Computer Communications 80 (2016) 1–15

Contents lists available at ScienceDirect

Computer Communications

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

A modified ACO algorithm for virtual network embedding based on graph decomposition

Fangjin Zhu, Hua Wang*

School of Computer Science and Technology, Shandong University, PR China

ARTICLE INFO

Article history:
Received 17 October 2014
Revised 22 May 2015
Accepted 11 July 2015
Available online 17 July 2015

Keywords:
Network virtualization
Virtual network embedding
Ant colony optimization algorithm
Virtual network topology invariance
Topology decomposition

ABSTRACT

Network virtualization is a promising scheme to solve Internet ossification. A challenging problem in this scheme is virtual network embedding (VNE), which involves efficiently embedding multiple heterogeneous virtual networks into one or more physical networks. The VNE problem is known to be NP-hard and thus requires an approximate algorithm as a solution. This study models the VNE problem based on virtual network topology invariance and analyzes the shortcomings of a general embedding algorithm under different network topologies. A modified ant colony optimization algorithm is proposed based on network topology decomposition. A pre-computation algorithm is first proposed based on ant random walking to accelerate the recognition of the ring characteristics of a network topology. Pre-computation results are used to guide the decomposition of virtual networks and the embedding process of ring structures. The topology of a virtual network is decomposed into a combination of ring structures and tree structures, which have different characteristics. Different embedding algorithms are then designed for these structures. Point-disjoint paths are searched for any two virtual links to ensure the reliability of the network topology in the embedding process. The proposed algorithm shows an enhanced optimization performance, which is better than those of the ViNE-LB and GN-SP algorithms.

1. Introduction

An increasing number of applications with various qualities of service (QoS) requirements are being developed over the Internet. However, supporting different data delivery mechanisms suitable for these QoS requirements is difficult for traditional TCP/IP models. This problem is referred to as Internet ossification [1]. As a promising scheme to address this ossification problem, network virtualization is proposed to build a diversified Internet that supports a variety of network services and architectures through a shared substrate by decoupling network functionalities and infrastructures [2].

In this scheme, traditional Internet service providers (ISPs) are separated into infrastructure providers (InPs) and service providers (SPs). InPs maintain network resources, and SPs lease network resources to construct their own virtual networks (VNs). InPs need to embed VNs into their physical networks by mapping one physical node for a virtual node and searching one loop-free path for a virtual link. Thus, many VNs of different SPs would coexist in a physical network. Different mapping methods result in different resource consumption. Hence, InPs face the important problem of efficiently mapping these VNs into a physical network to gain the highest revenue. This problem is referred to as the virtual network embedding (VNE) problem.

The VNE problem involves both node mapping and link mapping. A well-known NP-hard multiple separator problem can be reduced to the VNE problem [3,4]. Therefore, the VNE problem is known to be NP-hard. Single node or link mapping can also be NP-hard, and solving them requires heuristic and approximate algorithms. Several methods have been proposed to address the VNE problem [5]. These techniques are summarized into the following categories: two-stage mapping methods [4,6–9], coordinated node and link mapping methods [10–12], and mapping methods based on graph theory [6,11–14]. However, common problems in research need to be discussed and addressed.

(1) ***General embedding algorithms are used for different topologies of VN requests***. Some of the aforementioned methods employ a general mapping technique for any VN requests and ignore the differences among VN topologies [4,6–12]. These differences always indicate different key relations that affect the embedding process (details are discussed in Section 3). A general algorithm cannot reflect these differences efficiently.
(2) ***Existing embedding algorithms based on graph decomposition are too simple***. Other methods attempt to reduce the

* Corresponding author. Tel.: +86 53188061653.
E-mail addresses: zhufj@sdu.edu.cn (F. Zhu), wanghua@sdu.edu.cn (H. Wang).

http://dx.doi.org/10.1016/j.comcom.2015.07.014
0140-3664/

complexity of mapping algorithms and propose some techniques based on graph decomposition and combination. In some works [6], network topology is divided into combinations of star structures, but the hardness of closed characteristics remains. In other works [13], network topology is divided into core networks and edge networks, and different existing mapping algorithms are used. However, the algorithms for mapping core and edge networks still fail to consider different types of topologies.

(3) ***Existing embedding algorithms may decrease the reliability requirement of SPs***. The topology of a VN always implicitly contains some requirements of the owners of VNs (SPs), such as the requirement of redundancy degree. However, most mapping methods ignore this requirement, resulting in different virtual links sharing one or more physical links or nodes [4,6–11]. This phenomenon can decrease the reliability of a VN, and some mutual backup virtual links in the design may fail simultaneously.

We propose an embedding algorithm that guarantees that the mapping networks generated in a substrate network (SN) satisfy the reliability requirements of SPs. The proposed algorithm attempts to improve the acceptance ratio while maximizing the revenue of an InP. The following are our main ideas and contributions to the VNE problem:

- ***Improved optimization performance for InPs***. To improve the optimization performance, a pre-computation algorithm is designed to obtain the ring characteristics of a graph (a VN request topology or SN topology). A VN topology is randomly divided into a combination of ring structures and tree structures by using the ring characteristics of the VN request. Different construction algorithms are then designed for these structures. An algorithm for ring structures is designed based on the ring characteristics of the SN. An algorithm for tree structures is designed based on tree-indexed random walking.
- ***Realization of reliability requirements of SPs***. To maintain the reliability of a VN, the concept of candidate node is proposed to design point-disjoint paths for the virtual links in the mapping process. In this way, a node (or a link) is prevented from being shared by different virtual links of the VN.
- *Convenience of resource management and protocol deployment for InPs*. Resource management and protocol deployment in VNs are eventually realized in SNs. In our algorithm, the unique disjoint SN path for each virtual link facilitates explicit resource management (such as resource reservation) and is convenient for deploying special data transmission protocols (such as multiprotocol label switching or MPLS) in SNs. In existing works, the sharing of nodes or links among SN paths may cause routing loops, which result in extra mechanisms (such as mechanisms for avoiding packet disorder). This will make resource management and protocol deployment increasingly difficult for InPs.

A modified ant colony optimization (ACO) algorithm is introduced to improve optimization performance. The ACO algorithm is a population-based algorithm, inspired by the foraging behavior of ants and proposed by Dorigo [15]. Ants communicate with one another through pheromone; based on this, the ACO algorithm uses pheromone to accelerate knowledge in problem solving; this algorithm has been successfully applied in many combination optimization problems [16]. In our algorithm, each ant randomly constructs a complete solution according to pheromone and heuristic information. The solutions are then evaluated with a fitness function that guides them toward optimization through multiple iteration processes.

Simulation results show that the idea of random selection in tree structure mapping (random root selection and random selected subtree growing) and ring structure mapping (random adjacent node selection) results in a flexible and sufficient graph traversal. Moreover, the ACO algorithm improves optimization performance. Thus, reliability is guaranteed, and high acceptance rate and revenue are obtained.

This paper is organized as follows. Section 2 summarizes related works on the VNE problem. Section 3 describes the background of the VNE problem and defines the optimization problem to be addressed. Section 4 proposes a modified ACO algorithm to solve the optimization problem. Section 5 discusses the simulation results and analysis. Section 6 concludes the study.

2. Related works

The VNE problem is an NP-hard problem with different optimization objectives; it involves node mapping and link mapping [5]. In the first part of this section, we mainly summarize existing works with the same objective as that of our study. In the second part, other works with objectives that differ from that of our study to a certain degree are introduced. Lastly, the relations between these works and our work are analyzed.

2.1. Related works with the same optimization objective

The objectives of the works in [4,6–14,17–22] are the same as that of the present study, i.e., to minimize the cost of a VN request, or maximize the revenue of an InP. In these studies, the cost of a VN request is generally computed as the weighted sum of the CPU costs of all mapped nodes and the bandwidth costs of all mapped paths; thus, the general solutions attempt to determine the mapped SN nodes that are close to one another and to identify the shortest SN path between any two mapped nodes if a virtual link exists between them in a VN request [6–9], or use multiple paths that simultaneously provide bandwidth for a virtual link if the SN supports splitting data transmit mechanisms [4,10]. To realize these objectives, some “sufficient capacity” nodes with more nodes and edges, or more CPU and bandwidth resources, in their neighboring area, are determined. “Sufficient capacity” nodes imply that their neighboring areas have more opportunities to embed other nodes and links of a VN request.

Based on the aforementioned greedy idea, some forms of heuristic information are defined with several network topology attributes [6–9], and a single path (the shortest or the k-shortest path) is used for a virtual link. In [6], Zhu et al. proposed a basic VN assignment scheme, which defined neighbor resource availability as heuristic information to select nodes greedily, and used the shortest path for each virtual link. In [7], Cheng et al. referenced the Markov random walk model and proposed ‘NodeRank’ as a measurement for node resource based on this model. The NodeRank value of a node was used as heuristic information, and two kinds of algorithms were designed: RW-MaxMatch (a two stage algorithm), and RW-BFS (a coordinated node and link mapping algorithm). To improve the performance of their two stage algorithm, they used a particle swarm optimization (PSO) algorithm to search for the optimal solution among all feasible solutions. In [8], Li et al. proposed a two stage scheme, which used multiple factors of a network topology that included hop count as a top-k dominating model to greedy node mapping and used k-shortest path to link mapping. In [9], Cui et al. proposed an algorithm based on the maximum convergence degree, to ensure that the topology of a virtual network gathered together when it was mapped onto an SN, and the k-shortest path was used for link mapping. In these works, heuristic information could improve the performance, but load balance was not considered. The load balance of an SN is beneficial to accepting successive VN requests and gaining additional revenues. However, load balance with a single path for a virtual link is NP-hard [4].