

一种支持硬件加速的虚拟网络功能部署模型

胡宇翔 范宏伟* 兰巨龙 段通

(国家数字交换系统工程技术研究中心 郑州 450002)

摘要: 为解决以软件实现的虚拟网络功能(VNF)性能受限问题,软件定义网络和网络功能虚拟化(SDN/NFV)等新型网络架构引入了硬件加速资源。硬件加速资源的部署,使得VNF能够为日益增长的数据流量提供服务保障。该文针对已有研究未考虑具有高性能数据处理需求的服务链VNF部署问题,提出一种支持硬件加速的VNF部署模型。该模型基于硬件加速资源的承载特性,在保证未加速VNF到商用服务器的优化部署下,优先实现交换机中加速资源的复用,并根据网络业务的性能需求,灵活调整加速资源与VNF的映射约束。仿真实验表明,与其他典型部署方法相比,在引入相同硬件加速资源的情况下,该模型可以承载更多的业务流量,满足服务链高性能数据处理需求,有效提高了部署在网络中加速硬件的资源利用率。

关键词: 软件定义网络;网络功能虚拟化;虚拟网络功能部署;硬件加速

中图分类号: TP393

文献标识码: A

文章编号: 1009-5896(2019)08-1893-09

DOI: 10.11999/JEIT180861

A Model for Virtualized Network Function Placement with Hardware Acceleration Support

HU Yuxiang FAN Hongwei LAN Julong DUAN Tong

(National Digital Switching System Engineering & Research Center, Zhengzhou 450002, China)

Abstract: In order to deal with the limited capacity of Virtualized Network Function (VNF), hardware acceleration resources are adopted in Software-Defined Networking and Network Function Virtualization (SDN/NFV) architecture. The deployment of hardware acceleration resources enables VNF to provide service guarantees for increasing data traffic. To overcome the ignorance of the requirements for VNF with high processing throughput in service chain in existing researches, a model for VNF placement with hardware acceleration support is proposed. Based on the bearing characteristics of hardware acceleration resources, the model prioritizes the reuse of acceleration resources in the switch under the optimal placement of VNF without acceleration to commercial servers. The mapping correlation between hardware acceleration resources and VNF is flexibly adjusted according to the requirements of network services. Simulation results show that the proposed model can bear more service flows and meet the high processing throughput needs of service chains than typical policies in the case of the same amount of resources, which improves effectively the resource utilization of the acceleration hardware deployed in the network.

Key words: Software-Defined Network (SDN); Network Function Virtualization (NFV); Virtualized Network Function (VNF) placement; Hardware acceleration

1 引言

传统网络功能基于专用硬件实现,灵活性差,

维护复杂,难以满足未来网络的需求。为实现网络功能的灵活部署,以软件定义网络(Software-Defined Networking, SDN)和网络功能虚拟化(Network Function Virtualization, NFV)为代表的虚拟化技术,为当前网络结构面临的僵化问题提供了解决途径。SDN和NFV增强了网络的可扩展性和网络配置的灵活性,推进了整个网络体系的开放虚拟化发展。然而,在SDN/NFV架构中,以软件形式实现的虚拟网络功能(Virtualized Network Function, VNF)会带来显著的性能开销受限的数据处理速率

收稿日期: 2018-09-04; 改回日期: 2019-01-04; 网络出版: 2019-01-16

*通信作者: 范宏伟 fhwxd@foxmail.com

基金项目: 国家网络空间安全专项基金(2017YFB0803204), 国家自然科学基金(61521003)

Foundation Items: The National Network Security Special Program of China (2017YFB0803204), The National Natural Science Foundation of China (61521003)

和不稳定的处理时延,无法满足数据处理速率较高和时延敏感的网络业务需求^[1]。此外,随着大数据、5G^[2]和人工智能等领域的发展,数据流量在未来几年内将呈指数级增长,对高性能VNF的需求日益迫切。

为提升VNF处理性能,基于硬件加速的实现方法成为学术界和工业界的研究热点^[3]。根据加速硬件的特性,可将当前的硬件加速机制研究总结为两个方向:(1)利用转发节点内的交换机承载无状态VNF和有状态的轻量级VNF^[4,5]; (2)利用加载到NFV服务器内的可编程硬件如图形处理器(GPU)^[6]、网络处理器(NP)^[7]和FPGA^[8,9],实现加速。其中,FPGA因为更高的能耗比和功能定制能力,成为学术界服务器端加速研究的主流平台。

目前,VNF的硬件加速研究处于初始阶段,仍有许多问题亟待研究。为满足业务需求,网络流量通常要按特定路径,遍历部署在不同服务器内的多个VNF,即服务功能链(Service Function Chain, SFC)。本文主要关注SDN/NFV架构下,如何在硬件加速资源约束的条件下保证VNF的合理部署和编排,实现服务功能链的加速处理。实现VNF编排和部署在不同应用场景下的优化目标,一直是SDN/NFV领域的研究热点。文献^[10]针对用户服务的可靠性需求及电信服务提供商(Telecommunication Service Providers, TSPs)的成本管理问题,从适当降低SFC的可靠性角度出发,提出一种可靠性感知的VNF部署模型,实现用户服务和TSPs经济策略之间的平衡。文献^[11]研究了VNF部署和流量路由的联合问题,基于对虚拟机资源的复用,提出一种数学方法来估算路径长度和复用因子,并以此为指导完成既定目标约束下的VNF部署和路由

选择。文献^[12]从运营商角度出发,提出一种减少资源碎片的SFC构建模型,该模型将路径选择和服务链中多个VNF的部署进行融合,优化了资源的使用效率,实现了容纳更多业务流量的优化目标。文献^[13]对部署于同一服务器上的VNF性能进行测量研究,从资源分配、位置选择和动态缩放3个角度给出了VNF编排和部署的建议。文献^[14]则针对5G移动通信场景的VNF编排问题,将时延等作为优化目标,采用马尔可夫模型从时间和空间两个维度完成VNF的部署。文献^[15]提出一种在分布式云网络场景下的快速VNF部署算法,将服务链路由和VNF部署的联合问题表示为多模式链模型,通过双随机次梯度算法对线性优化目标进行求解。现有研究更多关注计算和网络资源,其部署算法并未考虑对硬件加速资源的调用,不能满足服务链的加速需求;而且,由于加速资源与计算和网络资源之间形态的差异性,基于通用服务器的VNF部署方法无法适用到加速硬件上。因此,为实现高性能需求的服务功能链部署,本文提出一种支持硬件加速的虚拟网络功能部署模型,本模型在充分利用硬件承载特性的基础上,合理复用网络中交换机内的表项资源,灵活调整映射约束,满足了网络业务的性能需求,实现了包括硬件加速资源在内的物理基础设施以最优方式承载服务功能链。

2 问题描述与模型建立

2.1 问题描述

首先,为更清楚地描述本文所要解决的支持硬件加速的VNF部署问题,对现有研究中的两类加速资源对VNF的承载特性进行分析。如图1(a)所示,第1类加速资源利用SDN交换机的硬件处理能

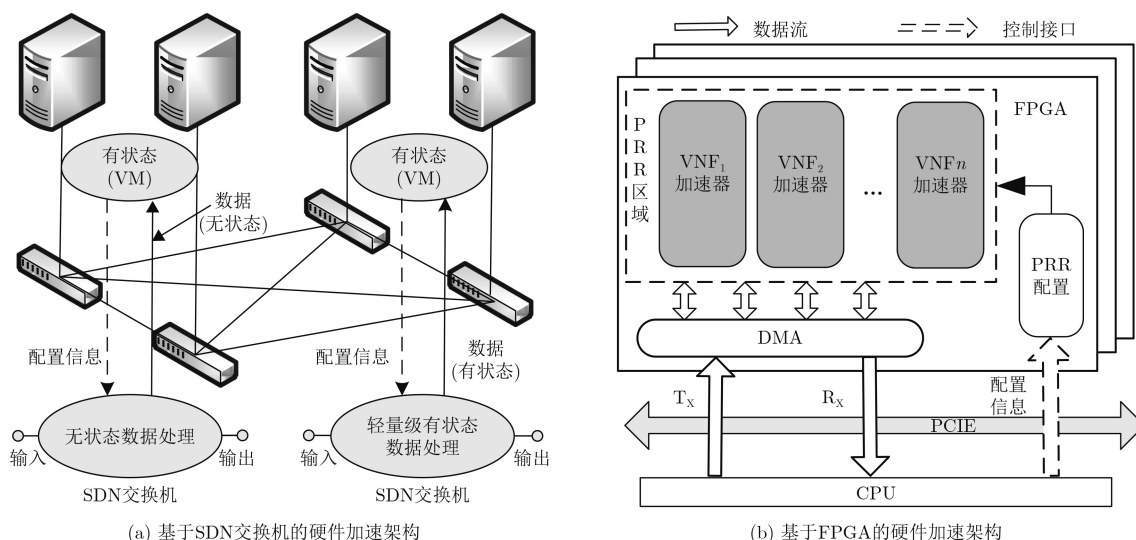


图1 硬件加速资源对网络功能的承载

力，以表项形式承载VNF的部分数据处理逻辑^[4]，但受限于交换机的硬件结构，具有加速能力的SDN交换机所能承载的VNF类型有限，单个交换机只能支持单一类型的VNF加速处理。如图1(b)所示，第2类加速资源是指加载到通用服务器内部的可编程性和高速数据处理能力能够支持各种复杂VNF实现^[8,9]单块加速卡动态地承载多个VNF。相较于交换机，FPGA的成本和功耗较大。因此，为节省开销和提高硬件加速资源效用，大部分有加速需求的VNF可以优先共享交换机内的加速资源，少数超出表项承载范围的VNF，则由服务器端的加速卡处理。

基于以上分析，支持硬件加速的VNF部署问题(VNF Placement problem with Hardware Acceleration support, VPHA)可以描述为：为满足业务流量的性能需求，提高包括加速资源在内的网络基础设施的利用率，对于用户的SFC请求，首先，在合理的链路开销约束下，尽可能共享和利用交换节点内的加速资源，承载有硬件加速需求的VNF；然后，基于这些交换节点选取合适的转发路径，将不需加速的和超出加速表承载范围的VNF按顺序部署到与转发路径相连的服务器和内部加载的加速卡上，完成对SFC的承载。

2.2 模型建立

底层网络：底层网络中的基础设施包含SDN交换机、商用服务器和硬件加速资源，其中硬件加速资源以加速表和加速卡的形式分布在交换机和服务器上。如图2所示，采用无向图 $G=(N, S, E)$ 描述底层网络。其中， N 表示服务器集合，用 C_n 表示服务器 $n(n \in N)$ 的计算存储资源； S 表示交换机集合，用 C_s 表示交换机 $s(s \in S)$ 的交换能力； E 表示交换机之间的链路集合，用 C_e 表示链路 $e(e \in E)$ 的带宽资源。 $A=\{A_n, A_s\}$ 表示硬件加速资源集合，其中 A_n 表示位于服务器内的加速卡资源， A_s 表示位于交换机内的加速表资源。 $\alpha(a, n) \in \{0, 1\}$ 和 $\alpha(a, s) \in \{0, 1\}$ 分别表示服务节点 n 和转发节点 s 是否包含加速资源 A 。 $H_{n,a}$ 表示服务节点 n 中加速资源

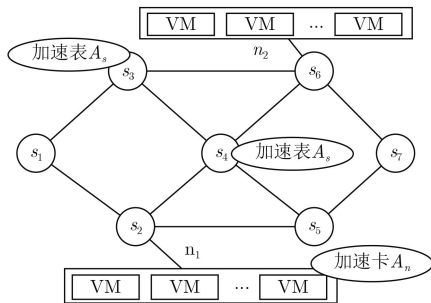


图2 VPHA底层网络示意

A_n 的数量集合， $H_{s,a}$ 表示转发节点 s 中加速资源 A_s 的数量集合。交换机和服务节点之间通过高速光纤直接相连，可以忽略带宽限制，并用表示 $\beta(n, s) \in \{0, 1\}$ 表示服务节点 n 和转发节点 s 之间的连接关系。

服务请求：定义 $D = \{d_1, d_2, \dots, d_m\}$ ($d_i \in D$, $1 \leq i \leq m$)为服务请求集合； $F = \{F_N, F_{A_n}, F_{A_s}\}$ 为VNF集合，分别表示需要商用服务器、加速卡和加速表承载的VNF类型，并用 $c_n(f)$ 、 $c_{n,a}(f)$ 和 $c_{s,a}(f)$ 的值表示 $f \in F$ 的计算存储资源或加速资源开销。服务请求 d_i 可以用4元组 $\{o_i, t_i, L_i, R_i | o_i, t_i \in S\}$ 来描述，其中 o_i 和 t_i 分别为请求 d_i 的流入、流出节点， $L_i = \{f_{i,1}, f_{i,2}, \dots, f_{i,k}\}$ ($f_{i,j} \in L_i$, $1 \leq j \leq k$)表示请求 d_i 需要按顺序处理的VNF集合， R_i 为该请求的流量速率。

VPHA的目标是当服务请求进入网络后，在基础设施资源的约束下，通过优化VNF部署策略，最大化承载服务请求的数据流量之和，即 $\max \sum R_i$ 。此时，各约束条件可以表述为：

VNF实例化约束：服务请求中VNF的实例化，要通过网络中服务节点和转发节点内的计算存储资源或硬件加速资源实现，定义 $\gamma(f_{i,j}, n) \in \{0, 1\}$ 和 $\delta(f_{i,j}, n) \in \{0, 1\}$ 表示功能 $f_{i,j}$ 与服务节点 n 中计算存储资源和加速资源的承载关系， $\varepsilon(f_{i,j}, s) \in \{0, 1\}$ 表示功能 $f_{i,j}$ 与转发节点 s 中加速资源的承载关系。对于网络承载的服务请求，要将集合 L_i 中的每个VNF实例化，对于任意的VNF约束条件如式(1)所示，表示服务链中每个VNF都必须通过底层物理资源的支持而实现，可由且仅由商用服务器、转发节点内加速表或服务器端加速卡中的一种物理基础设施进行承载。对于任意一个节点能提供的基础资源要大于等于承载VNF所需要的计算存储资源或加速资源，约束条件如式(2)–式(4)所示。此外，规定网络中任意一个服务或转发节点最多只能承载集合 L_i 个VNF，约束条件如式(5)所示。

$$\sum_{n \in N} \gamma(f_{i,j}, n) + \sum_{n \in N} \delta(f_{i,j}, n) + \sum_{s \in S} \varepsilon(f_{i,j}, s) = 1, \quad \forall n \in N, s \in S \quad (1)$$

$$\sum_{i=1}^m \sum_{j=1}^k \gamma(f_{i,j}, n) \cdot c_n(f_{i,j}) \leq C_n \quad (2)$$

$$\sum_{i=1}^m \sum_{j=1}^k \delta(f_{i,j}, n) \cdot c_{n,a}(f_{i,j}) \leq H_{n,a} \quad (3)$$

$$\sum_{i=1}^m \sum_{j=1}^k \varepsilon(f_{i,j}, s) \cdot c_{s,a}(f_{i,j}) \leq H_{s,a} \quad (4)$$

$$\sum_{j=1}^k \gamma(f_{i,j}, n) \leq 1, \sum_{j=1}^k \delta(f_{i,j}, n) \leq 1, \sum_{j=1}^k \varepsilon(f_{i,j}, s) \leq 1 \quad (5)$$

路径传输约束：因为忽略服务器与交换机之间链路的带宽限制，业务流量的传输路径约束主要考虑交换机的交换能力约束和相连的链路带宽约束。定义 $\eta(d_i, s) \in \{0, 1\}$ 表示业务流量 d_i 与交换机 s 之间的转发关系，对于任意交换机的转发能力要大于等于所转发的业务流量之和，约束条件如式(6)所示。定义 $\eta(d_i, e) \in \{0, 1\}$ 表示流量 d_i 与链路 e 之间的传输关系，对于任意一段链路，传输的业务流量之和要小于等于该链路的带宽，约束条件如式(7)所示。规定业务流量从源节点到目的节点有且只有1条传输路径，不能对同一段链路复用。传输路径中所有节点都应遵循流量守恒原则，定义 $\eta^*(d_i, s) \in \{0, 1\}$ 表示业务流量 d_i 是否流入交换机 s ， $\eta^*(s, d_i) \in \{0, 1\}$ 表示业务流量 d_i 是否流出交换机 s 。对于任意一条业务流 d_i 的源节点 o_i 和目的节点 t_i 而言，只流出或流入相应服务链承载的业务流，约束条件如式(8)和式(9)所示。而对于业务流 d_i 传输路径中的任意中间节点 s 而言，交换机的流入流量与流出流量相等，约束条件如式(10)所示。

$$\sum_{i=1}^m \eta(d_i, s) \cdot R(i) \leq C_s, \eta(d_i, s) \in \{0, 1\}, \forall s \in S \quad (6)$$

$$\sum_{i=1}^m \eta(d_i, e) \cdot R(i) \leq C_e, \eta(d_i, e) \in \{0, 1\}, \forall e \in E \quad (7)$$

$$\eta^*(o_i, d_i) - \eta^*(d_i, o_i) = 1, d_i \in D, i = 1, 2, \dots, m \quad (8)$$

$$\eta^*(d_i, t_i) - \eta^*(t_i, d_i) = 1, d_i \in D, i = 1, 2, \dots, m \quad (9)$$

$$\sum_{s \in S} \eta^*(d_i, s) - \sum_{s \in S} \eta^*(s, d_i) = 0, d_i \in D, \\ i = 1, 2, \dots, m \quad (10)$$

在SDN/NFV架构中，VNF的部署和路径的选择相互影响，VNF的部署位置限制了流量的转发路径；链路资源的开销约束又会限定VNF对部署节点的选择。因此，对于VPHA模型，要联合求解服务链中3类VNF的部署参数及路径选取参数。此外，出于成本和功耗的限制，网络中只有部分节点内含有加速资源，相比于部署到服务器，部署至加速硬件的VNF可能会带来更高的链路开销，这也为路径选择参数的求解带来额外的困难。

要求解VPHA中多个服务请求的最优化部署方案，是十分困难的。实际上，VPHA是一个NP-hard问题。已知VNF部署和路径选择的联合问题(Joint VNF placement and Path selection problem,

JVP)是NP-hard问题^[12]。相比于JVP，VPHA在对VNF部署和路径选择的联合求解中，增加了加速资源的约束条件。显然，只有在JVP存在可行解的情况下，VPHA才可能求出部署方案。若将VPHA问题中的加速资源限制放宽，则该问题可以归结为JVP问题。因此，VPHA问题也是NP-hard问题。

3 算法描述

3.1 算法设计

相较于其他场景下VNF部署问题，VPHA模型求解的关键在于，在设计加速的VNF高效部署方法同时，还要保证剩余VNF到通用服务器的最优部署，实现两者的统一。其中，两类需要加速的网络功能 F_{An} 和 F_{As} 之间存在一定的交叉关系，由于加速卡能支持较多种类VNF，部署到转发节点上的网络功能 F_{As} 同样能够由服务节点内的加速卡实现。因此，设定在部署过程中，为提高资源利用率和部署成功率，在有加速卡资源闲置的情况下，可根据需要将 F_{As} 映射到服务节点的加速卡资源上。基于以上分析，本文提出一种基于加速硬件承载特性的复用资源优先部署算法(Hardware-acceleration Multiplexing Resources Prior placement algorithm, HMRP)，伪代码如表1所示。算法内容分为3个阶段，具体描述如下。

第1阶段，确定集合 L_i 中可以对交换节点内加速表项复用的网络功能 $f_{i,j} \in F_{As}$ 的部署位置，并构造初始路径。首先更新网络物理资源，剔除网络中不能满足带宽资源约束的链路和交换机，构造新的物理网络 G' 。从源节点 o_i 按距离依次搜索网络 G' 中含有加速资源的交换节点，并用 g^* 表示搜索到转发节点与 o_i 的最近距离。当发现可以复用的转发节点 s^* 时，即 $\varepsilon(f_{i',j'}, s^*)=1$ 且 $f_{i,j} = f_{i',j'}$ ($i' < i$)，判断彼此间距离 $|o_i, s^*|$ 与链路约束距离 $j \cdot \theta g^*$ 之间的大小关系，确定节点 s^* 是否可用，其中， j 为 $f_{i,j}$ 在 L_i 中序号， θ ($1 < \theta < 2$)为链路约束因子。设定链路约束距离是为防止因过分复用加速资源而造成链路资源的开销过大。当搜索完毕 $\{s^*\} \neq \emptyset$ 时，部署到节点上的VNF按照在 L_i 中排序，从源节点 o_i 出发，按最短路径依次连接各交换节点和目的节点 t_i ，形成初始路径 P_i ；当搜索完毕 $\{s^*\} = \emptyset$ 时，则按距离升序依次选取搜索到空白交换节点，并添加到 $\{s^*\}$ 中，将 L_i 中所有 $f_{i,j} \in F_{As}$ 按顺序部署到这些转发节点上，并以最短路径连接源节点和目的节点，形成初始路径 P_i 。记 $\{s^*\}$ 中的节点个数为 S^* ，以 o_i 为起始点， $\{s^*\}$ 中各交换节点和目的节点为终点，将 P_i 分为 $(p_{i,1}, p_{i,2}, \dots, p_{i,S^*+1})$ 共 S^*+1 段子路径。

第2阶段，在 P_i 的上 S^*+1 段子路径上尽可能多

表 1 HMRP算法

输入：服务请求集合 D ，底层网络 G ，VNF参数集合 F
 输出：VNF部署方案 P_d

- (1) 根据服务请求 d_i ，更新底层网络 G'
- (2) for $x \leftarrow 1$ to $|S|$ do //搜索网络中转发节点
- (3) $s_temp \leftarrow o_i, g^* \leftarrow 0, \{s^*\} \leftarrow \Phi$; //初始化搜索节点、链路约束和复用节点集合
- (4) find $(s|\alpha(a, s) = 1)$, update g^* ; //从 o_i 出发搜索含有 A_s 的交换节点，并更新链路约束
- (5) if $\varepsilon(f_{i,j}, s) == 1 \ \&\& \ f_{i,j} = f_{i,j} \ \&\& \ H_{s,a} > c_{n,s}(f_{i,j}) \ \&\& \ |o_i, s| \leq j \cdot \theta g^*$ then //s满足复用约束
- (6) $\{s^*\} \leftarrow s$; //将复用 s 加入 $\{s^*\}$
- (7) if $\{s^*\} = \Phi$ then
- (8) reset空白交换节点 $(s|\sum \gamma(f, s) = 0)$ to $\{s^*\}$; //将搜索到的空白交换节点置入 $\{s^*\}$
- (9) deploy $(f_{i,j} \in F_{A_s}, \{s^*\})$, addroute $(P_i, \{s^*\})$; //构造初始路径
- (10) for $v \leftarrow 1$ to $S^* + 1$ do // P_i 分为 $S^* + 1$ 段子路径， L_i 分为 $S^* + 1$ 个子集
- (11) if $\{f_{i,j} \in F_{A_n} | f_{i,j} \in l_{i,v}\} \neq \Phi$ then
- (12) deploy $(f_{i,j} \in F_{A_n}, (n|\alpha(a, n) = 1, \beta(n, s) = 1, s \in p_{i,v}))$; //优先部署各子集中加速功能到加速卡
- (13) deploy $(q_0 | \max(q_j), (n, s | s \in p_{i,v}, \beta(n, s) = 1))$; //部署功能最多的方案到子路径上
- (14) if $\{f_{i,j} \in F_{A_s} | f_{i,j} \in l_{i,v}\} \neq \Phi$ then //若各子集中仍有未部署的加速表网络功能
- (15) deploy $(f_{i,j} \in F_{A_s}, (n|\alpha(a, n) = 1, \beta(n, s) = 1, s \in p_{i,v}))$; //放宽加速功能映射位置
- (16) if $\{f_{i,j} \in F_{A_s} \cup F_{A_n} | f_{i,j} \in l_{i,v}\} \neq \Phi$ then //仍有未部署的加速网络功能
- (17) reject d_i ; //拒绝该服务请求
- (18) break;
- (19) First-fit_deploy $(f_{i,j} \in L_i, n)$; //利用First-Fit部署剩余网络功能

地部署VNF。根据每段子路径中已放置的VNF序号，将 $(l_{i,1}, l_{i,2}, \dots, l_{i,S^*+1})$ $S^* + 1$ 个子集。尝试将 $S^* + 1$ 个子集的网络功能组合，按照VNF顺序映射到每个子路径中的交换节点或者与交换节点中直连的服务节点中。若子集中存在 $f_{i,j} \in F_{A_n}$ ，则优先映射该类网络功能到服务节点的加速卡后，再尝试剩余组合的放置，将放置网络功能个数最多的组合，确定为子路径的部署方案。所有的子路径最多尝试 $(S^* + 1) \cdot (2^{\Delta-1} - 1)$ 次组合放置， Δ 表示 $S^* + 1$ 个子集中VNF序号差值的最大值。当各子路径的部署方案确定后，为节省链路资源，当子集中仍存在 $f_{i,j} \in F_{A_s}$ 未部署时，可放宽约束，将其映射到各子路径中与交换机直连的服务节点内空闲加速卡上。若调整过后，子集中仍有加速需求的网络功能未部署时，拒绝该服务请求。

第3阶段，调整路径规划，放置剩余VNF。经过阶段1, 2的算法部署后，各子集中会存在少量网络功能 $f_{i,j} \in F_N$ 因路径约束还未映射到服务器内。此时，可以利用最先匹配(First-Fit)^[16]思想对剩下的VNF进行部署，从上一个已部署的节点出发，在避免路径复用和满足带宽约束的情况下，连接第1个能满足计算需求的服务器作为部署节点，并以

最短路径返回下一个已部署的服务节点，完成最终的服务链映射。

3.2 算法的时间复杂度分析

HMRP部署算法第1阶段 $f_{i,j} \in F_{A_s}$ 的映射部分，需对含有加速资源的交换节点进行遍历，最多要搜索 $|S|$ 个交换节点，节点间最短距离用基于Fibonacci堆的Dijkstra算法实现，则该阶段的算法复杂度为 $O(l(|E| + |S| \lg |S|))$ ；第2阶段是对网络功能组合的尝试，网络功能组合中会包含全部类型的VNF，当 S^* 为0时尝试次数最多，组合方案会达到 2^{l-1} 次，再依次尝试映射到路径中，该阶段的时间复杂度为 $O(2^l \cdot (|S| + |N|))$ ；第3阶段对剩余 $f_{i,j} \in F_N$ 部署位置的搜索也用Dijkstra算法实现，最多需搜索 $|N|$ 个服务节点，则该阶段算法复杂度为 $O(l(|E| + |N| + |S| \lg |S|))$ 。因此，在省去低阶项后，HMRP算法的时间复杂度为 $O(2^l \cdot (|S| + |N|) + l(|E| + |S| \lg |S|))$ 。

4 实验结果及评估

4.1 算法仿真

仿真测试采用如图3所示的底层网络，其中13个交换节点，6个直连的服务节点。各服务节点中计算存储资源服从[400, 600]随机生成，交换节

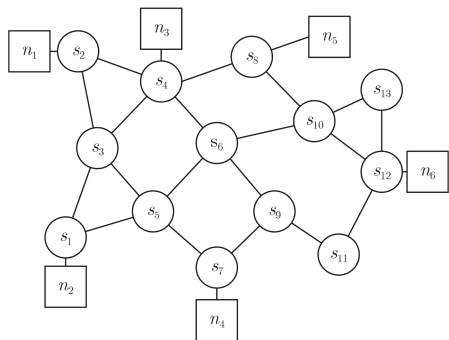


图3 算法仿真网络拓扑

点间链路带宽服从[300 B/Mbps, 500 B/Mbps]。交换节点的交换能力按相连的链路资源和服务节点计算资源的一定比例生成。在此基础上，加速表和加速卡资源总量分别设定为400和135，根据服务节点的计算资源数量和交换节点的交换能力合理分布于网络中交换节点和服务节点内。

文献[17]中指出服务链中常用的VNF种类有11个，若这些VNF需要被加速时，设定其中8个可被加速表承载，3个只能被加速卡承载，需要的计算存储资源和加速资源如表2所示。设每个服务请求中VNF个数服从[3, 5]随机分布，从表2中随机组合，再从中随机选取35%~45%的VNF(向下取整)需要加速。每个服务请求的流量速率服从[3 Mbps, 5 Mbps]。此外，设定链路约束因子 θ 为1.5。仿真测试平台用Matlab搭建，计算机处理器为Inter® Core™ i7-3770, 3.50 GHz, 16 GB RAM。

表2 VNF资源参数

VNF类型	1	2	3	4	5	6	7	8	9	10	11
计算存储资源开销 c_n	3	4	6	5	5	2	7	6	4	7	5
加速表资源开销 $c_{s,a}$	2	3	-	5	2	3	-	4	2	-	2
加速卡资源开销 $c_{n,a}$	1	1	33	2	1	2	2	2	1	3	1

由于目前业内还缺乏对硬件加速的VNF部署算法研究，因此本节选择服务链映射研究方向内较为常用的部署算法做为评估对象，分别是随机策略(random)、贪婪策略(greedy)和禁忌搜索(Tabu Search, TS)^[18]3类方法。随机策略在路径约束下，随机选取网络中满足资源约束的服务节点和交换节点；贪婪策略根据目标设定的不同，可分为贪婪快速映射(Greedy Fast Processing, GFP)算法和贪婪均衡映射(Greedy Least Loaded, GLL)算法，是在链路带宽和交换机能力约束下，根据VNF顺序，依次对服务节点和交换节点进行搜索，分别选取距离最近和剩余资源最大的节点进行部署；禁忌搜索通过对全局优化目标的设定，多次搜索实现部署节点资源的负载均衡。在生成不同数目的服务请求下，用5种算法完成50次部署并求取平均值，得到各算法的平均性能指标如图4所示。

从图4(a)中可见，当服务请求较少时，5种部署算法的部署成功率较为接近，随着服务请求数量的增多，HMRP算法的优势逐渐突显，通过综合考虑加速硬件的承载特性，改变了加速网络功能的部署优先级，而且根据资源分配情况，灵活调整功能约束。因此，HMRP算法的部署成功率最高，可容纳更多的服务请求。其次是TS的部署成功率，算法通过对设定目标的反复迭代计算，尝试更多可能性，从而得到更加合理的部署方案。基于贪婪思想的GFP和GLL算法在随着服务请求数目的增长，出现了较为快速的下降，是因为对部署节点的选择陷入了局部相对最优，在资源受限时，这种影响会被显著放大。其中，相较于GFP，GLL以节点资源容量为导向，实现了更高的部署成功率。随机策略的部署成功率最低，甚至当服务链数目接近100时，网络已经无法容纳更多的服务请求。图4(b)显示各部署算法下，网络中加速硬件承载的数据流量。与

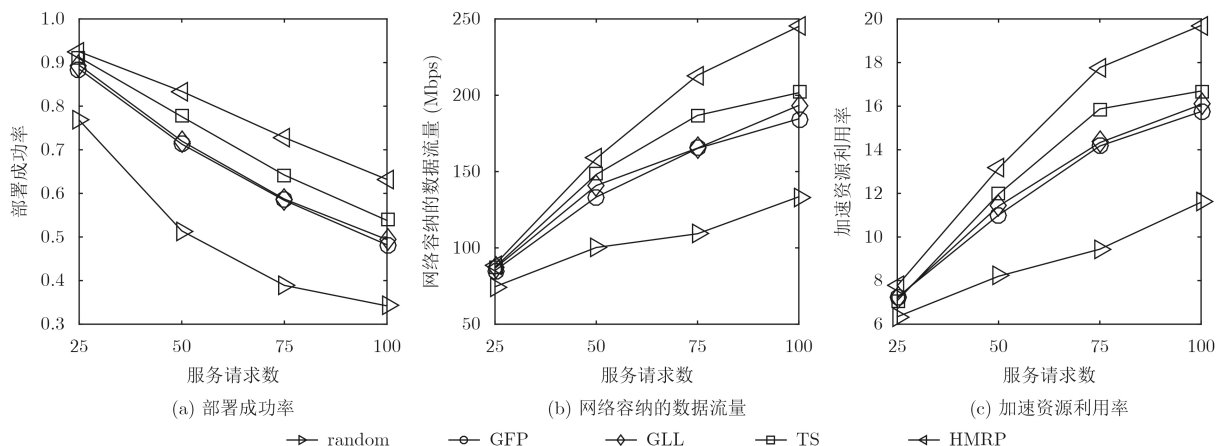


图4 算法仿真性能对比

部署成功率相一致，HMRP算法中加速资源所能承载的数据流量最大，而且随着服务请求的增多呈线性增加。图4(c)显示了部署算法在不同服务链请求数目下对加速资源的利用率 ω 。加速资源利用率 ω 反映了加速资源的单位负载，表示为

$$\phi = r_n \frac{\sum_{i=1}^m \left[R(i) \cdot \sum_{k=1}^L \delta(f_{i,j}, n) \right]}{\sum_{i=1}^m \sum_{j=1}^k c_{n,a} \cdot \delta(f_{i,j}, n)} + r_s \frac{\sum_{i=1}^m \left[R(i) \cdot \sum_{j=1}^k \varepsilon(f_{i,j}, s) \right]}{\sum_{i=1}^m \sum_{j=1}^k c_{s,a} \cdot \varepsilon(f_{i,j}, s)} \quad (11)$$

其中， $\sum_{i=1}^m \left[R(i) \cdot \sum_{k=1}^L \delta(f_{i,j}, n) \right]$ 和 $\sum_{i=1}^m \left[R(i) \cdot \sum_{j=1}^k \varepsilon(f_{i,k}, s) \right]$ 分别表示加速表和加速卡资源承载的数据流量， $\sum_{i=1}^m \sum_{j=1}^k c_{n,a} \cdot \delta(f_{i,j}, n)$ 和 $\sum_{i=1}^m \sum_{j=1}^k c_{s,a} \cdot \varepsilon(f_{i,j}, s)$ 是加速表和加速卡承载VNF所需的资源开销， r_n 和 r_s 是调节系数，跟资源总量将两类加速资源进行归一，仿真中 $r_n=0.747$ ， $r_s=0.253$ 。由图4(c)可知，利用率 ω 随着业务流数目的增多而增大，受加速资源容量的限制，增加幅度不断减少，接近每种部署算法下的加速资源承载上限，而且HMRP算法的利用率一直高于其他对比算法。对于算法复杂度而言，随机策略是通过依次选取满足约束条件的节点和链路进行映射的，对于不满足带宽约束条件的链路，可通过更新网络进行剔除，而转发节点和服务节点的选取需要通过遍历完成，算法复杂度为 $O(E+(S+N)^4)$ 。基于贪婪思想的GFP算法和GLL算法均以邻接表的形式进行广度优先遍历，并根据VNF对物理资源类型的需求，分别选取距离最近和资源容量最多的转发节点和服务节点，算法复杂度为 $O(l(S+N+E))$ 。TS算法的时间复杂度与设定的最大迭代次数 M_iter 和禁忌表的长度 TL 相关，算法复杂度为 $O(M_iter \times (l^2 + l + TL))$ 。分析发现，由于HMRP算法的时间复杂度中存在指数项，导致其复杂度高于贪婪算法和TS算法。但是，HMRP算法复杂度中指数项中的指数 l 为服务请求中VNF个数，最大值为5，且底数为定值，所以算法的运行时间在实际应用可接受的

范围内，而且输出结果的部署成功率和容纳的数据流量等性能指标显著优于对比算法。

4.2 模型验证

本文首次将加速卡和加速表结合，同时引入SDN/NFV加速框架中，提出相应的VPHA部署模型，为表明模型的可行性，建立了如图5所示的模型验证系统。其中，采用NetFPGA-10G板卡^[19]实现支持加速的OpenFlow交换机，以及作为服务器内的加速板卡，未加速的VNF以软件形式实现在NFV服务器(Intel Xeon E5-2420 CPU 1.9 GHz, 8 GB DDR3 RAM)内。

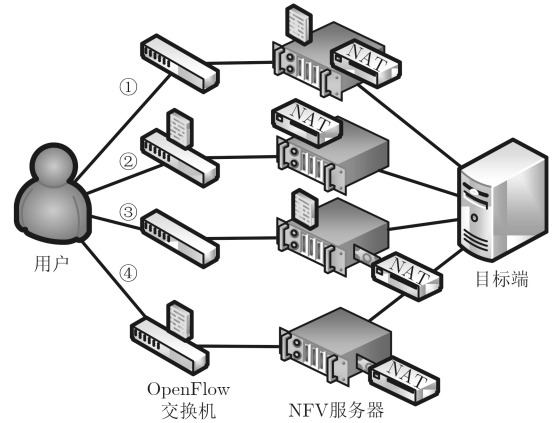


图5 模型验证示意

用户通过发送数据流，经过{Firewall, NAT}两个网络功能达到目标节点。根据文献[4,20]的实现方案，分别利用交换机内加速表和服务器内加速卡实现数据处理加速，其中交换机中的加速表最大支持容纳128条加速规则，服务器中加速卡最大可容纳64条自定义加速规则，并支持32 bit的正则匹配^[21]。实验分别测试在纯软件、单加速表、单加速卡和两种加速资源共用的4种模式下，不同数据报长度下数据流到目标节点的传输时延。其中，纯软件模式下两个VNF都由软件实现；单加速表模式下Firewall由交换机实现硬件加速，NAT由软件实现；单加速卡模式下NAT由服务器中加速卡实现加速，Firewall在服务器上软件实现；联合加速模式下将加速表和加速卡同时实现Firewall和NAT的硬件加速。实验结果如图6所示，加速卡和加速表结合的模式下，用户数据流的传输时延最低，证明了本文所提的VPHA部署模型具有可行性。

5 结束语

本文研究了在SDN/NFV网络中引入加速硬件后的虚拟网络功能部署问题，首先将加速网络功能部署统一到服务链映射的数学模型中，并证明了该问题的复杂性。然后为实现加速硬件对网络功能的

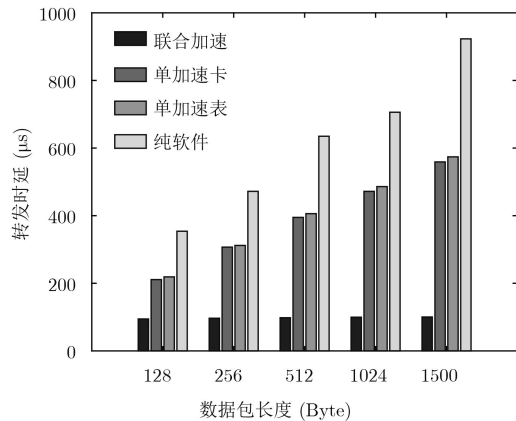


图6 数据包转发性能对比

最优承载, 在多种硬件资源约束下, 提出一种支持硬件加速的启发式网络功能部署策略, 对问题进行了求解。仿真测试和实验模型证明, 所提算法能有效提升服务链的部署成功率, 最大化网络中能够承载的数据流量, 对未来网络中有硬件加速需求的服务链映射机制具有一定实用性。后续将针对硬件加速的服务链缩放问题进行研究, 以满足数据业务的动态性要求。

参考文献

- [1] HONG Chiyao, CAESAR M, and GODFREY P B. Finishing flows quickly with preemptive scheduling[C]. ACM SIGCOMM 2012 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, Helsinki, Finland, 2012: 127–138. doi: [10.1145/2342356.2342389](https://doi.org/10.1145/2342356.2342389).
- [2] BOURAS C, KOLLIA A, and PAPAZOIS A. SDN & NFV in 5G: advancements and challenges[C]. The 20th Conference on Innovations in Clouds, Internet and Networks, Paris, France, 2017: 107–111. doi: [10.1109/ICIN.2017.7899398](https://doi.org/10.1109/ICIN.2017.7899398).
- [3] HAN Bo, GOPALAKRISHNAN V, JI Lusheng, et al. Network function virtualization: challenges and opportunities for innovations[J]. *IEEE Communications Magazine*, 2015, 53(2): 90–97. doi: [10.1109/MCOM.2015.7045396](https://doi.org/10.1109/MCOM.2015.7045396).
- [4] MATIAS J, GARAY J, TOLEDO N, et al. Toward an SDN-enabled NFV architecture[J]. *IEEE Communications Magazine*, 2015, 53(4): 187–193. doi: [10.1109/MCOM.2015.7081093](https://doi.org/10.1109/MCOM.2015.7081093).
- [5] BI Jun, ZHU Shuyong, SUN Chen, et al. Supporting virtualized network functions with stateful data plane abstraction[J]. *IEEE Network*, 2016, 30(3): 40–45. doi: [10.1109/MNET.2016.7474342](https://doi.org/10.1109/MNET.2016.7474342).
- [6] YI Xiaodong, DUAN Jingpu, and WU Chuan. GPUNFV: A GPU-accelerated NFV system[C]. The 1st Asia-Pacific Workshop on Networking, Hong Kong, China, 2017: 85–91. doi: [10.1145/3106989.3106990](https://doi.org/10.1145/3106989.3106990).
- [7] SONG Haoyu. Protocol-oblivious forwarding: unleash the power of SDN through a future-proof forwarding plane[C]. The 2nd ACM SIGCOMM Workshop on Hot Topics in Software Defined Networking, Hong Kong, China, 2013: 127–132. doi: [10.1145/2491185.2491190](https://doi.org/10.1145/2491185.2491190).
- [8] LI Bojie, TAN Kun, LUO Layong, et al. ClickNP: Highly flexible and high performance network processing with reconfigurable hardware[C]. 2016 ACM SIGCOMM Conference, Florianopolis, Brazil, 2016: 1–14. doi: [10.1145/2934872.2934897](https://doi.org/10.1145/2934872.2934897).
- [9] LI Xiaoyao, WANG Xiuxiu, LIU Fangming, et al. DHL: Enabling flexible software network functions with FPGA acceleration[C]. The IEEE 38th International Conference on Distributed Computing Systems, Vienna, Austria, 2018: 1–11. doi: [10.1109/ICDCS.2018.00011](https://doi.org/10.1109/ICDCS.2018.00011).
- [10] SUN Jian, ZHU Guangyang, SUN Gang, et al. A reliability-aware approach for resource efficient virtual network function deployment[J]. *IEEE Access*, 2018, 6: 18238–18250. doi: [10.1109/ACCESS.2018.2815614](https://doi.org/10.1109/ACCESS.2018.2815614).
- [11] KUO Tungwei, LIOU Bangheng, LIN K C J, et al. Deploying chains of virtual network functions: on the relation between link and server usage[C]. The 35th Annual IEEE International Conference on Computer Communications, San Francisco, USA, 2016: 1–9. doi: [10.1109/INFOCOM.2016.7524565](https://doi.org/10.1109/INFOCOM.2016.7524565).
- [12] 陈卓, 冯钢, 刘蓓, 等. 运营商网络中面向资源碎片优化的网络服务链构建策略[J]. *电子与信息学报*, 2018, 40(4): 763–769. doi: [10.11999/JEIT170641](https://doi.org/10.11999/JEIT170641).
- [13] CHEN Zhuo, FENG Gang, LIU Bei, et al. Construction policy of network service chain oriented to resource fragmentation optimization in operator network[J]. *Journal of Electronics & Information Technology*, 2018, 40(4): 763–769. doi: [10.11999/JEIT170641](https://doi.org/10.11999/JEIT170641).
- [14] ZENG Chaobing, LIU Fangming, CHEN Shutong, et al. Demystifying the performance interference of co-located virtual network functions[C]. 2018 IEEE Conference on Computer Communications, Honolulu, USA, 2018: 765–773. doi: [10.1109/INFOCOM.2018.8486246](https://doi.org/10.1109/INFOCOM.2018.8486246).
- [15] 汤红波, 袁泉, 卢干强, 等. 一种支持节点分割的vEPC虚拟网络功能部署模型[J]. *电子与信息学报*, 2017, 39(3): 546–553. doi: [10.11999/JEIT160507](https://doi.org/10.11999/JEIT160507).
- [16] TANG Hongbo, YUAN Quan, LU Ganqiang, et al. A model for virtualized network function deployment based on node-splitting in vEPC[J]. *Journal of Electronics & Information Technology*, 2017, 39(3): 546–553. doi: [10.11999/JEIT160507](https://doi.org/10.11999/JEIT160507).
- [17] FENG Hao, LLORCA J, TULINO A M, et al.

- Approximation algorithms for the NFV service distribution problem[C]. 2017 IEEE Conference on Computer Communications, Atlanta, USA, 2017: 1–9. doi: [10.1109/INFOCOM.2017.8057039](https://doi.org/10.1109/INFOCOM.2017.8057039).
- [16] ERAMO V, AMMAR M, and LAVACCA F G. Migration energy aware reconfigurations of virtual network function instances in NFV architectures[J]. *IEEE Access*, 2017, 5: 4927–4938. doi: [10.1109/ACCESS.2017.2685437](https://doi.org/10.1109/ACCESS.2017.2685437).
- [17] SUN Chen, BI Jun, ZHENG Zhilong, *et al.* NFP: Enabling network function parallelism in NFV[C]. Conference of the ACM Special Interest Group on Data Communication, Los Angeles, USA, 2017: 43–56. doi: [10.1145/3098822.3098826](https://doi.org/10.1145/3098822.3098826).
- [18] MIJUMBI R, SERRAT J, GORRICO J L, *et al.* Design and evaluation of algorithms for mapping and scheduling of virtual network functions[C]. The 2015 1st IEEE Conference on Network Softwarization, London, UK, 2015: 1–9. doi: [10.13140/RG.2.1.4088.6885](https://doi.org/10.13140/RG.2.1.4088.6885).
- [19] ZILBERMAN N, AUDZEVICH Y, KALOGERIDOU G, *et al.* NetFPGA: rapid prototyping of networking devices in open source[C]. 2015 ACM Conference on Special Interest Group on Data Communication, London, UK, 2015: 363–364. doi: [10.1145/2785956.2790029](https://doi.org/10.1145/2785956.2790029).
- [20] GE Xiongzi, LIU Yi, DU D H C, *et al.* OpenANFV: Accelerating network function virtualization with a consolidated framework in openstack[C]. 2014 ACM Conference on SIGCOMM, Chicago, USA, 2014: 353–354. doi: [10.1145/2619239.2631426](https://doi.org/10.1145/2619239.2631426).
- [21] 段通, 兰巨龙, 胡宇翔, 等. 面向SDN/NFV架构的VNF硬件加速资源编排机制[J]. *通信学报*, 2018, 39(6): 98–108. doi: [10.11959/j.issn.1000-436x.2018108](https://doi.org/10.11959/j.issn.1000-436x.2018108).
- DUAN Tong, LAN Julong, HU Yuxiang, *et al.* Orchestration mechanism for VNF hardware acceleration resources in SDN/NFV architecture[J]. *Journal on Communications*, 2018, 39(6): 98–108. doi: [10.11959/j.issn.1000-436x.2018108](https://doi.org/10.11959/j.issn.1000-436x.2018108).

胡宇翔：男，1982年生，副教授，研究方向为未来网络关键技术和网络智能化。

范宏伟：男，1994年生，硕士生，研究方向为网络功能虚拟化和硬件加速。

兰巨龙：男，1962年生，教授，博士生导师，研究方向为未来通信网络关键理论与技术。

段通：男，1992年生，博士生，研究方向为网络功能虚拟化和可编程硬件。