

基于节点多属性的虚拟网络映射算法

张鑫,王珺,王晓璇

(南京邮电大学 江苏省无线通信实验室 江苏 南京 210003)

摘要: 基于现有的两阶段虚拟网络映射算法,分析节点综合资源计算标准,改进选取节点映射的评估方法。该虚拟网络映射算法基于虚拟网络中节点的多种属性,即节点CPU计算资源、链路带宽资源、节点相邻节点的数目及拓扑属性。节点映射阶段采用改进之后的资源评估方式进行节点映射,同时采用可重用技术,实现物理网络节点可重复映射,即物理网络中的同一个节点可以被虚拟网络中的多个节点多次映射。该算法不仅有效地减少了映射过程中出现的资源瓶颈问题,而且降低了链路映射的成本,节约了部分映射带宽资源,从而使得网络基础设施提供商可以接受到更多的虚拟网络服务,提高网络运营的网络收益。仿真实验数据表明,该虚拟网络映射算法在虚拟网络请求接受率和网络开销比等参数指标上具有显著提升。

关键词: 虚拟网络映射; 物理网络; 虚拟网络请求接受率; 网络开销比

中图分类号: TP393

文献标识码: A

文章编号: 1673-629X(2018)12-0053-06

doi: 10.3969/j.issn.1673-629X.2018.12.011

Virtual Network Embedding Algorithm Based on Multiple Node Attributes

ZHANG Xin, WANG Jun, WANG Xiao-xuan

(Jiangsu Key Lab of Wireless Communication, Nanjing University of Posts and Telecommunications,
Nanjing 210003, China)

Abstract: Based on the existing virtual network embedding algorithm with two stages, we analyze the comprehensive resource calculation standard of node and improve evaluation method of selecting node mapping. The virtual network embedding algorithm is based on multiple attributes of nodes, involving CPU computing resource, link bandwidth resource, the number of neighbor nodes of virtual nodes and its attributes of topology. In the node mapping stage, the improved resource assessment method is adopted for the node mapping. Meanwhile, reusable technology is adopted to realize the repeated mapping of physical network nodes, that is, the same node in the physical network can be mapped multiple times by multiple nodes in the virtual network. This algorithm not only effectively reduces the link bottleneck in the virtual network embedding, but also decreases the cost of link mapping and saves part of the link bandwidth. Thus, more and more virtual network requests can be accepted by the substrate network and Internet infrastructure providers would earn much money. The simulation suggest that the embedding algorithm has a significant improvement in performances such as virtual network request acceptance ratio and network revenue/cost ratio.

Key words: virtual network embedding; substrate network; virtual network request acceptance ratio; network revenue/cost ratio

0 引言

随着互联网规模和应用范围越来越广泛,传统的TCP/IP网络结构难以应对日益复杂和多样化的信息社会的发展。同时智能手机、平板电脑的普及,使得互联网服务提供商需要面临更加复杂和多样化的用户需求与业务,而现有的网络体系无法快速、便捷地实现业

务和服务的更新与迭代。针对当前互联网发展所面临的困境,Stanford Nick McKeown教授等颠覆传统的网络结构,提出一种新型、革命式的网络模型,即软件定义网络(software defined network, SDN)及OpenFlow协议^[1-2]。SDN与TCP/IP网络相比较,能够实现控制平面和转发平面的分离,网络具备可编程的特性。

收稿日期: 2018-01-04

修回日期: 2018-05-06

网络出版时间: 2018-07-04

基金项目: 国家自然科学基金(61401234)

作者简介: 张鑫(1993-),男,硕士研究生,研究方向为软件定义网络、网络虚拟化、云计算;王珺,硕士,研究方向为无线传感器网络、网络安全。

网络出版地址: <http://kns.cnki.net/kcms/detail/61.1450.TP.20180703.1511.026.html>

因此 SDN 网络模型主要包括 SDN 网络应用、SDN Controller、支持 OpenFlow 协议网络设备、南北向接口等。

目前,网络虚拟化技术受到工业界和学术界的高度青睐,它不仅是解决互联网发展瓶颈的关键技术,也是云计算的核心技术。同时集中控制网络架构 SDN 日益成熟和完善,尤其在云计算数据中心等场景下,为了实现动态调整与快速部署,必须实现业务部署的自动化。采用集中控制的方式,网络运营商可以通过控制器编程,实现自动化动态业务部署,缩短业务部署周期与成本。近些年,随着 IaaS(Infrastructure as a Service)的发展,大多数主流的数据中心网络基本采用基于 SDN 网络模型构建虚拟网络实现网络虚拟化。

网络虚拟化技术聚焦的重点是在共享的、资源有限的物理网络中实现、分配各种不同用户或者组织要求的逻辑网络,同时保证每个逻辑网络之间能够隔离并且安全、正常地对外提供应用与服务。同时具体实现网络虚拟化技术又面临很多问题,首先就是逻辑网络映射到物理网络,即虚拟网络映射问题。目前虚拟网络映射问题已经得到广泛关注,同时因其映射过程中会受到各种条件的限制,因此属于 NP-hard 问题。

1 相关工作

文献[3]在不限制地理位置的条件下,底层物理网络支持路径迁移与路径分裂的特性,节点映射阶段利用 Greedy 算法来映射,链路映射阶段采取多商品流算法完成链路映射。文献[4]基于马尔可夫随机游动模型,利用 K-核分解技术将虚拟网络划分为核心网络和边缘网络,优先考虑生存时间最短的虚拟网络请求(区别于以收入为导向的调度策略)。文献[5]在节点映射阶段引入网络拓扑属性,即在考虑 CPU 和带宽的同时考虑节点的度(度是节点最基本最重要的拓扑属性,可衡量在拓扑结构中节点的相对影响和相对重要性,反映了与网络剩余部分的连接程度,节点的度等于其与邻居节点的直接链路数),以增强节点与其他节点联系能力的影响。

文献[6]基于一种 QoSMap 机制,通过选择高质量保证的物理链路满足虚拟网络的 QoS 要求,同时基于一跳距离的中继路由节点构建备份路径,最终为虚拟链路映射提供服务质量保证。文献[7]在离线且未考虑节点映射阶段,提出将隐藏跳数(虚拟链路映射到的底层路径经过的中间物理节点)纳入到链路映射阶段。算法将虚拟网络请求分成有特定资源请求和无明确资源请求两类,先处理有特定需求的请求,使得虚拟网络映射过程中剩余网络资源较多。

文献[8]从已有算法平等地对待底层网络的节点

和链路资源,实际上这些物理资源中某些资源相比其他的资源更重要,这些关键资源的耗尽会对未来的请求接受率有较大的影响。拓扑感知映射在已有算法的成本函数中引入比例因子,对资源进行区分。引入重优化机制,周期性的重优化机制不适用于实际的网络,文献中提出当有虚拟网络请求被拒绝时执行重优化算法,以有效地均衡负载。文献[9]基于图论和数据结构的理论基础,采取广度优先搜索的原则完成虚拟网络的链路映射,尽可能使得虚拟网络中节点映射到物理网络中,其位置上仍然能够保持在同一个区域内,最终使得 BFS-VNE 虚拟网络映射算法在相应的虚拟网络映射算法性能参数上优于其他虚拟网络映射算法。同时文献[10-11]中分别描述目前现有的虚拟网络映射模型及 K 最短路径算法。目前一些虚拟网络映射算法已经开始考虑确保网络正常稳定接纳虚拟网络请求,例如文献[12-15]。

现有的虚拟网络映射算法仅仅考虑单个节点所具有的 CPU 资源和带宽资源,同时未能从邻居节点及相关节点来全局考虑整个网络拓扑,导致后续的链路映射成功率下降,使得虚拟网络请求的数目变少,网络运营商的网络收益下滑。文中将基于节点多个属性,改进传统的节点映射度量方法,提出基于节点属性的虚拟网络映射方法。

2 虚拟网络映射问题模型

虚拟网络映射问题就是将网络服务提供商需要的虚拟网络请求,映射到满足其计算和链路资源的网络运营商的物理网络中。

2.1 物理网络

物理网络拓扑基于加权无向图可以表示为 $G^s = (N^s, L^s, A_N^s, A_L^s)$,其中 N^s 表示物理网络中的节点集合, L^s 表示物理网络中的链路集合, A_N^s 表示物理网络节点属性集合,一般来说有剩余的计算资源、可用的内存资源等, A_L^s 表示物理网络链路的属性集合,具体包括带宽资源、传输时延等。

2.2 虚拟网络

与物理网络相似,虚拟网络也可以表示为 $G^v = (N^v, L^v, C_N^v, C_L^v)$,其中 N^v 表示虚拟网络中虚拟节点的集合, L^v 表示虚拟网络中虚拟链路的集合, C_N^v 表示 CPU 资源请求, C_L^v 表示链路带宽资源请求。

2.3 虚拟网络映射问题描述

虚拟网络映射问题 $E(G^v): (N^v, L^v, C_N^v, C_L^v) \rightarrow (N', P', R_N, R_L)$,其中 $N' \subseteq N^s$, $P' \subseteq P^s$, R_N, R_L 分别表示分配给虚拟网络请求的节点资源和链路资源。具体而言节点映射: $M_N: \forall_{n^v} \in N^v, M_N(n^v) \in N^s (C(n^v) <$

$R_N(M_N(n^v))$, 链路映射: $M_L: \forall l^v = (m^v, n^v) \in L^v$, $M_L(l^v) \subseteq P^S(M_N(m^v), M_N(n^v))$, constraint: $b(l^v) \leq R_L(P)$, $\forall P \in M_L(l^v)$ 。

2.4 虚拟网络映射问题的主要性能参数

虚拟网络映射问题最主要解决的是物理网络资源利用效率, 希望能够接受更多的虚拟网络请求, 同时进一步提高网络基础设施提供商的收益, 其有关性能参数定义如下所述。

2.4.1 虚拟网络收益

由某时间段全部映射成功的虚拟网络请求的节点 CPU 计算资源和链路的带宽资源的总和决定, 在某个时刻一个虚拟网络请求所带来的网络收益可以表示为:

$$R(G^v, t) = \alpha \sum_{n^v \in N^v} c(n^v) + (1 - \alpha) \sum_{l^v \in L^v} b(l^v) \quad (1)$$

其中, 调节系数 $\alpha \in (0, 1)$, 一般情况取 $\alpha = 0.5$; $c(n^v)$ 表示某一时刻虚拟网络 G^v 中节点 n^v 的 CPU 计算资源数值; $b(l^v)$ 表示虚拟网络链路 l^v 的带宽资源数值。

2.4.2 网络映射开销

虚拟网络映射过程中全部的虚拟网络请求所需要的资源消耗即底层网络为满足虚拟网络中节点、链路资源约束而消耗的物理网络资源, 可以表示为:

$$C(G^v, t) = \alpha \sum_{n^v \in N^v} c(n^v) + (1 - \alpha) \sum_{l^v \in L^v} \sum_{l^s \in L^s} b(f_{l^s}^{l^v} l^v) \quad (2)$$

其中, $f_{l^s}^{l^v} \in (0, 1)$, 当 l^s 为 l^v 分配带宽资源时, $f_{l^s}^{l^v} = 1$, 反之 $f_{l^s}^{l^v} = 0$; $b(f_{l^s}^{l^v} l^v)$ 表示 l^s 为 l^v 分配的带宽大小。

2.4.3 物理网络平均收益

定义如下:

$$\lim_{T \rightarrow \infty} \frac{\sum_{t=0}^T R(G^v, t)}{T} \quad (3)$$

2.4.4 虚拟网络请求接受率

定义如下:

$$\lim_{T \rightarrow \infty} \frac{\sum_{t=0}^T \text{VNR}_A}{\sum_{t=0}^T \text{VNR}} \quad (4)$$

其中, 分子表示某单位时间范围内虚拟网络映射过程中成功接受到虚拟网络请求数量, 分母表示某单位时间范围内虚拟网络映射过程中总的虚拟网络请求数量。

3 虚拟网络映射算法描述

本节分析现有虚拟网络映射算法存在的缺点与不

足, 同时在原有 Two-Stage 虚拟网络映射算法的基础上实现 MNA-VNE 虚拟网络映射算法。首先介绍传统虚拟网络映射过程中节点综合资源评估计算, 针对现有评估准则中的缺陷进行改进与优化, 同时提出新的虚拟网络映射算法。

虚拟网络映射数学模型如图 1 所示。

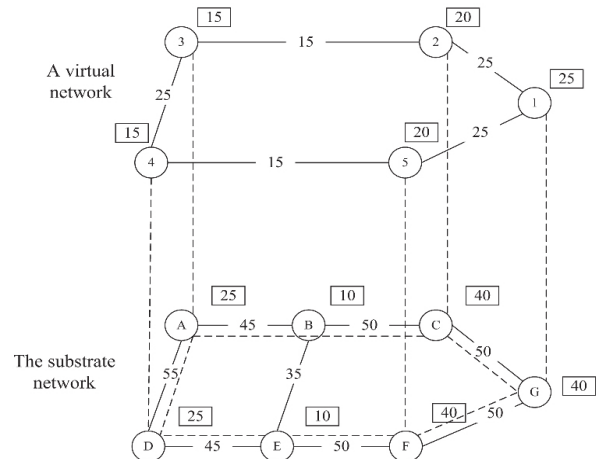


图 1 虚拟网络映射数学模型

3.1 节点度量准则

传统的虚拟网络映射算法中, 节点映射阶段中节点的度量不仅取决于单个节点所具有剩余 CPU 计算资源, 同时也需要考虑网络中剩余链路带宽资源, 因此网络中衡量节点资源的计算公式为:

$$\text{NR}(n) = \text{CPU}(n) \cdot \sum_{m \in \text{neighbour}} \text{BW}(m) \quad (5)$$

其中, $\text{NR}(n)$ 表示节点 n 的综合资源数值; $\text{CPU}(n)$ 表示节点 n 的 CPU 资源数值; $\text{BW}(m)$ 表示节点 n 连接的链路带宽资源之和。

传统虚拟网络映射算法中关于节点资源评估计算存在诸多不足:

(1) 节点的计算资源属性。

现有的虚拟网络映射算法中对于节点资源计算评估大多基于网络中的单个节点, 未从全局角度考虑网络拓扑结构, 不能真实反映出节点映射阶段中物理节点的优先级和重要性。

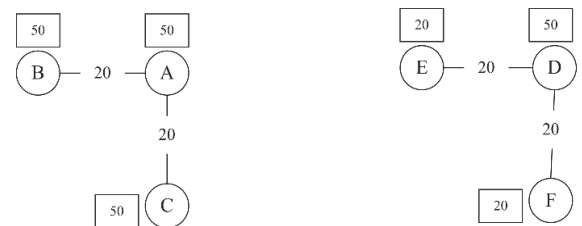


图 2 网络拓扑结构 I

从图 2 可以计算得出, 物理节点 A 与物理节点 D 的 NR 值相等, 传统的虚拟网络映射算法无法区分节点 A 和 D 的优先级, 认为节点 A 的邻居节点具备更多的 CPU 计算资源, 当有虚拟网络请求到达时, 虚拟节

点应该优先选择物理节点 A 作为映射的节点。

(2) 节点的带宽资源约束。

未能考虑与节点相连接的链路带宽资源相加不能准确地反映该节点在映射过程中的优先级及重要性,同时也会在映射过程中造成链路瓶颈,从而导致后续链路映射无法进行。

从图 3 中可得出,物理节点 A 和物理节点 B: $90 * (10+10) = 30 * (20+20+20) = 1\ 800$ 。由于 $NR(A) = NR(B)$, 虚拟网络映射过程中的虚拟节点 e 将会在节点 A 与节点 B 中进行选择。如果选择了节点 A 在其后的链路映射中,如果要求链路带宽需要大于 10, 由于节点 A 的链路带宽不符合节点 F 的带宽要求, 必须重新进行节点映射。



图 3 网络拓扑结构 II

(3) 网络中节点的度。

根据图 4 可知, 当一个虚拟网络请求到达物理网络中, 虚拟网络映射过程中需要完成节点映射。物理网络中节点 A 的相邻节点数目为 3, 而节点 F 的相邻节点数目为 4, 因此虚拟节点 g 在完成虚拟网络映射过程中, 会优先考虑节点 F, 因为物理网络中节点 F 周围的节点资源更加丰富, 以便后续的链路映射能够更好地完成。

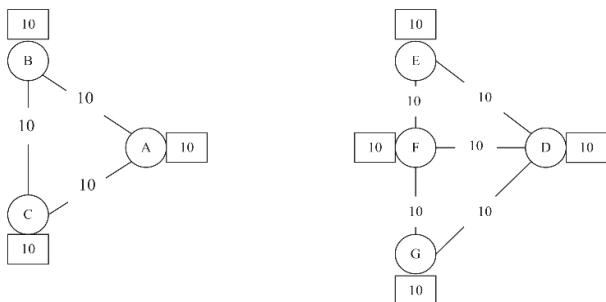


图 4 网络拓扑结构 III

针对上述不足, 首先给出节点映射阶段中的几个概念, 然后重新定义一种基于节点多属性的评估公式, 以更好地进行虚拟节点的映射。

首先, 计算节点的计算资源属性。节点的计算资源值不仅与单个节点自身的 CPU 计算资源有关, 而且还与其邻居节点的 CPU 计算资源有关。定义如下:

$$RC(n^v) = CPU(n^v) + CPU(n^v) \cdot$$

$$\log \sum_{n_{neighbour}^v \in N^v} n_{neighbour}^v / \log CPU(n^v) \quad (6)$$

其中, $RC(n^v)$ 表示节点具有的资源综合评估值;

$n_{neighbour}^v$ 表示 n^v 相邻的节点。

然后, 改进节点的链路带宽资源属性。针对链路带宽资源存在的不足, 对其进行重新修改, 改进公式如下:

$$BW(n^v) = \sum_{l^v \in L(n^v)} \theta_i bw(l^v) \quad (7)$$

其中, $BW(n^v)$ 表示该节点的链路带宽资源值; $bw(l^v)$ 表示满足该节点的映射链路带宽值; θ_i 为链路带宽的加权系数, 根据实际网络具体情况设置。例如网络中链路带宽资源分布在 $[1, 100]$ 之间, 考虑链路带宽资源丰富的加权系数越大, $[1, 10]$ 范围内为 0.1, $(10, 20]$ 范围内为 0.2, 依次类推。

最后, 考虑节点的拓扑属性。充分考虑节点的邻居节点的数目对虚拟网络映射的影响, 因此用节点的度表示该节点邻居节点的个数, 其定义如下:

$$N(n^v) = Neighbour(n^v) \quad (8)$$

其中, $N(n^v)$ 表示网络中节点的度; $Neighbour(n^v)$ 表示节点的相邻节点的数目。

综上, 得出基于节点多属性的评估方法:

$$NR(n^v) = RC(n^v) * BW(n^v) * N(n^v) \quad (9)$$

其中, $NR(n^v)$ 表示该节点的属性综合度量值; $RC(n^v)$ 表示节点的计算资源值; $BW(n^v)$ 表示节点的链路带宽资源值; $N(n^v)$ 表示节点的度。

3.2 节点映射算法步骤

文中提出的 MNA-VNE 虚拟网络映射算法是在原有两阶段虚拟网络映射算法基础上的改进与优化。下面主要介绍 MNA-VNE 虚拟网络映射算法的节点映射算法与链路映射算法。

节点映射阶段算法的详细描述及流程如下:

Step1: 对到达的每一个虚拟网络请求按照网络收益值从大到小依次排序, 其具体公式为: $R(G^v, t) = \alpha \sum CPU(n^v) + (1 - \alpha) \sum BW(l^v)$, 其中 $\alpha \in (0, 1)$, $CPU(n^v)$ 为虚拟网络请求中节点 n^v 的 CPU 计算资源值, $BW(l^v)$ 为该节点 n^v 某条链路带宽, 并选取收益值最大的虚拟网络请求进行映射。

Step2: 对虚拟网络请求中的每个节点按式 9 进行计算, 然后需要针对物理网络中的物理节点按式 9 进行计算评估, 将物理节点按照 $NR(n^s)$ 数值从大到小依次排序形成一个队列, 如果虚拟网络中 $NR(n^v)$ 的最大值小于等于物理网络中节点 $NR(n^s)$ 的最大值, 则满足节点映射的条件, 将虚拟节点 n^v 映射到物理节点 n^s 上。

Step3: 第一个虚拟节点映射成功之后, 优先选择刚刚已经完成映射过的物理节点进行映射, 如果不满足虚拟网络映射的资源需求, 则将虚拟网络中第二个

映射节点的资源能力值与该物理节点邻居节点进行比较,如果符合映射要求,则将虚拟节点映射到该物理节点的邻居节点,否则将从 Step2 的队列中选取最大满足要求的节点,并将其映射到该物理节点上。

Step4: 如果在虚拟网络映射中,虚拟网络中节点 $NR(n')$ 的最大值大于物理网络中节点 $NR(n')$ 的最大值,则该虚拟网络映射无法进行,虚拟网络映射失败。

Step5: 重复上述 Step1,将之前已经映射的物理网络中物理节点的剩余资源再一次进行计算,并再一次按照从大到小排序,最后直到虚拟网络请求中所有节点映射完成。

3.3 链路映射算法步骤

在 3.2 节的节点映射算法的基础上,实现虚拟网络中的链路映射。虚拟网络链路映射过程中将继续根据最短路径算法的思想实现虚拟链路的部署,首先排除不满足其虚拟网络带宽资源需求的物理网络链路,在此基础上采用 K 最短路径算法完成链路映射。

3.4 仿真实验

将 MNA-VNE 算法与文献[3]和文献[9]中的虚拟网络映射算法进行仿真实验对比及理论分析。其中文献[3]的 Greedy-VNE 和文献[9]的 BFS-VNE 均采用式 5 中的节点资源评估准则,并利用 Python Matplotlib 绘图,将所得实验数据在虚拟网络请求接受率、网络收益/开销比两个性能指标上进行比较。

3.4.1 实验环境

仿真实验中设置一个中等规模的物理网络,其中包括 100 个物理节点和大概 500 条物理链路。为了能够与上述虚拟网络映射算法进行比较与分析,其虚拟网络请求等一系列参数均按照文献[3]中 GT-ITM 实验中的参数进行设置。同时每次仿真实验运行 80 000 单位时间,取 10 次实验结果的平均值(单位时间为 ms)。

3.4.2 仿真结果分析

由图 5 可以得出,这 3 种虚拟网络映射算法在仿真实验初始阶段的虚拟网络请求接受率比较稳定。但随着仿真时间的不断推移,3 种算法的虚拟网络请求接受率都出现了不同程度的下降,其中 Greedy-VNE 算法下降得最快。主要原因是刚开始实现虚拟网络映射时,由于网络中剩余的 CPU 计算资源与链路带宽资源较为充裕,物理网络中有足够的物理网络资源能够满足其虚拟网络的资源请求,但随着到达的虚拟网络请求的数量越来越多,物理网络中的可用资源在逐渐减少,最终导致虚拟网络请求接受率逐步下降。由于 MNA-VNE 算法在节点映射阶段采用全新的度量评估方法,充分考虑网络节点拓扑,同时采用物理节点可

重映射的方法,代替部分链路映射,减少了底层物理网络中链路带宽资源的消耗,从而得到更多的虚拟网络请求。因此,当虚拟网络映射过程到最后时,MNA-VNE 虚拟网络映射算法相比其他两种算法具有明显的优势。

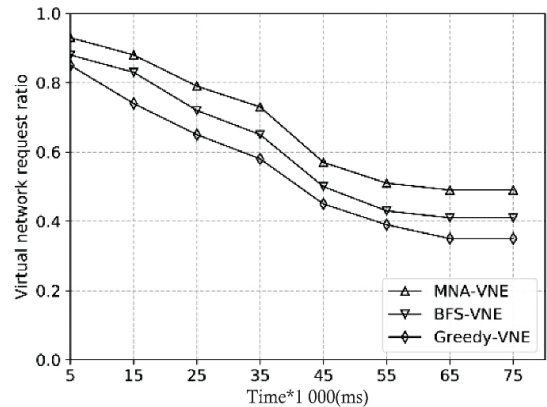


图5 虚拟网络请求接受率

从图 6 可以看出,3 种算法的网络收益/开销比随着仿真时间不断增加,都出现了不同程度的下降。其主要原因是随着虚拟网络请求数目的增多,物理网络可以接受的虚拟网络请求数目越来越少,剩余的物理网络资源可能无法满足虚拟网络请求的资源需要,虚拟网络请求中的链路映射需要消耗更多的时间和资源,导致虚拟网络映射的开销增加。由于 MNA-VNE 算法采用物理节点可重用,进一步降低物理网络中链路带宽资源的消耗,使得网络资源开销减少,从而接受到更多的虚拟网络请求,增加了底层网络的收益,从而提升了网络收益/开销比。所以从图中的曲线变化走势可以得出,MNA-VNE 虚拟网络映射算法在虚拟网络收益/开销比性能参数上优于其他两种虚拟网络映射算法。

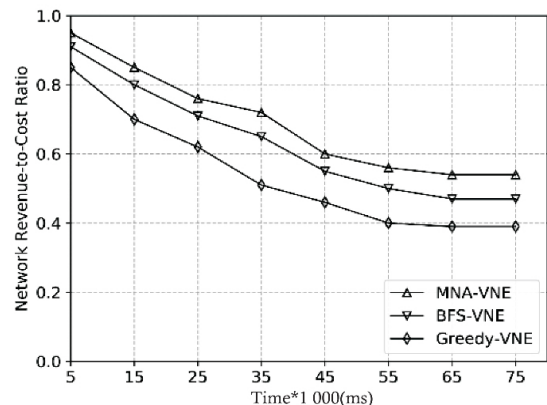


图6 虚拟网络收益/开销比

4 结束语

在传统两阶段静态虚拟网络映射算法的基础上,充分考虑网络中节点所具有的多种属性,具体包括节

点的资源属性及网络拓扑属性,并且采取物理节点可重用技术,尽可能考虑链路映射过程,提出 MNA-VNE 虚拟网络映射算法。该算法考虑单个节点的邻居节点的资源属性,提高了后续链路映射的成功率。仿真实验表明,MNA-VNE 虚拟网络映射算法在虚拟网络请求接收率、网络收益/开销比等性能参数上更具有优势。

参考文献:

- [1] MCKEOWN N, ANDERSON T, BALAKRISHNAN H, et al. Openflow: enabling innovation in campus networks [J]. ACM SIGCOMM Computer Communication Review, 2008, 38(2): 69-74.
- [2] KREUTZ D, RAMOS F M V, VERISSIMO P E, et al. Software-defined networking: a comprehensive survey [J]. Proceedings of IEEE, 2014, 103(1): 14-76.
- [3] YU M, YI Y, REFOXD J, et al. Rethinking virtual network embedding: substrate support for splitting and migration [J]. ACM SIGCOMM Computer Communication Review, 2008, 38(2): 17-29.
- [4] CHOWDHURY M, RAHMAN M R, BOUTABA R. ViNE-Yard: virtual network embedding algorithms with coordinated nodes and link mapping [J]. IEEE/ACM Transactions on Networking, 2012, 20(1): 206-219.
- [5] CHENG Xiang, SU Sen, ZHANG Zhongbao, et al. Virtual network embedding through topology-aware node ranking [J]. ACM SIGCOMM Computer Communication Review, 2011, 41(2): 38-47.
- [6] SHAMSI J, BROCKMEYER M. QoSMap: QoS aware mapping of virtual networks for resiliency and efficiency [C]//Proceedings of the IEEE GLOBECOM workshop. Washington, DC, USA: IEEE, 2007: 1-6.
- [7] 姜明, 王保进, 吴春明, 等. 网络虚拟化与虚拟网映射算法研究 [J]. 电子学报, 2011, 39(6): 1315-1320.
- [8] BUTT N F, CHOWDHURY N M M K, BOUTABA R. Topology awareness and reoptimization mechanism for virtual network embedding [C]//International conference on research in network. [s. l.]: [s. n.], 2010: 27-39.
- [9] 彭利民. 基于广度优先搜索的虚拟网络映射算法 [J]. 四川大学学报: 工程学科版, 2015, 47(2): 117-122.
- [10] ANDERSON T, PETERSON L, SHENKER S, et al. Overcoming the Internet impasse through virtualization [J]. Computer, 2005, 38(4): 34-41.
- [11] EPPSTEIN D. Finding the K shortest paths [J]. SIAM Journal on Computing, 1998, 28(2): 652-673.
- [12] 蔡志平, 刘强, 吕品, 等. 虚拟网络映射模型及其优化算法 [J]. 软件学报, 2012, 23(4): 864-877.
- [13] 陈青云. 虚拟化环境下网络生存性方案的设计与实现 [D]. 北京: 北京邮电大学, 2014.
- [14] 刘光远, 双锴, 苏森. 区分服务 QoS 的可生存虚拟网络映射算法研究 [J]. 通信学报, 2013, 34(12): 79-83.
- [15] 张鑫, 贾莹莹, 王晓璇. 基于 SDN 的虚拟网络映射问题研究综述 [J]. 微型机与应用, 2017, 36(22): 11-14.
- [12] CARROLL J J, DICKINSON I, DOLLIN C, et al. Jena: implementing the semantic web recommendations [C]//Proceedings of the 13th international world wide web conference on alternate track papers & posters. New York, NY, USA: ACM, 2004: 74-83.
- [13] HURTADO C, VAISMAN A. Reasoning with temporal constraints in RDF [C]//International conference on principles and practice of semantic web reasoning. [s. l.]: [s. n.], 2006: 164-178.
- [14] 沈文南. 一个 RDF 存储与查询系统的设计与实现 [D]. 南京: 东南大学, 2006.
- [15] SAKR S, AL-NAYMAT G. Relational processing of RDF queries: a survey [J]. ACM SIGMOD Record, 2010, 38(4): 23-28.
- [16] GUO Y, PAN Z, HEFLIN J. An evaluation of knowledge base systems for large OWL datasets [C]//Third international conference on the semantic web. Hiroshima, Japan: [s. n.], 2004: 274-288.
- [17] MANJUNATH G, BADRINATH R, SAYERS C, et al. Temporal views over rdf data [C]//International conference on world wide web. [s. l.]: ACM, 2008: 1131-1132.

(上接第 52 页)

- [5] PUGLIESE A, UDREA O, SUBRAHMANNIAN V S. Scaling RDF with time [C]//17th international conference on world wide web. Beijing, China: ACM, 2008: 605-614.
- [6] KHURANA U, DESHPANDE A. Efficient snapshot retrieval over historical graph data [C]//IEEE international conference on data engineering. Brisbane, QLD, Australia: IEEE, 2013: 997-1008.
- [7] GAHA M, ZINFLOU A, BOUFFARD A, et al. Temporal RDF system for power utilities [C]//The ninth international conference on advances in semantic processing. [s. l.]: [s. n.], 2015: 32-37.
- [8] 汤庸, 汤娜, 叶小平. 时态信息处理技术研究综述 [J]. 中山大学学报: 自然科学版, 2003, 42(4): 4-8.
- [9] 汤庸, 刘海, 郭欢, 等. TempDB: 时态数据管理系统 [J]. 计算机研究与发展, 2010, 47: 442-445.
- [10] SLIVINSKAS G, JENSEN C S, SNODGRASS R T. Adaptable query optimization and evaluation in temporal middle-ware [R]. [s. l.]: [s. n.], 2002.
- [11] 王涓. 基于 TimeDB 的时态数据库研究与应用 [D]. 淮南: 安徽理工大学, 2012.