Computer Communications 57 (2015) 25–36

Contents lists available at ScienceDirect

Computer Communications

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

Probabilistic inference-based service level objective-sensitive virtual network reconfiguration

Li Xu [a,b,*], Guo-zhen Tan [a], Xia Zhang [b]

[a] Department of Computer Science and Technology, DaLian University of Technology, No. 2 Ling Gong Road, Ganjingzi District, Dalian 116024, Liaoning, China
[b] State Key Laboratory of Software Architecture (Neusoft Corporation), No. 2 Xinxiu Street Neusoft Park, Hunnan New District, Shenyang 110179, China

ARTICLE INFO

Article history:
Received 2 June 2014
Received in revised form 1 October 2014
Accepted 9 October 2014
Available online 20 October 2014

Keywords:
Probabilistic inference
SLO assurance
Virtual network
Virtual network reconfiguration

ABSTRACT

Network virtualization enables multiple service providers to share the same physical infrastructure, and allows physical substrate network (SN) resources to be used in the form of a virtual network (VN). However, there are many obstacles to the application of this technology. One of the more challenging is the reconfiguration of SN-embedded VNs to adapt to varying demands. To address this problem, we propose a service level objective (SLO)-sensitive VN reconfiguration (VNR) method. A Bayesian network learning and probabilistic reasoning-based approach is proposed to automatically localise reconfiguration points and generate VN resource requests. To determine an optimal reconfiguration solution, we design a heuristic VNR algorithm with a virtual node and virtual link swapping strategy. We validate and evaluate this algorithm by conducting experiments in a high-fidelity emulation environment. Our results show that the proposed approach can effectively reconfigure a VN to adapt to a changed SLO. A comparison shows that our reconfiguration algorithm outperforms existing solutions.

1. Introduction

1.1. Background

Network virtualization [1] is a key technology in the decoupling of network resources from physical infrastructure. The Infrastructure as a Service (IaaS) [2] paradigm has been proposed to enable the self-service provision of resources in a physical substrate network (SN) in the form of a logical virtual network (VN). The efficiency of the physical infrastructure can be significantly improved by the simultaneous sharing of a physical network. Services deployed in a VN environment are also much more dynamic and flexible.

In the business model introduced in [3], network virtualization (NV)-driven IaaS is considered to be the foundation of future internet architectures. Current internet service providers will have two new roles: infrastructure provider (InP) and service provider (SP). The InP will be responsible for providing and operating physical infrastructure devices, with resources in the physical SN provisioned through an IaaS portal, whereas the SP will concentrate on providing business services to its customers by renting resources from the InP. In a virtualized environment, service descriptions and service resource requirements are encapsulated into a service template, which is described using a standardised modelling language such as OASIS TOSCA [4] or DMTF OVF [5]. The quality of service is assured through a service level agreement (SLA) with a customer. The specific measurable characteristics of an SLA are known as service level objectives (SLOs). Service deployment and resource procurement are provided in an on-demand, self-service manner, giving the SP unprecedented flexibility to obtain VN resources for deploying services and increasing the InP's SN resource utilisation rate. However, new challenges emerge as the management complexity increases.

One challenge posed by the network virtualization-driven service model is the efficient embedding of an SP-requested VN into an SN. Thus, we require an algorithm to minimise physical infrastructure allocation costs while maximising the SN resource utilisation rate. These are categorised as VN-embedding algorithms. There is a considerable body of work on the design of VN-embedding algorithms [6–8].

To provision resources to the SP in a self-service manner, IaaS platforms such as CloudStack [9] and OpenStack [10] are being developed that are capable of provisioning resources to support on-demand service deployment. With the development of Software-Defined Networking (SDN) and its enabling protocol OpenFlow [11], network functions are being decoupled from the substrate network. Furthermore, the emergence of SDN controllers

* Corresponding author at: Department of Computer Science and Technology, DaLian University of Technology, No. 2 Ling Gong Road, Ganjingzi District, Dalian 116024, Liaoning, China. Tel.: +86 13604119605.
E-mail address: xuli@neusoft.com (L. Xu).

http://dx.doi.org/10.1016/j.comcom.2014.10.005
0140-3664/

such as OpenDaylight [12] and network virtualization experiment environments such as FITS [13] have paved the way for the provisioning and reconfiguring of VNs through the IaaS platform. However, none of these can offer network quality-of-service (QoS) guarantees. In addition, VN performance requirements change with the service workload, meaning that previously constructed VNs are inadequate for current demands. To ensure the service performance complies with changing SLOs, SPs must reconfigure the VN mapping. The problem of reconfiguring a VN mapped in an SN to adapt to time-varying demands is another challenging problem. The irrational allocation of a VN wastes resources and causes data fragmentation, whereas inaccurate VN resource demands can result in performance degradation.

VN reconfiguration (VNR) process could be sponsored by the InP or SP. InP-sponsored VNR concentrates on decreasing the VNR request (VNRR) rejection rate, reducing resource fragmentation, and maximising revenue, whereas SP-sponsored reconfiguration focuses on releasing or procuring additional resources to adapt to a changed SLO. The pattern of reconfiguration can be classified as reactive or proactive. A reactive reconfiguration is triggered by events, such as a node failure or SLO violation. A proactive reconfiguration is executed periodically in anticipation of possible future failures. The literature primarily focuses on reactive InP-sponsored scenarios [14,15] with known specific reconfiguration resource requirements. To the best of our knowledge, SP-sponsored VNR issues have not been addressed. The accurate definition of VNRR such that the SP-sponsored reconfiguration effectively ensures compliance with a changed SLO definition, without excess provisioning, remains unsolved. In this paper, we propose a proactive SP-sponsored VNR approach to tackle this problem effectively and efficiently.

In our approach, we define the virtual nodes and virtual links that need to be reconfigured in a VN as reconfiguration points (RPs). VNR is designed as a two-step process. First, we find RPs and generate VNRRs for the target VN. An SLO-sensitive RP-localisation VNRR generation mechanism is designed to adapt to dynamically changing SP demands. In the second step, the VN is reconfigured according to the VNRR. We use a Bayesian network (BN) to model the uncertainty in an SLO and monitor performance metrics. Probabilistic inference on the BN determines the RPs and generates VNRRs. After the VNRRs have been generated, the VN is reconfigured. To increase the InP's revenue and reduce the fragmentation caused by reallocating additional resources to the VN, we propose a heuristic algorithm with virtual node and virtual link migration and swapping strategies. We evaluate the performance of our heuristic by conducting extensive experiments in a high-fidelity emulation environment. The effectiveness and efficiency of the proposed method are confirmed by comparison with existing solutions.

The contributions of this work are as follows:

(1) We propose an automatic probabilistic inference technique, that largely reduces the difficulty of RP localisation and VNRR generation for a given SLO and VN.
(2) The proposed SP-sponsored proactive approach can detect under-provisioned virtual nodes and virtual links, thus reducing the SLO violation risks.
(3) We design an RP localisation strategy that switches among performance-priority, cost-priority, and balanced modes. The strategy enables SPs to flexibly elaborate the reconfiguration of objectives.
(4) We improve existing VNR algorithms by adding virtual node and virtual link swapping mechanisms to the virtual node and virtual link migration phase. This hybrid migration and swapping strategy reduces resource fragmentation and increases revenue for the InP.

The remainder of this paper is organised as follows: Section 1.2 reviews related work on VNR. Section 2 formulates the problem as a mathematical model by defining a VN stress state, reconfiguration constraint, and VNRR. In this section, we introduce our proposed BN modelling and probabilistic inference-based RP localisation mechanism, and describe our VNR algorithm and migration and swapping strategies. The results of evaluation experiments are presented in Sections 3 and 4 concludes the paper.

1.2. Related work

Dynamically reconfiguring a VN that is already embedded in an SN is distinct from processes that only handle static VN-embedding. Reference [16] considers VN reconfiguration as a special VN-embedding process which is attempting to reconfigure the mapped VNRs in order to reorganize the resource allocation and optimise the utilisation of SN resources. There are few studies that have addressed this issue.

The existing literature models VN reconfiguration as a mathematical optimisation problem. The authors of [17,18] modelled the reconfiguration of an embedded VN using mixed-integer linear programming (MILP), and relatively optimal solutions were determined by solving the resulting optimisation problem. However, finding the best VN reconfiguration solution is considered to be NP-hard [14]. There are a number of works that address this issue in a more efficient manner.

To avoid the NP-hard problem, most works have focused on heuristic approaches with the objective of minimising the VN reconfiguration cost, reducing resource fragmentation, or increasing the InP's revenue [7,17–21]. The authors in [7] employ a greedy node mapping algorithm with node remapping capability. The motivation of the algorithm is to map the virtual nodes to the substrate nodes with the maximum substrate resources, thus minimising the use of resources at bottlenecked nodes and links. However, this only considers the reconfiguration of virtual links, and does not consider the migration of virtual nodes. Such a strategy limits the algorithm's efficiency and flexibility. To overcome this limitation and improve the efficiency of resource reallocation to VNs, [20] proposed a simulated annealing algorithm that reconfigures the VNs via node and link migrations and re-optimises the allocation of resources to VNs such that the load is balanced across the substrate network. Simulation results show that the proposed reconfiguration algorithm achieved a significant improvement in the acceptance ratio of VN requests.

Most of the existing literature on this topic neglects the possible service disruption during reconfiguration. The authors of [18] studied this problem by taking into account the cost incurred by the disrupted services. The service disruption-aware VN reconfiguration problem was formulated in terms of integer linear programming, and a cost-sensitive heuristic algorithm was developed. To further reduce the cost of reconfiguration, an algorithm that prevents the reconfiguration of the entire VN request and reduces the rejection rate has been presented [19]. This employs a high-utilisation virtual node selection algorithm. The algorithm takes the maximum utilisation within a given period of time as the key metric for selecting virtual nodes, but ignores virtual link stress and the virtual node resource release reconfiguration process. A reactive algorithm that reacts to any rejection of new VN requests was proposed in [21]. The algorithm is triggered by the VN-embedding rejection event, and reconfigures currently mapped networks so that new requests can be embedded. However, reconfiguring runtime VNs in a stochastic reactive manner is risky and costly.

To minimise the effects of congestion in an SN, a mechanism for assigning weights to substrate nodes and substrate links based on