Computer Networks 93 (2015) 448–459

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Energy aware virtual network embedding with dynamic demands: Online and offline☆

Zhongbao Zhang, Sen Su*, Junchi Zhang, Kai Shuang, Peng Xu

State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China

ARTICLE INFO

Article history:
Received 15 February 2015
Revised 10 August 2015
Accepted 19 September 2015
Available online 19 October 2015

Keywords:
Network virtualization
Virtual network embedding
Dynamic demands
Energy

ABSTRACT

In Network as a Service model in cloud computing, how to efficiently embed virtual networks with both node and link demands into a shared physical network, namely virtual network embedding, has attracted significant attention. Most of prior studies on this problem have the following two limitations: (i) they assumed that the virtual network demands are constants, which does not hold in real-world network since such demands may vary a lot over time; (ii) their primary goal was to achieve more revenues for the physical network, with no consideration of the energy cost, which has become a more and more critical issue. In this paper, we bridge the gaps and study the energy aware virtual network embedding problem with dynamic demands. Specifically, we first model the dynamics of virtual network demands as a combination of a Gaussian distribution and a daily diurnal pattern. We then design two efficient heuristic algorithms by leveraging the dynamic characteristic of virtual network demands to minimize the energy consumption while keeping a high revenue for the physical network. One algorithm processes the virtual network requests one by one while the other one processes them group by group. We implemented these two algorithms in C++ and performed side-by-side comparisons with the prior algorithm. Extensive simulations show that our algorithms significantly reduce the energy cost by up to 25% over the state-of-the-art algorithm, while maintaining near the same revenue.

1. Introduction

Cloud computing has emerged as a promising paradigm in several fields (*e.g.*, business, engineering and health-care) due to its flexibility, scalability and virtualized infinite resources [1]. Depending on the granularity or levels of providing services to the users, cloud computing offers several fundamental models, such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). Recently, network virtualization has been proposed to virtualize resources at a level of the network [2]. By leveraging such technique in cloud computing, the providers could offer another additional service based on the Network as a Service (NaaS) model [3].

In the NaaS model, two roles are involved: infrastructure providers (InPs) and the service providers (SPs): the InPs manage and operate a physical network (PN); as one of the multiple tenants, the SPs request and rent a virtual network (VN) on the PN from InPs for providing personalized services (*e.g.*, VoIP, online gaming or content distribution). The

☆ Some preliminary results of this paper has been accepted and will appear in the IEEE International Conference on Communications (ICC) in 2015. In this extended version, we have made the following improvements. First, besides the online virtual network embedding which processes requests one by one, we designed a new offline algorithm (Section 4) for processing requests group by group to improve the energy efficiency. Second, we evaluated our algorithms from more aspects in the experiment. Third, we added more reference papers in the related work (Section 6). Fourth, we significantly improved the presentation by rewriting several sections of this paper.

* Corresponding author. Tel.: +86 10 61198129.
E-mail addresses: zhongbaozb@gmail.com (Z. Zhang), susen@bupt.edu.cn (S. Su).

http://dx.doi.org/10.1016/j.comnet.2015.09.036
1389-1286/

key benefits of network virtualization in cloud computing are three-fold. First, it enables resource sharing at the network-level to make most efficient use of the PN. Second, it is possible for the VN to expand or shrink as needed. Third, it offers a pay-as-you-go pricing model for the users for renting their personalized topology network.

Two categories of constraints are present in a VN: node constraints and link constraints. The node constraints are typically on capacities of nodes (such as CPU, memory and storage capacity) and the relative location in the topology of the physical network. The link constraints are typically on communication bandwidth and latency. When the InP receives a sequence of VN requests from SPs, the InP needs to map the virtual nodes and links of these VNs onto the physical nodes and links in the PN, which is well known as the VN embedding problem.

Recently, the VN embedding problem has received significant attention. The primary goal of prior studies [4–14] is to design efficient embedding methods to maximize the revenue for accommodating the VNs in the shared PN. However, they ignored the energy cost, which has become an increasingly important factor for the InP. In our previous studies [15,16], we proposed to minimize the energy consumption when conducting VN embedding. Similar to most of prior art, these two studies still consider that the demands of virtual nodes and virtual links are constant values. However, in real-world applications, these demands may vary a lot over time and thus this consumption does not hold. In this paper, we revisit the VN embedding problem from both of the aspects of energy and dynamic demands. However, to address such a problem, there will be two challenges involved.

1. The first one resides in the modeling: how to capture and model the variation of VN demands in VN embedding.
2. The second one resides in the algorithm design: how to leverage the dynamic characteristic of VN demands to minimize the energy cost for the InP.

To address the first challenge, following the real demand traces and related studies [14,17–20], we observe that the VN demands exhibit a daily diurnal pattern and different demands have peaks at different time. Therefore, we model the VN demands as a combination of the Gaussian distribution and the daily diurnal pattern in this paper.

To address the second challenge, we formulate the VN embedding problem with dynamic demands to an integer linear programming (ILP) and design two heuristic algorithms to solve this formulation. One algorithm processes the VN requests one by one while the other one processes them group by group. The first algorithm is called EAD-VNE, which has two steps. The first step is node mapping and the second step is link mapping. In the node mapping, we strike the resource balance of the PN, exploit the dynamic characteristic of VN demands and pre-consider the subsequent link mapping to minimize the node energy consumption while achieving high revenue. In the link mapping, we design a weighted shortest path algorithm with the preference of active nodes and opposite variation trends between the bandwidth demand and the bandwidth load. Based on EAD-VNE, we design the other algorithm called EAD-VNE-G. This algorithm queues the VN requests and processes them group by group. The benefit of this algorithm lies in that it can further leverage the dynamic characteristic of VN demands for different VN requests and thus help benefit the energy conservation when performing VN embedding.

Through extensive simulations, we show that our algorithms outperform the state-of-the-art algorithm [16] in terms of long-term average revenue and energy consumption. In particular, our proposed two algorithms reduce up to 16 and 25% energy consumption than the state-of-the-art algorithm, respectively, while keeping nearly the same revenue.

We make the following major contributions in this work:

1. To the best of our knowledge, we make the first attempt to study the energy aware VN embedding problem with dynamic demands.
2. We formulate the problem into an ILP and design two heuristic algorithms, *i.e.*EAD-VNE and EAD-VNE-G, to optimize the energy consumption while generating high revenues for the InP.
3. We conduct side-by-side comparisons between our algorithms and the state-of-the-art algorithm to demonstrate the energy efficiency of our algorithms.

We organize the remainder of this paper as follows. Section 2 presents the modeling and problem formulation. Sections 3 and 4 present our two proposed heuristic algorithms: one processes the VN requests one by one and the other one processes the VN requests group by group. Section 5 evaluates these two algorithms. Section 6 reviews the related work. Finally, Section 7 concludes this paper.

2. Modeling and formulation

In this section, we first model the network in Section 2.1 and energy consumption in Section 2.2, respectively. Then we will present the formulation of this problem in Section 2.3. Finally, we will give the performance metrics in Section 2.4. Before that, we first summarize the notations in Table 1, which will be used throughout this paper.

2.1. Network modeling

Physical Network (PN): We model a physical network as a weighted graph $G_p = (N_p, L_p)$, where N_p denotes the set of physical nodes and L_p denotes the set of physical links. For physical nodes, the attributes generally include CPU, memory, disk and relative location in the network. For physical links, the attributes generally include bandwidth and latency. Similar to studies [9,10,15,16] , we mainly consider the CPU and location as the node attributes and bandwidth as the link attribute. For example, at the bottom of Fig. 1, (c) shows a PN where the numbers near the nodes and links denote the available CPU capacity and bandwidth capacity, respectively. Note that our modeling, analysis, and algorithm in this paper can be easily extended to incorporate other attributes.

Virtual Network (VN): A sequence of virtual networks (VNs) arrive and depart over time. Similarly to G_p, we model each VN as a weighted graph $G_v = (N_v, L_v, t_a, t_e)$. Differently, in G_v, we have t_a and t_e for denoting the arrival time and expiration time of the VN, respectively. In N_v, C_u and D_u denote the CPU demand and required location of the virtual node u. In L_v, B_{uv} denotes the bandwidth demand for the