

时间和能量感知的贝叶斯虚拟网映射

胡颖¹, 庄雷¹, 陈鸿昶², 马丁^{1,3}

(1. 郑州大学信息工程学院, 河南 郑州 450000;

2. 国家数字交换系统工程技术研究中心, 河南 郑州 450002;

3. 河南工业大学信息科学与工程学院, 河南 郑州 450000)

摘要: 针对虚拟网的节能映射问题, 建立了结合时间和能量感知的虚拟网映射算法。在对节点和路径的评价标准中加入了时间因素, 综合考虑了物理资源的运行时间等因素, 用概率理论辅助分析了每个虚拟节点的多个可用物理节点被选中的概率。在节点选择阶段, 综合考虑底层节点的剩余资源量、CPU 资源利用率增量、节点开启情况和是否延长使用时间等因素, 并使用条件概率理论辅助分析得到各可用节点的重要性; 在链路选择阶段, 综合考虑链路开启情况、延长使用时间和链路长度等因素。不仅使虚拟网请求映射在当前较小的节点和链路集合中, 而且映射到了延长使用时间较短的设备上。实验结果表明, 与未考虑时间因素的方法相比, 该方法能带来更好的性能和更低的能耗。

关键词: 虚拟网映射; 节能; 贝叶斯; 条件概率; 时间感知

中图分类号: TP393

文献标识码: A

Time and energy aware virtual network embedding using Bayesian theory analysis

HU Ying¹, ZHUANG Lei¹, CHEN Hong-chang², MA Ding^{1,3}

(1. School of Information Engineering, Zhengzhou University, Zhengzhou 450000, China;

2. National Digital Switching System Engineering & Technological R&D Center, Zhengzhou 450002, China;

3. College of Information Science and Engineering, Henan University of Technology, Zhengzhou 450000, China)

Abstract: Aiming at the energy consumption problem in virtual network embedding, a virtual-network-embedding algorithm was proposed by combining the time and energy aware. Taking the running time during the evaluation of physical nodes and physical paths into account, it considered multiple factors which included the processing time of physical devices, and used probability theory to help analyze the selected probability of each available physical node for a virtual node. During the selection of substrate nodes, the factors of remaining resources, the increment of CPU utilization, the switch state and the amount of extended time of physical nodes were considered. The theory of conditional probability was further used to analyze the importance of available nodes. The factors of the switch state, the amount of extended time and the length of physical paths were also considered. The proposed approach could effectively map the current virtual network request onto a smaller set of nodes and links which are switched on, and also the devices which have less amount of extended time. Experimental results show that the proposed approach has better performance, and can effectively decrease energy consumption comparing with the methods without taking the time factor into consideration.

Key words: virtual network embedding, energy-saving, Bayesian, conditional probability, time aware

收稿日期: 2015-08-06; 修回日期: 2016-03-10

通信作者: 庄雷, ielzhuang@zzu.edu.cn

基金项目: 国家重点基础研究发展计划(“973”计划)基金资助项目(No.2012CB315901); 国家自然科学基金资助项目(No.61379079); 河南省科技厅攻关基金资助项目(No.122102210042)

Foundation Items: The National Basic Research Program of China (973 Program)(No.2012CB315901), The National Natural Science Foundation of China (No.61379079), Science and Technology Key Project of Henan Province (No.122102210042)

1 引言

随着电力成本的不断上涨，网络运营商已经认识到能耗管理问题的重要性。据有关研究显示^[1,2]，在数据中心，能耗开销已经占数据中心总开销的12%~20%，占运营开销的40%~50%。在Internet中，能耗开销也已经成为Internet服务提供商总开销的重要组成部分。能耗开销过大是由于当前网络为高峰负荷设计，网络资源超需供给确保了网络正常运行，却也导致资源利用率低下。据统计，大型ISP骨干网的平均链路利用率为30%~40%^[5]。过低的利用率造成能源的极大浪费，使网络能耗问题成为研究人员关注的热点问题^[6,7]。

网络虚拟化技术是解决网络僵化问题的关键解决方案，是未来网络研究中的一种重要技术。在网络虚拟化环境中，多个服务提供商(SP, service provider)以虚拟网请求的方式请求租用同一个基础设施提供商(InP, infrastructure provider)提供的物理网络，从而向端用户提供服务；InP负责部署和管理底层物理网络，优化虚拟网映射策略以使利益最大化，即映射更多的虚拟网请求的同时，使占用底层网络资源量最小化，以减少运行成本。运行成本不仅包括资源的占用，还包括运行的能耗。在网络虚拟化环境中降低能耗的方式有2种：虚拟网映射时（建立虚拟网时）和虚拟网运行中（建立虚拟网后）。本文关注虚拟网映射时的节能问题，即虚拟网节能映射问题。

在虚拟网节能映射中包括2个问题：1)如何对虚拟网节能映射问题建立模型；2)如何确定虚拟网节能映射算法，即根据模型建立的目标确定节点和链路映射的方法，以整合资源开启较小的节点和链路集合，达到减少能耗的目的。

当前已有一些学者对虚拟网节能映射问题进行了有意义的研究。文献[8]将虚拟网映射问题成功扩展到虚拟网节能映射问题，并建立了混合整数线性规划模型，将节能映射的目标设置为开启的节点数量和开启的链路数量之和；文献[9]从节能和占用资源最小化2个角度对虚拟网映射问题建立了整数线性规划模型，并分别以其中一个为主要目标，另一个为次要目标进行了解决和分析；文献[10]从能耗、请求优先级和请求的位置限制等3个条件来限制虚拟网节能映射问题，分析并得到了较优的结果；文献[11]为数据中

心网络的虚拟网节能映射问题建立了模型，从实际因素出发，在计算资源和网络资源2个角度对问题进行了分析讨论；文献[12]应用最优化理论为虚拟网节能映射问题建立了模型；文献[13]为云计算网络的虚拟网节能映射问题建立了混合整数线性规划模型；文献[14]从节能、负载均衡等多个目标设计虚拟网映射模型，并针对云计算网络中的节点或链路故障问题，提出了一个容错的映射算法。

以上虚拟网节能映射算法的主要节能方式是尽可能地节点链路映射到活动的底层节点链路上，达到了较好的节能效果，但都没有考虑延长运行时间对映射效果的影响。本文在对节点和链路评价标准的设计中考虑时间因素，加入了对物理资源的延长运行时间这一要素，设计了基于时间和能量感知的虚拟网映射算法(TEAVNE, time and energyaware virtual network embedding)。算法中，用传统概率理论辅助分析，并确定了每个虚拟节点的多个可用物理节点将被选中的概率，并通过仿真实验对该算法进行了验证与分析。

2 虚拟网节能映射模型

底层网络能耗包括物理节点能耗和物理链路2个部分，对其分析前，首先对符号定义如表1所示。

表1 符号表示

符号	描述
N^s	底层网络的物理节点集合
L^s	底层网络的物理链路集合
N^v	虚拟网的虚拟节点集合
L^v	虚拟网的虚拟链路集合
N_w	映射当前虚拟网需要开启的物理节点个数
N_{lk}	映射当前虚拟网需要开启的物理链路个数
x_j	设备(可以是物理节点或物理链路) j 本来为关闭状态时值为1，否则为0
f_j^u	虚拟节点 u 映射在物理节点 j 上时，值为1，否则为0
f_{jk}^{uv}	虚拟链路 l_{uv} 映射在物理链路 l_{jk} 上时，值为1，否则为0
$E_M(j)$	映射在物理节点 j 上的虚拟节点集合
$E_L(l_{jk})$	映射在物理链路 l_{jk} 上的虚拟链路集合
$ReqCPU(v)$	虚拟节点 v 请求的CPU资源量
$ReqBW(l_v)$	虚拟链路 l_v 请求的带宽资源量
$CPU(j)$	物理节点 j 的CPU资源总量
$BW(l_{jk})$	物理链路 l_{jk} 的带宽资源总量
$duration_R$	虚拟网请求 R 的运行时间
NO_i	有效时间片 i 内处于开启状态的物理节点数
LO_i	有效时间片 i 内处于开启状态的物理链路数

2.1 物理节点功耗

同文献[15], 设置物理节点 j 的功耗 (ECN, energy consumption in node) 为

$$ECN_j = \begin{cases} P_1 util_j + P_b, & \text{宿主节点 } j \text{ 处于激活状态} \\ 0, & \text{其他} \end{cases} \quad (1)$$

其中, P_b 为节点的基本能耗; 令 P_m 为节点的最大能耗, 则 $P_1 = P_m - P_b$; $util_j$ 为节点 j 的 CPU 利用率。

2.2 物理链路功耗

同文献[15], 定义底层网络物理链路 l_{jk} 的功耗 (ECL, energy consumption in link) 为

$$ECL_{l_{jk}} = \begin{cases} P_n, & \text{物理链路 } l_{jk} \text{ 处于激活状态} \\ 0, & \text{其他} \end{cases} \quad (2)$$

其中, 物理链路的功耗 P_n 是常量。

2.3 底层网络总能耗

为方便分析描述, 首先做如下定义。

定义 1 如果在某个时间段内, 底层网络上没有发生虚拟网请求的到达或离开事件, 称此时段的底层网络处于稳定状态。

定义 2 如果在某时刻发生了虚拟网请求的映射或离开事件 (这里认为映射和离开事件的发生时间忽略不计), 称该时刻为分裂点。

定义 3 相邻 2 个分裂点间的时间段, 称为有效时间片。

由以上定义可知, 在有效时间片内, 底层网络处于稳定状态; 在不同的有效时间片间, 底层网络单位能耗量不同。于是, 计算底层网络总能耗前应先确定所有分裂点, 由各分裂点将运行时间分为多个有效时间片, 在各个有效时间片内分别计算底层网络的能耗, 底层网络的总能耗即各有效时间片的能耗总和为

$$E^{\text{sum}} = \sum_{i=1}^{ts} E^i T_d^i \quad (3)$$

其中, ts 为分裂点划分出的有效时间片总个数, E^i 为有效时间片 i 内底层网络的单位能耗, T_d^i 为有效时间片的时长。

在有效时间片内, 物理网络设备稳定运行, 底层网络能耗即物理节点和物理链路的能耗之和。由式(1)和式(2), 得到底层网络的单位能耗 E^i 为

$$E^i = NO_i P_b + P_1 \sum_{j \in N^s} util_j + LO_i P_n \quad (4)$$

2.4 虚拟网节能映射问题模型

根据式(4), 设置虚拟网节能映射问题的目标

如下。

目标函数为

$$\text{Minimize} \left(P_1 \sum_{\substack{j \in N^s \\ u \in N^v}} \left(f_j^u \frac{ReqCPU(u)}{CPU(j)} \right) + P_b N_w + P_n N_{lk} \right) \quad (5)$$

和虚拟网映射问题一样, 虚拟网节能映射问题的约束条件包括以下几项。

1) 容量约束

$$\begin{aligned} & (\forall u \in N^v) (\forall j \in N^s): f_j^u ReqCPU(u) \\ & \leq CPU(j) - \sum_{v \in E_N(j)} ReqCPU(v) \\ & (\forall l_{uv} \in L^v) (\forall l_{jk} \in L^s): f_{jk}^{uv} ReqBW(l_{uv}) \\ & \leq BW(l_{jk}) - \sum_{l_v \in E_L(l_{jk})} ReqBW(l_v) \end{aligned} \quad (6)$$

2) 传输约束

$$(\forall j \in N^s) (\forall l_{uv} \in L^v): \sum_{l_{jk} \in L^s} f_{jk}^{uv} - \sum_{l_{kj} \in L^s} f_{kj}^{uv} = \begin{cases} 1, & f_j^u = 1 \\ -1, & f_j^w = 1 \\ 0, & \text{其他} \end{cases} \quad (7)$$

一个虚拟节点只能映射到一个底层节点的约束

$$(\forall u \in N^v): \sum_{j \in N^s} f_j^u = 1 \quad (8)$$

3) 相同虚拟网请求的虚拟节点不能映射到同一个底层节点的约束

$$(\forall j \in N^s): \sum_{u \in N^v} f_j^u \leq 1 \quad (9)$$

4) 变量约束

$$\begin{aligned} & (\forall j \in N^s) (\forall u \in N^v): f_j^u \in \{0, 1\} \\ & (\forall l_{jk} \in L^s) (\forall l_{uv} \in L^v): f_{jk}^{uv} \in \{0, 1\} \end{aligned} \quad (10)$$

以上基于混合整数规划设计得到虚拟网节能映射模型, 下面针对该问题设计节点和链路的节能评价标准以及节能映射的贪婪算法, 找到虚拟网节能映射的较优解。

3 时间和能量感知的虚拟网映射算法

与传统的虚拟网节能映射算法相比, TEAVNE 算法最主要的不同点在于: 在为虚拟节点选取映射目标时, 将会把占用设备的时间纳入其评价要素,

并应用传统概率论方法分析可用节点的选中概率，使虚拟网请求优先映射在已开启、延长资源占用时间较短且增加资源利用率较少的物理节点上；在为虚拟链路选取物理路径时，兼顾了路径的长度、路径中物理链路的开启量和物理链路的延长使用时间等 3 个因素。下面首先给出节点和链路的评价要素和评价标准。

3.1 节点和链路的评价要素

由虚拟网映射的基本指标——占用资源量和请求接收率等，可知评价要素中应包括如资源剩余量、物理路径的长度等要素。由虚拟网节能映射的目标式(5)，应将物理节点和物理链路的开关状态以及增加的 CPU 资源利用率作为评价要素。加入时间因素后，还应将当前的虚拟网请求是否延长设备运行时间以及延长运行时间的大小加入评价要素，否则将可能造成较差的节能效果。一个虚拟网映射的例子如图 1 所示。

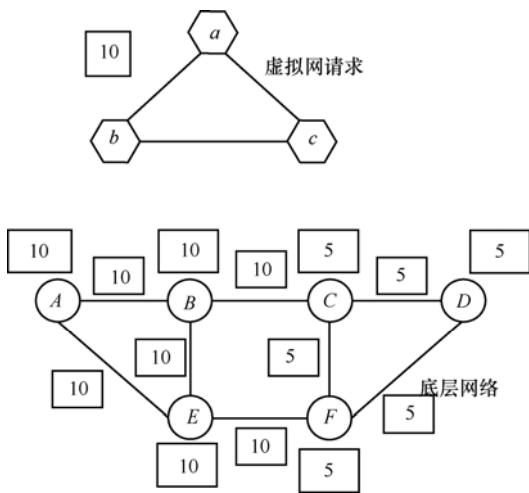


图 1 虚拟网映射的例子

图 1 中，六边形节点构成的拓扑代表虚拟网，圆形节点构成的拓扑代表底层物理网络，正方形框中的数字代表虚拟网的运行时间，矩形框中的数字表示底层网络节点或链路还要被占用的时间。

设图 1 中的底层网络的带宽和 CPU 剩余资源足够多，各物理节点或链路均可满足虚拟网请求的资源量，并设物理节点的 CPU 资源总量都相等。这里忽略执行映射操作的时间。

如果按照传统的节能映射目标和节点链路的评价因素，在节能映射时不考虑延长节点或链路的运行时间等因素，那么将 a、b、c 映射到 C、D、F 或将其映射到 A、B、E 的这 2 种方案的节点或链路

开启量都是零，CPU 资源上增加的总的资源利用率也相等。也就是说，2 种映射方案得到的当时的映射目标值是相等的。而在 5 个时间单位后，映射到 A、B、E 的方案下释放的节点和链路量较多。也就是说，不纳入时间因素的评价标准可能会增加长期的系统能耗。因此，在虚拟网节能映射中，为节点和链路设计评价标准时，应考虑时间因素，即将延长节点或链路的使用时间这一因素纳入评价要素。

综上，在对虚拟网节能映射时，对节点的评价要素包括设备是否已开启、增加的 CPU 资源利用率之和、是否延长设备运行时间以及延长运行时间的大小、资源剩余量等要素；对物理路径的评价要素包括需要开启的设备量、延长设备使用的时间和物理路径的长度等要素。

3.2 节点的评价标准

在为虚拟节点选取映射目标时，需要建立一种针对候选底层节点的评价标准，该标准对底层物理节点是否适合承载该虚拟节点进行评价，从而确定每个可用节点的重要性。

节点评价要素中有关节能的 2 个布尔型变量是物理节点是否已被开启和是否延长了物理节点的使用时间，用这 2 个布尔变量将可用节点集划分为 3 个节点集合：令 S_1 表示已开启且使用时间不会被延长的节点集； S_2 表示已开启且使用时间会被延长的节点集； S_3 表示未开启的节点集。这样，3 个节点集 S_1 、 S_2 和 S_3 是所有可选节点样本空间的一个划分。

令 A_j 表示选中节点集 S_j 的事件，若 N_i 表示选中可用节点 n_s^i 的事件，那么 $P(N_i A_j)$ 是联合概率，表示节点集 S_j 中的 n_s^i 被选中的概率。由条件概率公式，联合概率 $P(N_i A_j)$ 可由条件概率与边缘概率之积得到

$$P(N_i A_j) = P(N_i | A_j) P(A_j), n_s^i \in S_j, j \in \{1, 2, 3\} \quad (11)$$

其中， S_j 是节点 n_s^i 所隶属的节点集； $P(N_i | A_j)$ 表示已选中子节点集 S_j 的条件下选择节点 n_s^i 的概率； $P(A_j)$ 表示选择子节点集 S_j 的概率。

3.2.1 对边缘概率 $P(A_j)$ 的计算

为达到较好的节能效果，应优先选择已开启且未延长使用时间的节点，其次选择已开启且使用时间将被延长的节点，即 3 个节点集中元素的选中次序为 $S_1 \rightarrow S_2 \rightarrow S_3$ ，因此，应保证隶属于 3 个节点集的节点联合概率的平均值满足如下关系

$$\begin{aligned} \overline{P(N_q A_1)} &= \frac{\sum_{n'_x \in S_1} P(N_i A_1)}{|S_1|} > \overline{P(N_q A_2)} \\ &= \frac{\sum_{n'_x \in S_2} P(N_j A_2)}{|S_2|} > \overline{P(N_q A_3)} = \frac{\sum_{n'_x \in S_3} P(N_k A_3)}{|S_3|} \end{aligned} \quad (12)$$

其中, $|S_i|$ 为节点集 S_i 中节点的个数。

于是, 定义 $P(A_j)$ 为

$$P(A_j) = \frac{a_j |S_j|}{\sum_{i=1}^3 a_i |S_i|} \quad (13)$$

其中, 常数 a_1 、 a_2 和 a_3 分别为节点集 S_1 、 S_2 和 S_3 的经验权重系数。为满足式(12), 应设置权重系数大小满足 $a_1 > a_2 > a_3$ 。这里令 $a_1 + a_2 + a_3 = 1$ 。由式(13),

$$\sum_{i=1}^3 P(A_i) = 1。$$

3.2.2 对条件概率 $P(N_i|A_j)$ 的计算

当节点 n'_s 隶属于节点集 S_j 时, $P(N_i|A_j)$ 表示选中子节点集 S_j 的情况下选择节点 n'_s 的概率, 即节点集 S_j 内节点 n'_s 的相对重要性。下面对 3 个节点集的条件概率 $P(N_i|A_j)$ 分别进行讨论。

1) 节点集 S_1

节点集 S_1 中的节点都已被开启, 且使用时间未被延长, 那么, 节点集内节点间的相对重要性只与 CPU 资源利用率的增量和资源剩余量有关。

定义节点 CPU 资源剩余量如下

$$NR(n'_v, n'_s) = CPU(n'_s) - \sum_{n'_k \in E_N(n'_s)} ReqCPU(n'_k) \quad (14)$$

其中, 为虚拟节点 n'_v 选择可映射的物理节点时, n'_s 为可用的物理节点, 同前所述, $CPU(n'_s)$ 表示底层物理节点 n'_s 的 CPU 资源总量, $E_N(n'_s)$ 表示映射在物理节点 n'_s 上的虚拟节点集合, $ReqCPU(n'_k)$ 表示虚拟节点 n'_k 的 CPU 资源请求量。

节点周围链路的资源剩余量描述了所有与 n'_s 相连的物理链路 l_s 的资源剩余情况, 从另一个角度评价了 n'_s 是否适合承载虚拟节点 n'_v 。将链路资源剩余量定义如下

$$LR(n'_v, n'_s) = \sum_{l_s \in NL(n'_s)} \left(BW(l_s) - \sum_{l'_v \in E_L(l_s)} ReqBW(l'_v) \right) \quad (15)$$

为虚拟节点 n'_v 选择可映射的物理节点时, n'_s 为可用的物理节点之一, $NL(n'_s)$ 表示与 n'_s 相邻的物理链路集合。同前所述, $BW(l_s)$ 表示底层链路 l_s 的带宽资源总量, $E_L(l_s)$ 表示映射在物理链路 l_s 上的虚拟链路集合, $ReqBW(l'_v)$ 表示虚拟链路 l'_v 请求的带宽资源量。

对底层节点 n'_s 做出综合评价时, 应从物理节点的 CPU 资源剩余量及节点相邻链路的带宽资源剩余量这 2 个方面进行分析。于是, 定义资源剩余量为

$$RR(n'_v, n'_s) = NR(n'_v, n'_s) LR(n'_v, n'_s) \quad (16)$$

资源剩余量的值越大, 说明 n'_s 的剩余资源越多, 应增加资源剩余量值较大的节点被选中的可能性。

已知虚拟节点 n'_v 在物理节点 n'_s 上增加的 CPU 资源利用率为

$$\Delta Util_v^i = \frac{ReqCPU(n'_v)}{CPU(n'_s)} \quad (17)$$

在为虚拟节点 n'_v 选择物理节点时, 式(17)中的分子是定值, 影响 CPU 资源利用率增量的唯一因素是可用物理节点的 CPU 资源总量值。该值越大, 增加的资源利用率越小, 应被优先选择, 即该值与节点重要性呈正比例关系。

为评价 n'_s 是否适合承载虚拟节点 n'_v , 可定义节点集 S_1 中节点的相对重要性如下

$$\begin{aligned} R^1(n'_v, n'_s) &= b_1 \frac{RR(n'_v, n'_s) - RR(n'_v, n'_s^{\min}) + \varepsilon_1}{RR(n'_v, n'_s^{\max}) - RR(n'_v, n'_s^{\min}) + \varepsilon_2} + \\ & b_2 \frac{CPU(n'_s) - CPU(n'_s^{\min'}) + \varepsilon_1}{CPU(n'_s^{\max'}) - CPU(n'_s^{\min'}) + \varepsilon_2} \end{aligned} \quad (18)$$

其中, b_1 和 b_2 为经验权重系数。 n'_s^{\min} 表示其该节点集中资源剩余量最小的节点, 则 $RR(n'_v, n'_s^{\min})$ 表示该节点集中最小的资源剩余量。 n'_s^{\max} 表示该节点集中资源剩余量最大的节点, 则 $RR(n'_v, n'_s^{\max})$ 表示该节点集中最大的资源剩余量。 $n'_s^{\min'}$ 表示该节点集中 CPU 资源总量最小的节点, 则 $CPU(n'_s^{\min'})$ 表示该节点集中最小的 CPU 资源总量。 $n'_s^{\max'}$ 表示该节点集中 CPU 资源总量最大的节点, 则 $CPU(n'_s^{\max'})$ 表示该节点集中最大的 CPU 资源总量。 ε_1 和 ε_2 为 2 个相近的极小正数, $\varepsilon_1 < \varepsilon_2 \ll 1$, ε_2 用来避免 2 个最值相

等时, 使分母为 0 的情况; ε_1 用来避免分子为 0 的情况; 当所有节点延长时间相等时, ε_1 和 ε_2 的比值决定了对这种情况的评价值。

定义节点集 S_1 中节点 n_s^i 被选择的条件概率

$$P(N_i|A_1) = \frac{R^1(n'_v, n_s^i)}{\sum_{n'_v \in S_1} R^1(n'_v, n_s^i)} \quad (19)$$

2) 节点集 S_2

节点集 S_2 中的节点已被开启且将被延长使用时间, 延长使用时间的长短对节能效果是有影响的, 应优先选择延长时间较短的节点。另外, 加入节点集 S_1 中的 2 个评价要素, 可定义节点集 S_2 中节点的相对重要性为

$$R^2(n'_v, n_s^i) = bb_1 \frac{RR(n'_v, n_s^i) - RR(n'_v, n_s^{\min}) + \varepsilon_1}{RR(n'_v, n_s^{\max}) - RR(n'_v, n_s^{\min}) + \varepsilon_2} + bb_2 \frac{CPU(n_s^i) - CPU(n_s^{\min}) + \varepsilon_1}{CPU(n_s^{\max}) - CPU(n_s^{\min}) + \varepsilon_2} + bb_3 \frac{et_s^{\max} - et_s^i + \varepsilon_1}{et_s^{\max} - et_s^{\min} + \varepsilon_2} \quad (20)$$

其中, bb_1 、 bb_2 和 bb_3 为经验权重系数, et_s^{\max} 表示节点集 S_2 中节点的最长延长时间, et_s^{\min} 表示该节点集中节点的最短延长时间, et_s^i 表示节点 n_s^i 将被延长的时间。

类似地, 定义节点集 S_2 中节点被选择的条件概率为

$$P(N_i|A_2) = \frac{R^2(n'_v, n_s^i)}{\sum_{n'_v \in S_2} R^2(n'_v, n_s^i)} \quad (21)$$

3) 节点集 S_3

节点集 S_3 中的节点尚未被开启, 延长的使用时间为零, 其评价要素由节点的 CPU 资源总量以及映射虚拟节点 n'_v 后 CPU 资源利用率增量决定。

首先定义资源量

$$RR'(n'_v, n_s^i) = CPU(n_s^i) \sum_{l_s \in NL(n_s^i)} BW(l_s) \quad (22)$$

在节点集 S_3 中, 对 CPU 资源利用率增量的分析和节点集 S_1 中是一致的。类似于对条件概率 $P(N_i|A_1)$ 的定义, 定义节点集 S_3 中节点的相对重要性如下

$$R^3(n'_v, n_s^i) = b_1 \frac{RR'(n'_v, n_s^i) - RR'(n'_v, n_s^{\min}) + \varepsilon_1}{RR'(n'_v, n_s^{\max}) - RR'(n'_v, n_s^{\min}) + \varepsilon_2} +$$

$$b_2 \frac{CPU(n_s^i) - CPU(n_s^{\min}) + \varepsilon_1}{CPU(n_s^{\max}) - CPU(n_s^{\min}) + \varepsilon_2} \quad (23)$$

定义 S_3 中节点 n_s^i 被选择的条件概率

$$P(N_i|A_3) = \frac{R^3(n'_v, n_s^i)}{\sum_{n'_v \in S_3} R^3(n'_v, n_s^j)} \quad (24)$$

以上从节点的开关状态和节点是否被延长了使用时间 2 个角度将底层节点分类, 根据经验倾向和各节点集中元素的数量确定选择各集合的概率, 即确定边缘概率; 在各节点集中, 从剩余资源量、CPU 资源利用率增量以及所延长的节点使用时间这 3 个方面, 综合评价隶属各节点集中物理节点的相对重要性, 从而确定条件概率。

在节能目标下, 为某个虚拟节点选择映射节点时, 应首先为所有可用物理节点计算联合概率。联合概率值越大, 说明节点的重要性越高, 节点越适宜承载该虚拟节点。

3.3 物理路径的评价标准

在虚拟网映射中, 节点映射成功后, 通常是使用 k 最短路算法得到多条较短的物理路径。物理路径越长, 消耗的底层链路带宽资源越多, 所以从占用资源量的角度应选择最短物理路径。然而, 在虚拟网节能映射中, 最短路径上可能需要开启较多的物理链路, 因此不一定是最节能的路径。为达到较好的节能效果, 在选择物理路径时, 要兼顾路径的长度、物理链路的开启量和物理链路的延长使用时间等多个因素。

首先, 对物理链路 l_s^i 的节能长度定义如下

$$llen(l_s^i) = \begin{cases} 1, & l_s^i \text{ 未开启} \\ c \frac{et_{sl}^i}{rl}, & l_s^i \text{ 已开启且会延长运行时间} \\ 0, & l_s^i \text{ 已开启且不会延长运行时间} \end{cases} \quad (25)$$

其中, c 为经验权重系数, et_{sl}^i 表示物理链路 l_s^i 将被延长的运行时间, rl 表示该虚拟网的请求运行时间, 易知二者比值的值域为 $(0, 1)$ 。

最后, 通过定义节能路径长度 ELK 来对虚拟链路的多个可用物理路径进行评价。

$$ELK(l'_v, p) = \sum_{l'_s \in p} llen(l'_s) \quad (26)$$

由式(26)可以看出, ELK 从物理链路的延长运行时间、物理链路的开启量以及物理路径的长度这

3 个方面对底层物理路径做出综合评价。 ELK 值越大, 说明路径长度较长、需要开启的物理链路较多或延时较长, 越不适于承载虚拟链路 l'_v ; ELK 值越小, 评价结果越好, 因此, 取多个较短路径中 ELK 值最小的作为承载路径。

3.4 映射算法

现已设计了加入时间因素的节点和链路节能评价标准, 为了便于突出该评价标准的优势, 采用较为简单的算法——贪婪算法。算法中, 在多个可用节点中, 选择概率值最高的作为映射节点; 在 k 最短路径算法^[16]得到的多条最短路径中, 选择 ELK 值最小的作为映射路径。

下面对该算法描述如下。

算法 1 TEAVNE(time and energy aware virtual network embedding)

输入 虚拟网请求拓扑 $G^v(N^v, L^v)$, 底层网络拓扑图 $G^s(N^s, L^s)$;

输出 映射结果 $E(E_N, E_L)$ 。

1) 将虚拟节点按照请求资源量的大小降序排序, 得到虚拟节点映射序列: $n_v^{seq_1}, n_v^{seq_2}, \dots, n_v^{seq_i}, \dots, n_v^{seq_{|N^v|}}, i=1, 2, \dots, |N^v|$

2) For 序列中的 $n_v^{seq_i}$

3) 得到满足 $n_v^{seq_i}$ 资源约束的底层网络候选节点集合 $S = \{n_s^q | n_s^q \in G^s, n_s^q \text{ 未被映射}, n_s^q \text{ 的剩余 CPU 资源足够承载 } n_v^{seq_i}\}$;

4) 如果集合 S 中元素个数为 0, 返回 NODE_MAP_FAILED;

5) 对于 $\forall n_s^q \in S$, 由式(11)计算每个 n_s^q 的 $P(N_q A_j)$ 值, 将 $n_v^{seq_i}$ 映射到 $P(N_q A_j)$ 值最大的节点 n_s^{\max} 上, 即 $E_N(n_v^{seq_i}) = n_s^{\max}$;

6) End for

7) 将虚拟链路按照请求资源的多少降序排序, 得到虚拟链路映射序列: $l_v^{seq_1}, l_v^{seq_2}, \dots, l_v^{seq_i}, \dots, l_v^{seq_{|L^v|}}, i=1, 2, \dots, |L^v|$;

8) For each 序列中的 $l_v^{seq_i}$

9) 获取虚拟链路 $l_v^{seq_i}$ 两端的虚拟节点 $n_v^{e^b}$ 和 $n_v^{e^c}$;

10) 利用 k 最短路径算法求出前 k 条从 $E_N(n_v^{e^b})$ 到 $E_N(n_v^{e^c})$ 的最短路径;

11) 由式(26)计算每条路径的 ELK 值, 将 $l_v^{seq_i}$ 映射到 ELK 值最小的一条路径 P_{\min} 上, 即 $E_L(l_v^{seq_i}) =$

P_{\min} ;

12) 如果最短路径不存在, 返回 EDGE_MAP_FAILED;

13) End for

14) Return MAP_SUCCESS;

k 最短路径算法^[16]的时间复杂度为 $O(|N^v| \log |N^v| + k(|N^v| + |L^v|))$, 其中, $|N^v|$ 和 $|L^v|$ 分别为底层网络的节点个数和链路个数。由上述流程可知, TEAVNE 算法的时间复杂度为 $O(|L^v| |N^v| \log |N^v| + k(|N^v| |L^v| + |L^v| |L^v| + |N^v| |N^v|))$, 其中, $|N^v|$ 和 $|L^v|$ 分别为虚拟网请求的节点个数和链路个数。

4 实验

4.1 实验环境

同文献[17], 这里设置 3 种不同的实验场景分别测试算法的节能效果。实验在双核 CPU 3.4 GHz, 4 GB 内存的 PC 机上运行。底层网络拓扑和虚拟网请求拓扑由 GT-ITM^[18]工具生成。

1) 实验场景 1

将该实验环境标识为中等规模—低集中 (medium-sized & low-concentration), 实验设置同文献[19]。

底层物理网络拓扑: 底层网络共包含 100 个节点, 每两节点间连接的概率为 0.5, 底层节点的 CPU 资源量和底层链路的带宽资源量取值在 [50, 100] 内均匀分布。

虚拟网请求拓扑: 虚拟网请求的到达时刻服从平均 100 个时间单元到达 5 个请求的泊松过程, 持续时间服从均值为 1 000 的指数分布。每个请求的节点个数在 [2, 10] 内均匀分布, 节点连接率为 0.5, 虚拟节点请求的 CPU 资源取值在 [0, 20], 虚拟链路请求的带宽资源取值在 [0, 25] 之间均匀分布, 虚拟网请求共计 2 500 个, 实验共运行 50 000 个左右的时间单元。

2) 实验场景 2

将该实验环境标识为中等规模—中度集中 (medium-sized & moderate-concentration), 实验设置同文献[17, 20~23]。

底层物理网络拓扑: 网络设置同实验场景 1 的物理网络设置。

虚拟网请求拓扑: 虚拟网请求的到达时刻服从平均 100 个时间单元到达 5 个请求的泊松过程, 持续时间服从均值为 1 000 的指数分布, 每个请求的

节点个数在[2,20]内均匀分布，节点连接率为0.5，虚拟节点请求的CPU资源取值在[0,50]，虚拟链路请求的带宽资源取值在[0, 50]之间均匀分布，虚拟网请求共计2 500个，实验共运行50 000个左右的时间单元。

3) 实验场景3

将该实验环境标识为小规模—高集中 (small-sized & high-concentration)，设置同文献[17,22~24]。

底层物理网络拓扑：底层网络共包含50个节点，每两节点间连接的概率为0.5，底层节点的计算资源和底层链路的带宽资源取值在[50,100]内均匀分布。

虚拟网请求拓扑：虚拟网请求的到达时刻服从平均100个时间单元到达4个请求的泊松过程，持续时间服从均值为1 000的指数分布。每个请求的节点个数在[2,10]内均匀分布，节点连接率为0.5，虚拟节点请求的CPU资源取值在[0,20]，虚拟链路请求的带宽资源取值在[0, 50]之间均匀分布。虚拟网请求共计2 000个，实验共运行50 000个左右的时间单元。

4) 其他参数设置

在参数设置上，同文献[25~28]中的设置，得到节点和链路能耗中常数的值为： P_b 为150 W， P_m 为300 W， P_n 为15 W。其他参数设置为 $\varepsilon_1=10^{-6}$ 、 $\varepsilon_2=10^{-9}$ ， $a_1=0.5$ ， $a_2=0.3$ ， $a_3=0.2$ ， $b_1=0.6$ ， $b_2=0.4$ ， $bb_1=0.4$ ， $bb_2=0.4$ ， $bb_3=0.2$ ， $c=0.2$ ， $k=7$ 。

5) 比较算法

TEAVNE算法是以节能为目标分两阶段映射的贪婪算法，为检验加入时间因素的节点和链路评价标准的节能效果，并保证比较的公平性，实验选取2个较经典的同为两阶段映射的贪婪算法作为比较对象，即以最大化收益成本比为目的的贪婪映射算法SP^[20]和同样以节能为目的的贪婪映射算法EAVNE^[8]。

4.2 实验结果分析

4.2.1 对度量指标的定义

1) 能耗度量指标

在能耗度量指标上做如下5个定义。

定义平均节点开启量 (ANON, average number of open nodes) 为

$$ANON = \frac{\sum_{i=1}^{N_T} NO_i}{N_T} \quad (27)$$

其中， N_T 表示在时间单元0到T之间的有效时

间片个数； NO_i 表示有效时间片*i*内节点的开启个数。

定义平均链路开启量 (ANOL, average number of open links) 为

$$ANOL = \frac{\sum_{i=1}^{N_T} NL_i}{N_T} \quad (28)$$

其中， NL_i 表示有效时间片*i*内链路的开启个数。

定义平均节点资源利用率 (AUCPU, average utilization of CPU) 为

$$AUCPU = \frac{\sum_{i=1}^{N_T} NRU_i}{N_T} \quad (29)$$

其中， NRU_i 表示有效时间片*i*内节点资源的CPU利用率。

定义平均链路资源利用率 (AUBW, average utilization of BW) 为

$$AUBW = \frac{\sum_{i=1}^{N_T} LRU_i}{N_T} \quad (30)$$

其中， LRU_i 表示有效时间片*i*内链路资源的带宽利用率。

定义平均能耗量 (AAEC, average amount of energy consumption) 为

$$AAEC = \frac{\sum_{i=1}^{N_T} E^i}{N_T} \quad (31)$$

同前面的定义， E^i 为有效时间片*i*内的底层网络能耗。

从以上定义可以看出，每次平均值的计算都是从起始时刻开始的。

2) 资源使用情况的度量指标

虚拟网映射算法一方面要满足服务提供商的构建需求，另一方面需要高效地使用基础设施提供商的物理网络资源，因此可用虚拟网映射的请求接受率和收益成本比来衡量映射算法的资源使用情况。请求接受率用来衡量虚拟网请求的映射成功率，收益成本比用来衡量已映射虚拟网的资源占用情况。请求接受率和收益成本比从不同角度反映算法对资源的使用情况，其值越高，映射算法对资源的使用越充分。这里用收益成本比来衡量资源的使

用情况。

对平均收益成本比 (ARRC, average ratio of revenue and cost) 定义如下

$$ARRC = \frac{\sum_{i=1}^{A_T} Revenue^i}{\sum_{i=1}^{A_T} Cost^i} \quad (32)$$

其中, A_T 表示从起始时刻到时间 T 的时间段内接受的虚拟网请求数量, $Revenue^i$ 和 $Cost^i$ 分别表示接受的虚拟网请求 i 的收益和成本, 对其分别定义如下

$$Revenue^i = \sum_{n_v \in N_v^i} CPU(n_v) + \sum_{l_v \in L_v^i} BW(l_v) \quad (33)$$

$$Cost^i = \sum_{n_s \in E_N(N_v^i)} CPU(n_s) + \sum_{l_v \in L_v^i} \sum_{l_s \in E_L(l_v)} BW(l_s) \quad (34)$$

由上式可以看出, 在映射某个虚拟网请求时, 收益成本比只和物理路径的长度有关, 映射的物理路径越短, 收益成本比越高。

3) 映射时间长度的度量指标

定义平均映射时间(AMT, average mapping time)为

$$AMT = \frac{\sum_{i=1}^{A_T} MT^i}{A_T} \quad (35)$$

其中, MT^i 为映射虚拟网请求 i 所用的时间。

4.2.2 对实验场景 1 的结果分析

在实验场景 1 (medium-sized & low-concentration) 中, 各算法的请求接受率均为 100%, 此时只需要比较各算法的节能情况。图 2 给出了在该场景下 TEAVNE 算法、EAVNE 算法和 SP 算法的映射结果。

本文的 TEAVNE 算法降低了底层网络节点和链路的开启数量, 很大程度地减少了底层网络能耗。图 2(a)~2(c)展示了节点和链路的平均开启量以及底层网络的平均能耗量随时间变化的情况。图中表明了, 在开启的节点数量和能耗量上, TEAVNE 算法较 SP 和 EAVNE 有较为明显的优势。这是由于 TEAVNE 算法在每次映射时不仅考虑本次映射的节能效果, 在节点和路径的选择标准中, 还加入了延长节点 (或链路) 的时长等因素, 考虑了对后续映射的节能效果, 从而使总体的节能效果更好。

TEAVNE 算法提高了节点和链路的资源利用

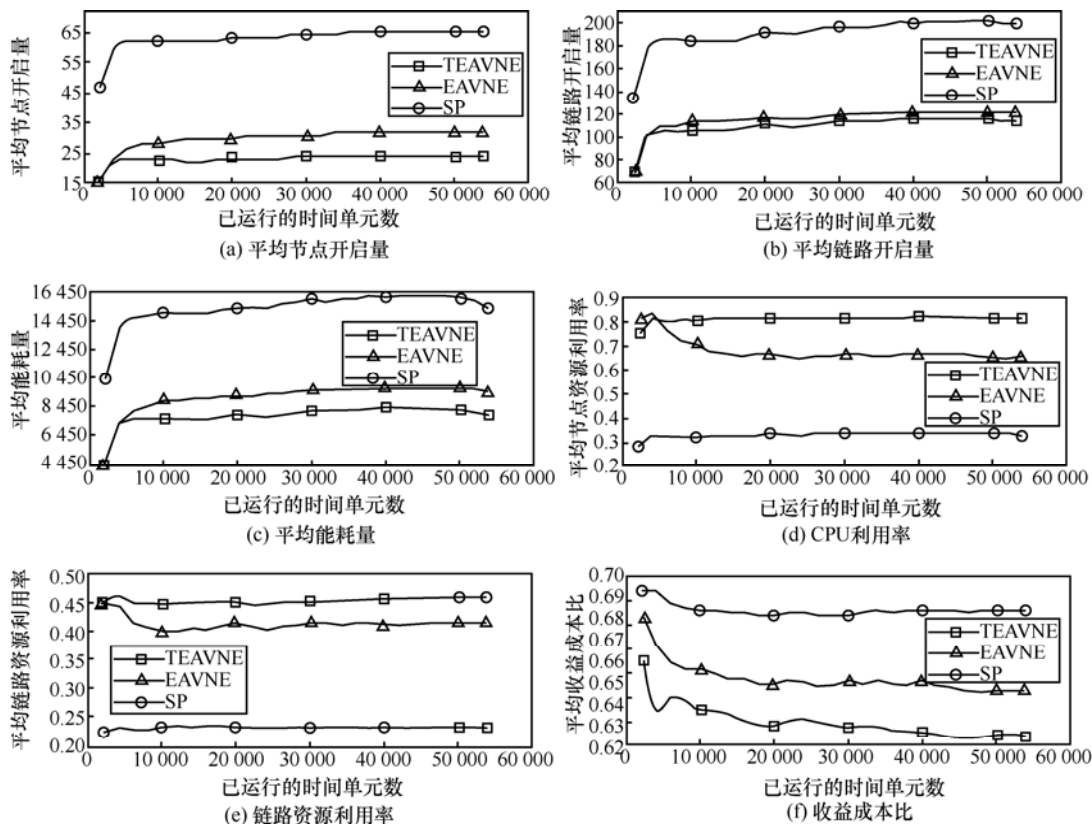


图 2 实验场景 1 的映射结果

率。图 2(d)、图 2(e)展示了节点的 CPU 资源利用率和链路的带宽资源利用率随时间变化的情况。图中表明了 在提高资源利用率方面, TEAVNE 算法较 SP 和 EAVNE 有较明显的优势。在该场景下, 各算法所接受 的请求完全相同, 节能算法的目标是开启尽可能少的节点和链路, 即要在更少的节点和链路集合上满足相同的虚拟网请求, 从而使节能效果越好, 资源利用率越高。

TEAVNE 算法较 EAVNE 算法和 SP 算法降低了收益成本比。图 2(f)展示了平均收益成本比随时间变化的情况。图中表明了 在最大化收益成本比这一目标上, TEAVNE 算法较 SP 和 EAVNE 算法有较差的效果。

根据式(32)~式(34), 对于相同的虚拟网请求, 收益成本比的大小只与虚拟链路映射到的物理路径长度(即虚拟链路的长度)有关。虚拟链路长度越长, 收益成本比越小。在节能映射的初始阶段,

将开启较小的物理链路集合, 使部分物理链路的资源利用率过高, 从而无法继续承载虚拟链路, 这样的物理链路越多, 虚拟链路将越长, 甚至无法映射成功。因此, 在接受相同虚拟网请求的情况下, 算法的节能效果越好, 收益成本比越低。

4.2.3 对实验场景 2 的结果分析

在实验场景 2 (medium-sized & moderate-concentration) 中, 各算法的请求接受率不再是 100%。图 3 给出了该场景下 TEAVNE 算法、EAVNE 算法和 SP 算法的映射结果。

在请求接受率不再为 100%的场景下, 节能算法 TEAVNE 依然有较好的节能效果; 和另一节能算法 EAVNE 一样, 其平均收益成本比较低。

从图 3(a)~图 3(e)可以看出, 3 种算法中, TEAVNE 算法有最好的节能效果。

在收益成本比上, 从图 2(f)和图 3(f)的对比中可以看出, 请求接受率的下降伴随着收益成本比的

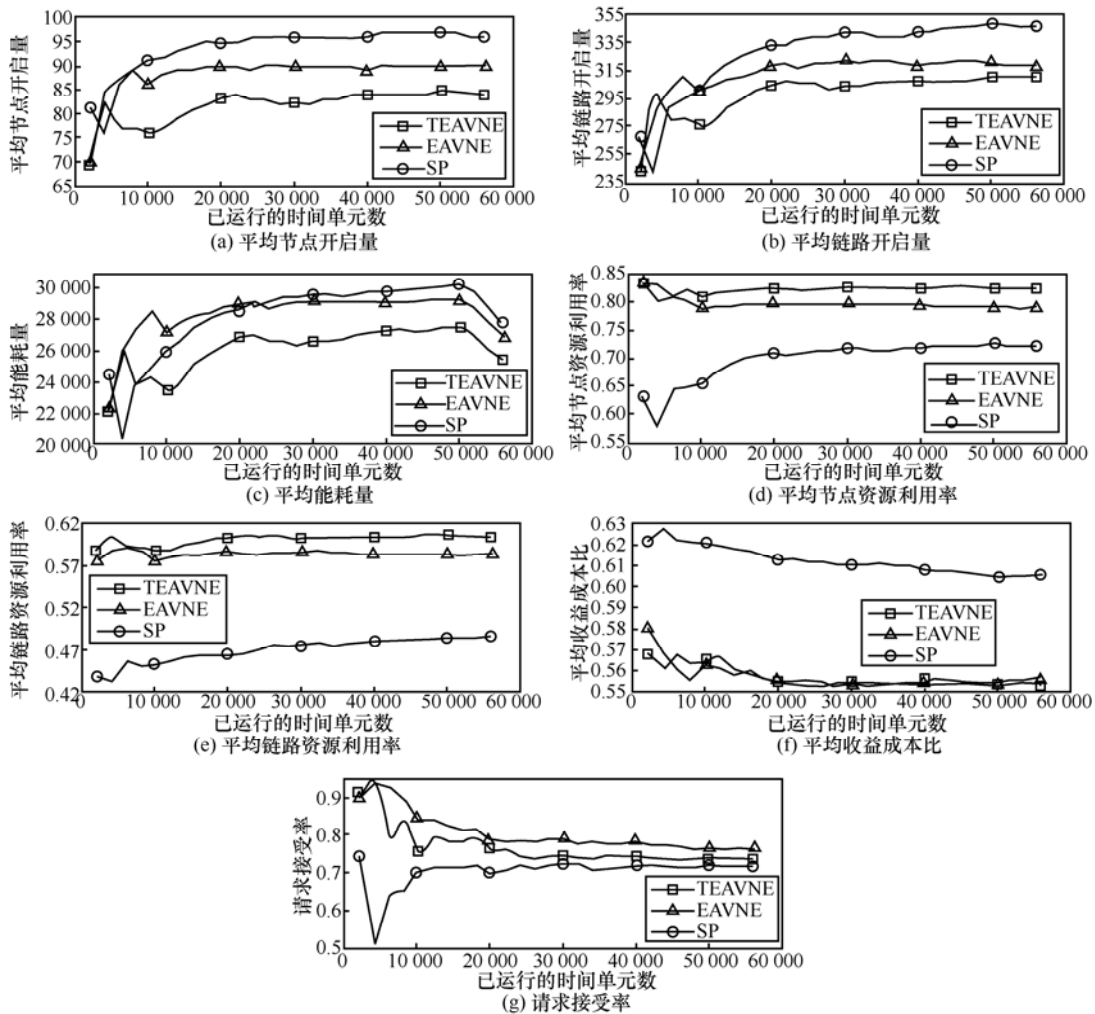


图 3 实验场景 2 的映射结果

下降。其中，SP 算法的收益成本比下降幅度较小，TEAVNE 和 EAVNE 算法的收益成本比下降幅度较大。这是由于节能算法集中映射的方式使部分链路的利用率过高，平均链路利用率越高（对比图 2(e) 和图 3(e)），链路利用率过高的物理链路越多，虚拟链路的长度越长，从而使节能算法的收益成本比下降的幅度越大。

3 种算法的请求接受率差距越来越小。从图 3(g) 中可以看出，SP 算法的请求接受率随时间的变化不大，而节能算法 TEAVNE 和 EAVNE 的请求接受率随时间增长的下降趋势较为明显。在运行 5 000 个左右的时间单元后，三者的平均请求接受率相差无几。

这是由于集中映射的方式使部分节点或链路的资源利用率较高，节点或链路的负载越不均衡，请求接受率越低。

4.2.4 对实验场景 3 的结果分析

在实验场景 3（small-sized & high-concentration）中，各算法的请求接受率较低。图 4 给出了该场景下 TEAVNE 算法、EAVNE 算法和 SP 算法的映射结果。

两节能算法 TEAVNE 和 EAVNE 较 SP 算法的节能效果更加明显，TEAVNE 算法较 EAVNE 算法在后期有更优的节能效果。从图 4(a)~图 4(c)可以看出，TEAVNE 算法较 EAVNE 算法在后期有更优的节能效果。两节能算法使节点或链路的开启更加集中，节能效果更加显著。由于 TEAVNE 算法中加入了时间因素，纳入了对后续映射的节能效果的考虑，从而在后期，TEAVNE 算法较 EAVNE 算法有更优的节能效果。

两节能算法较 SP 算法的收益成本比更低。从

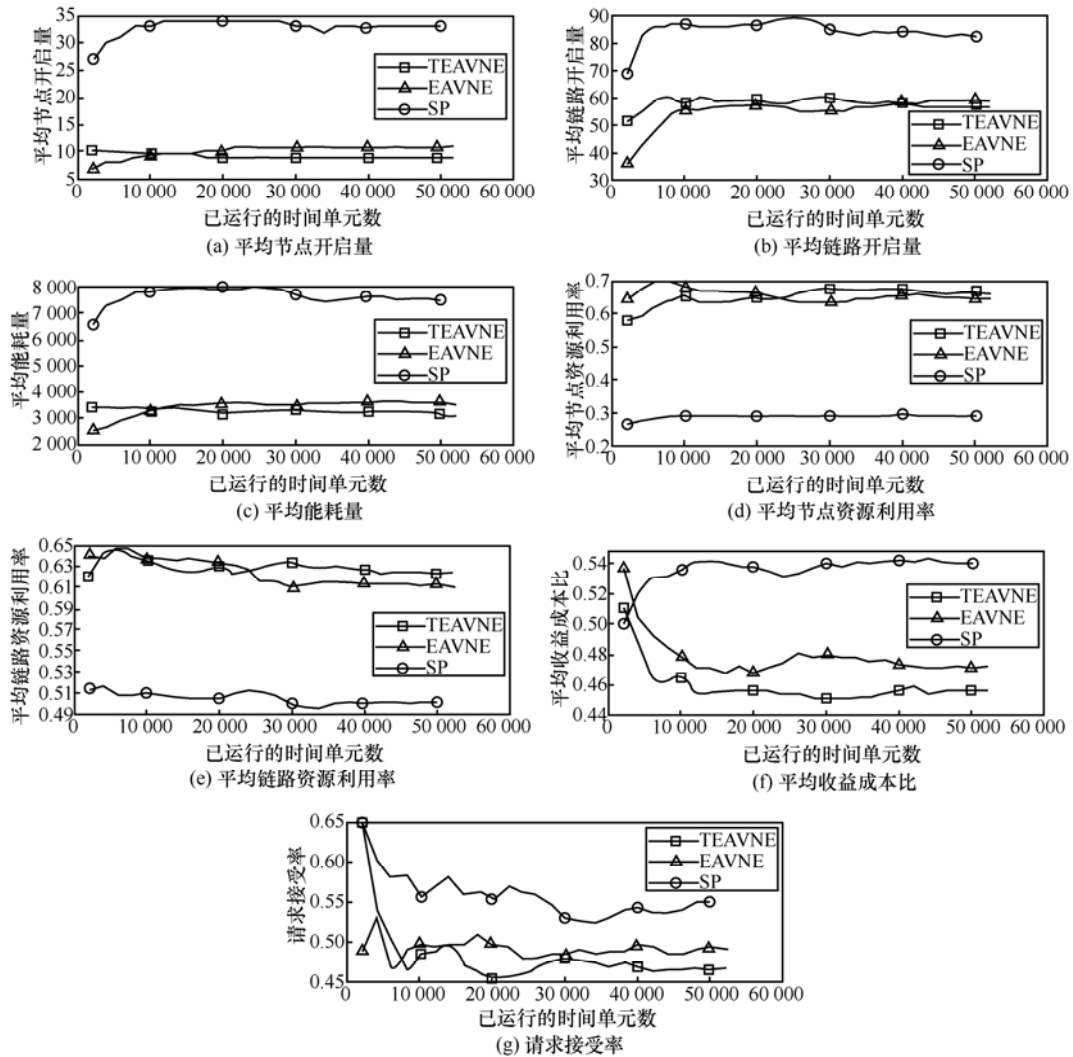


图 4 实验场景 3 的映射结果

图 2(f)、图 3(f)和图 4(f)可以看出，随着请求接受率的下降，两节能算法的收益成本比下降更为显著。这是由于集中映射的策略使有些链路的资源利用率过高，无法承载虚拟链路。物理链路的平均资源利用率越高（对比图 2(e)、图 3(e)和图 4(e)），链路利用率过高的链路越多，虚拟链路的长度越长，从而降低了收益成本比。

节能算法的请求接受率较低。从图 4(g)可以看出，节能算法较 SP 算法的请求接受率较低。其中，TEAVNE 算法的请求接受率最低。这是由于集中映射的策略使有些节点或链路的资源利用率过高，无法承载虚拟节点或虚拟链路。节点或链路的平均资源利用率越高，节点或链路的利用率过高的节点或链路越多，不均衡的现象越为严重，从而降低了请求接受率。

4.2.5 算法映射时间的比较

TEAVNE 算法与同为贪婪算法的 SP 算法和 EAVNE 算法运行时间相当。

实验中，底层网络分别在 3 种拓扑环境下运行所用的平均映射时间如图 5 所示。

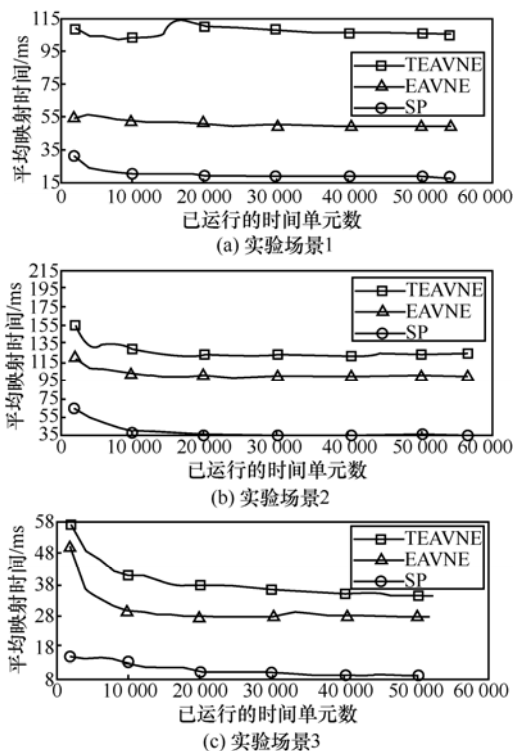


图 5 各时间段的平均映射时间

可以看出，同为贪婪算法的 TEAVNE 算法增加了运行时间，这是因为其在节点选择时增加了对每个节点选择概率的计算。但从图中也可以看出，平

均运行时间仍保持在毫秒级，且增加量比较小。

5 结束语

本文研究了网络虚拟化环境下的系统能耗问题，根据节能运行的实际需要，设计了节能模型，然后借助条件概率理论辅助分析，在对节点和链路的节能评价中加入时间因素，最终设计了时间和能量感知的虚拟网映射算法，把虚拟网映射在一个较小的节点和链路集合中，提高关闭的节点和链路数量，以实现高效节能。实验结果表明了 TEAVNE 算法能够有效降低底层网络的能量消耗，由于节能算法集中映射的特点，在资源请求量较底层网络的资源剩余量较多的情况下，节能算法会降低请求接受率。于是，以节能为目标的映射是有一定的适用场景的，在确定的场景下，如何平衡节能和提高请求接受率 2 个相对矛盾的目标，以最大化商家的总收益，将是下一步有待深入研究的问题。

参考文献：

- [1] CHUN B, IANNACCONE G, IANNACCONE G, et al. An energy case for hybrid datacenters [J]. *ACMSIGOPS Operating Systems Review*, 2010,44 (1): 76-80.
- [2] APC American Power Conversion. Determining total cost of ownership for data center and network room infrastructure[EB/OL]. <http://www.apcmedia.com/salestools/CMRP-5T9PQG\R4EN.pdf>, 2003.
- [3] QURESHI A, WEBER R, BALAKRISHNAN H, et al. Cutting the electric bill for internet-scale systems[C]//The ACM SIGCOMM 2009 Conference on Data Communication. Barcelona, Spain, c2009: 123-134.
- [4] China Mobile Research Institute[EB/OL]. http://v7vjw.greentouch.org/uploads/documents/Chih-Lin_I%20-%20TIA%20Green%20from%20a%20Service%20Provider%20Perspective.pdf, 2012.
- [5] FISHER W, SUCHARA M, REXFORD J. Greening backbone networks: reducing energy consumption by shutting off cables in bundled links[C]//The 1st ACM SIGCOMM Workshop on Green Networking. New Delhi, India, c2010: 29-34.
- [6] 林闯, 田源, 姚敏. 绿色网络和绿色评价:节能机制、模型和评价[J]. *计算机学报*, 2011,34(4):593-612.
- [7] 叶可江,吴朝晖,姜晓红,等.虚拟化云计算平台的能耗管理[J]. *计算机学报*, 2012,35(6):1262-1285.
- [8] BOTERO J F, HESSELBACH X, DUELLI M, et al. Energy efficient virtual network embedding[J]. *IEEE Communications Letters*, 2012, 16(5): 756-759.
- [9] MELO M, SARGENTO S, et al. Optimal virtual network embedding:

- energy aware formulation[J]. Computer Networks, 2015, 91(C): 184-195.
- [10] TRIKI N, KARA N, BARACHI M E, et al. A green energy-aware hybrid virtual network embedding approach[J]. Computer Networks the International Journal of Computer & Telecommunications Networking, 2015, 91(C):712-737.
- [11] GUAN X J, CHOI B Y, SONG S. Energy efficient virtual network embedding for green data centers using data center topology and future migration[J]. Computer Communications, 2015, 69: 50-59.
- [12] CHEN X H, LI C Z, JIANG Y L. Optimization model and algorithm for energy efficient virtual node embedding[J]. IEEE Communications Letters, 2015, 19(8): 1327-1330.
- [13] NONDE L, EL-GORASHI T E H, ELMIRGHANI J M H. Energy efficient virtual network embedding for cloud networks[J]. Journal of Lightwave Technology, 2015, 33(9): 1828-1849.
- [14] HOUIDI I, LOUATI W, ZEGHLACHE D. Exact multi-objective virtual network embedding in cloud environments[J]. Computer Journal, 2015, 58(3): 403-415.
- [15] 陈晓华,李春芝,陈良育,等.主动休眠节点链路的高效节能虚拟网络映射[J]. 软件学报, 2014, 25(7): 1416-1431.
CHEN X H, LI C Z, CHEN L Y, et al. Energy efficient virtual network embedding based on actively hibernating substrate nodes and links[J]. Journal of Software, 2014, 25(7): 1416-1431.
- [16] EPPATEIN D. Finding the k shortest paths[J]. SIAM Journal on Computing, 1998,28(2):652-673.
- [17] CHANG X L, MI X M, MUPPALA J K. Performance evaluation of artificial intelligence algorithms for virtual network embedding[J]. Engineering Applications of Artificial Intelligence, 2013, 26(10): 2540-2550.
- [18] ZEGURA E, CALVERT K, BHATTACHARJEE S. How to model an Internet work[C]//IEEE INFOCOM '96. Conference on Computer Communications. San Francisco, CA, USA, c1996: 594-602.
- [19] 江逸茗,兰巨龙,程东年,等.网络虚拟化环境中面向服务聚合的映射算法[J]. 软件学报, 2014, 25(6): 1328-1338.
JIANG Y M, LAN J L, CHENG D N, et al. Mapping algorithm for service aggregation in network virtualization [J]. Journal of Software, 2014,25(6): 1328-1338.
- [20] YU M, YI Y, REXFORD J, et al. Rethinking virtual network embedding: substrate support for path splitting and migration[J]. ACM SIGCOMM Computer Communication Review, 2008,38(2):19-29.
- [21] CHENG X, SU S, ZHANG Z B, et al. Virtual network embedding through topology awareness and optimization [J]. Computer Networks, 2012, 56(6): 1797-1813.
- [22] CHOWDHURY N M M K, RAHMAN M R, BOUTABA R. ViNEYard: virtual network embedding algorithms with coordinated node and link mapping [J]. IEEE/ACM Transactions on Networking, 2012, 20(1): 206-219.
- [23] MI X M, CHANG X L, LIU J Q, et al. Embedding virtual infrastructure based on genetic algorithm[C]//2012 13th International Conference on Parallel and Distributed Computing Applications and Technologies(PDCAT). Beijing, China, c2012: 239-244.
- [24] MOSHARAF N M, CHOWDHURY K, RAHMAN M R, et al. Virtual network embedding with coordinated node and link mapping[C]//28th Conference on Computer Communications(IEEE INFOCOM 2009). Rio de Janeiro, Brazil, c2009: 783-791.
- [25] LU G, GUO C, LI YL, et al. Serverswitch: a programmable and high performance platform for datacenter networks[C]//The 8th USENIX Conference on Networked Systems Design and Implementation. Berkeley: USENIX Association, c2011: 1-14.
- [26] SIVARAMANV, VISHWANATH A, ZHAO Z, et al. Profiling per-packet and per-byte energy consumption in the NetFPGA Gigabit router[C]//2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS 2011). Shanghai, China, c2011: 331-336.
- [27] UNNIKIRISHNAN D, VADLAMANI R, LIAO Y, et al. Scalable network virtualization using FPGAs[C]//18th ACM SIGDA International Symposium on Field-Programmable Gate Arrays(FPGA'10). c2010: 219-228.
- [28] BARROSO L A, HOLZLE U. The datacenter as a computer: an introduction to the design of warehouse-scale machines[J]. Synthesis Lectures on Computer Architecture, 2009, 6: 1-107.

作者简介:



胡颖 (1982-), 女, 河南商丘人, 郑州大学博士生, 主要研究方向为未来网络、网络虚拟化、虚拟网映射等。



庄雷 (1963-), 女, 山东日照人, 郑州大学教授、博士生导师, 主要研究方向为未来网络、网络虚拟化等。



陈鸿昶 (1964-), 男, 河南郑州人, 国家数字交换系统工程技术研究中心教授、博士生导师, 主要研究方向为未来网络、网络安全等。



马丁 (1978-), 男, 河南郑州人, 郑州大学博士生, 河南工业大学讲师, 主要研究方向为未来网络、网络功能虚拟化等。