IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 2, 2016, accepted November 21, 2016, date of current version January 4, 2017.

Digital Object Identifier 10.1109/ACCESS.2016.2632421

Virtual Network Embedding Based on the Degree and Clustering Coefficient Information

PEIYING ZHANG[1,2], HAIPENG YAO[1], AND YUNJIE LIU[1]
[1]State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China
[2]College of Computer & Communications Engineering, China University of Petroleum (East China), Qingdao 266580, China

Corresponding author: P. Zhang (25640521@qq.com)

This work was supported in part by the Shandong Provincial Natural Science Foundation, China, under Grant ZR2014FQ018, in part by the BUPT-SICE Excellent Graduate Students Innovation Fund, in part by the Fundamental Research Funds for the Central Universities of China University of Petroleum (East China) under Grant 13CX02031A, in part by the National Natural Science Foundation of China under Grant 61471056, in part by the China Jiangsu Future Internet Research Fund under Grant BY2013095-3-1, and in part by the BUPT Youth Research and Innovation Plan under Grant 2014RC0103.

ABSTRACT The issue of virtual network (VN) embedding constitutes an important aspect of network virtualization, which is considered to be one of the most crucial techniques to overcome the Internet ossification problem. The main purpose of VN embedding is to efficiently utilize the limited physical network resources to offer the supporting of virtual nodes and virtual links from the VNs. Due to the fact that the VN embedding problem is proved to be NP-hard, previous works have put forward some of heuristic algorithms to solve this VN embedding problem. However, most of the existing research works only consider the local resources of nodes, ignoring the topological attributes of its neighborhood nodes, and lead to lower resource utilization of the substrate network. To address this issue, we proposed an approach of VN embedding algorithm called *VNE-DCC*, which based on the node degree and the clustering coefficient information, we adopted the technique of node importance metric to rank the substrate nodes aim to select the node with the most embedding potential for every virtual node in each VN requests, and exploited the breadth-first-search algorithm to embed the virtual nodes aiming at reducing the resource utilization of substrate links so as to increase the acceptance ratio of VN requests and increase the revenues of operational providers. Extensive simulations have shown that the efficiency of our algorithm is better than the other state-of-the-art algorithms in terms of Revenue/Cost ratio and acceptance ratio.

INDEX TERMS Virtual network embedding, degree and clustering coefficient, network virtualization, virtual node mapping, virtual link mapping.

I. INTRODUCTION

Over the past years, the Internet technology has developed rapidly and becomes one of the most important techniques and it is changing the people's life style. The increasing number of end users and the large volume of business has brought challenges to the network availability, flexibility, reliability, and scalability; the notion of data transmission to the best of its abilities has been unable to satisfy the diverse requirements of future network services because different customers would require different services with different Service Level Agreements (SLAs). The existing interconnection network architecture is rigid, cannot meet the demand of future network business upgrade, does not support the deployment of new protocols and the innovations of future network. To address the aforementioned problems, the researchers have proposed the network virtualization technique which is termed as one of the ways to overcome the current network ossification problems [1]–[4].

Due to the emergence of network virtualization technology, the traditional internet server provides are decoupled into service providers (SPs) and infrastructure providers (InPs) according to their performed roles. A plurality of service providers hire underlying physical resources from the InPs, provide customized services for end users by creating multiple virtual networks [5]–[7]. By means of network virtualization technology, researchers can design and program a new protocol on multiple heterogeneous physical networks. Each VN request (VNR) is composed of the set of virtual nodes and the set of virtual links. The virtual node can be mapped onto a substrate node, and the virtual link can be mapped onto a substrate network path which may be consists of more than one physical links. Virtual network embedding is the process of mapping virtual nodes and virtual links onto the corresponding substrate nodes and substrate paths meanwhile meeting the requirements of the underlying physical resources constraints. Because all of virtual networks share the same substrate network, an efficient and effective virtual network embedding algorithm is of great importance to improve the resources utilization of physical network and increase the income of infrastructure providers (InPs).

In these previous works [7]–[13], [18], some of heuristic algorithms are put forward to maximize the revenue by accommodating more VN requests and reducing the mapping cost of the VN requests. Most of these approaches adopt heuristic algorithms or greedy algorithms to accomplish the mapping of virtual nodes, and then to perform the mapping of virtual links using k-shortest path or multiple commodity flow algorithm. These VN embedding algorithms can be roughly divided into the following categories.

The approaches that based on strategy. The most representative of these algorithms are [4] and [14]–[17]. Feng *et al.* [14] concentrated primarily on two types of the VN embedding problems, one is the virtual network embedding approach without reconfiguration and the other is the virtual network embedding approach with reconfiguration. To address the first issue, the authors exploited subdividing heuristics and adaptive optimization strategies to solve the VN embedding problems; To address the second issue, the authors adopted a selective VN reconfiguration mechanism which gives the priorities to the most critical virtual network requests to solve the VN embedding problems. In [4], the authors proposed a different approach which rethinking the design of the substrate network to enable simpler embedding algorithms and more efficient resources utilization without restricting the problem space. The authors mainly on the basis of these two principles: (i) The embedding algorithm allows the substrate network to split a virtual link over multiple substrate paths and (ii) The embedding algorithm can re-optimize the resource utilization of the substrate network periodically by means of path migration strategy. Fajjari *et al.* [15] proposed an approach of virtual network reconfiguration to eliminate the bottleneck caused by physical links through the observations that the acceptance ratio of virtual network requests is affected by the bottlenecked physical resources of substrate links. The authors suggested a heuristic algorithm to address this problem and proposed a greedy algorithm called *VNR* to reconfigure the substrate links aiming at increasing the acceptance ratio of virtual network requests. Hesselbach *et al.* [16] suggested a new path algebra-based embedding strategy for coordination of virtual nodes and virtual links to bridge the gap of node mapping stage and link mapping stage. The proposed mapping strategy of virtual network requests has significantly improved the performance of algorithm through extensive simulations. Wang and Hamdi [17] formulated the VN embedding problem as a new multiple objective linear programming optimization program, and solved it by a preemptive strategy. The authors utilized the abstract artificial intelligence resource model (BI) to promote the efficiency of online heuristic VN embedding algorithm called *Presto*.

The approaches that based on resource allocation mechanism. Juan *et al.* [18] put forward a heuristic resource allocation algorithm called *HVNE* to address the issues of network load balancing. In order to conserve network resources of physical links, the proposed approach named *HVNE* allows multiple virtual nodes to embed onto a single substrate node, and it makes up a connection between the theory of k-regional divided optimization and traditional topology division theory. Two existing vertex mapping approaches are proposed by [19], one that takes into consideration the situations in which abundant node resources are available for the current VN embedding process, and one that only takes into account the degree of a node which is already utilized by existing VN embedding algorithm. Razzaq *et al.* [20] devised a new approach for mapping the virtual nodes with the aim of minimizing the complete exhaustion of substrate nodes while maintaining the overall higher utilization of physical resources.

The approaches that based on topological attributes of substrate network and virtual networks. Feng *et al.* [14] primarily took advantage of topological attributes of substrate and virtual networks so as to improve the resource utilization of physical network. They advocated utilizing topological attributes of substrate network and virtual networks with the purpose of improving the long-term average revenue, acceptance ratio and R/C ratio. In this work, a topology-aware VN embedding algorithm based on the degrees of nodes is proposed, extensive simulations illustrated that the proposed algorithm is better than the other algorithms. Ding et. al. [21] improved the virtual network embedding algorithm based on degrees of node by means of node connectivity. They formulated a mathematical model of the connectivity between each pair of nodes to evaluate the resource ranking value of the nodes, and proposed a new two-stage VN embedding algorithm. Cheng *et al.* [7] inspired by the PageRank algorithm, proposed an approach of topology-aware node ranking by means of Markov Random Walk and devised two VN embedding algorithms called as *Max-Match* and *RW-BFS*, respectively. The proposed algorithms not only take into consideration the node CPU's capacity and its associated link bandwidth capacity, but also take into account the influence of its surrounding nodes. The authors improved the ranking method of node's importance, and therefore enhanced the performance of VN embedding algorithm in terms of the acceptance ratio and R/C ratio.

The approaches that based on heuristic. Chowdhury *et al.* [5] formed a cluster which satisfy the constraints of geographic position for both of virtual nodes and substrate nodes, and they used the augmented graph to perform the node mapping and link mapping. The authors presented an approach of VN embedding based on the connectional characteristics of virtual network topology in [6], they exploited the topological connection features to evaluate the node's importance and gave the priority to the virtual nodes with largest ranking value during the mapping procedure, took advantage of local resource of substrate nodes and their associated hops between two nodes to evaluate the substrate nodes. The proposed algorithm can make the mapped substrate nodes are closer each other aiming at saving the physical resources of link mapping. Jin-Peng and Tian-Yu [22] organized the substrate nodes into a cluster that meet the constraints of the geographic location, they chose the first node in a cluster for the corresponding virtual node, meanwhile choosing a compact substrate node as its mapping node, with the purpose of reducing the mapping cost of physical paths.

In the process of virtual network embedding, if a node is more important than the other nodes, the node should be given priority to choose aim to facilitate the subsequent link mapping procedure. Although these algorithms have improved the metrics of node to some extent, they do not take into account the topological structures around the nodes, the proposed algorithm can be further improved. Distinguished from previous studies, inspired by the literature [23], we not only take the CPU capacity and bandwidth into considerations, but also take advantage of node degree and clustering coefficient information for measuring the node importance so as to improve the performance of VN embedding algorithm. This paper proposed a two-stage virtual network embedding algorithm called *VNE-DCC* based on node degree and clustering coefficient information to evaluate the node importance, and exploited the breadth-first-search (BFS) algorithm to embed the virtual nodes with the purpose of reducing the resource utilization of substrate links. Our proposed algorithm uses the metric of *NIM* value to choose more significant nodes, and uses the K-shortest path algorithm to map virtual links.

Distinguished from previous studies, this paper presents the following major contributions:

(1) Different from prior research works, we proposed a node importance evaluating method named *NIM* based on the node degree and clustering coefficient information to measure the embedding potential of substrate nodes aiming at increasing the probability of the successful node mapping.

(2) With the aim of reducing the mapping costs of virtual links from virtual networks, we utilized the breadth-first-search (BFS) algorithm to embed the virtual nodes so as to increase the acceptance ratio of virtual network requests and increase the revenues of operational providers.

(3) Two experiments and six aspects of the comparison are performed through extensive simulations have demonstrated that our proposed *VNE-DCC* algorithm outperforms the other algorithms *Greedy* and *RW* in terms of R/C ratio and acceptance ratio.

The remainder of this paper is organized as follows. In Section 2, we presented the problem statement and network model. In Section 3, we introduced the measurement of node importance by means of node degree and clustering coefficient information. In Section 4, we gave the details of *VNE-DCC* algorithm. In Section 5, we discussed the experimental results and analyzed the reason. In Section 6, we concluded the paper.

II. PROBLEM STATEMENT AND NETWORK MODEL

In this section, we first introduce the network model including substrate network model and virtual network request (VNR) model, and formulate the general definition of virtual network embedding problem, and in the last subsection, we present the evaluation index of virtual network embedding.

A. THE MODEL OF SUBSTRATE NETWORK

The substrate network can be abstracted as undirected weighted diagram and it can be denoted as $G_s = (N_s, L_s, A_s^N, A_s^L)$, in which N_s stands for the collection of substrate nodes, L_s stands for the collection of substrate links, A_s^N stands for the attribute set of substrate nodes' resources, such as their locations, the speed of switching, the storage capacity and the CPU capacity and so forth, A_s^L stands for the attribute set of substrate links' resources, such as the propagation delay time, the available bandwidth and so on. Similar to some previous literatures, we only take the CPU capacity of nodes and available bandwidth of links into considerations. As demonstrated in Fig. 1, the available CPU capacity of node A is 20 units, the available bandwidth of link (A, C) is 30 units.

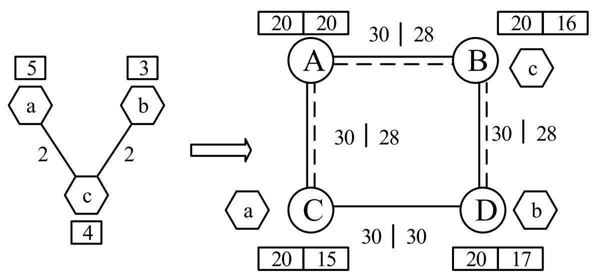


FIGURE 1. The virtual network request (VNR) and substrate infrastructure.

B. THE MODEL OF VIRTUAL NETWORK REQUEST

A virtual network request (*VNR*) can be modeled as undirected weight diagram and can be denoted as $G_v = (N_v, L_v, A_v^N, A_v^L, T_a, T_d)$, where N_v represents the collection of virtual network nodes, N_v represents the collection of virtual network links, A_v^N represents the resources constraint collection of virtual nodes, A_v^L represents the resources constraint collection of virtual links, T_a indicates the arrival time of each virtual network request, T_d indicates the duration of the virtual network request remaining in the substrate network. From the Fig. 1 we can see that the CPU capacity of virtual node denoted by a is 5 units, the bandwidth requirement of virtual link denoted by (a, c) is 2 units.

C. THE PROBLEM OF VIRTUAL NETWORK EMBEDDING

Virtual network embedding is a process in which virtual nodes are mapped onto substrate nodes and virtual links are mapped onto substrate paths, on the premise of meeting the certain physical network resource constraints. The mapping process consists of two stages: one is the node mapping and the other is link mapping. The task of node mapping is to assign virtual node onto the concrete substrate node, the mapping process must satisfy the resource constraints, a virtual node can only mapped onto a physical node, a physical node can only host a virtual node from the same of *VNR*. The task of link mapping is to assign virtual link onto the concrete substrate path, the mapping process must satisfy the link resources constraints such as bandwidth requirements, and its mapped physical path cannot be formed into a loop.

1) NODE MAPPING STAGE

We use a function to describe the process of node mapping, the function of Map_N can be denoted by $N_v \to \{N_s, \forall n_v \in N_v\}$, where N_v represents the set of virtual nodes, N_s represents the set of substrate nodes. As illustrated by Fig.1, the mapping result of virtual nodes is $\{a \to C, b \to D, c \to B\}$.

2) LINK MAPPING STAGE

We use a function to describe the process of link mapping, the function of Map_L can be denoted by $L_v \to \{\psi, \forall l_v \in L_v\}, \psi = \{\varphi \in P_\varphi | A_\varphi \geq A_{l_v}, \forall l_v \in L_v\}$, where P_φ represents the physical path set between two nodes, A_φ represents the available bandwidth of physical path φ, $A_\varphi = \min_{l_s \in \varphi} A_s^{l_s}$, in which $A_s^{l_s}$ represents the available bandwidth of physical link l_s. As illustrated by Fig.1, the mapping result of virtual links is $\{\{a, c\} \to \{C, A, B\}, \{b, c\} \to \{D, B\}\}$.

D. THE OBJECTIVES OF VNE

In this section, we introduce three kinds of evaluation indexes including the scaling factor of virtual link, revenue/cost ratio and acceptance ratio. No matter which evaluation index is used, the ultimate goal is to take advantage of infrastructure resources efficiently by means of mapping the virtual network requests into the shared physical network infrastructures while satisfying the resources constraints of nodes and links.

1) SCALING FACTOR OF VIRTUAL LINK

The mapping path length of virtual link in substrate network can affect the physical link bandwidth resource consumption. The scaling factor of virtual link can represent the average value of mapping path length in all of the virtual links, and takes the form:

$$SF = \sum_{n_i, n_j \in N_v} \frac{hop(M(n_i), M(n_j)) \times vlink(n_i, n_j)}{\sum_{n_i, n_j \in N_v} vlink(n_i, n_j)} \tag{1}$$

Where $M(n_i)$ and $M(n_j)$ represent the mapped substrate node of virtual node n_i and n_j, respectively, $hop(M(n_i), M(n_j))$ represents the physical path length between substrate node $M(n_i)$ and substrate node $M(n_j)$; $vlink(n_i, n_j)$ is a binary variable, its value is 1 when virtual node n_i and virtual node n_j is adjacent, otherwise its value is 0.

2) REVENUES AND COST RATIO

At t moment, the obtained revenues of accepting a *VNR* by a substrate network can be defined as the sum of virtual nodes' CPU capacity and virtual links' bandwidth requirements; the cost of accommodating a *VNR* by a substrate network can be defined as the sum of substrate nodes' CPU capacity and substrate paths' bandwidth resources. Hence, the revenues and cost ratio can be formulated as follows:

$$R/C = \frac{\sum_{n_v \in N_v} CPU(N_v) + \sum_{l_v \in L_v} BW(l_v)}{\sum_{n_v \in N_v} CPU(N_v) + \sum_{l_v \in L_v} BW(l_v) \times hop(l_v)} \tag{2}$$

Where $CPU(n_v)$ represents the CPU capacity of virtual node n_v, $B(l_v)$ represents the bandwidth requirements of virtual link l_v, $hop(l_v)$ represents the mapping path length of virtual link l_v.

3) THE ACCEPTANCE RATIO OF *VNR*

The acceptance ratio of *VNR* is the ratio of the number of virtual network request which have been successfully mapped and the total number of requests. The calculation formula is as follows:

$$AcceptanceRatio = \frac{\sum_{t=0}^{T} Number_{acc}}{\sum_{t=0}^{T} Number_{arr}} \tag{3}$$

In the above formula, the numerator represents the number of virtual network request that have been successfully mapped, and the denominator represents the total number of virtual network requests.

III. THE RANKING METHOD OF NODE IMPORTANCE

The *VNE* issue can be made up of two stages including node mapping and link mapping. In the first stage of node mapping, the task of it is to select some substrate nodes for virtual nodes while satisfying the CPU capacity requirements; the task of link mapping stage is to select some physical paths for virtual links while satisfying the bandwidth requirements. Most previous studies accomplish the embedding process by measuring the local resources for nodes and links respectively, aim to measure the nodes' importance in substrate network and virtual networks, which is calculated as follows:

$$NR(n) = CPU(n) \sum_{l \in nbr(n)} BW(l) \tag{4}$$

Where $CPU(n)$ represents the CPU capacity of node denoted by n, $nbr(n)$ represents the set of links that directly connected to the node n, $BW(l)$ represents the current available bandwidth of link denoted by l. The two stage of node mapping and link mapping are not considered independently in this manner.

The main shortcoming of this method is lack of considering its neighborhood node connectivity, to address this issue, we utilize the degree of node and clustering coefficient information to rank the importance of nodes.

A. NODE DEGREE

The node degree describes the number of its neighborhood nodes, and can be formulated as follows:

$$degree_i = \sum_{j \in N} \delta_{ij} \tag{5}$$

Where the parameter δ_{ij} takes the value 1 if two nodes denoted by i and j are directly connected, otherwise it takes the value 0. The node degree can reflect the ability to establish a direct link between the node and the surrounding nodes, but cannot reflect its neighborhood node connectivity.

B. CLUSTERING COEFFICIENT INFORMATION

The clustering coefficient describes the ratio of neighbors to each other in the network, and can be calculated as follows:

$$c_i = \frac{2e_i}{degree_i \times (degree_i - 1)} \tag{6}$$

In the above formula, e_i represents the number of triangles which formed by a node i with any of its two neighbors. In contrast to the node degree, the clustering coefficient can only reflect the node connectivity of its neighborhood nodes, but cannot reflect the scale of its neighborhood node. Therefore, we not only take advantage of the information of its neighborhood nodes, but also take node's clustering coefficient into consideration, put forward a novel node metric method and take the form:

$$p_i = \frac{f_i}{\sqrt{\sum_{j=1}^{N} f_i^2}} + \frac{g_i}{\sqrt{\sum_{g=1}^{N} g_i^2}} \tag{7}$$

Where f_i represents the sum of its own node degree and its neighborhood node degree, denoted by:

$$f_i = degree_i + \sum_{j \in nbr(i)} degree_j \tag{8}$$

Where $degree_j$ represents the node degree for node j, $nbr(i)$ represents neighbor set of node denoted by i. The function of g_i can be formulated as follows and take the form:

$$g_i = \frac{\max\limits_{j=1..N}\{\frac{c_j}{f_j}\} - \frac{c_j}{f_j}}{\max\limits_{j=1..N}\{\frac{c_j}{f_j}\} - \min\limits_{j=1..N}\{\frac{c_j}{f_j}\}} \tag{9}$$

Where c_i represents the above mentioned called the node clustering coefficient. the node's clustering coefficient can only reflect the close degree between its neighborhood nodes, but cannot reflect the scale of its neighborhood node. We normalize the value of $\frac{c_j}{f_j}$ by using the formula (7). Due to the fact that f_i can reflect the node degree and its neighborhood nodes degree, g_i can reflect the close degree between its neighborhood nodes, we use the function of $u(x) = \frac{x}{\sqrt{x^2}}$ to deal with the f_i and g_i, make the index p_i can reflect the combined results of the two factors.

A motivational example is illustrated in Fig. 2, we can see that the destructional effect of deleting the node 4 on the network is much more than deleting the node 2.

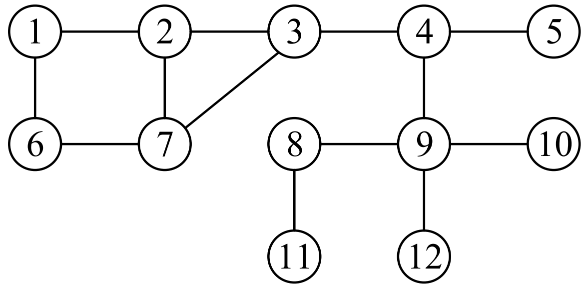


FIGURE 2. A Motivational Example of Node Degree and Clustering Coefficient Information.

However, when we only take the node degree and its neighborhood nodes degree into consideration, $degree_2 = degree_4 = 3, f_2 = f_4 = 11$. Furthermore, we consider the relationship between the nodes' neighborhood nodes, we can calculate that $p_2 = 0.38$, $p_4 = 0.72$. The calculation results show that the close degree between its neighborhood nodes has some influence on the node ranking, hence, the importance of a node needs to consider the combined effects of the node's neighbor information and the clustering coefficient.

C. THE NODE IMPORTANCE METRIC

Based on the above analysis, the node importance metric can be formulated as the product of $NR(n_i)$ and p_i, the calculation formula can be expressed as follows:

$$NIM(n_i) = NR(n_i) \times \{\frac{f_i}{\sqrt{\sum_{j=1}^{N} f_j^2}} + \frac{g_j}{\sqrt{\sum_{j=1}^{N} g_j^2}}\} \tag{10}$$

Where $NIM(n_i)$ represents the node importance metric value of the node n_i, $NR(n_i)$ represents the node ranking value which can be calculated by formula (4), the part of formula (10) denoted by $\{\frac{f_i}{\sqrt{\sum_{j=1}^{N} f_j^2}} + \frac{g_j}{\sqrt{\sum_{j=1}^{N} g_j^2}}\}$ represents the node metric value based on node degree and clustering coefficient information.

IV. THE ALGORITHM OF VNE BASED ON THE NODE DEGREE AND CLUSTERING COEFFICIENT INFORMATION

On the foundation of node degree and clustering coefficient information analysis, we devised a novel two-stage VNE algorithm called *VNE-DCC* to efficiently utilize the physical network resources. The *VNE-DCC* algorithm is a two-stage VN embedding algorithm which includes node mapping stage and link mapping stage. In the subsequent subsections, we describe the details of node mapping algorithm and link mapping algorithm.

A. NODE MAPPING STAGE

In the first stage of node mapping, we measure the node's importance by means of its *NIM* value with the purpose of choosing the most important substrate node so as to facilitate the subsequent link mapping. In order to make the adjacent virtual nodes can map the adjacent substrate nodes, we use

Algorithm 1 The Sorting Algorithm of Virtual Nodes in *VNRs*

1: Sort the virtual nodes by $NIM(n_i)$ in non-increasing order.
2: Choose the virtual node which has greatest *NIM* value as the *Root*.
3: Using *Root* as the root node, traverse the graph of virtual network request using breadth first search algorithm, and get the breadth first search tree denoted by *Tree*.
4: Sort the every layer's nodes of *Tree* according to *NIM* value in non-increasing order.
5: Return the sequence of sorted nodes.

graphical breadth-first-search (*BFS*) algorithm to sort the virtual nodes, make continuous mapping procedure of virtual nodes has greater relevance. The sorting algorithm of virtual nodes are list in Algorithm 1.

With the purpose of reducing the overhead of link mapping stage, we use the breadth first search algorithm in the node mapping stage. The specific mapping algorithm is shown in Algorithm 2.

Algorithm 2 The Node Mapping Algorithm Based on Breadth First Search

1: Sort the virtual nodes by Algorithm 1.
2: Sort the substrate nodes according to the *NIM* value in non-increasing order.
3: **for** each of *VNR* **do**
4: **if** the virtual node is root **then**
5: map the virtual node to the substrate node which owns the greatest *NIM* value.
6: **else**
7: find the parent node of the virtual node denoted by *P*.
8: find the mapped substrate node denoted by *S* for the virtual node's parent node *P*.
9: find the set of *S*'s neighbor nodes as the candidate substrate nodes.
10: choose the substrate node which owns the greatest *NIM* value meanwhile satisfying the CPU capacity constraints.
11: **end if**
12: return the node mapping result.
13: **end for**

B. LINK MAPPING STAGE

In the second stage of link mapping, we use k-shortest algorithm aim to select the most suitable physical paths while satisfying the bandwidth resource requirements. In order to facilitate the calculation, we firstly remove these links that do not meet the constraints of link's bandwidth, and then use the Floyd algorithm to calculate the shortest path. The specific implementation details can refer to the Algorithm 3.

Algorithm 3 The Link Mapping Algorithm Based on K-Shortest Path

1: Sort the virtual links according by its bandwidth.
2: **for** each virtual link **do**
3: calculate the required bandwidth of virtual link denoted by *BW*.
4: remove the links whose bandwidth is lesser than *BW*.
5: find the two virtual nodes for each virtual link.
6: find the two substrate nodes for the two virtual nodes according to the node mapping result.
7: calculate the substrate path using Floyd algorithm between two substrate nodes.
8: return the link mapping result.
9: **end for**

Algorithm 4 VNE Algorithm Based on Degree and Clustering Coefficient

1: Calculate the revenues of all of the virtual network requests.
2: Sort all the *VNRs* (Virtual Network Request) in non-increasing order according to their revenues.
3: **for** each of *VNRs* **do**
4: sort the virtual nodes by Algorithm 1.
5: do node mapping by Algorithm 2.
6: do link mapping by Algorithm 3.
7: return the mapping result.
8: **end for**

C. OUR PROPOSED ALGORITHM

Based on the Algorithm 1, Algorithm 2 and Algorithm 3, we give the mapping procedure of our algorithm called *VNE-DCC* in Algorithm 4.

D. TIME COMPLEXITY ANALYSIS

The time complexity of the Algorithm 1 is $O(|N_v|^2)$, where $|N_v|$ represents the number of virtual nodes in each virtual network request. The time complexity of the Algorithm 2 is $O(|N_v| \times |N_s|^2)$, where $|N_s|$ represents the number of substrate nodes in substrate network. The time complexity of the Algorithm 3 is $O(|E_v| \times |N_s|^3)$, where $|E_v|$ represents the number of virtual links. The time complexity of the Algorithm 4 is $O(N \times |E_v| \times |N_s|^3)$, where N represents the number of virtual network requests.

The time complexity of the Algorithm *Greedy* is $O(N \times |E_v| \times |N_s|^3)$, where N represents the number of virtual network requests, $|E_v|$ represents the number of virtual links, $|N_s|$ represents the number of substrate nodes in substrate network. The time complexity of Algorithm *RW* is also $O(N \times |E_v| \times |N_s|^3)$, but it actually consumes a little more running time than Algorithm *Greedy* in the process of evaluating the node's importance using random walks. The time complexity of our Algorithm is the same as these above two algorithms, but the R/C ratio can be significantly increased.

V. SIMULATION RESULTS AND ANALYSIS

In this section, the simulation environment settings and the simulation results are discussed. We use the measurement index of performance metrics which proposed in section 2.4 to evaluate our *VNE-DCC* algorithm. Two state-of-the-art algorithms which called as *Greedy* [24] and *RW* [7] are compared with our proposed algorithm, respectively. In this paper, our *VNE-DCC* algorithm did not support path splitting, so we did not consider the case of supporting path splitting.

A. EXPERIMENTAL ENVIRONMENT SETTINGS

Similar to the previous literature [24], we use GT-ITM tool to generate the topology of the virtual network requests and substrate network. The specific parameter settings are shown in Table 1.

TABLE 1. The Settings of Parameters

Parameter Items	The Range
Substrate network:	
The number of physical network nodes	100
The probability of connectivity between two nodes	0.5
The distribution of CPU capacity	$U[50, 100]$
The distribution of bandwidth requirement	$U[50, 100]$
Virtual network requests:	
The number of virtual nodes	$U[2, 20]$
The connectivity probability of virtual network	0.5
The distribution of CPU capacity	$U[0, 20]$
The distribution of bandwidth requirement	$U[0, 50]$
The VNR arrives at Poisson distribution	100 units / 4

B. EXPERIMENTAL RESULTS AND ANALYSIS

Due to the fact that the bigger the scaling factor is, the smaller the R/C ratio is, i.e., the R/C ratio can reflect the effect of the scaling factor. Therefore, similar to pervious literatures, we use R/C ratio and acceptance ratio to evaluate our algorithm compared with the other algorithms.

Experiment 1: In order to evaluate the performance of VN embedding algorithms with different CPU capacity requirements, we let the CPU capacity of virtual nodes uniformly distributed in $[0, C_n]$ and C_n increasing from 10 to 100 step 10. The experimental results are shown in Fig. 3 and Fig. 4.

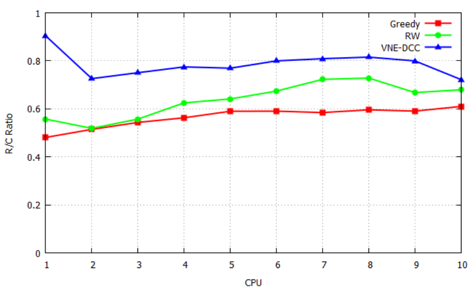


FIGURE 3. The Revenue/Cost Ratio with increasing number of CPU.

1) R/C RATIO COMPARISON

As illustrated in Fig. 3, the R/C ratio will be gradually increasing along with the increment of the CPU capacity requirements, the main reason is that all of our compared three algorithms are based on node ranking methods, the bigger the CPU capacity requirement is, the more effective the algorithms are. The first point's value of our proposed algorithm is much bigger than other algorithms, the reason is that our algorithm is based on breadth first search algorithm, when the demand of node's CPU capacity is lower, our algorithm performs much better. The fact that the free CPU capacity of substrate network nodes becoming scarce can account for the sudden decline of the last point's value. We carried out this experiment many times, but the second point's value of the *RW* still near to the value of the *Greedy*, we guessed that it due to the specific topology of substrate network in which the node importance metric cannot be improved using the random walk approach. From the overall trend as illustrated in Fig. 3, our algorithm called *VNE-DCC* is better than the other two algorithms.

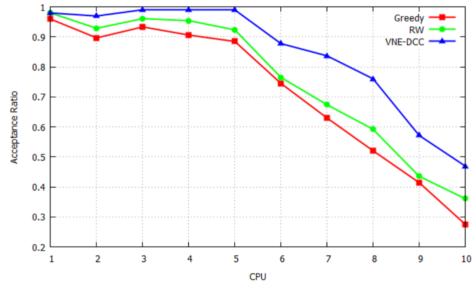


FIGURE 4. The Acceptance Ratio with increasing number of CPU.

2) ACCEPTANCE RATIO COMPARISON

As illustrated in Fig. 4, the acceptance ratio will be gradually increasing along with the increment of the CPU capacity requirements, through deep investigation and analysis, we found that the available CPU capacity of substrate nodes and the available bandwidth capacity of the substrate links are gradually reducing with the number of virtual network requests (VNRs) which have been successfully mapped increases during the mapping process, the remained network resources that can be mapped are gradually decreased with the mapping process, the acceptance ratio will be decreasing with the changing. The acceptance ratio of *RW* is the same as the acceptance ratio of *VNE-DCC*, the reason is that when the demand of CPU requirements is less, while the substrate network has abundant available resources, the acceptance ratio of these two algorithms is much higher than the Greedy algorithm, since both of these two algorithms have improved the node's ranking methods. As demonstrated in Fig. 4, our approach performed better when the demand of CPU requirements was increased. From the Fig. 4, we can see that the performance of our algorithm called *VNE-DCC* is superior to the other two algorithms.

Experiment 2: With the purpose of measuring the effect on different bandwidth requirements, we set the virtual nodes' CPU capacity uniformly distributed in $U[0, 100]$, and the

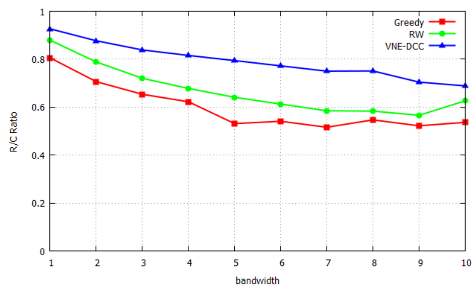


FIGURE 5. The Revenue/Cost Ratio with increasing bandwidth.

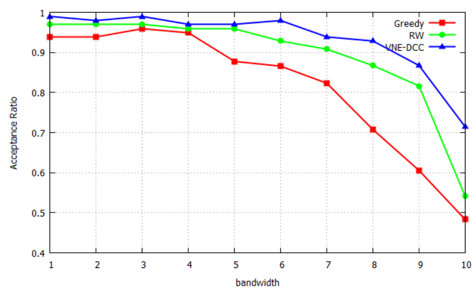


FIGURE 6. The Acceptance Ratio with increasing bandwidth.

bandwidth of the virtual link is subject to uniform distribution of $U[0, B_l]$, we let B_l is ranging from 10 to 100 step 10. We evaluate the performance of our algorithm compared with the other algorithms in terms of R/C ratio and acceptance ratio. The simulation results are illustrated in Fig. 5 and Fig. 6.

3) R/C RATIO COMPARISON

As demonstrated by Fig. 5, the R/C ratio will be gradually decreasing along with the increment of the demand of bandwidth requirements, the main reason is that the available physical network resources will be gradually reducing with the increasing demand of bandwidth requirements. From the Fig. 5, we can see that our algorithm outperforms the other two algorithms, the acceptance ratio of our algorithm outperforms *RW* algorithm, the acceptance ratio of *RW* algorithm outperforms Greedy algorithm, and our proposed algorithm is the best of these three algorithms.

4) ACCEPTANCE RATIO COMPARISON

As demonstrated by Fig. 6, the acceptance ratio of three algorithms will be gradually decreasing with the increment of the demand of bandwidth requirements, due to the fact that the infrastructure resources that have not been occupied are gradually reduced with the number of virtual network requests which have been successfully mapped increasing, and it leads to the increasing number of substrate nodes and substrate links which cannot meet the resource constraints. The first half of difference is not as obviously as the second half, the main reason is that the substrate network has sufficient link bandwidth resources in the first half, but doesn't have enough link bandwidth resources in the second half. From the Fig. 6, we can see that our algorithm is preferred over the other algorithms.

5) VERTICAL COMPARISON

In order to discuss our experimental results in detail, we compared the R/C Ratio and Acceptance Ratio between two experiments, respectively. We called it Vertical Comparison.

6) R/C RATIO COMPARISON

As depicted in Fig. 3 and Fig. 5, we compare the first part of the two diagrams, the more the demand of CPU capacity is, the bigger the R/C ratio is; the more the demand of link bandwidth requirement is, the smaller the R/C ratio is. Through deep investigation, we found that the reason is that both our proposed algorithm and the compared algorithms are based on node ranking methods, the more the demand of CPU capacity is, the more effective the algorithms are.

7) ACCEPTANCE RATIO COMPARISON

As demonstrated in Fig. 4 and Fig. 6, we can see that the trend of the first diagram is basically consistent with the trend of the second diagram. The differences between our algorithm and the other two algorithms in the second half of two diagrams are more obvious than in the first half of two diagrams, the reason is that the resources of substrate network are gradually decreased with the increasing demand of CPU capacity or bandwidth requirements.

VI. CONCLUSION

In the environment of network virtualization, virtual network embedding is one of the most challenging task aim to address the issue of network ossification. We use the node degree and clustering coefficient information to improve the metric of nodes importance. Based on the metric of node importance, we proposed a novel two-stage algorithm called *VNE-DCC* and performed extensive simulations to validate our proposed algorithm. Extensive experiments and results analysis have shown that our proposed algorithm improves the performance of VN embedding procedure in terms of R/C ratio and acceptance ratio, as well as decreases the scaling factor.

The merits of the provided algorithm is as follows: (i) The proposed algorithm presents a comprehensive metric of node importance named *NIM* to measure the embedding potential of substrate nodes. (ii) The proposed algorithm uses the breadth-first-search algorithm to embed the virtual nodes aim at reducing the bandwidth utilization of substrate links. The demerits of the provided algorithm is as follows: (i) The proposed algorithm considered the VN embedding as two separated procedures which would lead to lower acceptance ratio due to improper node mapping results. (ii) The proposed node importance metric does not consider the multiple features of the substrate nodes, hence the node ranking method can be further improved by means of other features of substrate nodes.

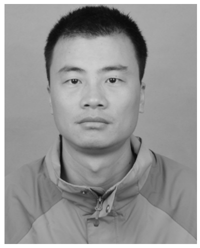
There are still some issues which should be further investigated. We focus mainly on the improvement of measurement of node importance, the improvement of link mapping algorithm is seldom considered. We should further optimize the link mapping stage to enhance the efficiency of virtual network embedding algorithm.

ACKNOWLEDGEMENTS

The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] J. S. Turner and D. E. Taylor, "Diversifying the Internet," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, vol. 2. Dec. 2005, p. 760.
[2] N. Feamster, L. Gao, and J. Rexford, "How to lease the Internet in your spare time," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 1, pp. 61–64, 2007.
[3] T. Anderson, L. Peterson, S. Shenker, and J. Turner, "Overcoming the Internet impasse through virtualization," *Comput.*, vol. 34, no. 4, pp. 34–41, 2005.
[4] R. Ricci, C. Alfeld, and J. Lepreau, "A solver for the network testbed mapping problem," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 33, no. 2, pp. 65–81, 2003.
[5] M. Chowdhury, M. R. Rahman, and R. Boutaba, "Vineyard: Virtual network embedding algorithms with coordinated node and link mapping," *IEEE/ACM Trans. Netw.*, vol. 20, no. 1, pp. 206–219, Jan. 2012.
[6] H. Cui, W. Gao, J. Liu, and Y. Liu, "A virtual network embedding algorithm based on virtual topology connection feature," in *Proc. 16th Int. Symp. Wireless Pers. Multimedia Commun. (WPMC)*, Jun. 2013, pp. 1–5.
[7] X. Cheng *et al.*, "Virtual network embedding through topology-aware node ranking," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 2, pp. 38–47, Apr. 2011.
[8] X. Cheng, Z.-B. Zhang, S. Su, and F.-C. Yang, "Virtual network embedding based on particle swarm optimization," *Dianzi Xuebao(Acta Electron. Sinica)*, vol. 39, no. 10, pp. 2240–2244, 2011.
[9] I. Houidi, W. Louati, W. B. Ameur, and D. Zeghlache, "Virtual network provisioning across multiple substrate networks," *Comput. Netw.*, vol. 55, no. 4, pp. 1011–1023, 2011.
[10] I. Houidi, W. Louati, and D. Zeghlache, "Exact multi-objective virtual network embedding in cloud environments," in *Proc. Workshop Enabling Technol. Infrastruct. Collaborative Enterprises Wet Ice*, vol. 58, no. 3, pp. 403–415, 2013.
[11] Z. Zhang, X. Cheng, S. Su, Y. Wang, K. Shuang, and Y. Luo, "A unified enhanced particle swarm optimization-based virtual network embedding algorithm," *Int. J. Commun. Syst.*, vol. 26, no. 8, pp. 1054–1073, 2013.
[12] S. Zhang, Z. Qian, J. Wu, S. Lu, and L. Epstein, "Virtual network embedding with opportunistic resource sharing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 3, pp. 816–827, Mar. 2014.
[13] Z. Zhang, S. Su, Y. Lin, X. Cheng, K. Shuang, and P. Xu, "Adaptive multi-objective artificial immune system based virtual network embedding," *J. Netw. Comput. Appl.*, vol. 53, pp. 140–155, Jul. 2015.
[14] M. Feng, L. Zhang, X. Zhu, J. Wang, Q. Qi, and J. Liao, "Topology-aware virtual network embedding through the degree," in *Proc. Nat. Doctoral Acad. Forum Inf. Commun. Technol.*, Aug. 2013, pp. 1–6.
[15] I. Fajjari, N. Aitsaadi, G. Pujolle, and H. Zimmermann, "VNR algorithm: A greedy approach for virtual networks reconfigurations," in *Proc. IEEE Global Telecommun. Conf.*, vol. 47. 2011, pp. 1–6.
[16] X. Hesselbach, J. R. Amazonas, S. Villanueva, and J. F. Botero, "Coordinated node and link mapping VNE using a new paths algebra strategy," *J. Netw. Comput. Appl.*, vol. 69, pp. 14–26, Jul. 2016.
[17] T. Wang and M. Hamdi, "*Presto*: Towards efficient online virtual network embedding in virtualized cloud data centers," *Comput. Netw.*, vol. 106, pp. 196–208, Sep. 2016.
[18] L. Juan, C. Lei, and L. Renfa, "A heuristic resource allocation algorithm for virtual network embedding," *Sci. China, Inf. Sci.*, vol. 42, no. 8, pp. 960–973, 2012.
[19] Z. M. Ren, "Node importance measurement based on the degree and clustering coefficient information," *Acta Phys. Sinica*, vol. 62, no. 12, p. 505, 2013.
[20] A. Razzaq, M. Hidell, and P. Sjödin, "Virtual network embedding: A hybrid vertex mapping solution for dynamic resource allocation," *J. Electr. Comput. Eng.*, vol. 2012, no. 1, 2012, Art. no. 5.
[21] J. Ding, T. Huang, J. Wang, W. Hu, J. Liu, and Y. Liu, "Virtual network embedding through node connectivity," *J. China Univ. Posts Telecommun.*, vol. 22, no. 1, pp. 17–23, 2015.
[22] L. X.-G. H. Jin-Peng and G. Q.-Y. W. Tian-Yu, "A virtual network embedding approach to preserving node closeness," *Chin. J. Comput.*, vol. 35, no. 12, pp. 2492–2504, 2012.
[23] R. Zhuo-Ming, S. Feng, L. Jian-Guo, G. Qiang, and W. Bing-Hong, "Node importance measurement based on the degree and clustering coefficient information," *Acta Physica Sinica*, vol. 62, no. 12, pp. 505–505, 2013.
[24] M. Yu, Y. Yi, J. Rexford, and M. Chiang, "Rethinking virtual network embedding: Substrate support for path splitting and migration," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 17–29, 2008.



PEIYING ZHANG is currently pursuing the Ph.D. degree in information and communication engineering with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications. He is also a Lecturer with the College of Computer & Communication Engineering, China University of Petroleum, East China. His research interests include network virtualization and future network architecture.

HAIPENG YAO is currently a Lecturer with the School of Information and Communication Engineering, State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications. His main research interests include future network architecture, big data for networking, and the architecture and key technology of the new generation mobile communication system.

YUNJIE LIU received the B.S. degree in technical physics from Peking University, Beijing, China, in 1968. He is currently the Academician of the China Academy of Engineering, the Chief of the Science and Technology Committee, China Unicom, and the Dean of the School of Information and Communications. His current research interests include next generation network, network architecture, and management.