

基于自适应协同进化粒子群算法的虚拟网节能映射研究

胡颖^{*①} 庄雷^① 兰巨龙^② 马丁^{①③}

^①(郑州大学信息工程学院 郑州 450000)

^②(解放军信息工程大学国家数字交换系统工程技术研究中心 郑州 450002)

^③(河南工业大学信息科学与工程学院 郑州 450000)

摘要: 该文针对虚拟网节能映射问题提出自适应的协同进化粒子群算法。首先,为虚拟网节能映射问题设置了聚合度,该聚合度被用于自适应地选择粒子的搜索方式,即随机搜索、种内搜索或种外搜索。其次,根据粒子群的进化结果,自适应地确定是否终止对子群的搜索。最后,在常用的测试环境下进行了仿真实验,对映射的能耗效果对比了结果,实验结果表明了所提算法的高效性。

关键词: 虚拟网节能映射; 协同进化; 自适应算法; 粒子群算法

中图分类号: TP311

文献标识码: A

文章编号: 1009-5896(2016)10-2660-07

DOI: 10.11999/JEIT151434

Energy Aware Virtual Network Embedding Using Particle Swarm Optimization Algorithm Based on Adaptive Cooperative Coevolution

HU Ying^① ZHUANG Lei^① LAN Julong^② MA Ding^{①③}

^①(School of Information Engineering, Zhengzhou University, Zhengzhou 450000, China)

^②(National Digital Switching System Engineering & Technology Research Center, The PLA Information Engineering University, Zhengzhou 450002, China)

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

Abstract: A novel adaptive co-evolutionary particle swarm optimization algorithm is presented for energy aware virtual network embedding problem. The polymerization degree is designed, which is used to adaptively select searching method, namely variation search, internal search or external search. Second, the algorithm adaptively determine whether to terminate the searching process of particle swarm according to the evolution result. Moreover, extensive simulation under common test environment compares results in energy consumption performing goal, and the results indicate the efficiency of the proposed algorithm.

Key words: Energy aware virtual network embedding; Co-evolution; Adaptive algorithm; Particle swarm optimization algorithm

1 引言

当前互联网自身结构“僵化”,可扩展性差,新功能的增加和新业务的部署非常困难;以及 OTT (Over The Top)业务飞速发展给网络运营商带来了巨大的挑战。种种迹象表明,需要一种革命式的新

型互联网体系结构。网络虚拟化技术是形成新型互联网体系结构的重要手段^[1]。网络虚拟化屏蔽了物理层的实现细节,在技术上,为在物理网络上运行多样化的协议和异构的应用提供了可能,使网络具有更好的开放性和灵活性;在运行维护上,使得物理网络具有了弹性特征,这对网络故障恢复、网络负载均衡都具有较大意义;在经济上,通过共享底层基础设施避免了服务提供商对基础设施的重复购买,并降低了能耗和运维成本,提高了资源的利用率。网络虚拟化既可以应用在未来互联网的中心,也可以应用在互联网的边缘,如数据中心^[2],移动网络^[3]等。因此,网络虚拟化是未来互联网、云计算和软件定义网络的重要技术^[1]。该技术通过整合网络基

收稿日期: 2015-12-17; 改回日期: 2016-07-01; 网络出版: 2016-08-26

*通信作者: 胡颖 huying_yx@126.com

基金项目: 国家 973 计划(2012CB315901), 国家自然科学基金(61379079), 河南省科技厅攻关项目(122102210042)

Foundation Items: The National 973 Program of China (2012CB315901), The National Natural Science Foundation of China (61379079), The Science and Technology Key Project of Henan Province (122102210042)

基础设施资源，能够合理有效地使用能量，使得智能能量感知的网络部署成为可能。

随着电力成本的不断上涨，商家已经意识到能耗管理的重要性。当前网络为高峰负荷而设计，资源的超量供给确保了网络的正常运行，但也导致了资源利用率低下。据统计，大型 ISP 骨干网的平均链路利用率大约为 30%~40%^[4]，数据中心服务器的平均利用率为 11%~50%^[5,6]，过低的资源利用率造成了电能的巨大浪费，促使绿色网络研究的兴起，网络能耗问题成为研究的热点^[7,8]。虚拟网映射技术根据虚拟网请求的拓扑结构及节点、链路的资源需求，分配底层网络资源构建虚拟网，因此，该技术是网络虚拟化技术实施的基础和关键。以节能为目标的虚拟网映射应在满足当前虚拟网请求的前提下，最小化能耗，如何在这种场景下对虚拟网请求进行以节能为目标的映射，成为降低系统能耗以减少商家开销的关键问题。

当前已有一些学者对虚拟网节能映射问题进行了很有意义的研究。文献[9]从节能和资源最小化两个角度对虚拟网映射问题建立了整数线性规划模型，并分别以其一为主要目标，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两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。

在映射算法上，对文献[15]的算法根据虚拟网映射问题的特点进行改进，并针对虚拟网映射这个具有离散解空间的混合整数规划问题，设计出多种群的自适应粒子群算法。该算法为避免粒子群算法自身的缺陷，比如易造成早熟停滞等问题，加入了随机搜索方式、种群内部精英指导搜索方式、种群外部的精英粒子指导搜索方式以及在多种搜索方式(包括粒子群固有的)间自适应切换等因素。最终提出了自适应的协同进化粒子群算法(Particle Swarm Optimization Algorithm based on Adaptive Cooperative Coevolution, PSOAACC)。实验结果表明，PSOAACC 算法在节能映射上有较好的性能。

2 节能映射模型与评价标准

2.1 虚拟网节能映射模型

虚拟网节能映射的目标是最小化能耗量，即使得映射和运行每个虚拟网请求所增加的能耗量减至最低。由文献[16]中底层网络的节点和链路能耗模型，设置虚拟网节能映射的目标：

$$\min(P_l \times \text{CPUUtil}(j) + P_b \times N_w + P_n \times N_{lk}) \quad (1)$$

在式(1)中， P_b 为节点的基本能耗，令 P_m 为节点的最大能耗，且 $P_l = P_m - P_b$ ； P_n 为开启链路的单位能耗。 N_w 为映射当前虚拟网需要开启的物理节点个数； N_{lk} 为映射当前虚拟网需要开启的物理链路个数。 $\text{CPUUtil}(j)$ 为物理节点 j 请求的 CPU 资源利用率。

约束：虚拟网节能映射的约束同文献[16]，包括容量约束、传输约束、一个虚拟节点只能映射到一个物理节点以及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。

2.2 节能映射评价标准

在选择节点和路径时，为了更有效地利用资源，除了节能这一主要目标，加入对最小化资源量(即最大化收益成本比)的考虑。

2.2.1 节点评价标准 应优先选择已被开启，剩余资源量较大，且 CPU 资源能力较大的节点作为宿主节点。定义节点节能度 ω ， ω 值越大，节点被选中的概率越高。

$$\omega(n_v, n_s) = d_1 \times \left(\text{CPURem}(n_s) + \sum_{l_{jk} \in L(n_s)} (\text{BWRem}(l_{jk})) + d_2 \times \text{CPU}(n_s) \right) \quad (2)$$

式中， d_1 为系数，令物理节点 n_s 已被开启或尚未开启的 d_1 取值不同； d_2 为常系数。 $L(n_s)$ 为与物理节点 n_s 相连的物理链路集合。 $\text{BWRem}(l_{jk})$ 为物理链路 l_{jk} 的剩余带宽资源量； $\text{CPU}(n_s)$ 为物理节点 n_s 的 CPU 资源总量； $\text{CPURem}(n_s)$ 为物理节点 n_s 的剩余 CPU 资源量。

2.2.2 链路评价标准 在路径选择时，要兼顾物理路径长度和物理路径中物理链路的开启量。本文通过定义路径耗能量 θ 来对虚拟链路 l_v 的可用物理路径 p 进行评价， θ 值越小，对 p 的评价越好。

$$\theta(l_v, p) = a_1 \times \text{len}(p) + a_2 \times N_{lk} \quad (3)$$

式中， a_1 和 a_2 是比例系数； $\text{len}(p)$ 为物理路径 p 的长度。

3 自适应的协同进化粒子群算法

3.1 适应度函数

在式(1)中给出了虚拟网节能映射的目标函数, 函数值越小, 节能效果越好, 该问题是最小化问题, 于是可直接设置目标函数为适应度函数, 即

$$\text{fitness} = P_1 \times \text{CPUUtil}(j) + P_b \times N_w + P_n \times N_{lk} \quad (4)$$

3.2 自适应协同进化粒子群算法的实现

PSOAACC 算法模型分上下两层, 底层为多个子种群, 高层为精英粒子。底层各子种群采用具有合作机制的自适应的粒子群算法分别进化。在进化过程中, 从底层的多个种群中提出适应度值最低的粒子组成高层精英粒子, 精英粒子连同自身的历史最优粒子参与对底层各子种群的指导操作, 即种间的指导搜索; 若子种群使用本群的历史最优粒子以及自身粒子的历史最优粒子作为指导搜索, 即为种内的指导搜索; 另外, 为了尽量避免种群陷入局部最优, 这里使粒子适时进行随机搜索。

3.2.1 子种群的聚集程度 子种群中粒子各分量上取值的集中程度决定了子种群的聚集程度, 且各分量上取值越多样化, 子种群越分散。

定义子种群 k 在位置 j 上的聚集度 H_j^k 为

$$H_j^k = \sum_{i=1}^{UC_j} \left(C_j^i - \left\lfloor \frac{tc}{UC_j} \right\rfloor \right)^2 \quad (5)$$

其中, UC_j 为位置 j 上可用符号数, C_j^i 为位置 j 上符号 i 出现的次数, tc 为子种群的个体数。

子种群 k 的聚集度 H_k 可取各位置分量聚集度的算术平均值:

$$H_k = \sum_{j=1}^{|N^s|} H_j^k / |N^s| \quad (6)$$

其中, $|N^s|$ 为个体向量的长度, 即虚拟网请求的节点数目。

为衡量子种群某时刻的聚集程度, 可将其与聚集程度的极端情况相比较。子种群的两个极端情况分别是所有个体都相同和个体各分量均匀地分散。

当均匀分散时, C_j^i 要么取值为 $\left\lfloor \frac{tc}{UC_j} \right\rfloor$, 要么取值为

$\left\lceil \frac{tc}{UC_j} \right\rceil$ 。其中, 取值为 $\left\lfloor \frac{tc}{UC_j} \right\rfloor$ 的个数为

$$c_1 = tc - UC_j \times \left\lfloor \frac{tc}{UC_j} \right\rfloor \quad (7)$$

C_j^i 取值为 $\left\lceil \frac{tc}{UC_j} \right\rceil$ 的个数为

$$c_2 = UC_j - c_1 = UC_j - tc + UC_j \times \left\lfloor \frac{tc}{UC_j} \right\rfloor \quad (8)$$

根据式(5), 式(7)和式(8), 得到子种群 k 的位置 j 在均匀分散情况下聚集度为

$$\begin{aligned} \min H_j^k &= \sum_{i=1}^{UC_j} \left(C_j^i - \left\lfloor \frac{tc}{UC_j} \right\rfloor \right)^2 = 0 + 1 \times c_1 \\ &= tc - UC_j \times \left\lfloor \frac{tc}{UC_j} \right\rfloor \end{aligned} \quad (9)$$

当所有位置都均匀分散时, 由式(6)和式(9), 得到子种群 k 在均匀分散情况下的聚集度为

$$\begin{aligned} \min H_k &= \frac{\sum_{j=1}^{|N^s|} H_j^k}{|N^s|} = \frac{\sum_{j=1}^{|N^s|} \min H_j^k}{|N^s|} \\ &= \frac{\sum_{j=1}^{|N^s|} \left(tc - UC_j \times \left\lfloor \frac{tc}{UC_j} \right\rfloor \right)}{|N^s|} \\ &= tc - \frac{\sum_{j=1}^{|N^s|} \left(UC_j \times \left\lfloor \frac{tc}{UC_j} \right\rfloor \right)}{|N^s|} \end{aligned} \quad (10)$$

当子种群 k 的所有个体都相同时, 可得到此时分量 j 的聚集度 $\max H_j^k$ 值为

$$\begin{aligned} \max H_j^k &= \sum_{i=1}^{UC_j} \left(C_j^i - \left\lfloor \frac{tc}{UC_j} \right\rfloor \right)^2 = \left(tc - \left\lfloor \frac{tc}{UC_j} \right\rfloor \right)^2 \\ &\times 1 + \left(\left\lfloor \frac{tc}{UC_j} \right\rfloor \right)^2 \times (UC_j - 1) = (tc)^2 \\ &- 2 \times tc \times \left\lfloor \frac{tc}{UC_j} \right\rfloor + UC_j \times \left(\left\lfloor \frac{tc}{UC_j} \right\rfloor \right)^2 \end{aligned} \quad (11)$$

此时子种群 k 的聚集度 $\max H_k$ 为

$$\begin{aligned} \max H_k &= \frac{\sum_{j=1}^{|N^s|} H_j^k}{|N^s|} = \frac{\sum_{j=1}^{|N^s|} \max H_j^k}{|N^s|} = (tc)^2 - 2 \\ &\times tc \times \frac{\sum_{j=1}^{|N^s|} \left\lfloor \frac{tc}{UC_j} \right\rfloor}{|N^s|} + \frac{\sum_{j=1}^{|N^s|} UC_j \times \left(\left\lfloor \frac{tc}{UC_j} \right\rfloor \right)^2}{|N^s|} \end{aligned} \quad (12)$$

将子种群 k 的聚集度 H_k' 归一化为

$$H_k' = \frac{H_k - \min H_k + \varepsilon_1}{\max H_k - \min H_k + \varepsilon_2} \quad (13)$$

其中, ε_1 和 ε_2 为两个相近的极小正数, $\varepsilon_1 < \varepsilon_2 \ll 1$ 。

H_k' 随 H_k 的增大而增大, H_k 随聚集程度的增大而增大。从而可根据 H_k' 的值判断聚集程度, 其值越大越聚集, 越应使子种群向着扩散的方向进化。

3.2.2 子种群种内外学习的方式 已知, 当子种群的聚集程度较大时, 应增大子种群的学习范围, 向着

扩散的方向进化, 因此, 此时应进行种间指导搜索; 否则, 应进行种内指导搜索。设置阈值 H_{th} , 若 $H'_i \geq H_{th}$, 进行种间学习; 否则, 进行种内学习。

进行种内学习时, 要使用种内最优个体指导粒子的方向。对分量 j 调整的概率 $p_1^{i,j}$:

$$p_1^{i,j} = \beta_1 \times p_0^{i,j} + \beta_2 \times x_{hb}^{i,j} + \beta_3 \times x_{gb}^{i,j} \quad (14)$$

其中, β_1, β_2 和 β_3 为权重, 满足 $\beta_1 \leq \beta_2 \leq \beta_3$, $\beta_1 + \beta_2 + \beta_3 = 1$; $p_0^{i,j}$ 为粒子 i 在分量 j 的上次的调整概率; 将粒子 i 的历史最优位置 hb 与粒子 i 的当前位置在分量 j 上比较, 相同时将 $x_{hb}^{i,j}$ 取值 0, 相异时将 $x_{hb}^{i,j}$ 取值 1; 将群内最优粒子 gb 与当前粒子 i 在分量 j 上比较, 相同时将 $x_{gb}^{i,j}$ 取值 0, 相异时将 $x_{gb}^{i,j}$ 取值 1。

种间学习时, 需要使用种间的精英粒子(即整体的最优粒子)对粒子方向的指导。于是, 对种群 k 中粒子 i 的分量 j 调整的概率 $p_1^{i,j}$ 为

$$p_1^{i,j} = \beta_1 \times p_0^{i,j} + \beta_2 \times x_{hb}^{i,j} + \beta_3 \times x_{el}^{i,j} \quad (15)$$

其中, $x_{el}^{i,j}$ 表示精英个体 el 与粒子 i 在分量 j 上比较, 相异取 1, 相同取 0。

3.2.3 随机搜索的方式 根据式(2)为每个可用物理节点计算出节能度, 和文献[15]中对粒子随机初始化的方式类似, 根据节能度得到依次增大的权重和。生成随机数, 随机数落到哪个区间即选择那个物理节点作为映射节点。这样, 节能度大的节点具有较高的选中概率。

3.2.4 确定粒子将进行随机搜索或学习搜索 当聚集程度较大, 应增加随机搜索个体的数量, 使种群向着扩散的方向进化。可定义随机搜索的个体数为

$$tcv = \gamma \times H'_k \times tc \quad (16)$$

式中, γ 为常系数。将 tc 个个体按适应度值升序排序, 设置适应度值较大的后 tcv 个个体为将进行随机搜索的个体, 剩余 $tcl = tc - tcv$ 个个体为将进行学习搜索的个体。

3.2.5 种群进化搜索终止条件 若子种群不再进化, 即连续 η 次没有更新子种群的最优个体, 说明结果已接近最优值, 或难以跳出局部优值, 此时, 可终止对该子种群的进化搜索。当所有子种群都被停止时, 便可终止整个种群的进化搜索。为避免出现搜索时间过长的情况。

另外, 可设置一个最大迭代次数 n_{max} , 依据经验并根据解向量的长度提前终止对整个种群的进化搜索。

3.3 自适应的协同进化粒子群算法流程

以上分析了 PSOAACC 算法的主要思想, 下面就具体算法实现流程描述如表 1。

子程序 $sub_ep(k, t, if_terminate_k)$ 负责子种群 k 的进化, 最终搜索出满足资源约束且使适应度

表 1 自适应的协同进化粒子群算法流程

算法 1 PSOAACC

输入: 底层网络 $G^s(N^s, L^s, A^s, A^l)$, 虚拟网络 $G^v(N^v, L^v, R^v, R^l)$

输出: 映射方案 f (包括节点映射方案 $nodespos$, 链路映射方案 $edgepaths$)

- (1) 初始化参数;
- (2) For $k = 1$ to n //初始化各子种群 k , n 为种群的个数
- (3) For $j = 1$ to tc //初始化子种群 k 中各粒子 j
- (4) 按 3.2.3 节对该粒子进行随机搜索, 生成映射的物理节点序列;
- (5) 使用 K 最短路径算法^[17]对节点映射序列进行链路映射, 按链路评价标准 2.2.2 节在 K 个物理路径中选择 θ 值最小的物理路径, 如果成功, 且按 3.1 节计算得到的适应度值更低, 则转步骤(6); 否则, 转步骤(8);
- (6) Particle[k][j] = f ; //记录粒子的位置;
- (7) 更新精英粒子和子种群 k 的最优个体, 转步骤(9);
- (8) Particle[k][j] = particle[k][j - 1]; //生成失败时, 取值为相邻粒子的位置
- (9) End for j ;
- (10) End for k ;
- (11) $t = 0$; //初始化迭代次数;
- (12) While (有种群未终止且迭代次数未超限) do //根据 3.2.5 节判断;
- (13) $t = t + 1$;
- (14) for $k = 1$ to n
- (15) if if_terminate_k == false //子种群 k 未终止
- (16) sub_ep($k, t, if_terminate_k$); //种群进化
- (17) End if
- (18) End for
- (19) End while
- (20) Return 精英粒子的位置

值尽可能小的映射方案如表 2。

在求解 K 条最短路径时, 采用 K 最短路径算法^[17], 该算法的时间复杂度为 $O(|N^s| \times \lg |N^s| + K \times |N^s| + |L^s|)$, 其中 $|N^s|$ 和 $|L^s|$ 分别为底层网络的节点个数和链路个数。那么, 由算法流程可看出, PSOAACC 算法的复杂度为多项式级别。

4 实验

4.1 实验环境

本实验在双核 CPU 3.4 GHz, 4 G 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, 实验设置同文献[19]: 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, 即到达时刻服从均值为 25 的泊松分布。持续时间服从均值为 1000 的指数分布。每个请求的节点个数在 [2,10]

表2 Sub_ep($k, t, \text{if_terminate_}k$)程序

```

PROCEDURE sub_ep( $k, t, \text{if\_terminate\_}k$ )
(1)根据 3.2.1 节得到子种群  $k$  的聚集度  $H'_k$ ;
(2)if  $H'_k \geq H_{th}$  then 子种群  $k$  进行种间指导搜索;
(3)else 子种群  $k$  进行种内指导搜索;
(4)End if
(5)按 3.2.4 节对子种群  $k$  中各粒子确定将进行指导搜索或是随机搜索;
(6)for  $j = 1$  to  $tc$  //对每个粒子  $j$  进化
(7) if 对粒子  $j$  进行指导搜索, 且子种群  $k$  为种内指导搜索 then
(8) 按 3.2.2 节对 Particle[ $k$ ][ $j$ ] 进行种内指导搜索, 生成映射的物理节点序列;
(9) 使用  $K$  最短路径算法[17]对节点映射序列进行链路映射, 按链路评价标准 2.2.2 节在  $K$  个物理路径中选择  $\theta$  值最小的物理路径, 如果成功, 且按 3.1 节计算得到的适应度值更低, 则转步骤(10); 否则, continue, 进入对下个粒子的搜索;
(10) Particle[ $k$ ][ $j$ ] =  $f$ ; //存储新的节点序列, 更新该粒子
(11) 更新精英粒子和子种群  $k$  的最优个体;
(12) Else if 对粒子  $j$  进行指导搜索, 且子种群  $k$  为种间指导搜索 then
(13) 按 3.2.2 节对该粒子种间指导搜索, 生成映射的物理节点序列;
(14) 使用  $K$  最短路径算法[17]对节点映射序列进行链路映射, 按链路评价标准 2.2.2 节在  $K$  个物理路径中选择  $\theta$  值最小的物理路径, 如果成功, 且按 3.1 节计算得到的适应度值更低, 则转步骤(15); 否则, continue, 进入对下个粒子的搜索;
(15) Particle[ $k$ ][ $j$ ] =  $f$ ; //存储新的节点序列, 更新该粒子
(16) 更新精英粒子和子种群  $k$  的最优个体;
(17) Else
(18) 按 3.2.3 节对该粒子进行随机搜索, 生成映射的物理节点序列;
(19) 使用  $K$  最短路径算法[17]对节点映射序列进行链路映射, 按链路评价标准 2.2.2 节在  $K$  个物理路径中选择  $\theta$  值最小的物理路径, 如果成功, 且按 3.1 节计算得到的适应度值更低, 则转步骤(20); 否则, continue, 进入对下个粒子的搜索;
(20) Particle[ $k$ ][ $j$ ] =  $f$ ; //存储新的节点序列, 更新该粒子
(21) 更新精英粒子和子种群  $k$  的最优个体;
(22) End if
(23) End for //对每个粒子  $j$  进化
(24) 按 3.2.5 节判断是否对该子种群  $k$  终止进化, 若终止, 设置  $\text{if\_terminate\_}k = \text{true}$ ; 否则, 设置  $\text{if\_terminate\_}k = \text{false}$ 。

```

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 1000 个, 实验共运行 25000 个左右的时间单元。

PSOAACC 是以节能为目标的智能搜索算法, 实验选取的比较对象有两个: 较经典的同样以节能为目标的贪婪映射算法 EAVNE^[20]和基于粒子群的映射算法 PSO^[15]。

在参数设置上, 同文献[18,21-23]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。由于链路功耗差异较大, 和文献[16]中设置一样, 这里比较了链路功耗在 1 W, 15 W 和 30 W 下的网络总能耗, 如图 1(d)所示。PSOAACC 算法的参数设置为子种群个体数 $tc = 30$, 最大迭代次数 $n_{\max} = 5$, 子种群数 $n = 3$; 其它的参数设置为 $\eta = 3$, $H_{th} = 0.5$, $\gamma = 1$, $a_1 = 1$, $a_2 = 1$, $\varepsilon_1 = 10^{-9}$, $\varepsilon_2 = 10^{-6}$, $K = 5$, $\beta_1 = 0.1$, $\beta_2 = 0.2$ 和 $\beta_3 = 0.7$, 物理节点已开启时设置 $d_1 = 100$, 物理节点尚未被开启时 $d_1 = 1$, $d_2 = 5$ 。PSO 算法的参数设置为种群个体数是 30, 最大迭代次数为 15。

为提高实验结果的可信度, 所有实验运行 10 组不同的虚拟网请求, 取平均值作为实验结果。

4.2 节能比较

本文的虚拟网节能映射方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, 并使得运行中的平均能耗增长较不明显。

为方便分析描述, 首先作如下定义:

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 相邻两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, 对底层网络计算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:

$$E^{\text{sum}} = \sum_{i=1}^{ts} E^i \times T_d^i \quad (17)$$

式中, 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 内底层网络的单位时间能耗, 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, 底层网络能耗即各类设备的能耗之和:

$$E^i = \sum_{j \in N^s} ECN_j + \sum_{l_{jk} \in L^s} ECL_{l_{jk}} = NO_i \times P_b + P_l \times \sum_{j \in N^s} CPUUtil_j + LO_i \times P_n \quad (18)$$

式中, NO_i 表示有效时间片 i 内处于开启状态的宿主节点数; LO_i 表示有效时间片 i 内处于开启状态的物理链路数。

定义平均节点开启量(Average Number of Open Nodes, ANON)、平均链路开启量(Average Number of Open Links, ANOL)和平均能耗量(Average Amount of Energy Consumption, AAEC)为总和与有效时间片个数之商(时间段均从初始时刻起)。

图 1(a), 图 1(b)和图 1(c)表示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。图中表明在开启节点数量, 开启的链路数量和能耗量上, PSOAACC 算法较 PSO 和 EAVNE 有较为明显的优势。

底层网络物理链路功耗参数对底层网络运行总能耗的有影响, 但本文算法 PSOAACC 和粒子群算法 PSO 相比在各种情况下均有效降低了能耗。

图 1(d)给出底层链路功耗参数分别取 1 W, 15

W 和 30 W 的底层网络总能耗(根据式(17))情况。图 1(d)表明 PSOAACC 算法在不同的链路能耗下都能取得较好的节能效果。当链路功耗为 1 W 时, PSO 较 EAVNE 的底层网络总能耗降低了 35.8%, PSOAACC 较 PSO 的底层网络总能耗降低了 7.72%; 当链路功耗为 15 W 时, PSO 较 EAVNE 的底层网络总能耗降低了 29.31%, PSOAACC 较 PSO 的底层网络总能耗降低了 10.53%; 当链路功耗为 30 W 时, PSO 较 EAVNE 的底层网络总能耗降低了 15.49%, PSOAACC 较 PSO 的底层网络总能耗降低了 12.59%。

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, 设计自适应的协同进化粒子群算法(PSOAACC), 把虚拟网映射在一个较小的节点和链路集合中, 提高休眠的节点链路数量, 以实现高效节能。实验结果表明了 PSOAACC 算法能够有效降低底层网络的能量消耗, 与经典节能算法相比, 提高了收益成本比。这里将问题限定在集中方式下的单域虚拟网节能映射问题, 下一步将对跨域映射问题进行进一步的研究。

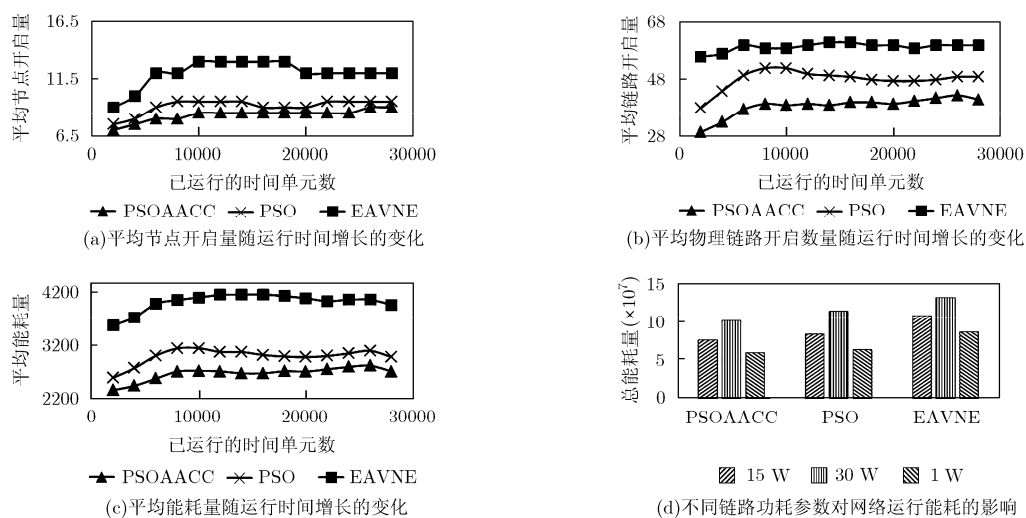


图 1 运行结果

参考文献

- [1] ANDERSON T, PETERSON L, SHENKER S, *et al.* Overcoming the internet impasse through virtualization[J]. *Computer*, 2005, 38(4): 34-41. doi: 10.1109/MC.2005.136.
- [2] CORREA E S, FLETSCHER L A, and BOTERO J F. Virtual data center embedding: a survey [J]. *IEEE Latin America Transactions*, 2015, 13(5): 1661-1670. doi: 10.1109/TLA.2015.7112029.
- [3] CHOCHLIDAKIS G and FRIDERIKOS V. Robust virtual network embedding for mobile networks[C]. 2015 IEEE 26th Annual International Symposium on Personal, Indoor and Mobile Radio Communications, Hong Kong, China, 2015: 1867-1871. doi: 10.1109/PIMRC.2015.7343603.
- [4] FISHER W, SUCHARA M, and REXFORD J. Greening backbone networks: Reducing energy consumption by shutting off cables in bundled links[C]. ACM SIGCOMM Workshop on Green Networking, India, 2010: 29-34. doi:

- 10.1145/1851290.1851297.
- [5] BARROSO L and HOLZLE U. The case for energy-proportional computing[J]. *Computer*, 2007, 40(12): 33–37. doi: 10.1109/MC.2007.443.
- [6] BOHRER P, ELNOZAHY E N, KELLER T, *et al.* The Case for Power Management in Web Servers[M]. New York, NY, USA, Kluwer Academic/Plenum Publishers, 2002: 261–289.
- [7] 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. doi: 10.3724/SP.J.1016.2011.00593.
- LIN Chuang, TIAN Yuan, and YAO Min. Green network and green evaluation: mechanism, modeling and evaluation[J]. *Chinese Journal of Computers*, 2011, 34(4): 593–612. doi: 10.3724/SP.J.1016.2011.00593.
- [8] 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. doi: 10.3724/SP.J.1016.2012.01262.
- YE Kejiang, WU Zhaohui, JIANG Xiaohong, *et al.* Power management of virtualized cloud computing platform[J]. *Chinese Journal of Computers*, 2012, 35(6): 1262–1285. doi: 10.3724/SP.J.1016.2012.01262.
- [9] MELO M, SARGENTO S, KILLAT U, *et al.* Optimal virtual network embedding: Energy aware formulation[J]. *Computer Networks*, 2015, 91: 184–195. doi: 10.1016/j.comnet.2015.08.011.
- [10] TRIKI N, KARA N, BARACHI M E, *et al.* A green energy-aware hybrid virtual network embedding approach[J]. *Computer Networks*, 2015, 91: 712–737. doi: 10.1016/j.comnet.2015.08.016.
- [11] GUAN X J, CHOI B Y, and SONG S. Energy efficient virtual network embedding for green data centers using data center topology and future migration[J]. *Computer Communications*, 2015, 69(9): 50–59. doi: 10.1016/j.comcom.2015.05.003.
- [12] CHEN Xiaohua, LI Chunzhi, and JIANG Yunliang. Optimization model and algorithm for energy efficient virtual node embedding[J]. *IEEE Communications Letters*, 2015, 19(8): 1327–1330. doi: 10.1109/LCOMM.2015.2442575.
- [13] NONDE L, EL-GORASHI T E H, and ELMIRGHANI J M H. Energy efficient virtual network embedding for cloud networks[J]. *Journal of Lightwave Technology*, 2015, 33(9): 1828–1849. doi: 10.1109/JLT.2014.2380777.
- [14] HOUIDI I, LOUATI W, and ZEGHLACHE D. Exact multi-objective virtual network embedding in cloud environments[J]. *Computer Journal*, 2015, 58(3): 403–415. doi: 10.1093/comjnl/bxu154.
- [15] ZHANG Zhongbao, CHENG Xiang, SU Sen, *et al.* A unified enhanced particle swarm optimization-based virtual network embedding algorithm[J]. *International Journal of Communication Systems*, 2013, 26(8): 1054–1073. doi: 10.1002/dac.1399.
- [16] 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 Xiaohua, LI Chunzhi, CHEN Liangyu, *et al.* Energy efficient virtual network embedding based on actively hibernating substrate nodes and links[J]. *Journal of Software*, 2014, 25(7): 1416–1431.
- [17] EPPATEIN D. Finding the k shortest paths[J]. *SIAM Journal on Computing*, 1998, 28(2): 652–673. doi: 10.1137/S0097539795290477.
- [18] ZEGURA E W, CALVERT K L, and BHATTACHARJEE S. How to model an Internetwork[C]. Proceedings of IEEE INFOCOM'96. Conference on Computer Communications, San Francisco, CA, USA, 1996: 594–602. doi: 10.1109/INFCOM.1996.493353.
- [19] CHOWDHURY N M M K, RAHMAN M R, and BOUTABA R. Virtual network embedding with coordinated node and link mapping[C]. 2009 IEEE INFOCOM 28th International Conference on Computer Communications, Rio de Janeiro, Brazil, 2009: 783–791. doi: 10.1109/INFCOM.2009.5061987.
- [20] SU Sen, ZHANG Zhongbao, CHENG Xiang, *et al.* Energy-aware virtual network embedding through consolidation[C]. IEEE Conference on Computer Communications, Orlando, FL, USA, 2012: 127–132. doi: 10.1109/INFOCOMM.2012.6193473.
- [21] SIVARAMAN V, VISHWANATH A, ZHAO Z, *et al.* Profiling per-packet and per-byte energy consumption in the NetFPGA Gigabit router[C]. IEEE INFOCOM 2011 - IEEE Conference on Computer Communications Workshops, Shanghai, China, 2011: 331–336. doi: 10.1109/INFCOMW.2011.5928833.
- [22] UNNIKRISHNAN D, VADLAMANI R, LIAO Y, *et al.* Scalable network virtualization using FPGAs[C]. 18th ACM International Symposium on Field-Programmable Gate Arrays, Monterey, CA, USA, 2010: 219–228.
- [23] BARROSO L A, CLIDARAS J, and HOLZLE U. The Datacenter as A Computer: An Introduction to the Design of Warehouse-scale Machines[M]. San Rafael, CA, USA, Morgan & Claypool Publishers, 2013: 1–154.
- 胡 颖: 女, 1982 年生, 讲师, 研究方向为网络虚拟化。
- 庄 雷: 女, 1963 年生, 教授, 研究方向为网络和自动机。
- 兰巨龙: 男, 1962 年生, 教授, 研究方向为新一代信息网络关机理论和技术。
- 马 丁: 男, 1978 年生, 讲师, 研究方向为服务路径组合和网络虚拟化。