

一种基于匹配博弈的服务链协同映射方法

张红旗 黄睿* 常德显

(中国人民解放军信息工程大学 郑州 450001)

摘要: 针对软件定义网络(SDN)/网络功能虚拟化(NFV)环境下服务链映射难以兼顾效率与物理资源利用率的问题, 该文提出一种基于匹配博弈的服务链协同映射方法。首先, 以最大化网络资源效用为目标, 建立服务链映射模型MUSCM; 然后, 分虚拟网络功能(VNF)部署和组链两个部分解决服务链映射问题, VNF部署部分, 设计基于多对一匹配博弈理论的算法协同服务链和服务节点双方互相进行选择, 有效提升了服务链的映射效率和物理资源的利用率, 在此基础上, 设计基于分段路由策略的算法实现各VNF实例间的流量牵引以完成VNF组链, 有效降低了链路传输时延。实验结果表明, 与经典算法相比, 该算法在保证映射请求接受率的同时, 有效降低了服务链的平均传输延时, 提升了系统的物理资源利用率。

关键词: 软件定义网络; 网络功能虚拟化; 服务链映射; 匹配博弈; 分段路由

中图分类号: TP393.08

文献标识码: A

文章编号: 1009-5896(2019)02-0385-09

DOI: 10.11999/JEIT180385

A Collaborative Mapping Method for Service Chain Based on Matching Game

ZHANG Hongqi HUANG Rui CHANG Dexian

(The Information Engineering University of PLA, Zhengzhou 450001, China)

Abstract: Considering that it is difficult to balance efficiency and resource utilization of Service Chain (SC) mapping problem in Software Defined Network (SDN)/Network Function Virtualization (NFV) environment, this paper proposes a collaborative mapping method for SC based on matching game. Firstly, it defines a SC mapping model named MUSCM to maximize the utility of network resources. Secondly, it divides the SC mapping problem into Virtual Network Function (VNF) deployment and connection parts. As for the VNF deployment part, an algorithm is designed to collaborate the selection of the SC and the service node based on many-to-one matching game, improving the mapping efficiency of SC and utilization of physical resource effectively. On the basis of it, an algorithm is designed based on segment routing strategy to accomplish the traffic steering between VNF instances to finish the VNF connection part, reducing the link transmission delay effectively. The experiment result shows that, compared with the classical algorithm, this algorithm ensures the mapping request received rate, and at the same time, it reduces the average transmission delay of the service chain and improves the physical resources utilization of the system effectively.

Key words: Software Defined Network (SDN); Network Function Virtualization (NFV); Service Chain (SC) mapping; Matching game; Segment routing

1 引言

近年来, 随着软件定义网络^[1](Software Defined Network, SDN)的发展和网络功能虚拟化^[2]

(Network Function Virtualization, NFV)技术的兴起, SDN/NFV架构越来越多地运用于运营商网络中。一方面, SDN通过逻辑平面和物理平面的分离, 将控制功能从传统的分布式网络设备迁移到集中化的控制平台, 进而实现网络的集中管控和开放可编程。另一方面, NFV将以专有硬件形式部署的网络功能转变为在通用服务器上运行的虚拟网络功能(Virtual Network Function, VNF), 解耦传统网络设备的软件功能和硬件载体, 减少了在网络特定位置部署专用中间盒子的开销, 增加了网络设备

收稿日期: 2018-04-25; 改回日期: 2018-09-11; 网络出版: 2018-09-25

*通信作者: 黄睿 xjhr1009@163.com

基金项目: 国家863计划(2012AA012704), 郑州市科技领军人才项目(131PLJRC644)

Foundation Items: The National 863 Program of China (2012AA012704), The Zhengzhou Science and Technology Talents Project (131PLJRC644)

部署的灵活性。二者结合,使得网络设备和网络架构更加敏捷^[3]。由此,借助SDN/NFV的服务链^[4](Service Chain, SC)技术应运而生。

服务链在利用NFV技术将传统网络功能虚拟化并部署在服务节点的基础上,借助SDN的流量集中管控功能,依据用户或服务请求引导流量按序经过服务节点上的VNF,进而为用户或服务提供定制的网络服务。每条服务链可以视作由一个或多个VNF实例按照既定次序串接而成的虚拟路径,当服务链为用户或服务提供服务时会占用物理基础设施中的相关资源,因此,如何设计合理的服务链映射方法在保证映射效率的同时提高运营商的网络资源利用率已成为该领域的研究热点。

在已有的相关研究工作中,通常依据服务链映射请求,在满足资源约束的条件下,确定VNF实例的部署位置,并规划流量路由路径,以达到相应的优化目标^[5]。Dwaraki等人^[6]从服务链的角度入手,将物理网络拓扑转换为分层图,采用Dijkstra算法逐层选择VNF实例和VNF路径,但未考虑大规模物理网络下分层图带来的存储开销。为此,Xiong等人^[7]着眼于物理基础设施的载荷,分别基于物理节点的服务能力和物理链路的容量约束设计优化算法选择可映射的服务节点和路由路径,但未考虑分阶段映射过程中由于服务节点间距离过大而引入的服务时延。为降低时延开销,Sekar等人^[8]提出基于合并策略的服务链映射方法,将服务链中的VNF实例尽可能多地映射在同一服务器上,但是该方法未考虑服务器的资源载荷和各服务器间的负载均衡。由此,Kuo等人^[9]基于物理平面中链路长度和服务器资源的使用情况,联合优化映射过程中的VNF部署和路径选择问题,该方法虽能满足用户或业务的映射需求,但未从运营商的角度考虑底层物理资源的优化使用。为此,Zhang等人^[10]以最大化平均资源利用率和最小化平均响应时间为目标建立服务链映射多目标优化模型,并根据网络情况和映射请求联合优化服务链部署问题,但该算法的计算复杂性较高,实际运用中映射效率较难保证。

综上所述,以上研究仅考虑服务链对服务节点的偏好或服务节点对服务链的偏好,这都是以最大化自身效用函数为目标的单向选择算法,并未考虑到映射双方的协同作用,因此难以兼顾映射效率与物理资源利用率。为此,本文以最大化网络资源效用为目标建立服务链映射模型,提出一种基于匹配博弈的服务链协同映射方法,同时考虑服务链和服务节点的偏好,运用匹配博弈理论,达到服务链和服务节点的双赢,在此基础上,利用分段路由策略

建立服务链中VNF实例间的连接,在保证映射效率的同时有效提升了物理资源的利用率。实验结果表明本文所提方法能够有效提高运营商网络的服务能力,优化物理资源的使用。

2 服务链映射模型MUSCM

本文首先给出物理基础设施、虚拟网络功能和服务链的定义,在此基础上以最大化网络资源效用为目标,建立服务链映射模型(Maximize Utility Service Chain Mapping, MUSCM),主要符号列表如表1所示。

表1 主要符号列表

符号	描述
G	物理基础设施无向图 $G = (N, E)$
N	物理节点集合, $n \in N$
E	物理链路集合, $e(n_i, n_j) \in E$
N^F	转发节点集合
N^S	服务节点集合
I_k	资源种类, $I_k = \{\text{CPU}, \text{Me}, \text{Th}\}$
$\text{vol}_n^{I_k}$	服务节点 n 的 I_k 类资源的容量
$\text{vol}_{e(n_i, n_j)}^{\text{band}}$	服务节点 n_i 和 n_j 间物理链路的带宽资源容量
P	网络中可提供的所有VNF种类集合
F^p	类型为 p 的VNF实例集合, $f^p \in F^p$
R^c	构成服务链 c 的VNF实例集合
$d_{f^p}^{I_k}$	VNF实例 f^p 的 I_k 类资源需求
C	服务链集合, $c \in C$
$d_c^{I_k}$	服务链 c 的 I_k 类资源需求
$d_{f^p, f^{p'}}^c$	服务链 c 中实例 f^p 和 $f^{p'}$ 间的带宽资源需求
$X_{f^p, n}^c$	服务链 c 中VNF实例 f^p 与服务节点 n 映射关系

2.1 概念定义

定义1 物理基础设施: 物理基础设施用无向图 $G = (N, E)$ 表示, 其中, N 代表物理节点集合, 可将其细分为转发节点集合 N^F 和服务节点集合 N^S , E 表示物理链路集合。对 $\forall n \in N^S$, 用 $\text{vol}_n^{I_k}$ 表示服务节点 n 的 I_k 类资源的容量, $I_k = \{\text{CPU}, \text{Me}, \text{Th}\}$ 。对 $\forall n_i, n_j \in N^S, e(n_i, n_j) \in E$, 用 $\text{vol}_{e(n_i, n_j)}^{\text{band}}$ 表示服务节点 n_i 和 n_j 间物理链路的带宽资源容量。

定义2 虚拟网络功能: 网络中可提供的所有VNF种类(如防火墙、IDS、代理等)用集合 P 表示, 对 $\forall p \in P$, 用 F^p 表示类型为 p 的VNF实例集合, $f^p \in F^p$ 表示集合中的某个VNF实例, $d_{f^p}^{I_k}$ 表示VNF实例 f^p 的 I_k 类资源需求。

定义3 服务链: 网络中的服务链集合用 C 表示, 服务链 $c \in C$ 是由一组有序的VNF实例组成,

$d_c^{I_k}$ 表示服务链 c 的 I_k 类资源需求。为便于讨论,假设每个VNF实例 $f^p \in F^p$ 只属于一条服务链 c (即不存在VNF实例的分割),用 R^c 表示构成服务链 c 的VNF实例集合。此外,用 $d_{f^p, f^{p'}}^c$ 表示服务链 c 中VNF实例 f^p 和 $f^{p'}$ 之间的带宽资源需求。定义布尔变量 $X_{f^p, n}^c$ 表示服务链 c 中VNF实例 f^p 是否映射在服务节点 n 上,是为1,否则为0。

2.2 模型建立

如图1所示,SDN/NFV环境下的服务链映射问题可划分为两个子问题:VNF部署和VNF组链。VNF部署主要解决如何选择合适的服务节点部署VNF实例而VNF组链主要解决如何利用SDN技术进行流量牵引从而完成服务链的构建。

以服务链SC1: FW→IDS为例, VNF部署阶段在确定了SN1上部署FW实例之后,主要解决在SN2还是SN3上部署虚拟IDS实例,而VNF组链阶段主要解决FW实例和IDS实例间流量路由路径的选择,从而完成服务链SC1的构建。下面,首先建立服务链映射模型MUSCM:

$$\text{MUSCM: } \max U(X) = \alpha u_n^N + \beta u_e^L \quad (1)$$

$$\text{s.t. } \sum_{c \in C} \sum_{n \in N^S} X_{f^p, n}^c = 1, \forall f^p \in F^p, p \in P \quad (2)$$

$$\left. \begin{aligned} \sum_{c \in C} \sum_{f^p \in F^p} X_{f^p, n}^c d_{f^p}^{I_k} &\leq \text{vol}_n^{I_k} \\ \forall n \in N^S, \forall p \in P \end{aligned} \right\} \quad (3)$$

$$\left. \begin{aligned} \sum_{c \in C} \sum_{\substack{f^p \in F^p \\ f^{p'} \in F^{p'}}} X_{f^p, n}^c X_{f^{p'}, n'}^c d_{f^p, f^{p'}}^c &\leq \text{vol}_{e(n, n')}^{\text{band}} \\ \forall n, n' \in N^S, \forall e(n, n') \in E, \forall p \in P \end{aligned} \right\} \quad (4)$$

$$X_{f^p, n}^c = \{0, 1\}, N^S \leq |N^S| \quad (5)$$

约束条件: 式(2)限定服务链 c 中的每个VNF实例只能映射在一个服务节点上, 式(3)确保承载VNF实例的服务节点满足节点资源约束(包含CPU、Me和Th 3个方面), 式(4)确保承载VNF实例间连接的物理链路满足链路带宽约束, 式(5)限定了变量的取值范围。以上5条为MUSCM问题的基本约束条件。

目标函数: MUSCM问题的目标是最大化网络资源效用 $U(X)$, $U(X)$ 分别由节点资源效用 u_n^N 和链路资源效用 u_e^L 构成, α 和 β 代表二者的权重系数。用指数函数 $\exp(\cdot)$ 将比值进行转化, 使得网络资源效用 $U(X)$ 和物理资源利用率成正相关。

节点资源效用 u_n^N 可定义为

$$u_n^N = \exp \left[\frac{\sum_{c \in C} \sum_{f^p \in F^p} X_{f^p, n}^c d_{f^p}^{I_k}}{\text{vol}_n^{I_k}} \right], \quad \forall n \in N^S, \forall p \in P \quad (6)$$

其中, $\omega_{I_k}^N$ 代表节点中 I_k 类资源的效用转化因子, 且 $\sum_{I_k} \omega_{I_k}^N = 1$ 。

链路资源效用 u_e^L 定义为

$$u_e^L = \exp \left[\frac{\sum_{c \in C} \sum_{\substack{f^p \in F^p \\ f^{p'} \in F^{p'}}} X_{f^p, n}^c X_{f^{p'}, n'}^c d_{f^p, f^{p'}}^c}{\text{vol}_{e(n, n')}^{\text{band}}} \right], \quad \forall n, n' \in N^S, \forall e(n, n') \in E, \forall p \in P \quad (7)$$

其中, ω_{band}^L 代表链路带宽资源的效用转化因子。

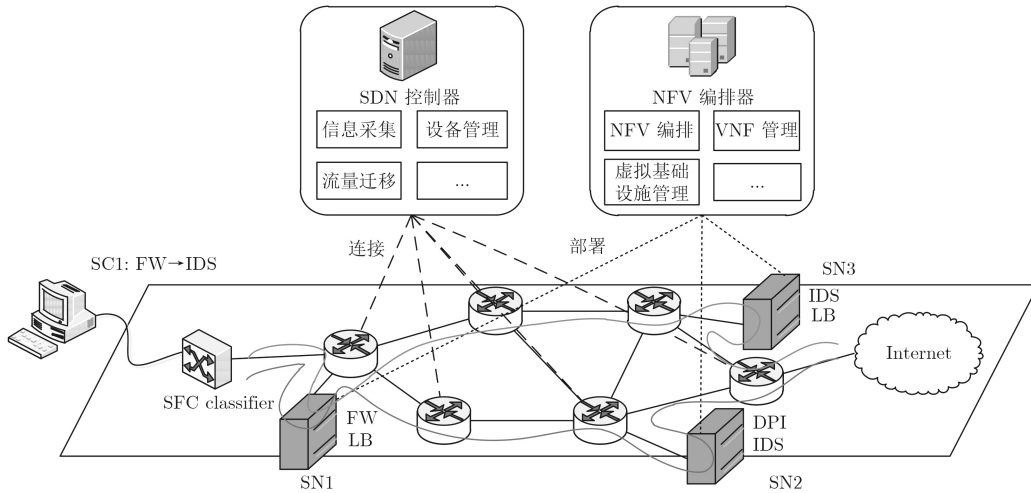


图1 SDN/NFV环境下服务链映射问题

MUSCM问题旨在寻求满足约束条件下,网络资源效用最大化的映射方案。这是一个离散解空间下的组合优化问题,同时也是一个NP困难问题,难以在多项式时间内得到解决。

3 基于匹配博弈的服务链协同映射方法

匹配博弈^[11]: (matching game)是研究和应用都非常广泛的一类合作博弈,主要研究双边市场的稳定匹配及匹配的最优稳定解问题。作为匹配博弈中的典型应用,多对一匹配模型曾成功应用于解决数据中心中虚拟机的部署问题^[12]。本文结合服务链映射过程中多个VNF实例可映射在一个服务节点上的特性,将由VNF实例构成的服务链和服务节点作为匹配博弈的双方,分别设计服务链匹配偏好表构建算法、服务节点匹配偏好表构建算法、博弈选择算法以及分段路由连接算法4个子算法解决VNF部署和VNF组链问题,以期高效地完成系统中的服务链映射。

3.1 算法准备

定义 4 偏好: 对 $\forall c \in C, n \succ_c n'$ 表示服务链 c 相较于服务节点 n' 更偏好于服务节点 n , $\phi \succ_c n$ 表示服务链 c 不接受服务节点 n ; 对 $\forall n \in N^S, p, p' \in P, f^p \succ_n f^{p'}$ 表示服务节点 n 相较于VNF实例 $f^{p'}$ 更偏好VNF实例 f^p , $\phi \succ_n f^p$ 表示服务节点 n 不接受VNF实例 f^p 。

定义 5 匹配: VNF实例在服务节点上的部署结果是产生一个确定的匹配。这种匹配关系可用匹配函数 $m: R^c \cup N^S \rightarrow 2^{R^c \cup N^S}$ 表述, m 是一个多值映射,表示构成服务链 c 的VNF实例集合 R^c 和服务节点集合 N^S 经匹配选择后,可产生一个多对一的映射关系:

(1) $\forall n \in N^S, f^p \in F^p, p \in P$, 当 $m(R^c) \subseteq N^S$ 时, $0 \leq |m(f^p)|^{I_k} \leq \text{vol}_n^{I_k}$, 其中 $|m(f^p)|^{I_k}$ 表示匹配到VNF实例 f^p 的资源总量, 当 $|m(f^p)|^{I_k} = 0$ 时, 说明VNF实例 f^p 未匹配成功;

(2) $\forall n \in N^S$, 当 $m(N^S) \subseteq R^c$ 时, $0 \leq |m(n)|^{I_k} \leq \text{vol}_n^{I_k}$, 其中 $|m(n)|^{I_k}$ 表示匹配到服务节点 n 的资源总量, 当 $|m(n)|^{I_k} = 0$ 时, 说明服务节点 n 未匹配成功;

(3) $\forall n \in N^S, f^p \in F^p, p \in P$, 当且仅当 $f^p \in m(n)$ 时, $m(f^p) = n$ 。

定义 6 阻塞对: 当匹配 m 中存在, $f^p \notin m(n)$ 且 $n \notin m(f^p)$, 但 $f^p \succ_n m(n)$ 且 $n \succ_c m(f^p)$ 时, 称 (f^p, n) 为一个阻塞对。

定义 7 饱和: 对服务链 $c \in C$, 当链中的所

有VNF实例都匹配成功时, 称之为服务链饱和; 对服务节点 $n \in N^S$, 当服务节点资源用尽时, 称之为服务节点饱和。为满足用户需求, 所有的服务链需要映射到服务节点中, 即系统中不能存在不饱和的服务链。

定义 8 稳定匹配: 当一个匹配 m 同时满足不存在阻塞对并且使得服务链饱和时, 称 m 为稳定匹配。

定义 9 分段路由路径: 对 $\forall \text{begin}, \text{end} \in N^S$, 一条从begin到end的分段路由路径是一个非空的转发图FG序列:

$$\{\text{FG}(\text{begin}, n_1), \text{FG}(n_1, n_2), \dots, \text{FG}(n_{i-1}, n_i), \text{FG}(n_i, \text{end})\}$$

其中, 前序子图的目的节点是后续子图的起始节点, 首图和末图的起始节点和目的节点分别对应分段路由路径的源节点和目的节点。

3.2 算法描述

VNF部署阶段主要涉及服务链匹配偏好表构建算法、服务节点匹配偏好表构建算法和博弈选择算法3个子算法, 前两个算法主要用于建立博弈双方的偏好表, 而博弈选择算法的目的在于依据映射双方的偏好表快速找到MUSCM问题的稳定解; VNF组链阶段的分段路由连接算法主要负责VNF部署的服务节点间路由路径的选择, 从而有效完成服务链的构建。下面, 我们将对这4个子算法分别展开描述。

3.2.1 服务链匹配偏好表构建算法

一条服务链是由一组VNF实例按序连接而成, 为在提高节点资源利用率的同时降低节点间通信时延开销, 服务链中的各VNF实例偏好部署于同一服务节点上。因此, 我们认为一条服务链中的所有VNF实例都具有与服务链相同的偏好, 构成服务链匹配偏好表 $\text{SC}(N^S)$ 。每条服务链 $c \in C$ 对服务节点集合 N^S 具有完整、严格、可传递的偏好关系。

为达到网络中服务等级协议(SLA)的要求, 每条服务链 c 旨在寻找满足资源约束条件下产生服务时延最短的服务节点进行映射。首先, 为保证服务链 c 在匹配之初的可行性, 基于文献^[13]中提出的规范指标选择满足服务链 c 资源需求的最佳服务节点作为偏好表中的初始节点:

$$\Delta R = \sum_{I_k} \omega_{I_k} (d_c^{I_k} - \text{vol}_n^{I_k})^2 \quad \left. \vphantom{\sum_{I_k}} \right\} \quad (8)$$

$$\forall c \in C, \forall n \in N^S$$

式中的 ΔR 可以作为度量服务链 c 和服务节点 n 之间

资源偏差的指标，其值越小，代表二者之间差异越小，因此更适合被选作服务链 c 的映射节点。

下面对服务时延问题进行讨论，假设服务链 c 以初始节点作为当前节点，其偏好选择满足资源约束条件(式(3)、式(4))下，距离当前节点最近的服务节点作为下一候选节点，以期产生最小的链路传输时延。该问题类似于终端节点不重合的旅行商问题(Traveling Salesman Problem, TSP)，可依据邻近算法^[14](Nearest Neighbor Algorithm, NNA)迭代计算，直至找出满足服务时延最短的所有服务节点，并将其按照偏序关系排列，从而构成服务链匹配偏好表：

$$SC(N^S) = \{n_0 > n_1 > \dots > n_m\} \quad (9)$$

其中， n_m 表示第 m 个候选服务节点，算法详细描述见表2。注意：该算法将针对一条特定的服务链 $c \in C$ 中VNF实例与服务节点的映射关系进行说明。

表2 服务链匹配偏好表构建算法

算法1 服务链匹配偏好表构建算法

输入：构成服务链 c 的VNF实例集合 R^c ，服务节点集合 N^S ，物理基础设施图 G

输出：服务链匹配偏好表 $SC(N^S)$

- (1) 根据输入初始化；
- (2) for each n in N^S do
 - 依据式(8)计算资源偏差 ΔR ；
 - 选择 ΔR 最小的服务节点 n_0 ；
 - $SC(N^S) \leftarrow \{n_0\}$ ；//将 n_0 作为起始节点，并加入偏好表
 - 将 n_0 标记为已读；
 - end for
- (3) while N^S 中存在未读节点 do
 - 依据NNA算法寻找图 G 中距离当前节点最近的且满足资源约束条件(式(3)、式(4))的下一服务节点 n_w ；
 - $SC(N^S) \cup \{n_w\}$ ；//将 n_w 加入偏好表
 - 将 n_w 标记为已读；
- (4) 所有服务节点已读则终止；
- (5) 返回步骤(3)；
- (6) 输出服务链匹配偏好表 $SC(N^S)$

3.2.2 服务节点匹配偏好表构建算法

为降低服务供应商的资本支出(CAPEX)和运营成本(OPEX)，映射过程中，应使得服务节点的资源碎片化程度最小，从而让服务节点上有限的物理资源发挥最大的作用。假定物理基础设施 G 中的所有服务节点都具有相同的偏好，构成服务节点映射偏好表 $N^S(SC)$ 。每个服务节点 $n \in N^S$ 对构成服务链 c 的VNF实例集合具有完整、严格、可传递的偏好关系。

基于服务链映射的合并策略^[4](consolidation policy)，每个服务节点 n 偏好部署尽可能多的VNF实例以提升节点的资源利用率，但这并不能保证各服务节点的资源碎片化程度最小。为此，采用最佳适应算法^[14](Best Fit Algorithm, BFA)，对服务链 c 中的VNF实例资源需求进行从大到小排序，按照贪心启发式思想优先选择使得服务节点剩余资源空间最小的VNF实例，构成服务节点匹配偏好表：

$$N^S(SC) = \{f_1^p > f_2^p > \dots > f_t^p\} \quad (10)$$

其中， f_t^p 表示服务链 c 中的第 t 个候选VNF实例，算法详细描述见表3。注意：该算法将针对一条特定的服务链 $c \in C$ 中VNF实例与服务节点的映射关系进行说明。

表3 服务节点匹配偏好表构建算法

算法2 服务节点匹配偏好表构建算法

输入：构成服务链 c 的VNF实例集合 R^c ，服务节点集合 N^S

输出：服务节点匹配偏好表 $N^S(SC)$

- (1) 根据输入初始化；
- (2) for each f^p in R^c do
 - 按资源需求对 f^p 进行排序；//资源需求优先级依次为CPU > Me > Th
 - 依据BFA算法选择剩余资源空间最小的VNF实例 f^p ；
 - $N^S(SC) \cup \{f^p\}$ ；//将 f^p 加入偏好表
 - 将VNF实例 f^p 标记为已读；
- end for
- (3) 所有VNF实例已读则终止；
- (4) 返回步骤(2)
- (5) 输出服务节点匹配偏好表 $N^S(SC)$

3.2.3 博弈选择算法

GS算法^[15]又称延迟接受算法，是解决多对一双边匹配问题的经典算法，其最早应用于大学招生领域。由于GS算法具有对匹配提出方的优化倾向性，在权衡MUSCM问题中匹配双方的重要性后，本文将由VNF实例构成的服务链作为匹配的提出方进行博弈。

首先，由算法1和算法2分别构建服务链匹配偏好表 $SC(N^S)$ 和服务节点匹配偏好表 $N^S(SC)$ ，服务链依据其偏好表向第1个服务节点提出匹配请求，服务节点可根据其偏好表决定是否接受该VNF实例，如果VNF实例不在其偏好表中或VNF实例的资源需求超过了服务节点的阈值，则拒绝接受该VNF实例，反之，则接受；未匹配的VNF实例可依据其偏好表向下一个服务节点提出匹配请求，如此迭代进行，直到所有VNF实例完成匹配。依据

GS算法中延迟接受的特性,一个服务节点可以拒绝接受当前VNF实例并选择其偏好表中排序更高的VNF实例,但当服务节点拒绝接受某一VNF实例后,其偏好表中排序低于此的VNF实例也将被拒绝,算法详细描述见表4。注意:该算法将针对一条特定的服务链 $c \in C$ 中VNF实例与服务节点的映射关系进行说明。

表 4 博弈选择算法

算法3 博弈选择算法
输入: 构成服务链 c 的VNF实例集合 R^c , 服务节点集合 N^S
输出: 服务链和服务节点的平稳匹配结果
根据输入初始化;
if 服务链 c 是不饱和的 do
for each f^p in R^c do
$n \leftarrow \text{get highest rank in SC}(N^S)$;
if $(\text{vol}_n^{I_k} > d_{f^p}^{I_k}) \&\& (f^p \text{ in } N^S(\text{SC}))$ then
将 f^p 匹配给 n ;
$\text{vol}_n^{I_k} = \text{vol}_n^{I_k} - d_{f^p}^{I_k}$;
end
else
找出所有满足 $f^p \succ_n f^{p'}$ 的 $f^{p'}$;
拒绝所有 $f^{p'}$ 并更新服务节点 n 的资源;
$\text{vol}_n^{I_k} = \text{vol}_n^{I_k} - d_{f^p}^{I_k}$;
将 $f^{p'}$ 从服务节点映射偏好表 $N^S(\text{SC})$ 中移除;
将 n 从服务链映射偏好表 $\text{SC}(N^S)$ 中移除;
end
end for
输出匹配结果;
else
return

3.2.4 分段路由连接算法

上述3个子算法确定了VNF实例与服务节点间的部署关系,下面将基于分段路由策略^[16]设计连接算法建立服务节点之间的最优路由路径。假设节点映射关系集合为 $\Omega = \{n_1^{\text{ma}}, n_2^{\text{ma}}, \dots, n_i^{\text{ma}}\}$, 设定节点 n_c^{sou} 和 n_c^{des} 为服务链 c 的起始节点和目的节点,那么其分段路由路径可表示为

$$\{\text{FG}(n_c^{\text{sou}}, n_1^{\text{ma}}), \text{FG}(n_1^{\text{ma}}, n_2^{\text{ma}}), \dots, \text{FG}(n_{i-1}^{\text{ma}}, n_i^{\text{ma}}), \text{FG}(n_i^{\text{ma}}, n_c^{\text{des}})\} \quad (11)$$

进而,依据最短路径算法^[14](Shortest Path Algorithm, SPA)计算每一个转发子图FG的连接路径,构成服务链 c 的流量传输通路,算法详细描述见表5。注意:该算法将针对一条特定的服务链 $c \in C$ 中VNF实例与服务节点的映射关系进行说明。

表 5 分段路由连接算法

算法4 分段路由连接算法
输入: VNF实例与服务节点间的映射关系 Ω
输出: 服务链 c 的路由路径
根据输入初始化;
依据式(11)表达 Ω 中服务节点的分段路径, 构成集合 S
for each FG $(n_{i-1}^{\text{ma}}, n_i^{\text{ma}})$ in S do
利用SPA算法计算连接路径;
end for
输出计算结果;
return

3.3 计算复杂度分析

本节将对本文所提的服务链协同映射方法的计算复杂度进行分析。为便于讨论,将网络中服务节点的个数记为 A ,某条链中VNF实例的个数记为 B (假设所有链中VNF实例的个数相等),网络中的服务链总数记为 C ,任意两个服务节点间的转发节点数记为 D (假设服务节点间的转发节点个数相等)。以一条特定的服务链 $c \in C$ 为例,服务链匹配偏好表构建算法的计算复杂度为 $A + A \log_2 A$,服务节点匹配偏好表构建算法的计算复杂度为 $AB \log_2 B$,博弈选择算法在最坏条件下的计算复杂度为 B^2 ,分段路由连接算法的计算复杂度为 $BD \log_2 D$,因此,整个方法的计算复杂度为 $A + B^2 + A \log_2 A + AB \log_2 B + BD \log_2 D$,考虑到网络中有 C 条服务链,总的计算复杂度为 $C(A + B^2 + A \log_2 A + AB \log_2 B + BD \log_2 D)$,即 $O(B^2)$ 。

文献^[10]提出的服务链映射方法计算复杂度为 $O(AB \log B)$,随着网络规模的增大,服务节点数目 A 会不断增加,产生结果是 $A \gg B$,从而对算法的计算复杂度产生较大的影响。因此,本文方法在计算复杂度方面优于文献^[10]。

4 实验结果及性能分析

本文将围绕以下4个方面评估模型可行性及算法有效性:

(1)服务链映射请求接受率。服务链映射请求接受率是指一定请求强度下,网络中接受的服务链数目占总的服务链映射请求数目的比例,反映了算法对映射请求处理的成功程度。

(2)服务链的平均传输时延。服务链的平均传输时延是指服务链整体在链路上的传输时延与服务链长度的比值,反映了算法对映射请求的响应效率。

(3)服务节点的平均资源利用率。服务节点的平均资源利用率是指某一时刻,网络中各服务节点资源使用值与资源额定值比值的均值(重点考虑服

务节点的CPU资源),反映了算法对服务节点的资源使用情况。

(4)物理链路的平均带宽利用率。物理链路的平均带宽利用率是指某一时刻,网络中各物理链路带宽资源使用值与额定值比值的均值,反映了算法对物理链路带宽资源的使用情况。

4.1 实验设置

实验在配置Intel Core i7-6300HQ, 3.60 GHz CPU, 16 GB内存的Linux系统PC机上运行,实验参数设置如表6和表7所示,底层物理网络拓扑由GT-ITM工具^[17]生成,每对物理转发节点之间以等概率生成物理链路。为评估算法的自适应性,假设服务链映射请求动态到达,且服从强度为 $[0, 1000]$ 的泊松分布,每个映射请求由一个或多个VNF组成,数量服从 $[2, 4]$ 的均匀分布。

表 6 服务节点参数设置

CPU	内存	吞吐量
32	100 GB	10 Gbps

表 7 VNF参数设置

VNF种类	CPU	内存(GB)	吞吐量(Mbps)
Firewall	8	4	200
IDS	8	4	80
IPS	4	4	268
WAN-opt	2	2	10

为便于讨论,模型中的权重系数 α 和 β 均取1。考虑到节点资源效用中CPU资源的重要性,可将其效用转化因子设置为0.5,内存和吞吐量的效用转化因子均设置为0.25。链路带宽资源的效用转化因子设置为1,实际运用中可根据需要进行调整。

4.2 实验结果分析

为对比验证本文所提算法的有效性,本文将基于合并策略^[8]的服务链映射方法(Co-A)、基于随机策略^[4]的服务链映射方法(Ra-A)和本文方法进行比较分析。Co-A方法采用经典的节点聚合策略,在优化物理资源利用率方面具有一定的代表性。Ra-A方法则是采用随机化的方法进行服务链的映射,可以作为对比验证的基准。本文方法中,首先通过算法1和算法2建立服务链和服务节点匹配偏好表,然后采用算法3完成服务链中VNF实例的部署,最后利用算法4建立实例间的路由路径,完成服务链的构建,具体实验结果如下:

(1)服务链映射请求接受率:在物理网络规模恒定条件下(设定为 25×25),随机生成500条服务

链,对各个映射方法的服务链映射请求接受率进行分析。如图2所示,当服务链映射请求强度较低时,各方法的接受率相近,而当请求强度进一步增加时,各个方法的接受率都有不同程度的下降,但本文方法在请求强度为1000时,仍能保证61%左右的接受率,明显高于另外两种方法,原因在于:本文方法分别从博弈双方(VNF实例和服务节点)角度考虑对象的选择问题,兼顾了二者需要满足的匹配条件,既避免了Ra-A方法的随机性,又降低了Co-A方法中由于服务节点拥塞而导致服务链映射请求被拒绝的概率。

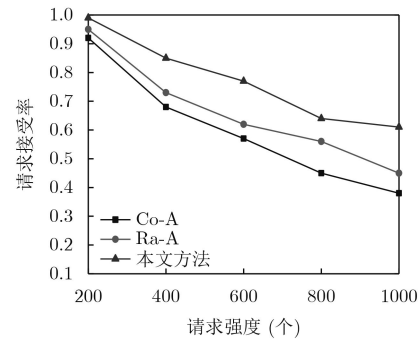


图 2 服务链映射请求接受率对比

(2)服务链的平均传输时延:在物理网络规模(设定为 25×25)及服务链映射请求强度(200个)恒定的条件下,随机生成500条服务链并采用3种不同的方法部署实验,对各个方法的平均传输时延进行分析。如图3所示,3种方法产生的平均传输时延差异较大,其中Co-A方法时延最小,仅为0.7 s, Ra-A方法时延最大,高达3.4 s,而本文方法位于二者之间,在2.1 s左右,原因在于:Co-A方法将一条服务链中尽可能多的VNF实例映射在同一个服务节点上,降低节点间链路传输的时延,而同一服务节点上各虚拟机间的时延几乎可以忽略,因此,Co-A方法的时延表现最优;Ra-A方法未考虑节点间传输时延的影响,在满足约束条件下随机进行VNF实例的映射,从而导致节点间距离过大而产生过高

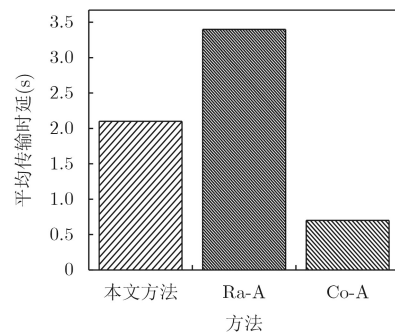


图 3 服务链的平均传输时延对比

的传输时延,因而时延表现最差。本文方法虽然达不到Co-A方法的时延表现,但较Ra-A方法而言,已有了较大程度的提升,是因为本文方法总是在匹配下一服务节点时,选择距离当前节点最近的节点,从而有效降低了链路间的时延开销。

(3)服务节点的平均资源利用率:在物理网络规模(设定为 25×25)及服务链映射请求强度(200个)恒定的条件下,随机生成500条服务链并采用3种不同的方法部署实验,对各个方法在不同时刻服务节点的平均资源利用率进行分析。如图4所示,Co-A方法的平均资源利用率约为21%,Ra-A方法的平均资源利用率约为50%,而本文方法的平均资源利用率在74%左右。原因在于:Co-A方法偏好将一条服务链中的VNF实例聚集在一个服务节点上,导致系统中某一服务节点的资源利用率过高,而其他服务节点利用率低下甚至闲置,因此该方法下服务节点的平均资源利用率表现最差;Ra-A方法中,VNF实例与服务节点随机映射,因此相较于其他方法,该方法下服务节点的平均资源利用率波动最为明显;本文方法首先从服务链的角度出发,将VNF实例尽可能映射在地理位置相对集中的不同服务节点上,均衡各服务节点的使用,然后从服务节点的角度考虑,尽可能选择资源碎片化程度小的VNF实例进行匹配,有效提升了系统中服务节点的平均资源利用率。

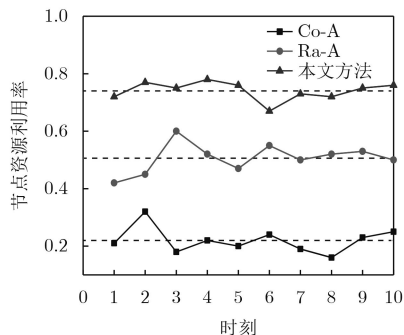


图4 服务节点的平均资源利用率对比

(4)物理链路的平均带宽利用率:在物理网络规模(设定为 25×25)及服务链映射请求强度(200个)恒定的条件下,随机生成500条服务链并采用3种不同的方法部署实验,对各个方法在不同时刻物理链路的平均带宽利用率进行分析。如图5所示,Co-A方法的平均资源利用率约为23%,Ra-A方法的平均资源利用率约为47%,而本文方法的平均资源利用率在67%左右。原因在于:Co-A方法将一条服务链中的VNF实例尽可能多地聚集在一个服务节点上,导致系统中与某一服务节点相连的

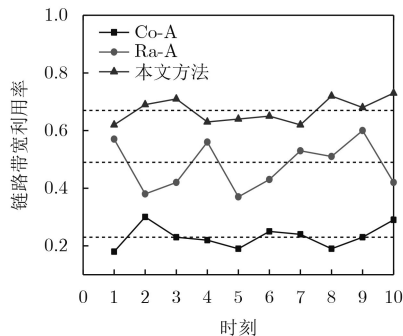


图5 物理链路的平均带宽利用率对比

物理链路带宽利用率过高,而其他链路的带宽资源未能得到充分使用,因此该方法下物理链路的平均带宽利用率表现最差;Ra-A方法中,映射节点选取的随机性较大,因此相较于其他方法,该方法下物理链路的平均带宽利用率波动最为明显;本文方法在选择了地理位置相对集中的服务节点基础上,通过分段路由策略构建转发子图,子图内采用最短路径算法,使得物理链路较为均匀地分布在网络中,因此获得了较高的物理链路平均带宽利用率。

5 结束语

服务链映射是SDN/NFV体系架构中的核心,重点解决网络服务资源的动态调度和灵活编排问题。本文首先以最大化网络资源效用为目标,建立服务链映射模型MUSCM。在此基础上,提出一种基于匹配博弈的服务链协同映射方法。将服务链映射问题分解为VNF部署和VNF组链两个部分,分别基于多对一匹配博弈理论和分段路由策略设计算法求解此问题,仿真实验对比说明了本文所提算法在服务链映射请求接受率、服务链的平均传输时延、服务节点的平均资源利用率和物理链路的平均带宽利用率方面取得了较好的优化效果。后续工作中将针对可靠性条件下的服务链映射问题展开深入研究。

参考文献

- [1] KREUTZ D, RAMOS F M V, VERÍSSIMO P E, *et al.* Software-defined networking: A comprehensive survey[J]. *Proceedings of the IEEE*, 2015, 103(1): 14-76. doi: 10.1109/JPROC.2014.2371999.
- [2] MIJUMBI R, SERRAT J, GORRICO J L, *et al.* Network function virtualization: State-of-the-art and research challenges[J]. *IEEE Communications Surveys and Tutorials*, 2017, 18(1): 236-262. doi: 10.1109/COMST.2015.2477041.
- [3] OCAMPO A F, GIL-HERRERA J, ISOLANI P H, *et al.* Optimal service function chain composition in network functions virtualization[C]. *International Conference on*

- Autonomous Infrastructure, Management and Security, Zurich, Switzerland, 2017: 62–76.
- [4] LI Yong and CHEN Min. Software-defined network function virtualization: A survey[J]. *IEEE Access*, 2015, 3: 2542–2553. doi: [10.1109/ACCESS.2015.2499271](https://doi.org/10.1109/ACCESS.2015.2499271).
- [5] BHAMARE D, JAIN R, SAMAKA M, *et al.* A survey on service function chaining[J]. *Journal of Network and Computer Applications*, 2016, 75: 138–155. doi: [10.1016/j.jnca.2016.09.001](https://doi.org/10.1016/j.jnca.2016.09.001).
- [6] DWARAKI A and WOLF T. Adaptive service-chain routing for virtual network functions in software-defined networks[C]. The Workshop on Hot Topics in Middleboxes and Network Function Virtualization, Salvador, Brazil, 2016: 32–37.
- [7] XIONG Gang, HU Yunxiang, LAN Julong, *et al.* A mechanism for configurable network service chaining and its implementation[J]. *KSI Transactions on Internet and Information Systems*, 2016, 10(8): 3701–3727. doi: [10.3837/tiis.2016.08.016](https://doi.org/10.3837/tiis.2016.08.016).
- [8] SEKAR V, EGI N, RATNASAMY S, *et al.* Design and implementation of a consolidated middlebox architecture[C]. Usenix Conference on Networked Systems Design and Implementation, Lombard, Italy, 2012: 24–34.
- [9] KUO Tunwei, LIOU Bangheng, LIN Chingju, *et al.* Deploying chains of virtual network functions: On the relation between link and server usage[C]. IEEE International Conference on Computer Communications, San Francisco, USA, 2016: 1–9.
- [10] ZHANG Qixia, XIAO Yikai, LIU Fangming, *et al.* Joint optimization of chain placement and request scheduling for network function virtualization[C]. IEEE International Conference on Distributed Computing Systems, Atlanta, USA, 2017: 731–741.
- [11] GALE D and SHAPLEY L S. College admissions and the stability of marriage[J]. *American Mathematical Monthly*, 1962, 69(1): 9–15. doi: [10.4169/amer.math.monthly.120.05.386](https://doi.org/10.4169/amer.math.monthly.120.05.386).
- [12] XU Hong and LI Baochun. Anchor: A versatile and efficient framework for resource management in the cloud[J]. *IEEE Transactions on Parallel and Distributed Systems*, 2013, 24(6): 1066–1076. doi: [10.1109/TPDS.2012.308](https://doi.org/10.1109/TPDS.2012.308).
- [13] ZHANG Yan and ANSARI N. Heterogeneity aware dominant resource assistant heuristics for virtual machine consolidation[C]. Global Communications Conference, Austin, USA, 2014: 1297–1302.
- [14] CORMEN T T, LEISERSON C E, RIVEST R L, *et al.* Introduction to Algorithms[M]. McGraw, USA, MIT Press, 2002: 145–167.
- [15] MANLOVE D. Algorithmics of Matching under Preferences[M]. Glasgow, UK, World Scientific Pub. Co, 2013: 266–278.
- [16] FILSFILS C, NAINAR N K, PIGNATARO C, *et al.* The segment routing architecture[C]. Global Communications Conference, San Diego, USA, 2015: 1–6.
- [17] CALVERT K L and BHATTACHARJEE S. How to model an internetwork[C]. IEEE Conference of Computer Societies, San Francisco, USA, 1996: 594–602.
- 张红旗：男，1962年生，教授，博士生导师，研究方向为网络安全和信息系统安全。
- 黄 睿：女，1993年生，硕士生，研究方向为软件定义网络和网络功能虚拟化。
- 常德显：男，1977年生，副教授，研究方向为网络安全和信息系统安全。