2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)

A Virtual Network Embedding Algorithm Based on RBF Neural Network

Hui Zhang[1,2], Xiangwei Zheng[1,2], Jie Tian[1,3], Qingshui Xue[4]
[1](School of Information Science and Engineering, Shandong Normal University, Jinan 250014, China)
[2] (Shandong Provincial Key Laboratory for Distributed Computer Software Novel Technology, Jinan 250014, China)
[3](Shandong Provincial Key Laboratory of Computer Networks, Shandong Computer Science Center (National Supercomputer Center), Jinan 250014, China)
[4](School of Computer Science and Information Engineering, Shanghai Institute of Technology, Shanghai, 201418, China)
E-mail: xwzhengcn@163.com

***Abstract*—With the emergence of network virtualization, the infrastructure can be effectively integrated to overcome the "ossification" of the Internet. The biggest challenge in network virtualization is the problem of virtual network embedding. Unfortunately, most of the researches on virtual network embedding only focus on static algorithms, which allocate fixed or invariable resources to virtual networks until the end of their lifetime. However, the demand of virtual requests for resources are dynamically changing and fluctuating in reality. Therefore, the traditional static schemes not only greatly reduce the utilization of substrate resources, but also decrease the revenue of the service providers. In this paper, we aim to satisfy the dynamic requirement of resources for virtual networks. We propose a dynamic embedding algorithm (RBF-VNE) which is based on RBF neural network to learn and predict the dynamic changes of resources, and then we dynamically adjust and allocate resources according to the predicted results. Simulation results show that our approach can well integrate substrate resources to solve above problems and performs well than static embedding algorithms.**

Keywords-network virtualization; virtual network embedding; static embedding algorithms; RBF neural network; dynamic resource allocation.

I. INTRODUCTION

Network virtualization [1] has been considered as a long-term solution to overcome the problem of the network "ossification"[2]. Under the environment of network virtualization, network service providers (SP) can build a number of virtual networks with different characteristics by using substrate resources rented from multiple infrastructure providers (IP). Then service providers can provide end-to-end differentiated services for end users.

One of the major challenge of network virtualization is known as the virtual network embedding (VNE) [3-4], which deals with efficient embedding of virtual networks (VN) with resource constraints onto the substrate network (SN). It is proved that the restricted VNE is NP-Hard [5]. How to reasonably allocate limited substrate resources and accept more virtual requests have become the main purpose of the research on VNE.

Unfortunately, most of current researches usually adopt the static two-stage [6-7] heuristic embedding methods, where the VN requests are known in advance and the demand for resources are unchanged over time. The VN requests will hold a fixed resources until the end of their lifetime in static embedding process. However, the resources requirement of VN requests represent fluctuating and changing characteristics in reality. Therefore, these traditional schemes cannot fully manage and mobilize the entire SN resources, which will lead to a waste of substrate resources and the low acceptance rate of VNE (as shown in Figure 1). As the existing literatures are limited to the traditional algorithms, even the most efficient static embedding algorithms cannot adapt to the dynamic changes in VNE [8]. It is exceedingly necessary to dynamically adjust the allocation of resources according to actual demand of VNs.

Research shows that many intelligent algorithms have been used to solve all kinds of problems, such as dynamic resources management. In literature [9], artificial neural networks and fuzzy systems have been applied to solve the problem of resources allocation. In order to improve the utilization of substrate resources and increase the revenue of operators, we proposed the RBF-VNE algorithm based on RBF neural network to adjust and allocate the resources dynamically in this paper.

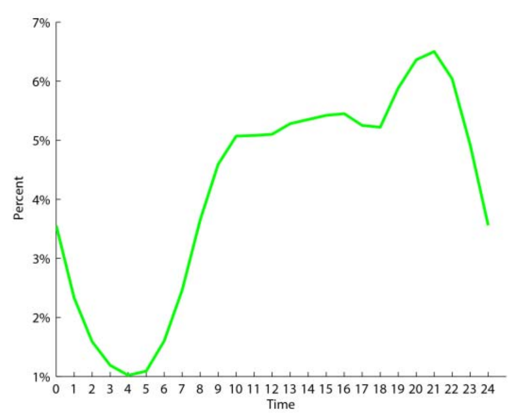

Figure 1. The distribution of the users' online time

978-1-5386-3221-5/17 $31.00 © 2017 IEEE
DOI 10.1109/CSE-EUC.2017.77

The remainder of this paper is organized as follows. The problem description is introduced in Section II. Section III elaborates the RBF-VNE algorithm under the environment of dynamic demands. Section IV shows the setup of the simulation and the performance analysis. Finally, Section V summarizes the paper and expounds the further research plan.

II. PROBLEM DESCRIPTON

A. Virtual network embedding

We model the SN as a weighted undirected graph $G_S = (N_S, E_S, C_S^N, B_S^E)$, where N_S and E_S denote the set of substrate nodes and paths respectively. The symbol C_S^N and B_S^E denote the attributes of substrate nodes (e.g. CPU) and substrate links (e.g. bandwidth).

Similarly, a VN request can also be shaped as a weighted undirected graph $G_V = (N_V, E_V, C_V^N, B_V^E)$, where N_V and E_V represents the set of virtual nodes and virtual links. Virtual nodes and links are associated with recourse constraints, which is denoted by C_V^N and B_V^E.

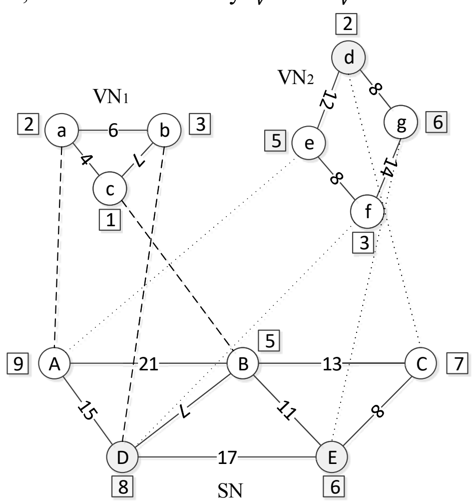

Figure 2. An example of VNE

The problem of VNE is to embed VNs onto a subset of SN (as shown in Figure 2), which is denoted by $M \in G_V(N_V, E_V) \to G_S(N_S, E_S)$. The process of embedding the virtual nodes to a subset of substrate nodes is called node embedding, which is defined as $M^N \in G_V(N_V, C_V^N) \to G_S(N_S, C_S^N)$. Similarly, the link embedding is denoted as $M^E \in G_V(E_V, B_V^E) \to G_S(E_S, B_S^E)$ .During the embedding process, the scheme must satisfy the resource constraints of virtual nodes and virtual links.

B. Evaluation indexes

In this paper, we use the following metrics to evaluate the performance of the experiments.

①Acceptance rate: AR $= \sum_t^T (G_V \to G_S)/\sum_t^T G_V$.

$\sum_t^T (G_V \to G_S)$ represents the number of VNs successfully embedded from t to T. $\sum_t^T G_V$ is the total number of arrived VNs .

②Average cost: AC $= \sum_t^T Cost(G_V \to G_S)/\sum_t^T G_V$.

$\sum_t^T Cost(G_V \to G_S)$is the total mapping cost from t to T, which includes the cost of node embedding and link embedding.

③Long-run average revenue: LR $= \sum_t^T Rev(G_V \to G_S)/T$.

$\sum_t^T Rev(G_V \to G_S)$ represents the total revenue obtained from successful embedding of VNs, which includes node embedding revenue and link embedding revenue.

III. DYNAMIC RBF-VNE ALGORITHM

The traditional static embedding algorithms always take each virtual request as a processing unit. That is to say, it will allocate requested resources to the VNs according to the arrived sequence and recycle resources until the end of VN requests. These schemes cannot fully manage and mobilize the entire SN resources, which will lead to excessive allocation of substrate resources and rejection of the forthcoming requests due to insufficient resources.

In this paper, we study the VNs with dynamic resources requirement and handle these requests according to the time series. We used a RBF-VNE algorithm based on RBF neural network to learn and predict the dynamic changes of VNs, and then we recycle and redistribute resources at every moment of the embedding process. The specific algorithm and flow chart of RBF-VNE is shown as follows:

Algorithm of RBF-VNE

Generate a arrived queue of VNs.
Take the time of the first arrived VN as the beginning time of the queue;
Calculate the end time of all requests and take the maximum value as the end time of the queue.
For (Every moment of queue)
  If (The queue of VNs is not processed)
   If (The current time is equal to arrival time of VNs)
     Take this virtual request from the queue
     If (Meet the resource constraint of this VN)
       Embed the VN and allocate the initial demand of resources to it;
       Dynamically change required resource of the VN and record its actual resource requirements at this moment.
     Else Reject this VN.
   Else
     Allocate resources to this VN according to the original plan and calculate the actual required resources at this moment.
     Take actual required resources of this VN in the first few moments as the training samples. Initialize parameters of RBF neural network;
     Train and optimize RBF neural network according to the algorithm 1. Get the prediction model;
     Use the learned model to predict required resources of this VN for next time;
     Adjust predicted results to meet the requirements of resources.
     Recover the resources allocated to this VN last moment, and assign it to the resources at the next moment according to the adjusted results.
End If

22: **End For**

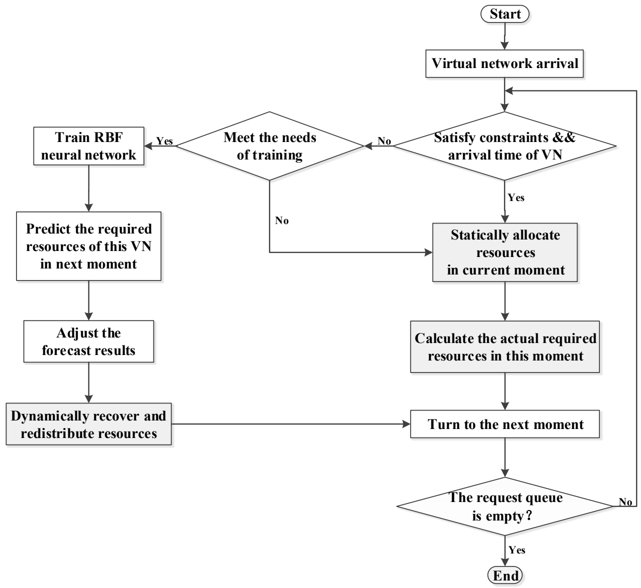

Figure 3. The flow chart of RBF-VNE

IV. EXPERIMENTAL ANALYSIS

A. Experiment Setting and Parameters

In the simulation experiment, we used the function $G(t) = 0.5 * R(1 + sin(0.01 * t) * sin(0.03 * t)) + e_t$ to simulate the actual resources requirement of VNs. e_t is the noise of normal distribution (mean value=0, standard deviation=0.1), R represents initial required resources of virtual nodes and virtual links and t represents a moment in the entire lifetime T of a VN. We used *k-means* clustering method to calculate the centers and widths of neurons and we took the first 10 training samples as the initial cluster centers. The initial number of hidden nodes M=10 and the threshold ε=0.01.

B. Performance evaluation

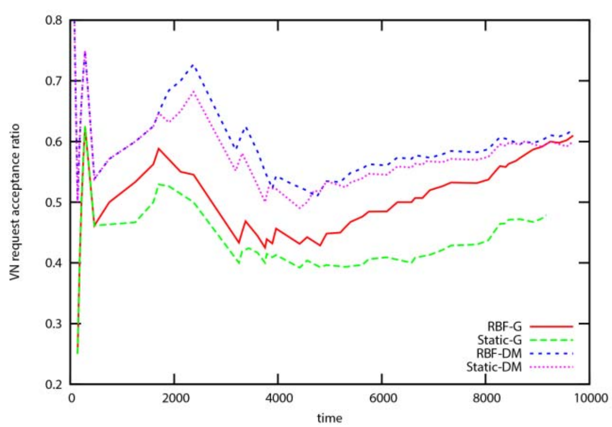

Figure 4. Acceptance rate

Figure 4 is the comparison of different algorithms in the VN acceptance rate (AR). As can be seen from the graph, whether the dynamic RBF-G (RBF Greedy) algorithm or RBF-DM (RBF D-Vine MCF) [10] algorithm are better than the static Static-G (static Greedy) algorithm and Static-DM (Static D-Vine MCF) algorithm respectively.

In the prior moments of the embedding, dynamic RBF-VNE algorithms have the same performance in the AR as corresponding static algorithms due to a small number of arrived VNs and abundant underlying substrate resources. With the increase of virtual requests, the substrate resources are gradually decreasing. Since the RBF-VNE algorithms can recover and redistribute the resources more accurately at every moment of the embedding process, the SN under the dynamic RBF-VNE algorithms always has more available resources. Therefore, the RBF-VNE algorithms are better than the static embedding algorithms in the AR in later periods. Due to the advent of virtual requests become more and more and the virtual requests successfully embedded have long-term possession of resources, which result in the shortage of resources and the upcoming VNs rejected. Therefore, the AR of dynamic algorithms and static algorithms decreased in some periods. However, because of the advantages of the dynamic algorithms, the performance of the RBF-VNE are always better than the corresponding static embedding algorithms.

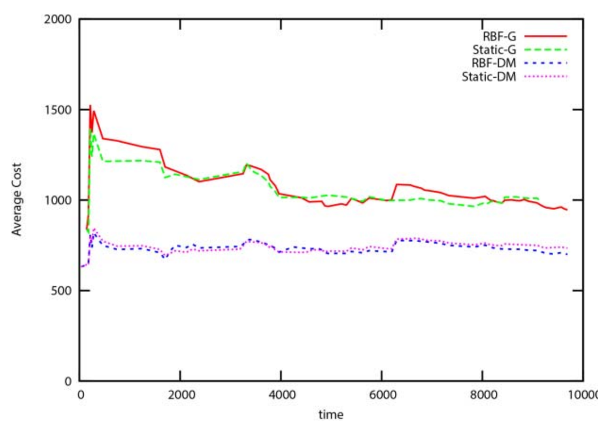

Figure 5. Average cost

Figure 5 shows the differences on the average cost (AC) of the algorithms. Since the dynamic algorithms and static algorithms have similar trends in AR in the previous periods, the performance of AC between the corresponding algorithms are similar too. Although the RBF-G algorithm and the RBF-DM algorithm enhance the management of the pool of substrate resources, the schemes increase the cost of due to dynamic treatments. The higher the AR, the higher the cost of the embedding. Therefore the AC of RBF-VNE algorithms may be higher than the corresponding static algorithms, which is also the reason that the curves of RBF-G and RBF-M become higher at some moments. From the graph we can also find that the curves of RBF-G algorithm and Static-G algorithm are close to each other, while the trends of RBF-DM algorithm and Static-DM algorithm have a big difference relatively. The reason is that the original static D-Vine algorithm is more complex than static Greedy algorithm, and the implementation of RBF-DM algorithm will be more complex too. But from the overall point of view, the trends of dynamic algorithm and static algorithm (especially the RBF-G algorithm and the Static-G algorithm) are close in AC, which also can be within the scope of our acceptance.

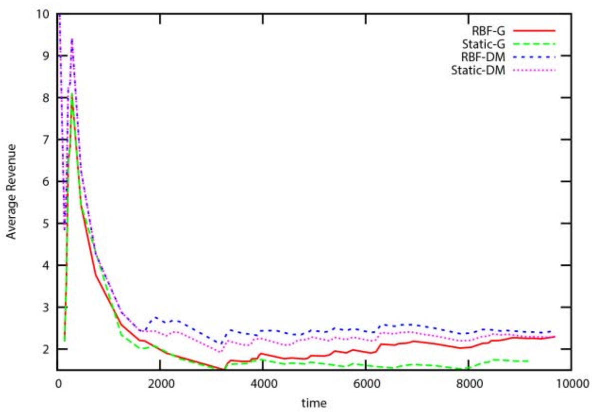

Figure 6. Average revenue

Embedding revenue is one of the most important indicators to network operators. The key factor that affects the embedding revenue is the network acceptance rate. Figure 6 shows the performance of the algorithms in the long-term average revenue (LR) of network embedding. In the early stage of embedding process, the LR of the corresponding algorithms are similar to the performance of the AR. Since the RBF-G algorithm and RBF-DM algorithm are generally higher than the Static-G algorithm and Static-DM algorithm in AR, the LR of them are higher than the static embedding algorithms too. In addition, in the case that both average revenue and network acceptance are higher than the static algorithms, the total revenue of the dynamic algorithms will be much greater than the static algorithms.

It can be seen that the RBF-VNE algorithm proposed in this paper can well integrate and utilize the SN resources. In the case of little changes in AC, the proposed algorithm can significantly improve the network acceptance and network revenue.

V. CONCLUSION

In this paper, a dynamic resource allocation algorithm called RBF-VNE has been proposed to study dynamic requirements of VNE. We took the actual required resources of VNs as the training samples of RBF neural network, then the obtained model was utilized to predict the demands of resources at the next moment. Finally, we recycled and redistributed network resources according to the predicted results. The simulation results show that compared with the traditional static algorithm, our method can make better use of SN resources and significantly improve the network acceptance and network revenue in the case of little changes in the average cost.

There are two main plans for further research:

(1) Since the initialization structure of RBF neural network is not optimal (especially the number of hidden layer neurons), the RBF neural network model can be optimized. In the complex embedding environment, due to the impact of the virtual request’s life cycle, training time and the number of training samples, the deviation of the predicted results may be bigger.

(2) In this paper, although the random noise is added to simulate the real resource requirements of users, the user's needs may be more variable in reality. This requires us to use the collected authentic data to train the prediction model.

ACKNOWLEDGMENTS

We are grateful for the support of the National Natural Science Foundation of China (61373149, 61170227), SIT Collaborative innovation platform under Grant No. 3921NH166033, and the Open Research Fund from Shandong Provincial Key Laboratory of Computer Network (No. SDKLCN-2016-02).

REFERENCES

[1] Xing Y, Zhan Y. Virtualization and cloud computing[M]//Future Wireless Networks and Information Systems. Springer Berlin Heidelberg, 2012: 305-312.

[2] Carapinha J, Jiménez J. Network virtualization: a view from the bottom[C]//Proceedings of the 1st ACM workshop on Virtualized infrastructure systems and architectures. ACM, 2009: 73-80.

[3] Haider A, Potter R, Nakao A. Challenges in resource allocation in network virtualization[C]//20th ITC Specialist Seminar. 2009, 18(2009).

[4] Chowdhury M, Samuel F, Boutaba R. Polyvine: policy-based virtual network embedding across multiple domains[C]//Proceedings of the second ACM SIGCOMM workshop on Virtualized infrastructure systems and architectures. ACM, 2010: 49-56.

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

[6] Zhu Y, Ammar M H. Algorithms for Assigning Substrate Network Resources to Virtual Network Components[C]//INFOCOM. 2006, 1200(2006): 1-12.

[7] Lu J, Turner J. Efficient Mapping of Virtual Networks onto a Shared Substrate[J]. Washington University in St Louis, 2006.

[8] Wenzhi L, Shuai L, Yang X, et al. Dynamically adaptive bandwidth allocation in network virtualization environment[J]. Advances in Information Sciences and Service Sciences, 2012, 4(1).

[9] Stubberud S C, Kramer K A. A game theoretic sensor resource allocation using fuzzy logic[J]. Advances in Fuzzy Systems, 2013, 2013.

[10] Papagianni C, Leivadeas A, Papavassiliou S, et al. On the optimal allocation of virtual resources in cloud computing networks[J]. IEEE Transactions on Computers, 2013, 62(6): 1060-1071.