

# Virtual Network Embedding Algorithm Based on Resource Locality Index

Bing WU<sup>1, a</sup>, Yuxing MAO<sup>2, b</sup>

<sup>1</sup>Air Defence Forces Academy, Zhengzhou 450052, China

<sup>2</sup>PLA University of Science and Technology, Nanjing, 210007, China

<sup>a</sup>email: lylglg@163.com, <sup>b</sup>email:kxu20112012@163.com

**Keywords:** Virtual Network; Embedding; Resource Aggregation

**Abstract.** We propose a node local resource aggregation index and then we also propose a virtual network embedding algorithm based on this index. Simulation results show that our proposed algorithm outperforms other embedding algorithms in virtual network request acceptance ratio, long-term average revenue and revenue cost ratio.

## Introduction

Network ossification is a big challenge to the development of the Internet [1]. Nowadays Network virtualization is getting more and more focus for its efficiency in helping network innovation. Network virtualization maintains multiple virtual networks in a shared physical infrastructure through virtualization technology, each virtual network uses its own routing strategy and network protocol. A virtual network is a service slice composed of virtual nodes and virtual links based on client's request and virtual network embedding aims to figure out how to match these virtual nodes/links to physical nodes/links in substrate network. Virtual network embedding is the main concern in network virtualization.

Because of its multiple objectives and constraints, virtual network embedding problem is proven NP-hard and heuristic algorithms are used to solve this problem. Zhang et al. propose a heuristic algorithm using node/link load balance method [4]. Yu et al. propose a multi-commodity embedding algorithm based on splittable substrate links [5]. Houidi et al. proposes two-stage embedding algorithm for the first time, it separates virtual network embedding process into node mapping stage and link mapping stage [6]. Chowdhury et al. introduce path cutting and migration technology into substrate network to enhance the flexibility of embedding process [7].

To keep balance between the performance of embedding algorithm and time complexity. In recent years, more and more researches tend to consider link mapping constraints during node mapping process. Such methods embed virtual network requests to substrate network with affluent resources using topology of substrate/virtual nodes, which increases acceptance ratio and utilization of substrate network. Chen et al. propose an embedding algorithm based on node degree, it takes nodes with low degree as edge nodes and tries to map virtual nodes to edge nodes [13]. The defect of this algorithm is that node degree is not good enough to evaluate the importance of nodes. Qing et al. propose an virtual network embedding algorithm based on K-core decomposition, the algorithm separates virtual nodes into core nodes and edge nodes and embeds these nodes respectively [14]. However, K-core decomposition separates nodes in a coarse grain and is not suitable for some topologies. Works in [15-29] design virtual network embedding algorithms using virtual nodes topology information so that some above mentioned problems are avoided. However, there are still following defects: (1) depending on overall information to evaluate the importance of nodes, as a result, increases the time complexity of the algorithm; (2) there is no controlling mechanism for the distance between carrying substrate nodes and more resources are consumed to accept virtual network requests.

To overcome above problems, this paper first introduces a node evaluation method based on its local information and resource aggregation index. The proposed method evaluates the importance of targeting nodes according to the available resources of this node and its two-layer neighbors. Secondly, this paper proposes a virtual network embedding algorithm, VNE-RC, based on the

proposed evaluation method. The proposed embedding algorithm maps virtual nodes to substrate nodes with affluent available resources such that embedding consumptions are reduced. In addition, our proposed embedding algorithm has higher acceptance ratio and lower average execution time.

## Problem Analysis

### Network Model

The system model of virtual network embedding problem is as follows:

**Substrate Network:** substrate network is modeled as a weighted undirected graph  $G^s = (V^s, E^s, C^s, B^s)$ , where  $V^s$  and  $E^s$  are the sets of substrate nodes and links, respectively.  $C^s$  and  $B^s$  are CPU and bandwidth capacity of substrate nodes and links, respectively.  $P_s$  is the set of loop-free substrate paths. Fig. 1(b) and (d) represent a substrate network, numbers in the rectangle indicate the available CPU resources of nodes and numbers on the line indicate the available bandwidth.

**Virtual Network Request:** virtual network request is modeled as a weighted undirected graph  $G^r = (V^r, E^r, C^r, B^r)$ , where  $V^r$  and  $E^r$  are the set of virtual nodes and virtual links, respectively.  $C^r$  is the CPU resource constraint of virtual nodes and  $B^r$  is the bandwidth resource constraints of virtual links.  $T_a$  and  $T_b$  indicates the arrival time and duration of virtual network requests. After a virtual network request arrives, substrate network allocates resources which satisfy its constraints  $C^r$  and  $B^r$ . After a virtual network completes its given period of time  $T_d$ , substrate network collects resources allocated to it. Fig. 1(a) and (c) indicate two virtual network requests, numbers in the rectangle indicates the CPU resource constraints of virtual nodes and numbers on the line indicates the bandwidth resource constraints of virtual links.

**Virtual Network Embedding:** virtual network embedding problem is defined as a mapping from virtual network request  $G^r$  to substrate network  $G^s$ , which satisfies  $C^r$  and  $B^r$ :

$$M : G^r (V^r, E^r) \mapsto G^s (V^{sub}, P_s'),$$

Where  $V^{sub} \subset V_s$ ,  $P_s' \subset P_s$ . There are two main components in the embedding process:

(1) Node mapping:

$$M_N : V^r \mapsto V^{sub},$$

Node mapping maps virtual nodes to substrate nodes whose available CPU capacity are more than their constraints.

(2) Virtual link mapping

$$M_L : E^r \mapsto P_s'$$

Virtual link mapping maps virtual links to substrate paths whose available bandwidth capacity are more than their constraints.

Fig. 1(b) presents a virtual network embedding result. The virtual node embedding result is  $\{a \mapsto B, b \mapsto C, c \mapsto F, d \mapsto E\}$ , the virtual link embedding result is  $\{(a,b) \mapsto (B,C), (a,c) \mapsto (B,F), (c,d) \mapsto (F,E), (d,b) \mapsto (E,C)\}$ . Fig. 1(d) presents another virtual network embedding result, the virtual node embedding result is  $\{a \mapsto A, b \mapsto C, c \mapsto E\}$ , the virtual link embedding result is  $\{(a,b) \mapsto (A,B,C), (a,c) \mapsto (A,F,E)\}$ .

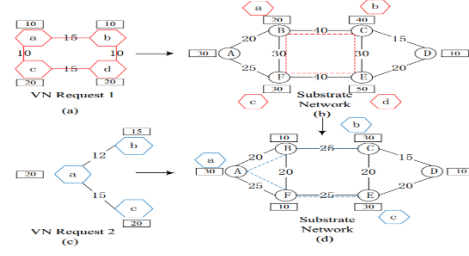


Fig. 1 Virtual Network Embedding Model

**Virtual network request revenue model:** given a virtual network request  $G^r$ , the revenue of infrastructure provider is defined as:

$$R(G^r) = \begin{cases} R_0(G^r) \cdot T_d, & \text{if } G^r \text{ accepted} \\ 0, & \text{otherwise} \end{cases}$$

where  $T_d$  is the duration of virtual network request  $G^r$ ,  $R_0(G^r)$  is the revenue per time unit and is defined as:

$$R_0(G^r) = \alpha \cdot \sum_{v^r \in V^r} C^r(v^r) + \beta \cdot \sum_{e^r \in E^r} B^r(e^r)$$

where  $\alpha$  and  $\beta$  are price of per unit CPU and bandwidth resource.

#### Embedding Objective

The main objective of this paper is to maximize the long-term revenue of infrastructure provider, which is defined as:

$$R_{G^s} = \lim_{T \rightarrow \infty} \frac{\sum_{G^s \in \Omega_T} R(G^r)}{T}$$

where  $\Omega_T = \{G_r | 0 \leq T_a \leq T\}$  represents virtual network requests which is accepted before time point  $T$ . We can see that the revenue of infrastructure provider is related to following two indexes:

#### Virtual network acceptance ratio:

$$p_a = \lim_{T \rightarrow \infty} \frac{\phi_a(T)}{\phi(T)}$$

where  $\phi_a(T)$  and  $\phi(T)$  are the number of accepted and arrival virtual network requests, respectively.

#### Virtual network requests revenue-cost ratio:

$$\varphi(G^r) = \frac{R_0(G^r)}{C_0(G^r)}$$

where  $C_0(G^r)$  is the resource consumed to accept virtual network request  $G^r$  in time unit, it is defined as:

$$C_0(G^r) = \sum_{v^r \in V^r} C^r(v^r) + \sum_{e^r \in E^r} |M_L(e^r)| \cdot B^r(e^r)$$

where  $|M_L(e^r)|$  is the number of hops of substrate path  $M_L(e^r)$ .

### Virtual network embedding algorithm based on node local resource index

#### Node Evaluation Method Based on Local Resource Aggregation Index

In complex networks, in order to evaluate the importance of nodes, its topology should be considered. [19] proposes a ranking method using semi-local centrality information. This method uses two-layer neighbor information to evaluate its importance and points out that the time complexity of this method increases linearly as the scale of the network increases. Experiment

results show that this method reaches better result than PageRank and LeaderRank. However, this algorithm only considers the weight of links between nodes but not evaluate the available resources of the nodes itself and cannot be applied to virtual network embedding problem directly. This paper improves this method and make it applicable to solve virtual network embedding problem.

**Definition 1** Node resource index:

$$Rf_L(v^s) = d \cdot pc(v^s) + (1-d) \cdot \frac{b(u, v^s)}{pb(u)} \cdot Q(u)$$

$$Q(u) = d \cdot pc(u) + (1-d) \cdot \sum_{w \in N(u)} \frac{b(u, w)}{pb(w)} \cdot pc(w)$$

$$pc(i) = \frac{C^s(i)}{\sum_{j \in V} C^s(j)}$$

$$pb(i) = \sum_{j \in N(i)} b(i, j)$$

In the above definition,  $b(i, j)$  is the available bandwidth of the link between node  $i$  and  $j$ , if there is no link between the two nodes, its value is 0.  $N(i)$  is the set of adjacent nodes of node  $i$  and  $V$  is the set of all nodes in the network.  $d$  is a constant between 0 and 1.

The aggregation index of node indicates the relationship between targeting node and its neighbors. The higher the aggregation index, the closer relationship between the targeting nodes and its neighbors.

**Definition 2** Node aggregation index:

$$cf(v_s) = \frac{\left( W^{\lfloor \frac{1}{3} \rfloor} \right)_{v_s v_s}^3}{d_{v_s} (d_{v_s} - 1)}$$

$$W^{\lfloor \frac{1}{3} \rfloor} = \left\{ w_{ij}^{\frac{1}{3}} \right\}$$

$$w_{ij} = \frac{b(i, j)}{\max bw}$$

In the above definition,  $d_{v_s}$  is the degree of node  $v_s$ ,  $\max bw$  is the maximum bandwidth value of the network topology.

This paper collaborates node local resource index and node local aggregation index together and proposes node local resource aggregation index to evaluate the node importance.

**Definition 3** Node local resource aggregation index

$$LRI(v_s) = Rf(v_s) \ln Cf(v_s)$$

The detail of node local resource aggregation index is presented in table 1. The time complexity of the Algorithm is  $O(|V| \cdot D^2(V))$ , where  $V$  is the number of nodes in the network topology and  $D(V)$  is the average number of nodes in the network topology.

**Algorithm 1: NLRAIA(Node Local Resource Aggregation Index Algorithm)**Input: network topology  $G(V, E)$ Output: node local resource aggregation index vector  $R$ 

```
(1)  $R \leftarrow 0$ ;  
(2) for each  $v_i \in V$  do  
(3)    $r_i = d \cdot pc(i)$ ;  
(4)   for each  $j \in neighbor(i)$  do  
(5)      $q_j = d \cdot pc(j)$   
(6)     for each  $k \in neighbor(j)$  do  
(7)       if  $k == i$   
(8)         continue;  
(9)       end if  
(10)       $q_j += (1-d) \cdot \frac{b(j,k)}{pb(k)} \cdot pc(k)$   
(11)     end for  
(12)      $r_i += (1-d) \cdot \frac{b(i,j)}{pb(j)} \cdot q_j$ ;  
(13)      $r_i = r_i \cdot Cf(i)$ ;  
(14) end for
```

**2-layer aggregation virtual node embedding algorithm**

To fully utilize the CPU and bandwidth resources of substrate network, we propose 2-layer aggregation virtual node embedding algorithm based on node local resource aggregation algorithm. The algorithm has following advantages: (1) it maps virtual nodes with high resource constraints to substrate nodes with affluent resource capacity, so that more virtual network requests can be accepted; (2) it compresses virtual network requests to substrate nodes and links with small distances so that less resources are consumed. The detail of the algorithm is presented in table 2.

**Algorithm 2: NM-LRAI(Node Mapping Algorithm based on Local Resource Aggregation Index)**Input:  $G_r(V_r, E_r)$ ,  $G_s(V_s, E_s)$ Output: virtual node embedding strategy  $M_N$ 

```
(1) sort virtual nodes and substrate nodes according to their local resource
    index;
(2) while there is virtual node unmapped in  $V_r$  do
(3)      $vNode \leftarrow$  virtual node in  $V_r$  with highest resource aggregation value
(4)     for each  $sNode \in V_s$  do
(5)         if  $sNode$  has already carried some virtual node
(6)             continue;
(7)         end if
(8)         else if  $C_{vNode} > C_{sNode}$ 
(9)             continue;
(10)        else
(11)            map  $vNode$  to  $sNode$ 
(12)            set  $vNode$  to already mapped, set  $sNode$  to already
            carried;
(13)             $Map\_neighbor(vNode, sNode)$ ;
(14)        end if
(15)    end for
(16)    If  $vNode$  is not successfully mapped do
(17)        return false;
(18)    end if
(19) end while
(20) return true
```

## Simulation Performance

### Simulation Settings

We use computing platform with 3.10GHz Intel core i7-4600 and use C++ program to simulate the arrival and departure of virtual network requests. We use GT-ITM topology generator [23] to construct substrate network topology and virtual network request topology. To facilitate our comparison, we generate a substrate network with 100 nodes and 2000 virtual network request topologies. The CPU and bandwidth resource of substrate nodes and links follows a uniform distribution in [50, 100] and the connect probability of substrate nodes is 0.2. The number of virtual nodes of a virtual network request follows a uniform distribution in [2, 20], the connect probability of virtual nodes is 0.5. The resource constraints of virtual nodes and links follow a uniform distribution in [0, 50].

### Evaluation Metrics

Similar to existing works [14-16], we use following evaluation metrics to estimate the algorithm we proposed in this paper.

- 1) Virtual network request acceptance ratio: this metric is used to evaluate the percentage that how many virtual networks requests are accepted by the substrate network. This metric represents the main performance of the embedding algorithm.
- 2) Long-term average revenue of substrate network: this metric evaluates the revenue the embedding algorithm brings to substrate network.
- 3) Long-term revenue-cost ratio: this metric is used to evaluate the revenue and cost that the

embedding algorithm generates.

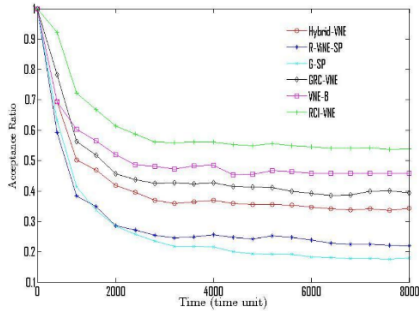


Fig. 2 Acceptance ratio

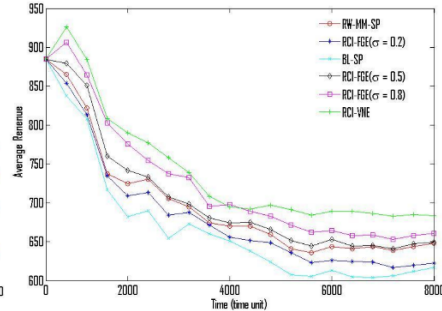


Fig. 3 long-term average revenue

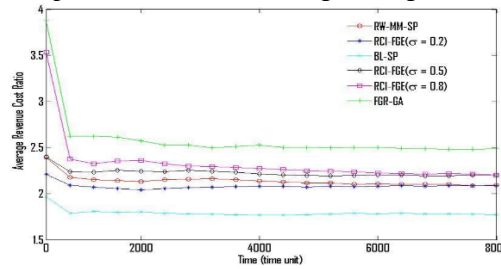


Fig. 4 Revenue cost ratio

Fig. 2 shows the acceptance ratio of different embedding algorithms, we can see that in the beginning of the embedding process the acceptance ratio of each embedding algorithm is high. The reason behind this scene is that the resource of substrate network is relative higher. However, the acceptance ratio of each embedding algorithm goes down as time continues. Our algorithm has the highest acceptance ratio among all the embedding algorithms.

Fig. 3 and Fig. 4 shows the long-term average revenue and revenue cost ratio of each embedding algorithm. We can see that our embedding algorithm reaches the highest revenue and revenue cost ratio for that our algorithm maps virtual nodes to substrate nodes with small distance and thus less resources are consumed.

Through experiments, we can see that our embedding algorithm outperforms other embedding algorithms in virtual network acceptance ratio, long-term average revenue and revenue cost ratio.

## Conclusion

In this paper, we propose a node local resource aggregation index and then we also propose a virtual network embedding algorithm based on this index. Simulation results show that our proposed algorithm outperforms other embedding algorithms in virtual network request acceptance ratio, long-term average revenue and revenue cost ratio.

## References

- [1] Turner J. S. and Taylor D. Diversifying the Internet [C]. Proceedings of IEEE Conference on Global Telecommunications, St. Louis, 2005: 755- 760.
- [2] Anderson T, Peterson L, Shenker S, et al.. Overcoming the internet impasse through virtualization [J]. IEEE Computer Magazine, 2005, 38(4): 34- 41.
- [3] Andersen D G. Theoretical approaches to node assignment [M]. New York: Computer Science Department, 2002: 86-123.
- [4] Y. Zhang, M. Ammar, et al.. Algorithm for assigning substrate network resources to virtual network components[C]. Proceedings of IEEE INFOCOM, Barcelona, 2006: 1- 12.
- [5] Yu M., Yi Y., Rexford J., et al.. Rethinking virtual network embedding: substrate support for path splitting and migration [J]. ACM SIGC OMM Computer Communication Review, 2008, 38(2): 17- 29.

- [6] Houidi I., Louati W., et al.. A distributed virtual network mapping algorithm [C]. IEEE International Conference on Communication, Beijing, 2008: 5634- 5640.
- [7] M. Chowdhury, M. Rahman, et al.. ViNEYard: Virtual network embedding algorithms with coordinated node and link mapping [J]. IEEE/ACM Transactions on Networking, 2012, 20(1): 206-219.
- [8] Melo M., Sargento S., Killat U., et al.. Optimal virtual network embedding: Node-link formulation [J]. IEEE Transactions on Network and Service Management, 2013, 10(4): 356- 368.
- [9] Jens Lischka, H Karl, et al.. A virtual network mapping algorithm based on subgraph isomorphism detection[C]. Proceedings of the 1st ACM workshop on Virtualized infrastructure systems and architectures, Barcelona, 2009: 81-88.
- [10] Chen X., Luo Yan, Jie Wang, et al.. Virtual network embedding with border matching [C]. IEEE Fourth International Conference on Communication Systems and Networks, Bangalore, 2012: 1-8.
- [11] Qing S., Liao J., et al.. Hybrid Virtual Network Embedding with K-core Decomposition and Time-oriented Priority[C]. IEEE International Conference on Communications, Ottawa, 2012: 2695- 2699.
- [12] Gong L., Wen Y., Zhu Z., et al.. Toward profit-seeking virtual network embedding algorithm via global resource capacity [C]. Proceedings of IEEE INFOCOM, Toronto, 2014: 1- 9.
- [13] Wang Z, Han Y, Lin T, et al.. Virtual network embedding by exploiting topological information [C]. IEEE Global Communications Conference, California, 2012: 2603- 2608.
- [14] Cheng X, Su S, Zhang Z, et al.. Virtual network embedding through topology-aware node ranking [J]. ACM SIGCOMM Computer Communication Review, 2011, 41(2): 38-47.
- [15] Cui H, Gao W, Liu J, et al.. A virtual network embedding algorithm based on virtual topology connection feature[C]. IEEE 16th International Symposium on Wireless Personal Multimedia Communications, Atlantic City, 2013: 1-5.
- [16] Huang T, Liu J, Chen J, et al.. A topology-cognitive algorithm framework for virtual network embedding problem [J]. Communications, China, 2014, 11(4): 73-84.
- [17] Cui H, Tang S, Huang X, et al.. A Novel Method of Virtual Network Embedding Based on Topology Convergence-Degree[C]. IEEE International Conference on Communications Workshops, Budapest, 2013: 246- 250.
- [18] Chen D, Lv L, Shang M S, et al.. Identifying influential nodes in complex networks [J]. Physica: a Statistical mechanics and its applications, 2012, 391(4): 1777-1787.
- [19] Fischer A, Botero J F, Till Beck M, et al.. Virtual network embedding: A survey [J]. IEEE Communications Surveys & Tutorials, 2013 , 15( 4): 1888-1906.