Dynamic Virtual Network Embedding Based on Service Request Aware

Ningning Liang[1], Julong Lan[1], Peng Wang[1], Kun Long[2]

[1] National Digital Switching System Engineering & Technological Research & Development Center, Zhengzhou, 450002, China

[2] Unit 95879, Chengdu, 610081, China

email: ning_ning216@163.com

Keywords: service request; dynamic embedding; aware; effective utilization

Abstract. Aiming at the virtual network embedding problem under dynamic service request, Service Request Aware-based Dynamic Virtual Network Embedding (SRAD) algorithm is proposed. We first present virtual network embedding model, and then define virtual network construction cost according to the dynamic characteristic of service request. Our goal is to find the optimal reconfiguration policies by awaring service request type and changing trends, which can minimize the overall construction cost using substrate resource. Experimental results show that the proposed algorithm satisfies the service request, achieves higher success ratio and gains higher revenue/cost ratio for substrate network comparing with the existing algorithms.

Introduction

Network virtualization is considered to be a prospective solution for solving network ossification problem. It allows multiple virtual networks to share the same physical infrastructure according to service request and the substrate network resource status, such as network topology, resource state, etc [1].

Current research on virtual network embedding has formed lots of mature algorithms and mechanisms [2-4]. Aiming at load balancing, Zhu et al. [5] presented an algorithm which maps virtual nodes to substrate closer and lower load nodes, and maps virtual links by shortest path. Qi et al. [6] proposed a balanced adaptive algorithm for constructing virtual network. It migrates by the principle of high link bandwidth consumption and minimum cost priority when link failure occurs. Koizumi et al. [7] presented a virtual network top ology controlling method which can adapt network environment automatically. This algorithm is based on the attractor model, which can dynamically reconfigure the structure of attractors according to the changing network environment.

Nowadays, virtual network embedding algorithms generally assume the static service requests, and dynamically update virtual network construction only for substrate network changes. However, not only the substrate resource, but also service request and corresponding virtual network request is dynamically changing over time. In this paper, we focus on the issues that how to dynamically construct virtual network in a range of changing service request. Firstly, we build the virtual network embedding model, and then with the goal of minimum construction cost, we present a virtual network embedding algorithm according to the service request types and diverse changing state.

Virtual Network Model

A. Substrate resource

We abstract the substrate network into an undirected graph $G^s=(N^s, L^s, C^s)$, where N^s and L^s respectively represent a set of substrate nodes and links. C^s denotes substrate resources, including node resources and link resources. We also introduce link resource unit cost and node resource unit cost.

B. Service request model

The service request is expressed as an undirected graph $G^r=(N^r, L^r, C^r)$, where N^r and L^r

respectively represent the set of virtual nodes and virtual links, and they are subset of N^s and L^s. R^r denotes the constraint to virtual nodes and virtual links.

C. Virtual network embedding

Virtual network embedding can be expressed as a mapping which satisfies the constraint R^r and a subset from G^r to G^s in an undirected graph G^r.

$$M: G^r \mapsto G^v, \quad G^v = (N^v, L^v, C^v)$$

where N^v and L^v are the subset of N^s and L^s, respectively. G^v represents the service ability of the substrate network. Virtual network embedding can be divided into node mapping and link mapping.

Node Mapping: $M^N: (N^r, R_N^r) \mapsto (N^v, C_N^v)$;   Link Mapping: $M^L: (L^r, R_L^r) \mapsto (L^v, C_L^v)$

where R_N^r and R_L^r represent the constraints on the virtual nodes and virtual links. C_N^v and C_L^v denote service supporting ability of substrate nodes and links respectively.

D. Constrain

For the limited substrate resources, the purpose of dynamic virtual network embedding is to minimize the construction cost. The construction cost of virtual network is composed of occupancy cost and reconstruction cost. Occupancy cost refers to cost for mapping virtual network based on service requests by using substrate resources. Reconstruction cost is the consumption caused by accordingly adjustment for mapped virtual network due to the dynamic change of service requests.

1). Occupancy Cost

$$Cost_u(\Delta t) = \int_0^{\Delta t} f(G^r, req(t))dt = \int_0^{\Delta t} [req_n(t)\cos t_n + req_l(t)\cos t_l]dt \tag{1}$$

Where $req_n(t)$ and $req_l(t)$ represent the service request of substrate node and link resources respectively. $cost_n$ and $cost_l$ denote node resource unit cost and link resource unit cost respectively.

2). Reconstruction Cost

In this paper, we use the number of changed substrate resource and remapping time as an approximate measurement for reconstruction cost.

$$Cost_r(\Delta t) = \sum f(cha(n,l), T) \tag{2}$$

Where $cha(n, l)$ denotes the sum of changed substrate nodes and links, and T is the remapping time.

3). Overall Cost

$$Cost(\Delta t) = (1-\beta) \ Cost_u(\Delta t) + \beta Cost_r(\Delta t) \tag{3}$$

where β is the variable to control the weight of occupancy cost and reconstruction cost.

Thus, the minimized overall cost can be expressed as follows.

$$\min Cost(\Delta t) \tag{4}$$

$$\text{s.t. } d_P < D_l, \quad \sum_{n=1}^N C_{bm_n}^r < C_{bm}^s, \quad \sum_{n=1}^N CP_{cpu_n}^r < CP_{cpu}^s, \quad \sum_{n=1}^N Bw_n^r < Bw^s \tag{5}$$

Where, Constraint (5) represents that the delay of selected path must less than the delay requirement of virtual network request, and the sum of resource which is consumed by virtual nodes or links must less than the maximal capacity of substrate nodes and links.

Service Request Aware-based Dynamic Virtual Network Embedding (SRAD)

The algorithm can apperceive the dynamic change of service requests and substrate network resource. When the service request increases over time, the increased resources are firstly provided by the mapped substrate node or link. If the corresponding node or link has not enough resources available to meet the service request, it will have to remap the virtual nodes or links to other substrate nodes or links with enough resources.

When the resource migration is needed, we select migration object based on the service type. For example, we prefer to use the closer nodes in the initial mapping when migrating delay-sensitive service. In this way, we ensure a smaller time delay for migrating virtual network. While migrating bandwidth-sensitive service, we will give preference to the nodes or links which have abundant substrate resource.

If the current resource demand is less than the original, the virtual network will release this part of the resources to other service in order to enhance the utilization rate of the substrate resources.

When the virtual network request life-time expires, the previous resources will be completely released, then the part of the original resource will be used to map virtual network which can meet other service request. The flow chart of the algorithm is shown in Fig.1.
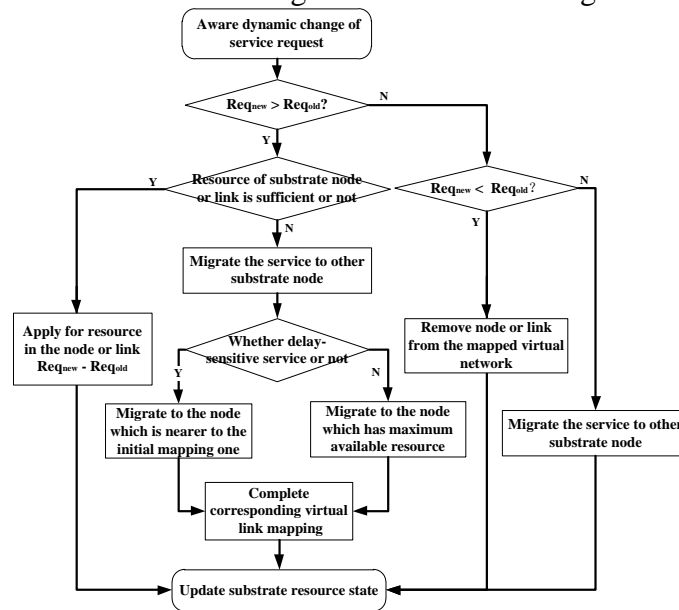

Fig.1. The flow chart of SRAD

Evaluation Results

Substrate network topology in the experiment consists of 100 nodes which are produced randomly by BRITE. Nodes connection degree is 0.5. Node and link resources are distributed uniformly between 50 and 100. The number of VN request nodes is distributed uniformly between 2 and 10. Link bandwidth, node CPU processing capacity and memory capacity are all in uniform distribution between 0 and 30. VN requests arrive in a Poisson process with an inter-arrival time of 20 time units and intensity is 10. Lifetimes of the VN requests follow an exponential distribution with mean 1000 time units. The unit cost of substrate nodes and links is 1. In order to ensure the accuracy of the simulation results, all the results presented for an experiment are an average of 20 runs of simulations.

(a) Revenue/cost rate

(b) Mapping success rate
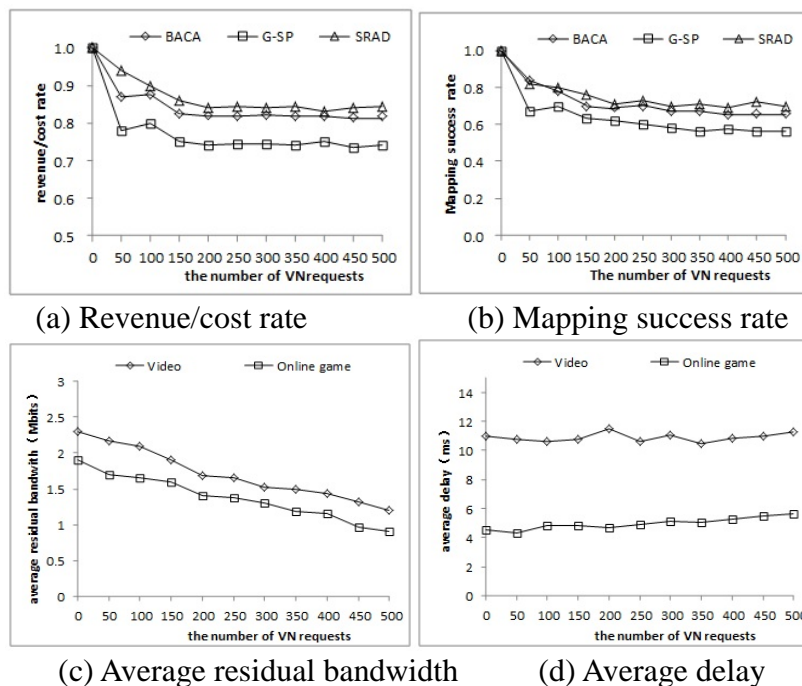
(c) Average residual bandwidth   (d) Average delay

Fig.2. The experimental results

Fig.2 (a) shows the revenue/cost rate of different virtual network mappings. With the number of virtual network requests increasing, the algorithms will gradually reach a balance. SRAD maps virtual network with the goal of lowest cost and dynamically construct the virtual network according to the changing service requests, so it gets higher revenue/cost rate than others. G-SP algorithm is the lowest one because it only considers land balancing. Fig.2 (b) shows the success rates of different virtual network mappings along with the increasing number of virtual network request. BACA and SRAD perform well because of the migration strategy, and better than G-SP which is on the basis of separating the node mapping and link mapping into two phases. The bandwidth and delay of different service is shown in Fig.2 (c) and Fig.2 (d). Because of dynamically awareness and migration according to type of services, the bandwidth-sensitive service and delay-sensitive service have got the corresponding supporting.

Conclusion

Aiming at the problem of mapping virtual network dynamical according to the change of service requests, we propose a service request aware-based dynamic virtual network embedding algorithm. The algorithm is aware of real-time service request, and adjusts the substrate resource allocation dynamically according to the change trend. Experimental results show that the proposed algorithm achieves corresponding service supporting and higher success ratio, and gains higher revenue/cost ratio, comparing with the existing algorithms.

Acknowledgement

References

[1] Andreas Fischer et al., Virtual Network Embedding: A Survey, IEEE Communications Surveys & Tutorials, 2013: 1888-1906.

[2] Rafael L. Gomes et al., A Bandwidth-Feasibility Algorithm for Reliable Virtual Network Allocation, IEEE 28th International Conference on Advanced Information Networking and Applications, 2014: 504- 511.

[3] Sarah Kareche et al., A New Robust Heuristic for Assigning Substrate Network Resources to Virtual Networks, International Conference on Advanced Networking Distributed Systems and Applications, 2014: 47- 52.

[4] Zhiming Wang et al., Coding-aware Virtual Network Mapping for Surviving Single Link Failure, IEEE ICC, 2014: 3025- 3030.

[5] Y. Zhu et al., Algorithms for assigning substrate network resources to virtual network components. In Proc. IEEE INFOCOM, April, 2006: 1- 12.

[6] Ning Qi et al., Research on Balanced Construction Algorithm of Virtual Network, Journal of Electronics&Information Technology, 2011: 1301- 1306.

[7] Yuki Koizumi et al., Adaptability of Virtual Network Topology Control Based on Attractor Selection against Multiple Node Failures. OptoElectronics and Communications Conference held jointly with 2013 International Conference on Photonics in Switching (OECC/PS), 2013: 1-2.