

基于条件反亲和度的网络功能虚拟化资源分配算法

林青¹, 戴慧璐², 焦纯¹

1. 西安培华学院中兴电信学院, 陕西 西安 710125; 2. 西安交通大学电信学院, 陕西 西安 710049

基金项目: 陕西省高等教育教学改革研究资助项目(15BY122)

通信作者: 林青, 530610232@qq.com 收稿/录用/修回: 2016-05-25/2016-08-18/2017-01-12

摘要

为了提高网络功能虚拟化(network function virtualization, NFV)资源分配的成功率, 针对运营商网络的高可靠性要求, 提出基于条件反亲和度的虚拟网络功能映射(virtual network function embedding, VNFE)算法, 实现NFV资源分配. 首先定义条件反亲和度, 放宽NFV资源分配技术反亲和性设计的约束条件; 其次, 建立双备份VNFE算法模型, 描述运营商网络NFV资源分配技术问题; 最后, 使用粒子群优化算法获得VNFE的解, 解决NFV资源分配问题. 仿真结果表明, 所提算法能够满足用户对网络可靠性的要求, 具有较高的资源利用, 并将NFV资源分配的成功率提高约46%.

关键词

条件反亲和度
网络功能虚拟化
资源分配
虚拟网络功能映射
中图分类号: TN915
文献标识码: A

Resource Allocation Algorithm for Network Function Virtualization Based on the Conditional Anti-affinity Degree

LIN Qing¹, DAI Huijun², JIAO Chun¹

1. College of ZTE Telecommunication, Xi'an Institute of PeiHua, Xi'an 710125, China;

2. School of Electronic Information, Xi'an Jiaotong University, Xi'an 710049, China

Abstract

To improve the success rate of resource allocation for network function virtualization (NFV), we propose a virtual network function embedding (VNFE) algorithm based on the conditional anti-affinity degree to accomplish resource allocation for the NFV with high availability requirements of carrier networks. First, it defines the conditional anti-affinity degree to relax the constraints of anti-affinity for designing resource allocation for the NFV. Then, we construct a backup VNFE model to describe the resource allocation problem for the NFV of the carrier network. Finally, we use a particle swarm optimization algorithm to obtain the solution of VNFE to resolve the resource allocation problem for the NFV. Simulation results show that the proposed algorithm satisfies the user's availability request for the network, has a high resource utilization, and improves the success rate of resource allocation for the NFV by approximately 46%.

Keywords

conditional anti-affinity degree;
network function virtualization;
resource allocation;
virtual network function embedding

1 引言

网络功能虚拟化(network function virtualization, NFV)实现了网络硬件与软件的分离, 通过将网络控制与管理功能部署在通用服务器中隔离的虚拟机上, 弱化运营商网络对专用网络硬件的依赖, 降低网络建设和维护的成本, 成为运营商网络发展和演进的关键技术^[1-2]. NFV资源分配技术通过设计服务功能链(service function chain, SFC)将虚拟网络功能(virtual network function, VNF)按顺序连接起来^[3], 再使用虚拟网络功能映射(virtual network function

embedding, VNFE)算法将SFC中包含的VNF映射在服务器上实例化SFC^[4-5], 组成网络控制平面, 作为一种服务提供给用户^[6]. 其中, VNFE能够确定将SFC中的VNF承载在哪些服务器上, 成为VNF资源分配技术研究的重点.

针对NFV和NFV资源分配技术的研究, 文[7]通过在光网络中设计SFC, 实现光网络的网络功能虚拟化. 文[8-10]将NFV技术应用到数据中心网络, 实现运营商网络的网络功能虚拟化. 文[11]将VNF定义为路由计算功能并在网络中部署VNF, 实现路由控制机制. 文[12]将VNF定义为网络控制器并在网络中部署VNF, 实例化网络

控制器. 考虑到 NFV 技术由运营商提出并主要在运营商网络中应用, 文[13]提出在 ETSI (European Telecommunications Standards Institute) 制定的 NFV 标准架构中加入安全性考虑; 文[14]提出具有鲁棒性的资源分配方法; 文[15]指出当在数据中心网络中应用 NFV 技术时, 需要保证其可靠性需求.

传统运营商网络强调 99.999% 的可靠性, 对网络控制与管理设备采用双备份部署, NFV 作为运营商网络演进的关键技术, 必须具有运营商网络的高可靠性特征^[16]. 因此, 在将 NFV 技术应用在网络中时也需要采用双备份方案, 即复制 SFC 得到 pSFC (备份 SFC), 再将 SFC 和 pSFC 同时映射在服务器上. 由于每个服务器都能隔离出多个虚拟机, 使得服务器能够承载多个 VNF, 一旦服务器发生故障, 其上承载的 VNF 同时失效. 如果将 SFC 和其对应的 pSFC 中包括的 VNF 映射在相同的服务器上, 则失去双备份部署的效果. 此时, 需要使用反亲和性设计, 将 SFC 和其对应的 pSFC 中的 VNF 映射在不同的服务器上, 避免服务器发生故障时, SFC 和 pSFC 同时失效.

NFV 技术应用到运营商网络时, 对应不同的业务会建立不同的 SFC. 与此同时, 网络中的服务器个数有限, 使得应用反亲和性设计时不能将 SFC 及其对应的 pSFC 都映射在服务器上, 造成 VNFE 算法失效. 针对此问题, 本文定义条件反亲和度, 描述服务器发生故障的概率, 如果该服务器不发生故障的概率大于网络对 SFC 及其对应 pSFC 的可靠度要求, 则可以将 SFC 和其对应的 pSFC 中的 VNF 映射在此服务器上, 放宽反亲和性设计时的约束条件, 提高 VNFE 算法的成功率. 考虑到 VNFE 算法是针对资源分配的优化算法, 而粒子群优化算法的参数较少, 其离散化方法较为成熟, 并且可以方便地推广到二值优化问题的求解中, 在优化问题解空间的多个域内同时展开搜索, 快速收敛到最优解^[17-20]. 因此, 本文在条件反亲和度定义的基础上, 提出基于条件反亲和度的 VNFE 算法, 先对双备份部署 VNFE 算法建模, 再使用粒子群优化算法求解双备份 VNFE 的解, 实现 NFV 资源分配. 本文所提双备份 VNFE 算法模型为首次提出, 论文主要贡献为:

- (1) 定义了条件反亲和度, 放宽了反亲和性设计的约束条件, 提高了 VNFE 算法的成功率;
- (2) 建立双备份 VNFE 模型, 为在运营商网络中完成 NFV 资源分配技术提供依据;
- (3) 提出基于条件反亲和度的 VNFE 算法, 使用粒子群优化算法获得 VNFE 的解, 实现 NFV 资源分配.

2 双备份 VNFE 算法建模

双备份 VNFE 是指在运营商网络中应用 NFV 技术建立包含 VNF 的 SFC, 复制 SFC 得到 pSFC, 将 SFC 及其对应的 pSFC 映射在服务器上, 如图 1 所示.

使用反亲和性设计完成 VNFE, 如图 1(a) 所示. 将 SFC 顺序连接用户归属服务器 (home subscriber server, HSS)、多媒体资源功能 (multi-media resource function, MRF)、呼叫会话控制功能 (call session control function, CSCF)、分

组/信令网关 (Packet/Signaling GateWay, P/S GW) 四个 VNF, 并将这 4 个 VNF 分别映射服务器 4、服务器 6、服务器 5、服务器 1 上. 此时, 只有服务器 2 和服务器 3 没有承载 SFC 中包含的 VNF, 但是服务器 2 和服务器 3 不能同时承载 pSFC 中的 4 个 VNF, 导致 VNFE 失效. 考虑条件反亲和度, 放宽反亲和性设计的约束条件, 若服务器 1 和服务器 4 不会同时发生故障的概率大于网络可靠性要求, 则可以将 SFC 和 pSFC 中包含的部分 VNF 映射在其上, 如图 1(b), 将 pSFC 顺序连接的 bHSS、bMRF、bCSCF、bP/S GW 分别映射服务器 4、服务器 3、服务器 2、服务器 1 上, 实现双备份 VNFE, 完成 VNF 资源分配. 表 1 给出 VNFE 中相关符号的定义.

双备份 VNFE 在反亲和性约束或条件反亲和性约束条件下, 同时将 SFC 及其对应的 pSFC 中的 VNF 映射到服务器上, SFC 映射失败或 pSFC 映射失败都会导致 VNFE 失效, 具体为:

SFC 映射定义为 $E(\text{SFC})$, 记为: $\forall i \in S(N_s), S_u \downarrow i$, 表示 SFC 中第 u 个 VNF 映射在第 i 个服务器上;

pSFC 映射定义为 $E(\text{pSFC})$, 记为: $\forall j \in S(N_s) \forall i \in S(N_s), S_v^p \downarrow j$, 表示 pSFC 中第 v 个 VNF 映射在第 j 个服务器上;

双备份 SFC 映射定义为: $E = E(\text{SFC}) \cap E(\text{pSFC})$.

双备份 VNFE 需要将 SFC 和 pSFC 中的 VNF 映射在服务器上, 由于每一个 VNF 都要完成网络控制功能, 对服务器的计算能力有一定的要求. 因此, 承载 VNF 的服务器的剩余计算资源要大于 VNF 请求的计算资源, 并且还要满足可靠性等约束条件:

- (1) 资源约束. 服务器上的剩余计算资源需要满足 SFC 和 pSFC 中 VNF 请求的计算资源:

$$\forall u \in \{S(N_v) \cup S_p(N_v)\}, i \in S(N_s), \quad x_i^u \cdot c(u) \leq c(i) \quad (1)$$

- (2) 可靠性约束. 完成 NFV 资源分配后, SFC 和 pSFC 发生故障的条件概率满足网络的可靠度需求 R_r :

$$f(v) = 1 - \prod_{u, u'=1}^{N(N_v)} (1 - f(S_u^p/S_u)) < 1 - R_r \quad (2)$$

其中, $f(S_u^p/S_u)$ 表示 SFC 中第 u 个 VNF 发生故障时, pSFC 中第 u' 个 VNF 也发生故障的条件概率. 如果 S_u 和 $S_{u'}$ 映射在同一个服务器上, 则 $f(S_u^p/S_u) = 1$, 表示 S_u 和 $S_{u'}$ 同时失效; 如果 S_u 和 $S_{u'}$ 没有映射在同一个服务器上, 则 $f(S_u^p/S_u) = f(S_{u'}^p) \cdot f(S_u)$ 为两者同时失效的概率. $f(S_u) = x_i^{S_u} \cdot f(i)$, $f(S_{u'}^p) = x_i^{S_{u'}^p} \cdot f(i)$, 表示 SFC 或 pSFC 中 VNF 的故障概率等于承载该 VNF 的服务器的故障概率.

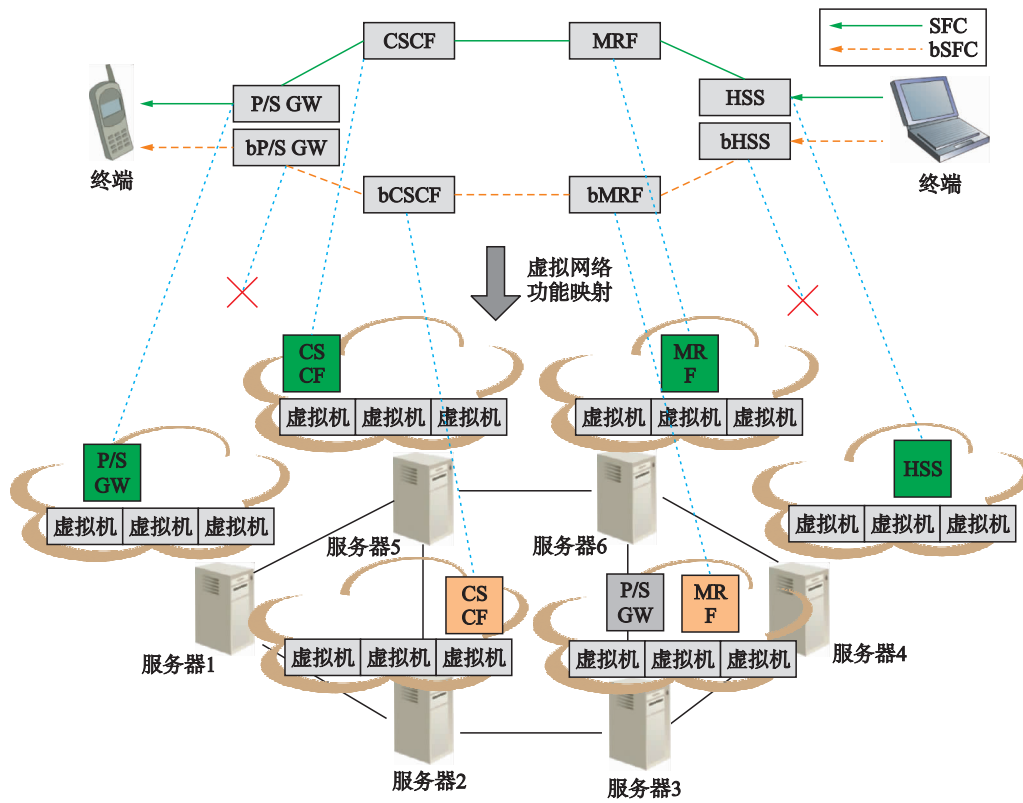
二值约束(1): SFC 和 pSFC 中的 VNF 只能映射在服务器上:

$$\forall i \in S(N_s), u \in \{S(N_v) \cup S_p(N_v)\}, x_i^u \in \{0, 1\} \quad (3)$$

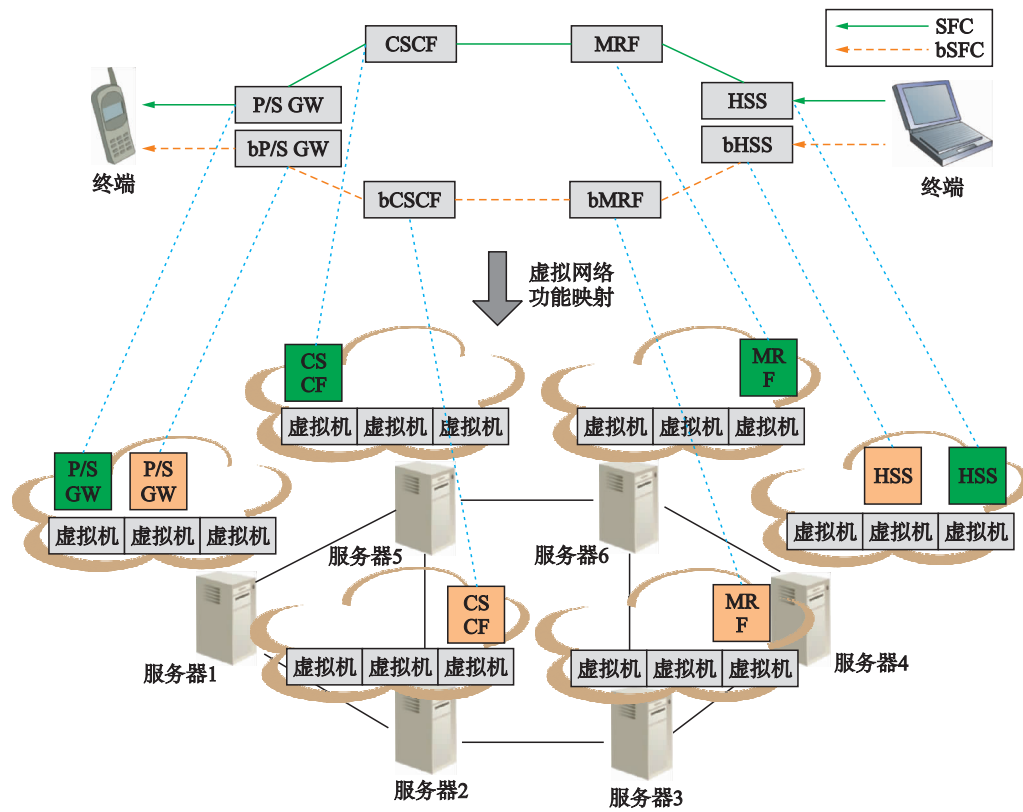
二值约束(2): 同一个 SFC 中的 VNF 不能映射在同一个服务器上:

$$\forall i \in S(N_s), \sum_{u \in S(N_v)} x_i^u \leq 1 \quad (4)$$

二值约束(3): 同一个 pSFC 中的 VNF 不能映射在同一个服务器上:



(a) 反亲和性设计



(b) 条件反亲和性设计

图 1 双备份 VNFE 算法

Fig.1 The double backups VNFE algorithm

表1 虚拟网络功能映射相关符号定义

Tab.1 The notations of the virtual network function embedding

符号	意义
$S(N_v)$	SFC 中所有 VNF 的集合
$S_p(N_v)$	pSFC 中所有 VNF 的集合
$S(N_s)$	服务器集合
S_u	SFC 中第 u 个 VNF
S_v^p	pSFC 中第 v 个 VNF 请求的计算资源
$c(S_u)$	SFC 中第 u 个 VNF 请求的计算资源
$c(S_v^p)$	pSFC 中第 v 个 VNF 请求的计算资源
$c(i)$	第 i 个服务器上的剩余计算资源
$N(N_v)$	SFC 中 VNF 的总数
$N(N_s)$	服务器总数
R_r	网络可靠度需求
$f(i)$	任意服务器的故障概率
$f(v)$	SFC 和 pSFC 发生故障的条件概率
x_i^u	如果第 u 个 VNF 请求映射在第 i 个服务器上, 则 x_i^u 为 1, 否则为 0

$$\forall i \in S(N_s), \sum_{u \in S_p(N_v)} x_i^u \leq 1 \quad (5)$$

二值约束(4): SFC 和 pSFC 中的 VNF 可以映射在同一个服务器上:

$$\forall i \in S(N_s), \sum_{u \in \{S_p(N_v) \cap S(N_v)\}} x_i^u \geq 0 \quad (6)$$

二值约束(5): 一个 VNF 只能映射在一个通用服务器上:

$$\forall u \in \{S(N_v) \cup S_p(N_v)\}, \sum_{i \in S(N_s)} x_i^u = 1 \quad (7)$$

3 基于条件反亲和度的 VNFE 算法

首先根据双备份 VNFE 算法模型对粒子群算法重新定义, 实现粒子群算法的二值化; 再设计粒子群算法适应度函数; 最后应用二值化后的粒子群算法获得 VNFE 算法的解, 完成 NFV 资源分配。

3.1 基于 VNFE 的粒子群二值化

粒子群算法中任意一个粒子 X_i 表示 VNFE 的一个可行解, 由于 SFC 按顺序连接 VNF, 并且 pSFC 为复制 SFC 所得, 因此对 SFC 及其对应的 pSFC 中的 VNF 排序, 得到 SFC 和 pSFC 集合 $S(N_v)$ 和 $S_p(N_v)$ 及任意集合中的元素 S_u 或 S_v^p 。

VNFE 要同时完成 SFC 中 VNF 及其对应的 pSFC 中 VNF 的映射, 因此将 X_i 的维数定义为 $2N(N_s)$ 。当 $1 \leq j \leq N(N_s)$ 时, 给定任意的 i , 如果 $x_{i,j}$ 为 X_i 中第 k 个 1, 则表示 SFC 中第 k 个 VNF 被映射在服务器 j 上, 对应表 1 的符号定义, $x_{i,j}^{S_k} = 1$ 。当 $(N(N_s) + 1) \leq j \leq 2N(N_s)$ 时, 给定任意的 i , 如果 $x_{i,j}$ 为 X_i 中第 k' 个 1, 表示 pSFC 中第 k' 个 VNF 被映射在服务器 j' 上, 对应表 1 的符号定义, $x_{i,j}^{S_{k'}^{p-N(N_s)}} = 1$ 。

粒子在进化过程中根据适应度函数修改粒子速度和位置, 逐渐收敛到全局最优解, 所得全局最优解即为 VNFE 的解。其中, p_{ij}^t 为任意一个粒子进化过程中的历史最优位置, p_{ij}^t 为所有粒子进化过程中的历史最优位置, v_{ij}^{t+1} 为粒子的速度, 粒子进化速度和位置更新函数为

$$v_{ij}^{t+1} = \omega v_{ij}^t + c_1 r_1 (p_{ij}^t - x_{ij}^t) + c_2 r_2 (p_{ij}^t - x_{ij}^t) \quad (8)$$

$$x_{ij}^{t+1} = x_{ij}^t + v_{ij}^{t+1} \quad (9)$$

$$\omega = \omega_{\max} - \frac{\omega_{\max} - \omega_{\min}}{t_{\max}} \cdot t \quad (10)$$

其中, ω 为惯性参数, 取实数; c_1 、 c_2 为加速因子, 取实数; r_1 、 r_2 为服从参数为 (0, 1) 均匀分布的随机数; ω_{\max} 、 ω_{\min} 为惯性参数的最大值和最小值; t 为粒子群算法的当前迭代次数; t_{\max} 为粒子群算法的最大迭代次数。

为了保证 X_i 中前 $N(N_s)$ 个和后 $N(N_s)$ 个元素中分别包括 $N(N_v)$ 个 1, 提出基于 VNFE 的粒子二值化方法, 将粒子位置转换为 0 或 1, 表示 SFC 及其对应的 pSFC 中 VNF 被成功映射在服务器上。

针对前 $N(N_s)$ 个元素, 需要使用二值粒子群优化算法求解 SFC 中包含的 VNF 的映射位置。基于 VNFE 的粒子二值化方法为:

```

初始化:  $j = 1$ ;
when  $j < N(N_v)$ 
{
  计算:  $s(v_{ij}^t) = \frac{1}{1 + e^{-v_{ij}^t}}$ ,  $q_{ij}^t = \frac{s(v_{ij}^t)}{\sum_j s(v_{ij}^t)}$ ;
  if ( $d < q_{ij}^t$ )
  {
     $x_{ij}^{t+1}$  取  $x_{ij}^t$  的补集;
    if ( $x_{ij}^{t+1} = 1$ )
       $S(N_v) = S(N_v) / S_j(N_v)$ ;
  }
  else
     $j++$ ;
}

```

其中, d 为 (0, 1) 之间服从均匀分布的随机数, $S_j(N_v)$ 其中表示集合 $S(N_v)$ 中的第 j 个元素, $S(N_v) = S(N_v) / S_j(N_v)$ 表示从集合 $S(N_v)$ 中去掉第 j 个元素 $S_j(N_v)$ 。

针对后 $N(N_s)$ 个元素, 需要使用二值粒子群优化算法求解 pSFC 中包含的 VNF 的映射位置。基于 VNFE 的粒子二值化方法为:

```

初始化:  $j = 1$ ;
when  $j < N(N_v)$ 
{
  计算:  $s(v_{ij}^t) = \frac{1}{1 + e^{-v_{ij}^t}}$ ,  $q_{ij}^t = \frac{s(v_{ij}^t)}{\sum_j s(v_{ij}^t)}$ ;
  if ( $d < q_{ij}^t$ )
  {
     $x_{ij}^{t+1}$  取  $x_{ij}^t$  的补集;
    if ( $x_{ij}^{t+1} = 1$ )
       $S_p(N_v) = S_p(N_v) / S_j^p(N_v)$ ;
  }
  else
     $j++$ ;
}

```

其中, d 为 $(0, 1)$ 之间服从均匀分布的随机数, $S_j^p(N_v)$ 其中表示集合 $S_p(N_v)$ 中的第 j 个元素, $S_p(N_v) = S_p(N_v)/S_j^p(N_v)$ 表示从集合 $S_p(N_v)$ 中去掉第 j 个元素 $S_j^p(N_v)$.

3.2 应用粒子群优化的 VNFE 算法

基于条件反亲和度的 VNFE 算法以 SFC 和 pSFC 发生故障的条件概率最大为优化目标, 同时满足可靠度约束、资源约束和二值约束, 应用粒子群优化算法求解 VNFE 的解. 尽最大可能放宽反亲和性设计的约束, 不要求 SFC 和 pSFC 中的 VNF 映射在不同的服务器上, 只需保证完成 NFV 资源分配后 SFC 和 pSFC 不会发生故障的概率满足网络可靠度要求即可, 提升 VNFE 的成功率.

此时, 根据基于条件反亲和度的 VNFE 算法的优化目标, 定义粒子群优化算法的适应度函数:

$$\min \left\{ 1 - \prod_{u, u'=1}^{N(N_V)} (1 - f(pS_{u'}/S_u)) \right\} \quad (11)$$

等价于:

$$\max \left\{ \prod_{u, u'=1}^{N(N_V)} (1 - f(pS_{u'}/S_u)) \right\} \quad (12)$$

应用粒子群优化的 VNFE 算法的流程如图 2 所示.

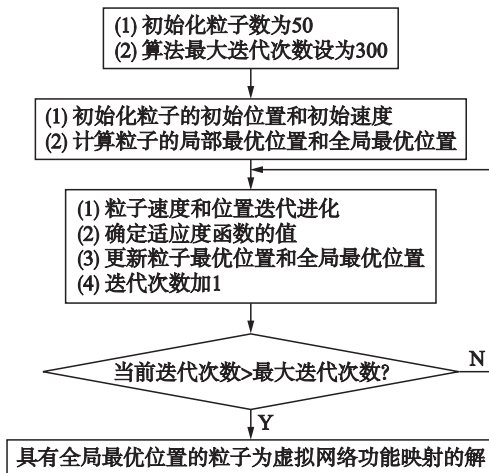


图 2 虚拟网络功能映射算法流程图
Fig.2 The flow chart of the virtual network function embedding algorithm

应用粒子群优化的 VNFE 算法的具体流程为:

Step 1 设定粒子数为 50, 最大迭代次数为 300.

Step 2 初始化例子的初始位置 x_{ij}^0 及初始速度 v_{ij}^0 , 以式(12)为二值粒子群算法的适应度函数, 应用粒子重构机制确定 x_{ij} 的值, 确定本地最优粒子 p_{ij}^0 和全局最优粒子 p_{gij}^0 .

Step 3 当粒子进化到第 t 代时, 使用式(8)~式(10)刷新粒子速度 v_{ij}^{t+1} 和粒子位置 x_{ij}^{t+1} , 应用粒子重构机制确定 x_{ij} 的值, 以适应度函数计算本地最优粒子 p_{ij}^t 和全局最优粒子 p_{gij}^t ; $t = t + 1$.

Step 4 如果 t 的值大于最大迭代次数, 转 Step 3; 否则, 转 Step 5;

Step 5 应用二值粒子群优化的虚拟网络功能映射算法的解为 p_{gij}^t .

4 仿真与性能分析

仿真过程中, 采用具有控制与传输分离特征的美国 Internet2 网络拓扑作为仿真拓扑. 该拓扑包括 34 节点, 每个节点上部署一台服务器, 服务器上的计算资源服从 4 000 到 5 000 的均匀分布, 每个服务器的故障概率服从 $(0.9, 1)$ 的均匀分布. 包括 VNF 的 SFC 相互独立, 每个 SFC 中的 VNF 个数服从 $(1, 10)$ 的均匀分布, 每个 VNF 请求的计算资源服从均值为 500、方差为 0.5 的正态分布. SFC 到达网络的时间间隔服从参数为 λ 的泊松分布, 被成功映射后其持续时间服从参数为 β 的负指数分布, 此时网络负载为 β/λ .

为了方便仿真, 设定 $\lambda = 1$, β 从 20 到 200 以 20 为间隔逐步增加, 模拟网络负载的变化. 为了减少随机数对仿真性能的影响, 针对每一个确定的网络负载, 模拟产生 10 000 个 SFC 并将其顺序加入网络, 即到即处理, 不对 SFC 排队, 如果 SFC 被阻塞, 则该 SFC 被直接丢弃.

由于本文首次提出应用双备份实现 NFV 资源分配, 当对其进行仿真分析时, 将其与反亲和性设计进行对比. 将本文所提算法标记为 C-VNFE, 将反亲和性设计标记为 A-VNFE, 通过映射成功率、服务器共享率、可靠性保障三个指标分析所提算法的性能, 同时给出所提算法的收敛性分析.

4.1 映射成功率分析

图 3 为所提基于条件反亲和度 VNFE 算法与反亲和度 VNFE 算法的映射成功率分析.

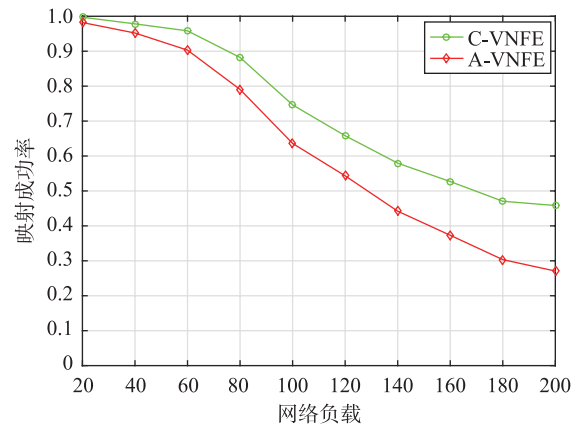


图 3 映射成功率分析

Fig.3 The analysis of the embedding success ratio

从图 3 可以看出, 在相同网络负载条件下, C-VNFE 算法的映射成功率大于 A-VNFE 算法的映射成功率. 这是因为 A-VNFE 算法必须将 SFC 中的 VNF 和 pSFC 中的 VNF 映射在不同的服务器上, 但是网络中的服务器个数有限, 如果服务器个数小于 SFC 和 pSFC 中所有 VNF 的总数, 则 SFC 和 pSFC 映射失败, 导致映射成功率较低. 与之相对的, 本文所提 C-VNFE 算法能够在满足网络可靠度要求的前提下, 将 SFC 及其对应 pSFC 中的 VNF 映射在相同的服务器上, 放宽了反亲和性 VNFE 设计的约束条件, 增加了 SFC 和 pSFC 映射成功的概率, 其映射成功率较高.

随着网络负载的增加,同时映射在服务器上的 SFC 及其对应 pSFC 越来越多,服务器上剩余的计算资源越来越少,当新的 SFC 及其对应 pSFC 到来时,服务器上剩余资源不能满足其要求的可能性增加,本文所提算法与对比算法的映射成功率都会下降.

4.2 服务器共享率分析

服务器共享率是指承载 pSFC 中 VNF 的服务器个数与承载 SFC 及其对应的 pSFC 中所有 VNF 的服务器个数的比值,描述在完成 VNFE 时对服务器的共享程度. 图 4 为所提算法与对比算法的服务器共享率分析.

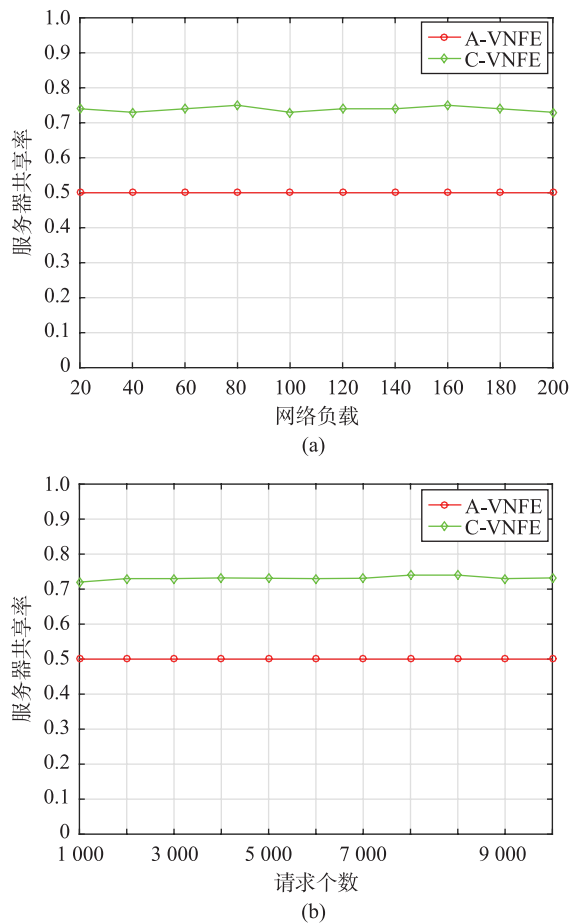


图4 服务器共享率分析

Fig.4 The analysis of server sharing ratio

从图 4 可以看出,在任何网络负载条件下, A-VNFE 算法的服务器共享率始终为 50%,这是因为 pSFC 为复制 SFC 所得,并且当应用 A-VNFE 算法映射 SFC 和 pSFC 中的 VNF 时,只要 SFC 和 pSFC 都能够映射成功,承载 SFC 和 pSFC 中 VNF 的服务器个数始终相等. C-VNFE 算法的服务器共享率在任何网络负载条件下都比 A-VNFE 算法高,这是因为 C-VNFE 算法放宽了反亲和性设计的约束条件,使得 SFC 中的部分 VNF 和 pSFC 中的部分 VNF 能够被映射在相同的服务器上,提高了服务器共享率. 由于网络的可靠度要求不同,使得当应用 C-VNFE 算法求解双备份 VNFE 解时,被映射在同一个服务器上的 SFC 中的 VNF 个

数与 pSFC 中的 VNF 个数不断变化,该算法的服务器共享率也会出现波动.

4.3 可靠性分析

可靠性保障是指 SFC 及其对应的 pSFC 被映射成功后,其不发生故障的概率能够满足网络的可靠度要求. 图 5 为所提算法与对比算法的平均条件故障概率及其与网络平均可靠度需求的对比曲线.

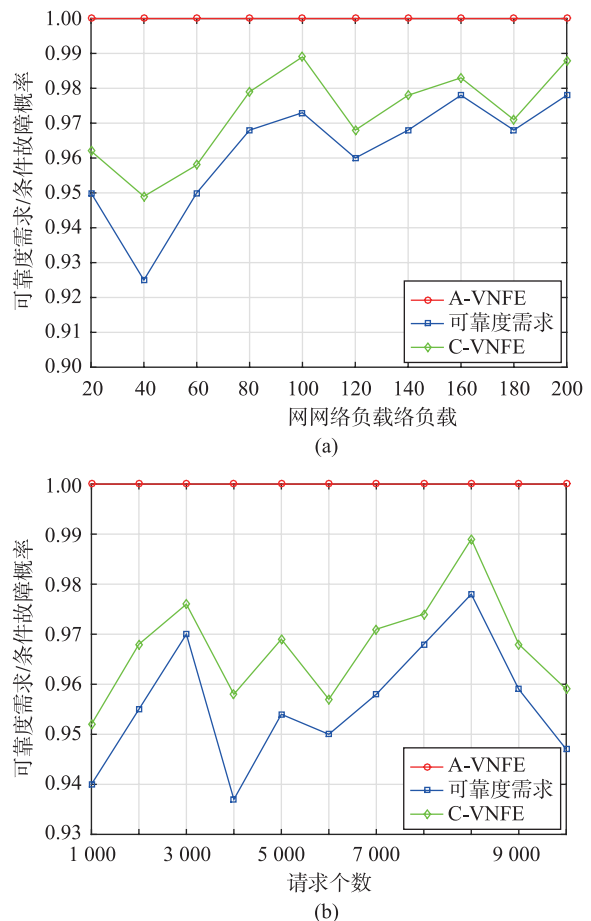


图5 可靠性分析

Fig.5 The analysis of reliability

从图 5 可以看出,在任何网络负载条件下, A-VNFE 算法的条件故障概率始终为 0,这是因为 SFC 中的 VNF 与 pSFC 中的 VNF 被映射在不同的服务器上,任何一个服务器发生故障时都不会影响其它服务器上承载的 VNF. 当计算 VNFE 的解时, C-VNFE 算法以 SFC 和 pSFC 条件故障概率小于网络可靠度请求为约束条件,以条件故障概率最大为优化目标,使得 C-VNFE 算法的条件故障概率虽然大于 0,但是能够满足网络的可靠度要求.

4.4 算法收敛性分析

由于本文所提算法应用粒子群优化算法获得基于条件反亲和度 VNFE 的解,图 6 给出典型网络负载条件下,任意一次应用所提算法求解 VNFE 时粒子群算法的收敛曲线. 其中,在所提粒子群算法中,粒子数和最大迭代次数分别设为 50 和 300,初始化粒子位置和速度时取(0, 1)之

间均匀分布的随机数。

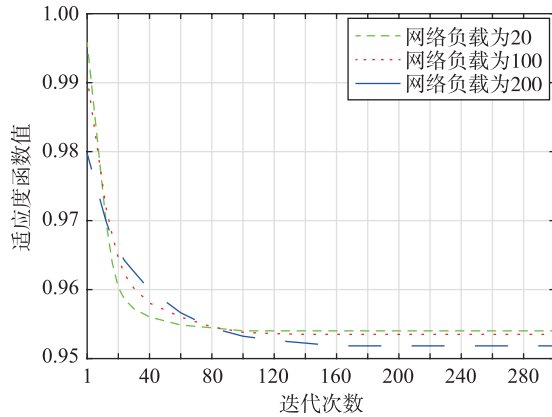


图6 收敛性分析

Fig.6 The analysis of convergence

从图6可以看出,负载越大所提算法收敛越快.这是因为负载越大,同时出现的SFC及其对应pSFC请求越多,服务器上剩余的计算机资源越少,可选择映射SFC和pSFC中VNF的服务器个数越少,所提算法中应用粒子群优化算法时的解空间越小,算法收敛速度越快.同时,随着粒子群算法解空间的缩小,计算其适应度函数时的可行解变少,最终收敛的值越小.

粒子群算法中不同的粒子数和最大迭代次数对算法的收敛性影响较大,而粒子数和最大迭代次数的选择与所求解优化问题及其适应度函数的定义关系密切.因此,针对本文仿真环境和所解决的问题,表2中给出不同网络负载条件下,不同粒子数和最大迭代次数对应的适应度函数值.

从表2中可以看出,针对不同的网络负载,不同的粒子数和最大迭代次数对粒子群算法的收敛性影响都一样.当粒子数较少时,即使迭代次数增加,算法也不能收敛到最优解.当粒子数设定为50,最大迭代次数为300时,算法能够收敛到最优解,但是当粒子数超过50且最大迭代次数数值较大时,算法对应的适应度函数值没有变化,但是其算法复杂度大大增加.因此,在本文所提仿真环境中,

针对SFC和pSFC的映射问题,选择粒子数50,最大迭代次数为300,不仅能够避免粒子群算法早熟,还能避免算法复杂度的增加.

表2 适应度函数值分析

Tab.2 The analysis of the value of fitness function

网络负载	粒子数	最大迭代次数	适应度函数值
20	20	300	0.961 7
		500	0.957 9
	50	300	0.954 6
		500	0.954 6
	70	300	0.954 6
		500	0.954 6
100	20	300	0.960 9
		500	0.955 5
	50	300	0.954 3
		500	0.954 3
	70	300	0.954 3
		500	0.954 3
200	20	300	0.960 8
		500	0.953 3
	50	300	0.952 1
		500	0.952 1
	70	300	0.952 1
		500	0.952 1

5 总结

本文提出基于条件反亲和度的网络功能虚拟化资源分配算法,在定义条件反亲和度的基础上,应用粒子群优化算法求解双备份SFC映射的最优解.仿真结果表明,所提算法在满足业务请求可靠度的基础上,能够将虚拟网络功能映射的成功率提升46%,为在电信网络中分配服务器资源、实例化SFC提供了参考价值.不仅可以保证电信网络的高可靠性要求,还能够提高网络资源利用率,减少电信网络的建设成本.

参考文献

- [1] Naudts B, Tavernier W, Verbrugge S, et al. Deploying SDN and NFV at the speed of innovation: Toward a new bond between standards development organizations, industry fora, and open-source software projects[J]. IEEE Communications Magazine, 2016, 54(3): 46-53.
- [2] Matias J, Garay J, Toledo N, et al. Toward an SDN-enabled NFV architecture[J]. IEEE Communications Magazine, 2015, 53(4): 187-193.
- [3] Quinn P, Guichard J. Service function chaining: Creating a service plane via network service headers[J]. Computer, 2014, 47(11): 38-44.
- [4] Faizul M, Chowdhury S, Ahmed R, et al. On orchestrating virtual network functions[C]//9th International Conference on Network and Service Management. Piscataway, NJ, USA: IEEE, 2015: 50-56.
- [5] Till M, Botero J. Coordinated allocation of service function chains[C]//Global Communications Conference. Piscataway, NJ, USA: IEEE, 2015: 1-6.
- [6] Xilouris G, Kourtis M, McGrath M, et al. T-NOVA: Network functions as-a-service over virtualised infrastructures[C]//IEEE Conference on Network Function Virtualization and Software-Defined Networks. Piscataway, NJ, USA: IEEE, 2015: 13-14.
- [7] Ming X, Meral S, Zhang Y, et al. Network function placement for NFV chaining in packet/optical datacenters[J]. Journal of Lightwave Technology, 2015, 33(8): 1565-1570.

- [8] Bellavista P, Callegati F, Cerroni W, et al. Virtual network function embedding in real cloud environments[J]. *Computer Networks*, 2015, 93(3): 506 – 517.
- [9] Gupta A, Habib M, Chowdhury P, et al. On service chaining using virtual network functions in network-enabled cloud systems[C]//3rd International Conference on Advanced Networks and Telecommunications Systems. Piscataway, NJ, USA: IEEE, 2016: 1 – 3.
- [10] Powen C, Huang Y C, Lin C L. Efficient NFV deployment in data center networks[C]//18th International Conference on Signal Processing for Communications. Piscataway, NJ, USA: IEEE, 2015: 5290 – 5295.
- [11] Addis B, Belabed D, Bouet M, et al. Virtual network functions placement and routing optimization [C]//4th International Conference on Cloud Networking. Piscataway, NJ, USA: IEEE, 2015: 1 – 7.
- [12] Munoz R, Vilalta R, Casellas R, et al. SDN/NFV orchestration for dynamic deployment of virtual SDN controllers as VNF for multi-tenant optical networks[C]//Optical Fiber Communications Conference and Exhibition. Piscataway, NJ, USA: IEEE, 2015: 1 – 6.
- [13] Jaeger B. Security orchestrator: Introducing a security orchestrator in the context of the ETSI NFV reference architecture[C]//IEEE Trustcom/BigDataSE/ISPA. Piscataway, NJ, USA: IEEE, 2015: 1255 – 1260.
- [14] Kaneko M, Okamoto M, Fukumoto T. A robust VNF allocation method in NFV[R]. Japan: IEICE technical report, 2015.
- [15] Herker S, Ai X L, Kiess W, et al. Data-center architecture impacts on virtualized network functions service chain embedding with high availability requirements[C]//Global Communications Conference. Piscataway, NJ, USA: IEEE, 2015: 1 – 7.
- [16] Tam F. Service availability standards for carrier-grade platforms; Creation and deployment in mobile networks[M]. Tampere, Finland: Tampere University of Technology, 2009: 156 – 159.
- [17] Jiang M, Luo Y, Yang S. Stochastic convergence analysis and parameter selection of the standard particle swarm optimization algorithm[J]. *Information Processing Letters*, 2007, 102: 8 – 12.
- [18] 杨淑云, 徐云霞, 李盼池. 基于 Bloch 球面搜索的量子鱼群算法[J]. *信息与控制*, 2014, 43(6): 647 – 653.
Yang S Y, Xu Y X, Li P C. Quantum-inspired artificial fish swarm algorithm based on the bloch sphere search algorithm[J]. *Information and Control*. 2014, 43(6): 647 – 653.
- [19] 乔俊飞, 王超, 魏静. 一种具有局部搜索的自适应粒子群算法[J]. *信息与控制*, 2015, 44(4): 385 – 392.
Qiao J F, Wang C, Wei J. An adaptive particle swarm optimization algorithm with local search[J]. *Information and Control*, 2015, 44(4): 385 – 392.
- [20] 肖红, 李盼池. 改进的量子行为粒子群优化算法及其应用[J]. *信息与控制*, 2016, 45(2): 157 – 164.
Xiao H, Li P C. Kinematic parameter identification of parallel manipulator based on improved particle swarm algorithm[J]. *Information and Control*, 2016, 45(2): 157 – 164.

作者简介

林 青(1979 –), 女, 硕士, 讲师. 研究领域为计算机网络.

戴慧珺(1979 –), 女, 博士, 工程师. 研究领域为 D2D, 移动用户安全, 位置隐私.