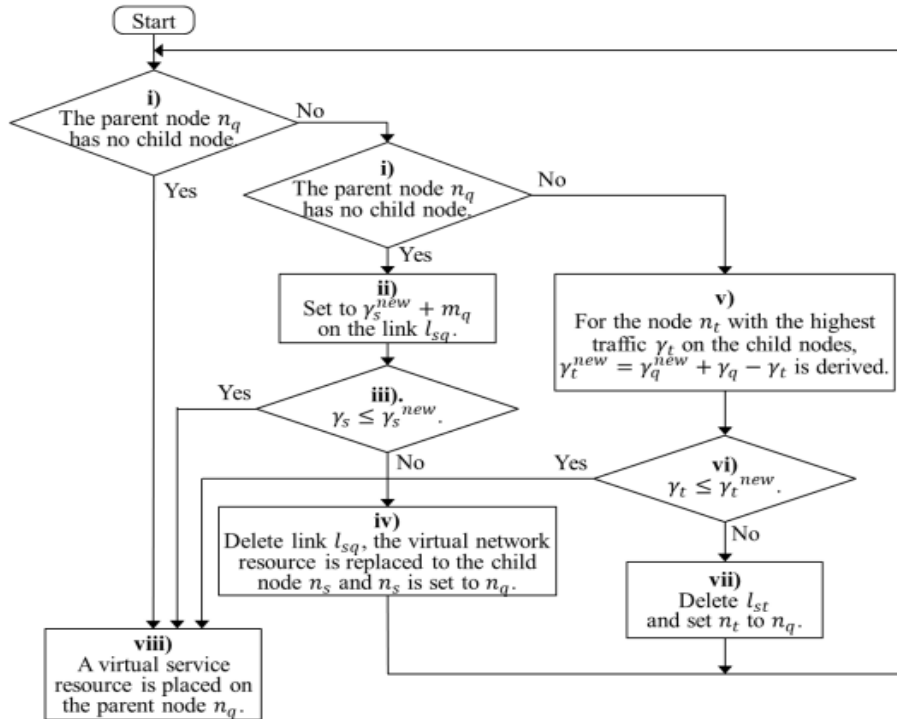


Fig. 4. Flowchart for 2^n virtual service resource placement.

A virtual network is logically divided into two virtual networks according to the traffic weight, using the algorithm of section 2.1; in each subdivided virtual network, a virtual service resource is placed using the flowchart in Fig. 4, where γ_i is the amount of traffic in node n_i and γ_i' is the traffic variation of node n_i . According to this diagram, the node that is chosen as the host of the virtual service resource is the one that has enough resources to support the amount of traffic that is generated by all its children nodes. The virtual service resource in each subdivided virtual network offers the same virtual service as the original virtual network and is replaced according to the method described in Section 2.1.

For example, considering Fig. 5, we get a network with the two extreme nodes n_4 and n_6 by applying the algorithm that was described in Section 2.1. These two nodes are the root nodes of the sub-trees graph A and graph B, which represent segments of the original virtual network.

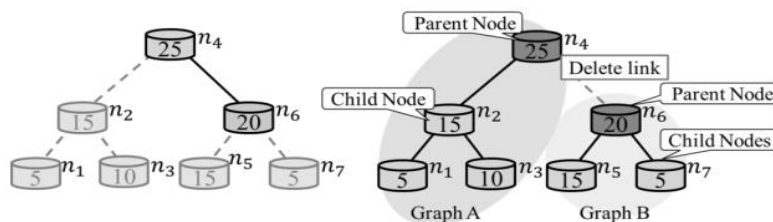


Fig. 5. Segmentation of the virtual network into two network segments.

The subdivision into sub virtual networks helps to reduce the traffic load of the original virtual node that hosted the amount of traffic for the proposed service.

The first drawback of this extended replacement approach is that there is no replacement solution for a number of virtual service resources $R \neq 2^n$. The dynamic replacement approach of [6] uses the subdivision of the original virtual network into 2^n segments according to the amount of traffic to perform a dynamic replacement of 2^n virtual service resources. This assumes that any traffic weight can be satisfied through at most 2^n virtual service resources, but this is not always verified. Thus, for any traffic weight requiring subdivision into an odd number of sub virtual networks where $R \neq 1$, Horiuchi and Tachibana [6] does not provide a solution.

Second, the heterogeneity of services is not considered in the 2^n virtual service resource replacement approach of [6]. User-generated traffic can involve several different services that are distributed across one or more virtual networks. In this case, replacing a virtual service resource can lead to the replacement of another dependent one, based on the type of traffic that is generated. A user can be simultaneously connected on a Video on Demand (VoD) service and a Voice over IP (VoIP) service. In [6], the authors consider a single virtual network service and not the dependency between multiple services.

3. Improvements of the Horiuchi and Tachibana Approach

In this section, we provide some solutions to the drawbacks of the virtual resource replacement approach in [6]. These solutions address the problems of equal traffic weight in the leaf nodes of the tree topology, automatic replacement and the establishment of a data migration mechanism of the virtual service.

3.1. Our Solutions to the Equal Traffic Weight Challenge in the Leaf Nodes

To address this problem with the tree topology, we can proceed as follows: select the leaf node whose parent node has the largest traffic weight. From here, two scenarios can arise:

If the direct parent nodes have equal traffic weights, we go back in the tree until we find a level where the traffic weights of the nodes are different. Once this level is reached, one of the leaf nodes of the node with the largest traffic weight is selected; then, we add the traffic weight of this leaf node to that of its parent node. We remove the previously selected leaf node from the research process and repeat this process until there are only two nodes in the tree. Finally, the node that is selected as new location for the virtual service resource is the one with the highest

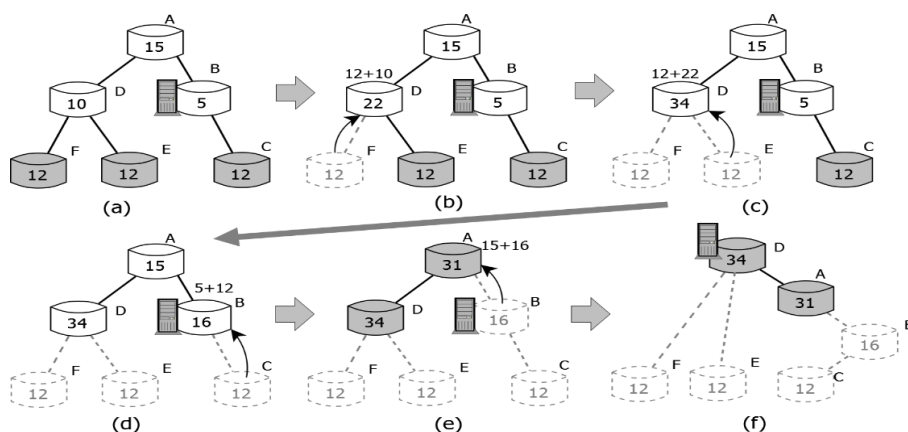


Fig. 6. First case for the problem of equal traffic weight in the leaf nodes with parents of different traffic weight.

If we have a tree with a root node followed directly by the leaf nodes (with equal traffic weights), we randomly select any leaf node (see Fig. 7). Generally, if the tree topology height is $h=1$, then any leaf node can be selected.

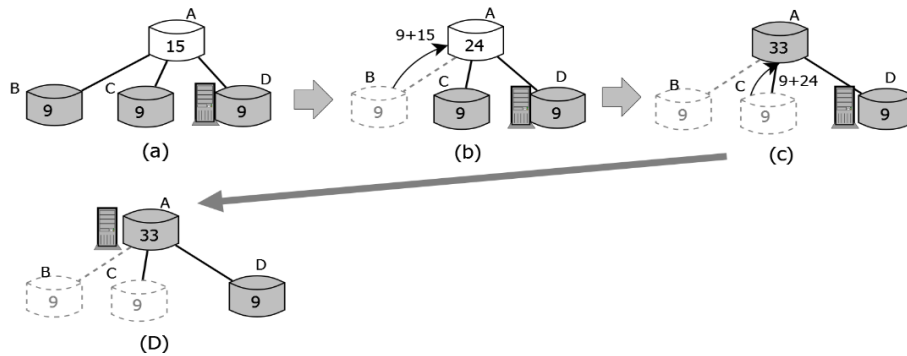


Fig. 7. Second case for the problem of equal traffic weight in the leaf nodes with tree topology height $h = 1$.

3.2. A Solution to Reduce the Number of Replacements

In [6], when the quality of service is not satisfied because of traffic change, a virtual service resource replacement is immediately performed; this replacement, which aims to improve the quality of service, may deteriorate it further because of the time that is needed to effectively replace a virtual service resource and the service disruptions in the network [11]. If multiple replacements are done simultaneously, the QoS will be greatly affected. Thus, a virtual service resource replacement approach is necessary, but only in the case of emergency. We define a delay beyond the replacement process that is effectively initiated. The case of the virtual service replacement emergency could be defined through some criteria that we mention here:

- *The type of service offered*: Depending on the services that are offered by the virtual networks, their resources and flow management policies requirements, we must consider their nature and the priorities between these services [4]. Thus, when the amount of traffic changes, the immediate reaction time is relative to the nature of the service that is involved. In addition, certain services such as VOIP and VOD are less tolerant to latency times than others; consequently, these services require a reaction time that is relatively fast in situations of failure.
- *The quantity of available resources in the considered virtual environment*: It can be possible that when we want to replace a virtual service resource, we do not find a node that is able to control the amount of traffic involved; therefore, it is necessary to wait before performing a replacement in this situation.

The reaction time that is set up to address this improvement path would avoid unnecessary replacements due to temporary traffic overweight. This reaction time must be a network setting that is defined by the administrator based on the type of service.

3.3. The Data Migration Mechanism

There are several mechanisms for migrating virtual entities from one node to another [12]:

- *Cold migration (or stop-and-copy)* [11]: In this approach, the service is first stopped, and then its data are copied to the new node; once the copy is complete in the destination node, the service is started in that node. The main advantage of this method is that it ensures the faultless migration of the server memory. In addition, the state of the server memory is not changed on the source host. On the other hand, the downtime and start times on the destination node are longer.
- *Live migration* [11], [13]: There are three approaches of live migration: pre-copy, post-copy and hybrid post-copy.

The pre-copy approach (see Fig. 8) mainly consists of transmitting all of the virtual service resource memory pages to the destination host while the service is running. After a given copy level, the service is stopped on the source host and the rest of the modified memory pages are copied to the destination host. The major problem occurs when the maximum interruption time is too low for sending the last modified

pages to the destination node [9]. In this case the migration time can be very high.

In contrast to the pre-copy, the post-copy (see Fig. 9) starts with the immediate shutdown of the virtual machine on the source host. Then, the subsidiary data are transferred before activating the service on the destination host. The major disadvantage compared to the pre-copy is based on the robustness of this approach. Indeed, since the VM is directly awakened on the destination host in an inconsistent state, the failure of one of the nodes, source or destination during migration causes the inevitable loss of the integrity of the VM memory state.

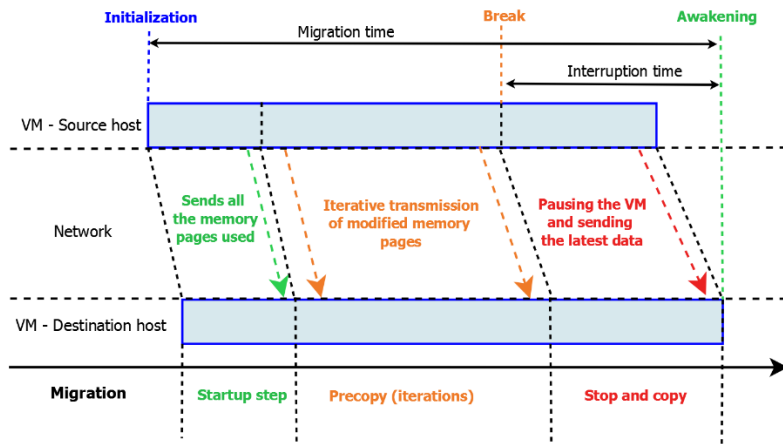


Fig. 8. Pre-copy algorithm.

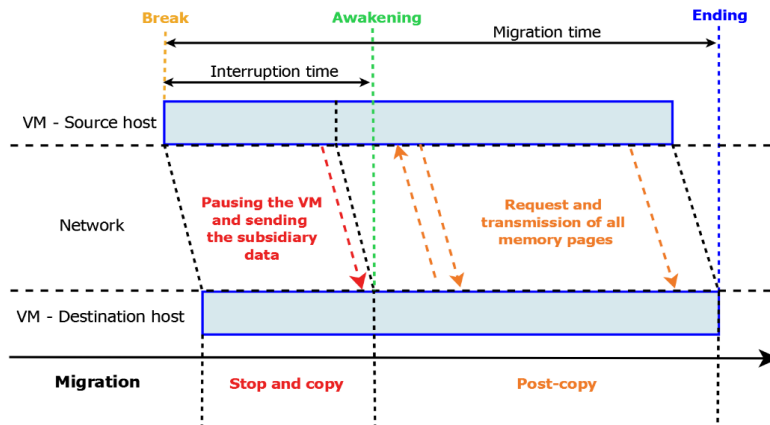


Fig. 9. Post-copy algorithm.

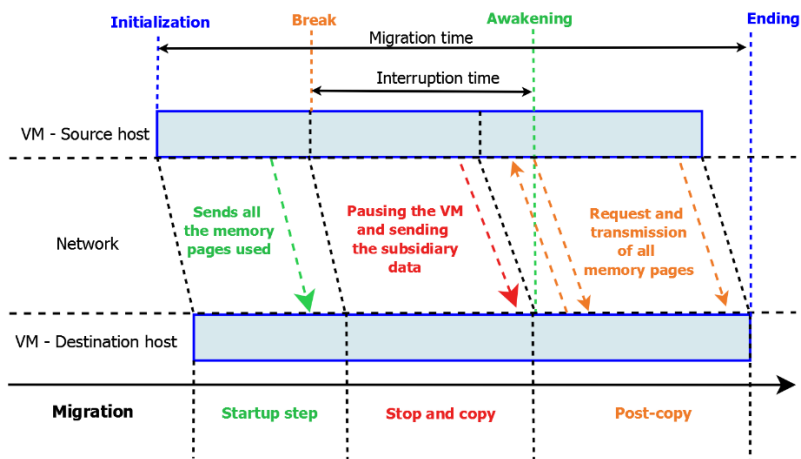


Fig. 10. Hybrid post-copy algorithm.

Hybrid post-copy [10] has been proposed to reduce post-copy performance issues. Fig. 10 shows the hybrid post-copy algorithm. Without interrupting the outstanding service, the progressively modified memory pages are transferred from the source node to the destination (pre-copy). The copy of the modified memory pages ends when a critical point is reached; therefore, the service is suspended on the source machine and its state is restored on the destination machine with consistent data (post-copy). We propose this hybrid migration approach, which substantially reduces both the downtime and the total migration time.

4. Simulations and Discussions

The objective of the simulations that follow is to evaluate the performances of our replacement approach, which is an improvement over the Horiuchi and Tachibana one. We compare our approach with the Horiuchi and Tachibana version. The performances are evaluated through the impact of the number of replacements on the overall QoS as well as the migration time of the services between a source node and a destination node. In this section, we present these performance results from the OMNET ++ simulator. The simulations were conducted on several network samples: Network1 is a small network (20 nodes and 31 links), and Network2 is a large network (60 nodes and 90 links). The aim of the tests on the different sized networks is to evaluate the effectiveness of both approaches according to the scalability of the network. The structure and resource weights of each network were randomly generated to show the adaptation of the tree topology construction strategy to any type of traffic weight within the nodes.

4.1. The Impact of the Number of Virtual Service Resource Replacements on the QoS

In this section, we present the influence of the number of replacements on the overall QoS in a network. Indeed, we have proposed in the improved version of the Horiuchi and Tachibana [6] approach that the number of replacements can be reduced by using a replacement starting delay k . For each network, we performed some tests for different replacement initialization delays: $k = 2$ sec, $k = 6$ sec and $k = 10$ sec. Fig. 11 and Fig. 12 illustrate the results of this study for networks of 20 and 60 nodes. In a small network (see Fig. 11), we found that the replacement rate is relatively low compared to that of large networks (see Fig. 12). This observation is reflected in many similarities in the number of replacements that are made after each variation of the replacement initialization delay k . This can be explained by the fact that a large number of nodes also induces a significant amount of traffic [6]. In Network2, there is almost no such similarity. This means that, as the number of nodes is high, many replacements are possible, and among them, many can be avoided. In all cases, these simulations show that our replacement approach, which imposes a replacement initialization delay, is better than that of Horiuchi and Tachibana, which performs many unnecessary replacements and has high amount of downtime compared to our approach.

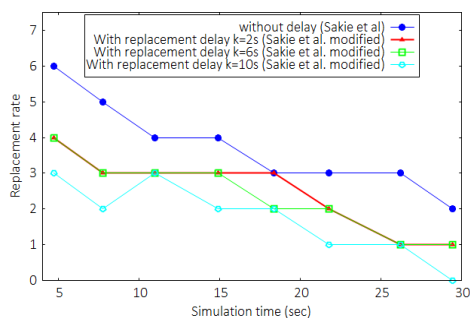


Fig. 11. Replacement rate for a 20-node network.

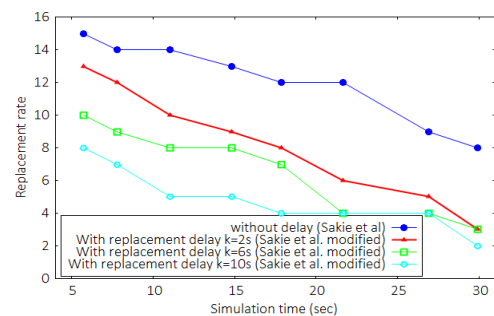


Fig. 12. Replacement rate for a 60-node network.

Concerning the reduction in the amount of traffic, Fig. 13 and Fig. 14 give a comparison of the traffic variation management method of Horiuchi and Tachibana with its modified version (Horiuchi and Tachibana

modified) that we proposed for Network1 (see Fig. 13) and Network2 (see Fig. 14). Both methods have almost the same traffic reduction rate, whether in a small or large network. Nevertheless, there are some differences, like the fact that the Horiuchi and Tachibana approach reduces the amount of traffic more than our approach. This difference is explained by the fact that, because of the replacement initialization delay that we impose, some replacements are not performed. In this case, the traffic weight is not greatly affected

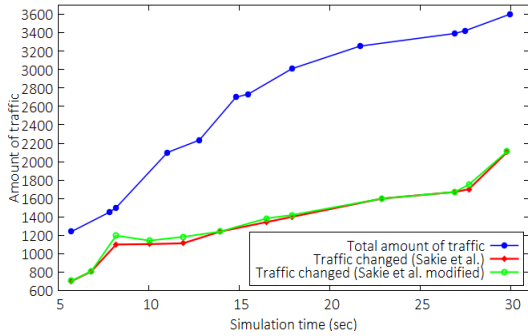


Fig. 13. Total amount of traffic variation for Network1.

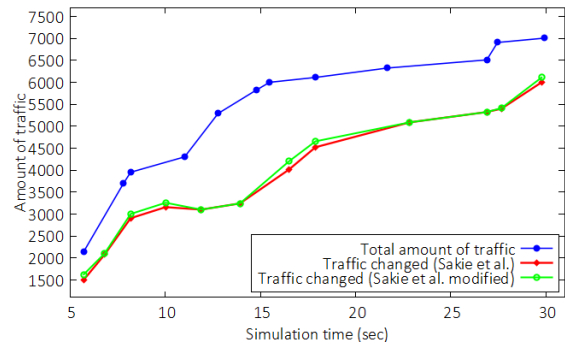


Fig. 14. Total amount of traffic variation for Network2.

We can also see that the traffic reduction rate is not very high in a large network compared to that of small networks. This can be explained by the difficulty to control all of the network traffic when the number of nodes and network users is important.

4.2. The Impact of the Number of Virtual Service Resource Replacements on the QoS

To measure the impact of migration time on QoS, we observed during the simulations the migration rate and the average time of migrations over a simulation time interval of 0 to 30 seconds. We compare our approach, which integrates migration, with that of Horiuchi and Tachibana, which does not consider it. Fig. 15 and Fig. 16 show that the average migration time of Horiuchi and Tachibana is lower than ours. This is explained by the fact that for each migration, the method of Horiuchi and Tachibana transfers all of the data on time, without a stop-and-copy step, which reduces the total migration time. Nevertheless, as we have shown previously, this method is not very realistic. This observation is the same in large networks (see Fig. 16).

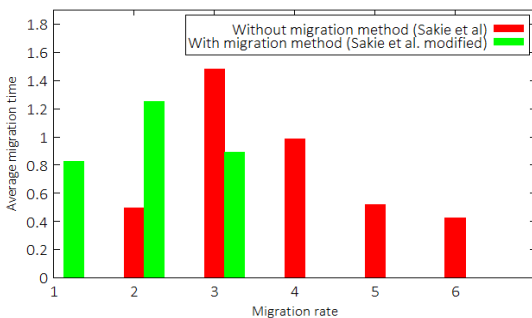


Fig. 15. Total amount of traffic variation for Network 1.

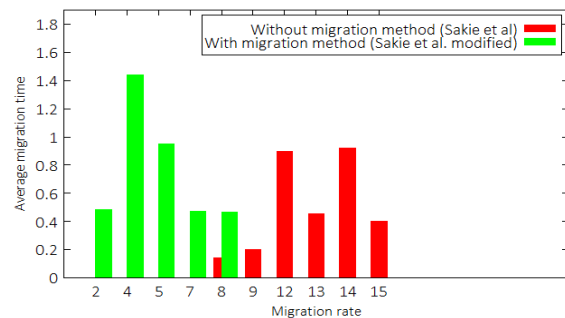


Fig. 16. Total amount of traffic variation for Network 2.

Based on the preceding results, we can conclude that the virtual resource replacement approach that we have proposed certainly has many points of similarity with the Horiuchi and Tachibana one, but it greatly improves upon it by integrating elements to better appreciate the replacement effects on QoS in a real virtual

network.

5. Conclusion

Our aim in this paper was to improve the QoS of virtual networks against traffic changes due to user mobility. We proposed a dynamic virtual resource replacement approach that helps to reduce the amount of traffic load in the nodes. This proposed approach is an improvement of the Horiuchi and Tachibana approach. It has the advantage of avoiding replacements due to temporary traffic changes that do not disturb the QoS for a long time. In addition, our approach preserves the consistent state of the replaced virtual service resource when it is restored to the destination host, using a data migration technique that we proposed. The numerical results of our simulations helped to better demonstrate the impact of these advantages on the QoS in the virtual network. Therefore, it appears that our replacement approach is better than that of Horiuchi and Tachibana and is closer to reality. However, the virtual service resource replacement idea could be just as useful in other research fields with the same QoS issues. For example, in the IoT (Internet of Things), equipment resources (storage space, processing power, energy) are generally limited. Consequently, there is a permanent need for resources from the equipment in this field to ensure QoS. Then, it would be interesting for the future work to deal with this dynamic virtual service resource replacement problem in the field of the IoT.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

JFM suggested this work and gave the main idea. YFY carried out analysis and experimentations. VKT and YFY wrote the first draft of the paper. JFM revised the first draft to get the final submitted manuscript. In addition, all authors read and approved the final version of the work.

References

- [1] Fernandes, N. C., Moreira, M. D., Moraes, I. M., Ferraz, L. H. G., Couto, R. S., Carvalho, H. E., *et al.* (2011). Virtual networks: Isolation, performance, and trends. *Annales des télécommunications*, 66(5-6), 339-355.
- [2] Chowdhury, N. M. K., & Boutaba, R. (2009). Network virtualization: state of the art and research challenges. *IEEE Communications magazine*, 47(7), 20-26.
- [3] Pham, T. S., Lattmann, J., Lutton, J. L., Valeyre, L., Carlier, J., & Nace, D. (2012). A restoration scheme for virtual networks using switches. *2012 IV International Congress on Ultra Modern Telecommunications and Control System* (pp. 800-805). IEEE.
- [4] Seddiki, M. S. (2015). *Allocation Dynamique des Ressources et Gestion de la Qualité de Service Dans la Virtualisation des Réseaux*. Doctoral dissertation, Université de Lorraine.
- [5] Shahriar, N., Ahmed, R., Chowdhury, S. R., Khan, A., Boutaba, R., & Mitra, J. (2017). Generalized recovery from node failure in virtual network embedding. *IEEE Transactions on Network and Service Management*, 14(2), 261-274.
- [6] Horiuchi, S., & Tachibana, T. (2018). Dynamic Replacement of virtual service resources based on tree topology for mobile users in virtual networks. *Journal of Computers*, 13(12), 1335-1349.
- [7] Koyanagi, Y., & Tachibana, T. (2014). *Dynamic Resource Allocation Based on Amount of Traffic in Virtual Networks* (Report Vol. 113, No. 473). Miyazaki, Japan: IEICE Technical Report.
- [8] Liu, J., Li, Y., & Jin, D. (2015). SDN-based live VM migration across datacenters. *ACM SIGCOMM Computer Communication Review*, 44(4), 583-584.
- [9] Kherbache, V. (2016). *Ordonnancement des Migrations à chaud de Machines Virtuelles*. Doctoral

dissertation, Université Côte d'Azur.

- [10] Luo, Y., Zhang, B., Wang, X., Wang, Z., Sun, Y., & Chen, H. (2008). Live and incremental whole-system migration of virtual machines using block-bitmap. *Proceedings of the 2008 IEEE International Conference on Cluster Computing* (pp. 99-106). IEEE.
- [11] Keshavamurthy, U., & Guruprasad, H. S. (2015). VM migration: A survey. *Global Journal of Engineering Science and Researches*, 2, 219-224.
- [12] Venkatesha, S., Sadhu, S., & Kintali, S. (2009). Survey of virtual machine migration techniques. *Memory*.
- [13] Clark, C., Fraser, K., Hand, S., Hansen, J. G., Jul, E., Limpach, C., & Warfield, A. (2005). Live migration of virtual machines. *Proceedings of the 2nd Conference on Symposium on Networked Systems Design & Implementation-Volume 2* (pp. 273-286). USENIX Association.

Copyright © 2020 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).



Jean Frédéric Myoupo is a professor of computer science in the University of Picardie-Jules Verne (UPJV), Amiens, France, where he leads the parallel and mobile computing research group in the Computer Science Lab-MIS. Professor Myoupo is the former dean of the Faculty of Mathematics and Computer Science of UPJV from 1999 to 2002. He received his Ph.D in applied mathematics from the Paul Sabatier University of Toulouse, France, in 1983 and his habilitation in computer science from the University of Paris 11, Orsay, France, in 1994. He held many positions in mathematics and computer science departments such as lecturer or associate professor in different universities, such as the University of Sherbrooke, Québec, Canada, the University of Yaounde, Cameroon, the University of Paris 11, Orsay, France and the University of Rouen, France. Professor Myoupo has served as member of program committee of international conferences, and he has been associate editor or member of editorial board of many international journals in computer science. His current research interests include parallel algorithms and architectures, and mobile computing and network management.



Yannick Florian Yankam is a Ph.D student at the University of Dschang, Dschang, Cameroon. He received his master degree in networks and distributed services from the Faculty of Science, University of Dschang, in 2016. His current research interests include computer networks, network virtualization, quality of service in virtual networks and cloud computing.



Vianney Kengne Tchendji is a senior lecturer of computer science at the University of Dschang, Dschang, Cameroon. He received his Ph.D in computer science from the University of Picardie-Jules Verne, Amiens, France, in June 2014. His current research interests include network virtualization, parallel algorithms and architectures, scheduling, wireless communication and ad hoc networking.