



面向物联网的能耗感知虚拟网络映射算法

赵季红^{1,2}, 吴豆豆¹, 曲桦², 季文君¹

(1. 西安邮电大学 通信与信息工程学院, 西安 710121; 2. 西安交通大学 电子与信息工程学院, 西安 710049)

摘要: 物联网中传感器节点间规模庞大的数据交互使得能耗过大问题日趋严重, 传统能耗感知算法无法适用于节点能耗不均的物联网环境。针对该问题, 重新构建基于无线传感器网络的能耗模型, 在考虑节点异构性和链路时效性的同时保证能耗最小。在此基础上, 提出一种改进的能耗感知虚拟网络映射算法, 在节点映射阶段, 基于最接近剩余容量原则将虚拟节点映射至同类型且能耗最小的物理节点上, 并为不同时延下的链路分配合适的资源。仿真结果表明, 相比 EA-VNE、EA-VNEH 算法, 该算法通过资源整合的方式, 可以提高底层资源利用率, 降低虚拟网络映射能耗, 且随着引入参数的增加, 能够实现更细粒度的资源分配。

关键词: 物联网; 网络虚拟化; 能耗模型; 虚拟网络映射; K 最短路径

开放科学(资源服务)标志码(OSID):



中文引用格式: 赵季红, 吴豆豆, 曲桦, 等. 面向物联网的能耗感知虚拟网络映射算法[J]. 计算机工程, 2020, 46(5): 19-25, 33.

英文引用格式: ZHAO Jihong, WU Doudou, QU Hua, et al. Energy-aware virtual network mapping algorithm for Internet of Things[J]. Computer Engineering, 2020, 46(5): 19-25, 33.

Energy-Aware Virtual Network Mapping Algorithm for Internet of Things

ZHAO Jihong^{1,2}, WU Doudou¹, QU Hua², JI Wenjun¹

(1. School of Communications and Information Engineering, Xi'an University of Posts and Telecommunications, Xi'an 710121, China;

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

[Abstract] The huge data interaction between sensor nodes in the Internet of Things (IoT) intensifies the problem of excessive energy consumption. However, the traditional energy-aware algorithm cannot be applied to the IoT environment with unbalanced node energy consumption. To address this problem, the energy consumption model based on wireless sensor network is redefined to ensure the minimum energy consumption while taking heterogeneity of nodes and timeliness of links into consideration. On this basis, this paper proposes an improved energy-aware virtual network mapping algorithm. In the node mapping phase, based on the principle of the closest remaining capacity, the virtual nodes are mapped to the same type of physical nodes with the least energy consumption and the appropriate resources are allocated to the links with different latency. Simulation results show that compared with the EA-VNE and EA-VNEH algorithms, the proposed algorithm can improve the utilization rate of underlying resources and reduce energy consumption of virtual network mapping by means of resource integration. Moreover, with the increase of introduced parameters, this algorithm can achieve more fine-grained resource allocation.

[Key words] Internet of Things (IoT); network virtualization; energy consumption model; virtual network mapping; K shortest path

DOI: 10.19678/j.issn.1000-3428.0055028

0 概述

物联网(Internet of Things, IoT)^[1]是物物相连

的互联网, 为满足万物互联的需求, 基础设施提供商(Infrastructure Provider, InP)依据网络虚拟化技术^[2-3], 在多样化现有网络的同时共享基板上的异构网络架

基金项目: 国家自然科学基金(61531013); 国家科技重大专项(2018ZX03001016)。

作者简介: 赵季红(1963—), 女, 教授、博士生导师, 主研方向为带宽通信网、新一代网络管理与控制; 吴豆豆, 硕士研究生; 曲桦, 教授、博士生导师; 季文君, 硕士研究生。

收稿日期: 2019-05-27

修回日期: 2019-08-11

E-mail: 630611797@qq.com

构,为多个服务提供商(Service Provider, SP)提供资源。随着业务的增加,节点间规模庞大的数据交换愈加频繁,因此,在虚拟网络映射^[4-5]时,合理高效地完成底层物理网络资源分配、减少网络能源消耗、降低网络运维成本并构建绿色节能的网络,成为相关学者关注的热点问题之一。

目前,关于节能的虚拟网络映射算法^[6-8]研究较多。其中,文献[9-10]以最小化底层工作节点的链路为目标,采用虚拟网络映射算法(EA-VNE)降低能源消耗。但大量虚拟请求的到来会增加映射排斥率,减少收益。文献[11]以最小化节点与链路能耗为目标,建立能耗模型,并通过能耗感知虚拟网络映射启发式算法(EA-VNEH)求解模型。该算法适应于节点异构网络,但其将所有网络资源的能耗视为相等,在实际应用中不可行。文献[12]提出一种基于流量拟合模型的自适应能量感知VNE方法,通过有选择地整合底层网络资源并关闭链路,从而保证QoS约束并实现节能。但是该算法采用整数线性规划求解问题,映射时间较长。文献[13]出于生态考虑,提出一种绿色能量感知混合虚拟网络映射算法,将其VN部署到与最小排放因子相关联的扇区资源上,该方法旨在解决能源效率和资源整合问题,但对各地区能源类型进行调查会相对困难。以上虚拟网络节能映射算法在互联网场景下达到了较好的节能效果,但无法适用于物联网中节点异构性更突出,初始状态、能量消耗速率不同等能耗特性。

本文针对物联网环境下节点异构性与链路时效性特点,提出一种改进的能耗感知虚拟网络映射算法(Modified-EA-VNE)。在虚拟网络映射之前根据不同业务的服务类型对虚拟请求做拆分处理,依据能耗模型,采用混合整数线性规划方法将虚拟节点映射至同类型且能耗最小的物理节点上,并通过改进的K最短路径算法为不同时延下的链路分配合适的资源。

1 问题描述与能耗模型

1.1 问题描述

为解决物联网环境下的能耗问题,本文提出一种改进的能耗感知虚拟网络映射算法,该算法根据业务的服务类型将每个到来的虚拟请求拆分为若干个小拓扑,并将拆分后的虚拟请求映射至满足其要求的能耗最小的物理节点上。其中,每个虚拟节点与CPU容量约束相关联,每条虚拟链路带宽容量和所需延迟相关联,每个物理节点与CPU剩余值和识别节点能源类型的地理位置相关联,每条物理路径(连接2个物理节点)与剩余带宽容量和延迟相关联。

1.2 网络拓扑

底层物理网络拓扑图定义为加权无向图 $G_s =$

(N_s, L_s, C_s^n, C_s^l) , 其中, N_s 和 L_s 分别表示底层物理网络节点和链路的集合, C_s^n 和 C_s^l 分别表示节点集和链路约束。每个底层节点 $n_s \in N_s$ 与 CPU 容量值 $C_{cpu}(n_s)$ 相关联, 其表示物理节点 n_s 上的可用 CPU 容量, $L_{loc}(n_s)$ 表示物理节点的地理位置。底层链路 $l_s \in L_s$ 与带宽容量 $b(l_s)$ 和时延 $d(l_s)$ 相关联, 从源节点 A 到目的节点 B 的一组底层路径由 $P_{path}(a, b)$ 表示。另外, 本文针对节点的异构性将节点的类型细化, 采用节点类别加序列号的形式表示节点的类别。将虚拟网络拓扑图标记为加权无向图 $G_v = (N_v, L_v, C_v^n, C_v^l)$ 。虚拟节点和虚拟链路与底层物理网络中的节点和链路具有相同的属性。

1.3 能耗模型

虚拟网络请求映射至物理网络的节点时, 总能耗 P_N 定义为^[14]:

$$P_N = N_w P_b + P_l \sum_{l_j \in N_v} U_{util}(n_j) \quad (1)$$

其中, N_w 表示需要从未激活状态变为激活状态的宿主节点数量, P_b 表示节点的基础功耗, P_l 表示转发节点的能耗, $U_{util}(n_j)$ 表示节点映射所需的 CPU 利用率。

虚拟网络请求映射至物理网络的链路时, 总能耗 P_L 定义为^[14]:

$$P_L = N_w P_n + N_n (P_b + P_n) + d(u, i) \alpha \quad (2)$$

其中, P_n 是一个值为恒定常量的链路功耗^[15], N_n 表示需要从未激活状态变为激活状态的转发节点数量, $d(u, i)$ 表示节点 u 到节点 i 的路径距离, α 表示距离因子。

1.4 虚拟网络映射数学模型

在建立虚拟网络映射能耗模型时, 以最小化总能耗为目标函数, 并满足虚拟网络资源的每个约束, 根据混合整数线性规划方法对虚拟网络映射问题进行建模。目标函数为:

$$\min (P_N + P_L) \quad (3)$$

约束条件为:

$$\forall i \in N_s, \forall j \in N_v, x_i^j \cdot C_{cpu}(j) \leq C_{cpu}(i) \quad (4)$$

$$\forall j \in N_v, D_{dis}(L_{loc}(j), L_{loc}(M_n(j))) \leq D(j) \quad (5)$$

$$\forall i \in N_s, \forall j \in N_v, G_{genre}(j) = G_{genre}(i) \quad (6)$$

$$\forall l_{ij} \in L_s, \forall l_{uv} \in L_v, \sum_{l_{uv} \in L_v} f_{ij}^{uv} \cdot b(l_{uv}) \leq b(l_{ij}) \quad (7)$$

$$\forall l_{ij} \in L_s, \forall l_{uv} \in L_v, \sum_{l_{uv} \in L_v} f_{ij}^{uv} \cdot d(l_{uv}) \geq d(l_{ij}) \quad (8)$$

$$\forall i \in N_s, \forall l_{uv} \in L_v, \sum_{l_{ij} \in L_s} f_{ij}^{uv} - \sum_{l_{ji} \in L_s} f_{ji}^{uv} = \begin{cases} 1, & x_i^u = 1 \\ -1, & x_i^v = 1 \\ 0, & \text{其他} \end{cases} \quad (9)$$

$$\forall i \in N_s, \forall j \in N_v, x_i^j \in \{0, 1\} \quad (10)$$

$$\forall l_{ij} \in L_s, \forall l_{uv} \in L_v, f_{ij}^{uv} \in \{0, 1\} \quad (11)$$

其中, $f_{ij}^{uv} \in \{0, 1\}$ 表示虚拟链路 l_{uv} 与物理链路 l_{ij} 之间的映射关系, 当 $f_{ij}^{uv} = 1$ 时, 则说明虚拟链路 l_{uv} 成功映射到物理链路 l_{ij} 上, 当 $f_{ij}^{uv} = 0$ 时, 则未成功。

虚拟映射模型的目标函数如式(3)所示, 式(4)~式(6)分别表示节点的 CPU 资源约束、位置约束和类别约束。其中, $C_{\text{cpu}}(n)$ 表示节点的 CPU 容量值, $G_{\text{genre}}(n)$ 表示节点类型, $D_{\text{dis}}(L_{\text{loc}}(j), l_{\text{loc}}(M_n(j)))$ 表示映射节点 j 与周围节点之间的欧式距离。式(7)~式(9)分别表示链路的带宽资源约束、延迟约束和连通性约束。其中, $b(l_{mn})$ 表示链路的带宽资源, $d(l_{mn})$ 表示链路的延迟。式(10)、式(11)表示该模型的变量范围约束。 $x_i^j \in \{0, 1\}$ 表示虚拟节点与物理节点之间的映射关系, 当 $x_i^j = 1$ 时, 表示虚拟节点 j 成功映射到物理节点 i 上, 当 $x_i^j = 0$ 时, 则未成功映射。

2 算法设计

本文虚拟网络映射算法包括虚拟请求拆分、节点映射以及链路映射^[16]3个部分。

2.1 虚拟网络请求拆分

由于物联网对实时性具有较高的要求, 因此在虚拟网络映射中需要考虑链路的时延问题。为了降低映射过程的复杂性并考虑链路的延迟特性, 本文根据时延约束, 对到来的虚拟网络请求做拆分处理, 将每个虚拟请求大拓扑拆分为一组虚拟请求小集群, 在后续的节点与链路映射中, 并行处理分割后的虚拟请求, 缩短映射运行的时间。上述拆分基于虚拟链路 $\text{Flag}(l_v)$ 的服务类型, 每条链路用 $f(l_v)$ 做标记以区分该链路对时延的要求。在链路映射中, 基于该拆分结果为每条虚拟链路分配满足其服务类型的物理链路。如图 1 所示, 根据业务的需求将虚拟网络拓扑图拆分成 S-V1 和 S-V2 2 个小拓扑图。

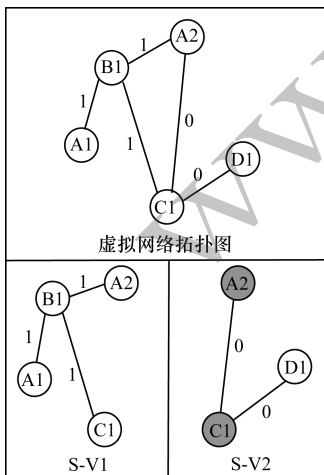


图 1 虚拟请求拆分示意图

Fig. 1 Schematic diagram of virtual request division

在图 1 中, S-V1 的 $f(l_v) = 1$ 表示该拓扑图具有延迟属性, S-V2 的 $f(l_v) = 0$ 表示该拓扑图具有带宽属性, 其中, 有灰色标记的 2 个节点为之前 S-V1 所映射过的节点, 不再考虑其节点映射问题。虚拟请求拆分过程伪代码如算法 1 所示。

算法 1 虚拟请求拆分算法

输入 $G_v = (N_v, L_v, C_v^n, C_v^l)$

输出 $G_v(i) = (N_v(i), L_v(i), C_v^n(i), C_v^l(i))$

1. 令 $G_i = G_v, i = 1, G_v(i) = 0$
2. while 任意一条链路 $L_r \neq \emptyset$ do
3. $l(u, v) \in L_r, u \in N_r, v \in N_r$
4. 将 G_i 分配给 $G_v(S-V(i))$, 并标记分配节点 $\text{Flag}\{u, v\}$
5. for 与 G_i 相连接的其他链路 $l(u', v') \in L_r$
6. if $(f(u', v') = f(u, v) \mid u' \in N_v(i) \ \& \ v' \in N_v(i))$
7. 将新链路分配给 $G_v(S-V(i))$, 并标记 $\text{Flag}\{u', v'\}$
8. end if
9. end for
10. $L_r = L_r - L_v(i)$
11. $i = i + 1$
12. end while

2.2 节点映射

由于物联网底层是一个自组织网络, 因此针对节点异构性问题, 本文依据节点的类型对底层资源进行分类。虚拟节点只能映射至同类型的物理节点上, 从而避免开启无关类型的节点。这种资源整合的方式可以减少底层资源的浪费, 进而达到节能的目的。在节点映射阶段, 基于最小能耗和最接近剩余容量原则, 将虚拟节点映射至同类型且能耗最小的物理节点上, 本文将节点的资源能力^[17]定义为:

$$N_{\text{NR}}(n) = \left[C_{\text{cpu}}(n) + C_{\text{cpu}}(n) \cdot \frac{\lg \sum_{n \in nb} n_{nb}}{\lg C_{\text{cpu}}(n)} \right] \cdot \sum_{l \in L(n)} b(l) \quad (12)$$

其中, $\left[C_{\text{cpu}}(n) \cdot \frac{\lg \sum_{n \in nb} n_{nb}}{\lg C_{\text{cpu}}(n)} \right]$ 表示该映射节点与周围节点的拓扑关系。为了将虚拟请求映射到能耗最小且刚好满足 CPU 需求的物理节点上, 需根据虚拟请求 n_v 的剩余 CPU 资源、位置关系以及节点类型, 确定候选映射节点集合 $\Omega(n_v)$ 以及映射节点 $n_s \in \Omega(n_v)$ 的综合资源能力。候选映射节点集合为:

$$\Omega(n_v) = \{ n_s \mid C_{\text{cpu}}(n_v) \leq C_{\text{cpu}}(n_s) \ \& \ D_{\text{dis}}(L_{\text{loc}}(n_v), L_{\text{loc}}(n_s)) \leq D(n_s), \ n_s \in N_s \} \quad (13)$$

映射节点的综合资源能力为:

$$C_{\text{CR}}(n_s) = \lg(N_{\text{NR}}(n_s) \cdot P_N(n_s) + \gamma) \quad (14)$$

其中, $\gamma \rightarrow 0$, 避免式(14)中对数函数内部出现 0 的情况。

节点映射算法通过将每个虚拟节点映射到能耗最小且满足 CPU 需求的最少资源量的物理节点上,以降低网络映射的能耗,提高物理资源的利用率。如图 2 所示,将虚拟请求 S-V2 映射至底层网络,首先考虑 CPU 为 10 的节点 D2,然后选择具有最少但足够的 CPU 资源的物理节点作为候选节点,最后比较节点类型,将虚拟请求 D2 映射至物理节点 D1 上,并计算物理节点 D1 的剩余资源量。算法伪代码如算法 2 所示,与以往的能耗感知映射算法相比,该算法通过将虚拟请求映射至 C_{CR} 最小但满足 CPU 要求的物理节点上,可有效节省能耗,提高物理资源的利用率。

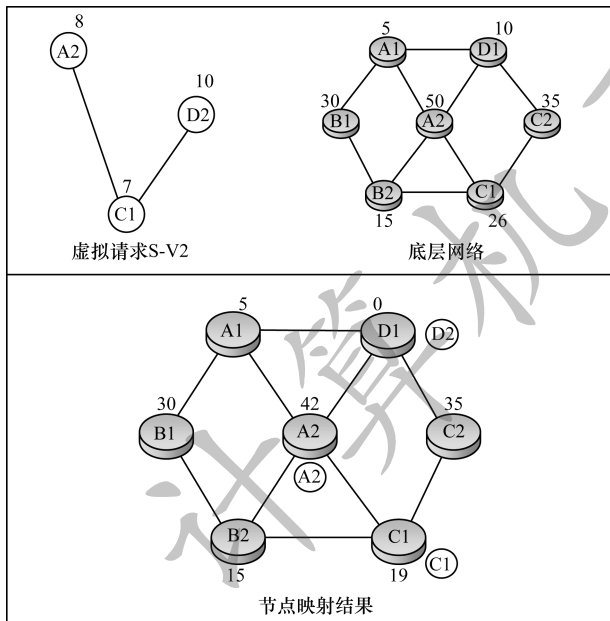


图 2 虚拟节点映射示意图

Fig.2 Schematic diagram of virtual node mapping

算法 2 节点映射算法

输入 $G_s = (N_s, L_s, C_s^r, C_s^l)$, $G_v = (N_v(i), L_v(i), C_v^r(i), C_v^l(i))$

输出 NodeMappingList, 并作为链路映射的输入参数

1. for 每一个虚拟节点 $n_v \in N_v$ do
2. 计算虚拟节点 n_v 的资源能力 $N_{NR}(n_v)$
3. end for
4. 将虚拟节点 n_v 按资源能力 $N_{NR}(n_v)$ 降序排列, 并将结果存入 VNL 中
5. for VNL 中的每一个虚拟节点 n_v do
6. 构建候选节点集合且每个 n_v 满足虚拟节点的 CPU 资源和位置约束
7. if $\Omega(n_v) \neq \varphi$ then
8. for $\Omega(n_v)$ 中的每一个候选节点 n_s do
9. 计算 $C_{CR}(n_s)$ 的值
10. end for

11. 将 n_s 按照 $C_{CR}(n_s)$ 的值降序存入 NodeList 中
12. if $G_{genre}(n_v(i)) = G_{genre}(n_s(j))$ then
13. 将虚拟节点 n_v 映射至 C_{CR} 最小且满足约束的物理节点 n_s 上, 并将结果集存入 NodeMappingList 中
14. end if
15. 计算 CPU 剩余资源 $C_{cpu}(n_s) = C_{cpu}(n_s) - C_{cpu}(n_v)$
16. end if
17. end for
18. 节点映射结束

2.3 链路映射

针对虚拟链路映射,通常采用 K 最短路径算法^[18-19],但在物联网环境下,需通过传感器实时、准确地传递物体的信息,对网络资源的能耗以及链路的实时性要求较高。因此,在链路映射阶段,本文通过整合链路映射过程中的能源消耗和链路延迟等因素,设计一种能耗优先且满足链路延迟的 K 最短路径算法。

如图 3 所示,虚线箭头表示 2 个节点之间的路径,虚拟网络中的 $x/y/z$ 分别表示带宽/时延阈值/是否对时延敏感。将虚拟请求 S-V1 映射至底层物理链路,首先根据节点映射算法确定映射成功的物理节点,然后依据带宽需求优先选取资源需求量较高的虚拟链路进行映射。对于虚拟链路 $l_{uv} \in L_v$,采用 K 最短路径算法计算最短路径集合 $\varphi(l_{uv})$,且对于任意路径 $P_{path} \in \varphi(l_{uv})$,需满足虚拟链路 l_{uv} 的带宽需求以及延迟容量,最后通过计算链路能耗 P_L ,将虚拟链路 l_{uv} 映射至能耗最小的路径上。

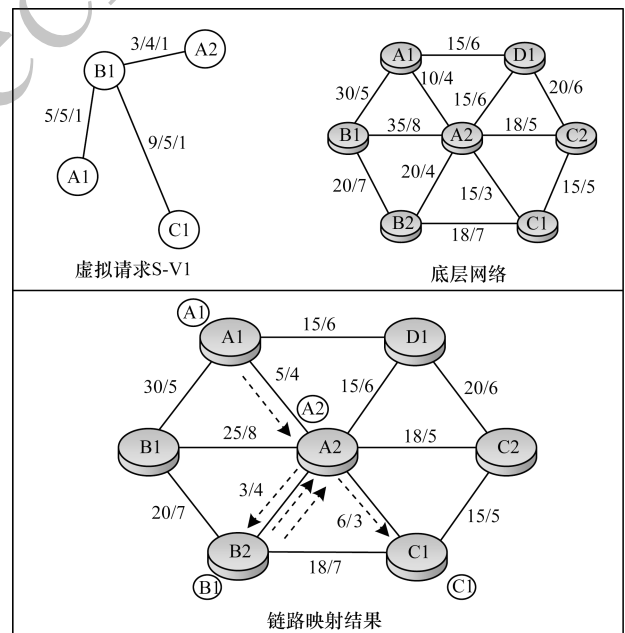


图 3 虚拟链路映射示意图

Fig.3 Schematic diagram of virtual link mapping

链路映射算法伪代码如算法 3 所示, 与传统算法相比, 该算法考虑链路的时延问题, 提高了算法的延展性, 能够更广泛地适用于物联网场景。在链路映射过程中, 针对不同延迟需求的链路进行分开处理, 可缩短链路映射的运行时间, 降低网络映射与链路映射的成本, 从而增加映射收益。

算法 3 链路映射算法

输入 $G_s = (N_s, L_s, C_s^m, C_s^l), G_v = (N_v(i), L_v(i), C_v^m(i), C_v^l(i))$

输出 LinkMappingList, 并计算评价指标
NodeMappingList

1. 将虚拟链路按带宽需求降序排序, 并将结果存入链表 VLL 中
2. for VLL 中的每条虚拟链路 l_{uv} do
3. 采用 K 最短路径算法计算底层节点的最短路径集合 $\varphi(l_{ij})$, 且对于任意路径 $P_{path} \in \varphi(l_{ij})$
4. if $\varphi(l_{ij}) = \varphi$ then
5. Return 映射失败
6. else
7. for $\varphi(l_{ij})$ 中的每一条最短路径 P_{path} do
8. 计算链路能耗 $P_L(l_{ij}^m)$, 并将其升序排列存入 LinkList 中
9. end for
10. switch (待映射链路 l_i (flag)) do
11. case 0
12. if $b(l_{uv}) \leq b(l_{ij})$ then
13. 将虚拟链路 l_{uv} 映射至物理链路 l_{ij} 上, 并将结果存入 LinkMappingList 中
14. case 1
15. if $b(l_{uv}) \leq b(l_{ij})$ & $d(l_{uv}) \geq d(l_{ij})$ then
16. 将虚拟链路 l_{uv} 映射至物理链路 l_{ij} 上, 并将结果存入 LinkMappingList 中
17. end switch
18. 计算带宽剩余资源 $b(l_s) = b(l_s) - b(l_{ij})$
19. end if
20. end for
21. 链路映射结束

3 仿真结果与性能分析

在本文仿真中, 由 GT-ITM^[20] 生成设定的网络拓扑, 通过 NS2^[21] 中的 Tk 工具生成可视化的网络拓扑, 对本文算法以及 2 种基于能耗感知的虚拟网络映射算法进行对比, 在底层物理资源使用数量、虚拟网络请求能耗、映射时间以及底层网络收益开销比等方面验证算法的性能。

3.1 仿真环境设置

本文通过 GT-ITM 生成网络拓扑, 具体实验参

数如表 1 所示, 其中, 虚拟节点与物理节点间的连通率均为 50%, 式(3)中的距离因子设置为 0.5, K 最短路径算法中 K 的取值为 5。针对节点的异构性特点, 令 $P_b = 150 \text{ W}, P_m = 300 \text{ W}, P_n = 15 \text{ W}$ 。每次实验运行时间为 30 000 个时间单位, 约有 1 200 个虚拟网络请求到达, 并运行 10 次实验, 取平均值作为最终结果。

表 1 实验参数设置
Table 1 Settings of experiment parameters

拓扑	节点数	CPU/带宽	延迟	到达时间	生存周期
物理网络	50	[50, 100]	[50, 300]	—	—
虚拟网络	[2, 8]	[0, 50]	[100, 350]	4 的泊松分布	参数为 25 的指数分布

3.2 性能分析

3.2.1 评价指标

本文实验评价指标具体如下:

1) 虚拟网络请求平均能耗:

$$\bar{E} = \frac{\sum_{i=1}^N E(i)}{N} \quad (15)$$

其中, $E(i)$ 为第 i 个虚拟请求消耗的能量, N 为虚拟请求的个数。

2) 收益开销比:

$$R_{\text{Rate}} = \frac{\sum_{i=1}^{A_T} R(i)}{\sum_{i=1}^N C_{\text{Cost}}(i)} \quad (16)$$

其中, A_T 为时间 T 内接收的虚拟请求数量, $R(i)$ 和 $C_{\text{Cost}}(i)$ 分别表示虚拟请求 i 的收益和成本, 两者具体定义如下:

$$R = \sum_{n_v \in N_v} C_{\text{cpu}}(n_v) + \sum_{l_v \in L_v} b(l_v) \quad (17)$$

$$C_{\text{Cost}} = \sum_{n_s \in E_N(N_v)} C_{\text{cpu}}(n_s) + \sum_{l_v \in L_v} \sum_{l_s \in E_L(L_v)} b(l_s) \quad (18)$$

3.2.2 结果分析

为了评估本文 Modified-EA-VNE 映射算法的性能, 本节将其与 EA-VNE、EA-VNEH 两种算法进行比较。EA-VNE 算法以最小化底层网络工作节点与链路数量为目标, 在节点映射阶段采用贪婪算法, 链路映射阶段采用最短路径算法。EA-VNEH 主要考虑节点能耗的异构特性, 通过建立能耗模型使节点和链路均映射至新增能耗最小的底层网络中。本节从工作节点和链路数量、虚拟网络请求映射时间、映射所花费的能耗以及物理网络收益开销比等方面对算法进行比较。

1) 工作节点与链路数量

图 4、图 5 分别表示不同算法中虚拟网络映射工作节点和链路的数量。从中可以得出, EA-VNEH 算

法所需的底层节点数量最多, Modified-EA-VNE 算法最少, 主要因为 EA-VNEH 算法着重考虑物理网络节点的异构性, 将虚拟请求映射至能耗最小的节点上, 而 Modified-EA-VNE 算法在节点映射时采用资源整合的方式且基于最接近剩余容量原则, 故所需的节点数最少。但在链路映射过程中, 由于 Modified-EA-VNE 算法需考虑链路的延迟特性, 故工作的链路略高于 EA-VNE。Modified-EA-VNE 算法考虑节点的异构性, 将节点分配给同类型的节点, 增加了使用同一物理节点作为映射节点的概率, 从而导致了资源整合。

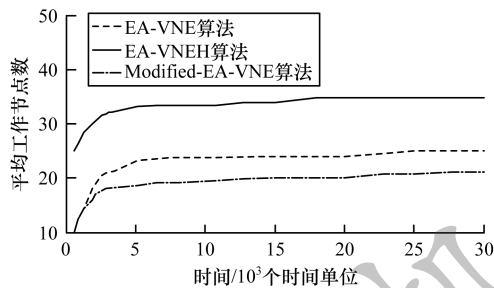


图 4 3 种算法平均工作节点数量对比

Fig. 4 Comparison of average number of working nodes of three algorithms

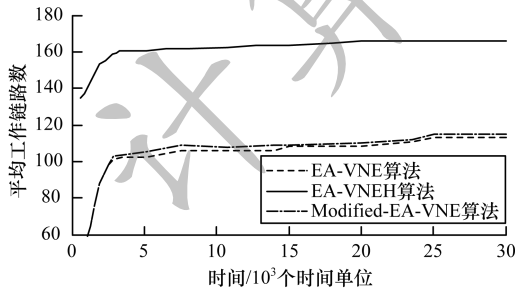


图 5 3 种算法平均工作链路数量对比

Fig. 5 Comparison of average number of working links of three algorithms

2) 虚拟网络请求能耗和映射时间

在物联网环境下, 3 种算法所消耗的能耗如图 6 所示。从图 6 可以看出, Modified-EA-VNE 算法的平均能耗约为 20.5 kW, 较 EA-VNE、EA-VNEH 分别降低 36.90% 和 18.65%, 其原因在于, EA-VNE 仅通过最小化工作节点和链路实现节能, 但是在节点异构的环境中, 该方法并不能实现能耗最优化, 而另外 2 种算法均适用于节点异构环境, 但由于 Modified-EA-VNE 算法不仅采用能耗模型, 将节点和链路映射至能耗最小的底层网络中, 又基于最接近剩余容量原则, 使得底层网络中工作的节点和链路数量尽可能少, 从而使得能耗最小。3 种算法虚拟网络的映射时间如图 7 所示, 从图 7 可以看出, EA-

VNE 算法的映射时间最短, 约为 50 ms, 而 Modified-EA-VNE 所需的时间较长, 约为 78 ms, 其原因在于, EA-VNE 和 EA-VNEH 采用启发式算法, 该算法有效降低了映射所需的时间, 而 Modified-EA-VNE 需考虑节点的异构性以及链路的时延特性, 映射节点分布较不均匀, 且映射节点之间的路径相对复杂, 此外, 将 VNR 拆分为 2 种类型的子 VNR, 仿真中只开启了 2 个线程, 并行处理 VNR 所减少的映射时间远小于在选取节点与路径时所花费的时间。

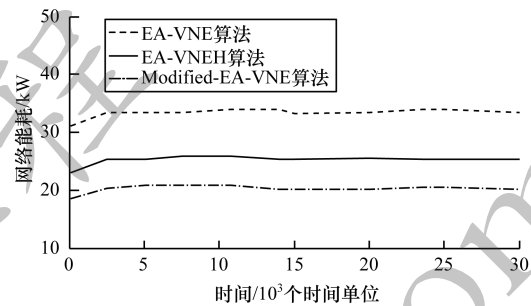


图 6 3 种算法虚拟网络请求平均能耗对比

Fig. 6 Comparison of average energy consumption of virtual network request of three algorithms

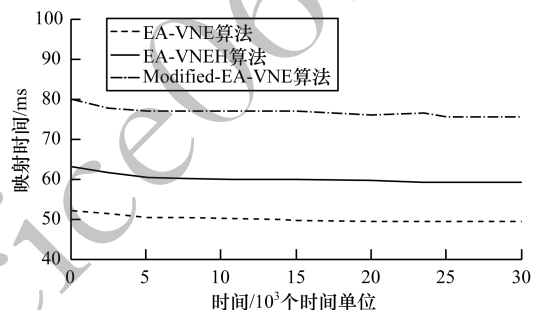


图 7 3 种算法虚拟网络映射时间对比

Fig. 7 Comparison of virtual network mapping time of three algorithms

3) 收益开销比

3 种算法的物理网络收益开销比对比情况如图 8 所示。从图 8 可以看出, 当虚拟请求较少时, 由于底层资源比较充足, 收益开销比较高, 但随着资源的逐渐消耗, 收益开销比也随之下降, 最终达到一个稳定的状态。Modified-EA-VNE 算法收益开销比相对较高的原因在于, 该算法在虚拟网络映射之前根据不同服务类型将虚拟请求做拆分处理, 分别对不同的虚拟请求分配资源, 提高了虚拟网络请求的接受率, 但由于需将节点分配给同类型的节点, 容易产生瓶颈效应, 导致其收益开销比趋于 EA-VNEH 算法, 而 EA-VNE 算法仅考虑工作节点和链路, 使得后续节点和链路过于拥塞, 降低了虚拟网络映射的成功率。

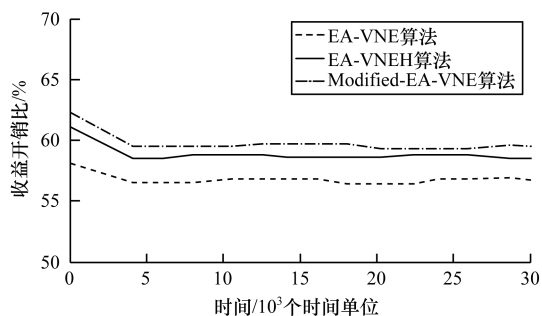


图8 3种算法物理网络收益开销比对比

Fig. 8 Comparison of benefit-expense ratio of physical network of three algorithms

4 结束语

为了解决物联网环境下的能耗优化问题,同时满足节点异构性与链路时效性的要求,本文设计一种能耗感知虚拟网络映射算法 Modified-EA-VNE。仿真结果表明,与EA-VNE和EA-VNEH算法相比,Modified-EA-VNE算法能够有效减少虚拟网络映射的能耗,增加底层网络资源利用率,提高物理网络收益,并更细粒度地完成虚拟请求映射。由于实际物联网的网络拓扑为动态拓扑,虚拟请求的到达和离开是动态且顺序不可预测的,因此下一步将结合节点重映射与路径分割技术来设计一种合理的调度算法。

参考文献

- [1] ZHANG Haiyu. The Internet of Things network node real-time detection of energy consumption simulation [J]. Computer Simulation, 2016, 33 (12): 453-456. (in Chinese)
张海玉. 物联网网络节点能耗实时检测仿真[J]. 计算机仿真, 2016, 33(12): 453-456.
- [2] CHENG X, SHUANG K, ZHANG Z, et al. Minimizing electricity cost in geographical virtual network embedding [C]// Proceedings of Global Communications Conference. Washington D. C., USA: IEEE Press, 2012: 15-23.
- [3] HU Ying, ZHUANG Lei, CHEN Hongchang, et al. Time and energy aware virtual network embedding using Bayesian theory analysis [J]. Journal on Communications, 2016, 37(6): 106-118. (in Chinese)
胡颖, 庄雷, 陈鸿昶, 等. 时间和能量感知的贝叶斯虚拟网映射[J]. 通信学报, 2016, 37(6): 106-118.
- [4] NONDE L, EL-GORASHI T, ELMIRGHANI J. Energy efficient virtual network embedding for cloud networks [J]. Journal of Lightwave Technology, 2015, 33(9): 1828-1849.
- [5] BOTERO J F, HESSELBACH X, DUELLI M, et al. Energy efficient virtual network embedding [J]. IEEE Communication Letters, 2012, 16(5): 756-759.
- [6] BOTERO J F, HESSELBACH X. Greener networking in a network virtualization environment [J]. Computer Networks, 2013, 57(9): 2012-2039.
- [7] QU Hua, FAN Bin, GUO Ya, et al. Green virtual network mapping algorithm using particle swarm optimization [J]. Telecommunications Science, 2016, 32(1): 11-17. (in Chinese)
曲桦, 樊斌, 郭涯, 等. 应用粒子群优化的绿色虚拟网络映射算法[J]. 电信科学, 2016, 32(1): 11-17.
- [8] XU Z, LIANG W, XIA Q. Efficient embedding of virtual networks to distributed clouds via exploring periodic resource demands [J]. IEEE Transactions on Cloud Computing, 2018, 6(3): 694-707.
- [9] ZHANG Z, SU S, ZHANG J, et al. Energy aware virtual network embedding with dynamic demands: online and offline [J]. IEEE/ACM Transactions on Networking, 2014, 22(5): 1607-1620.
- [10] SU S, ZHANG Z, CHENG X, et al. Energy-aware virtual network embedding through consolidation [C]// Proceedings of 2012 IEEE INFOCOM Workshops. Washington D. C., USA: IEEE Press, 2012: 25-30.
- [11] GONG Shuiqing, CHEN Jing, WANG Wei. Energy-aware virtual network embedding algorithm for heterogeneous nodes [J]. Journal of Electronics and Information Technology, 2015, 37(8): 2021-2027. (in Chinese)
龚水清, 陈靖, 王巍. 面向节点异构的能耗感知虚拟网络映射算法[J]. 电子与信息学报, 2015, 37(8): 2021-2027.
- [12] LIU H, CUI H, WANG J, et al. An adaptive energy-aware virtual network embedding scheme [C]// Proceedings of 2014 IEEE Vehicular Technology Conference. Washington D. C., USA: IEEE Press, 2014: 1-5.
- [13] TRIKI N, KARA N, BARACHI M E, et al. A green energy-aware hybrid virtual network embedding approach [J]. Computer Networks, 2015, 91(C): 712-737.
- [14] LIU Renlun. Research on energy saving algorithm of virtual network mapping [D]. Nanjing: Nanjing University of Posts and Telecommunications, 2017. (in Chinese)
刘认伦. 虚拟网络映射节能算法研究[D]. 南京: 南京邮电大学, 2017.
- [15] SIVARAMAN V, VISHWANATH A, ZHAO Z, et al. Profiling per-packet and per-byte energy consumption in the net FPGA gigabit router [C]// Proceedings of the 30th IEEE International Conference on Computer Communications Workshops. Washington D. C., USA: IEEE Press, 2011: 331-336.
- [16] CHOWDHURY M, 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.
- [17] ZHANG Peiying. Virtual network embedding based on node's multiple attributes ranking [J]. Computer and Modernization, 2016, 11(11): 33-37. (in Chinese)
张培颖. 节点多特征评价的虚拟网络映射算法[J]. 计算机与现代化, 2016, 11(11): 33-37.
- [18] HU Junhong, GUO Xiucheng, TAO Tao, et al. Optimization of modern tram network based on K shortest path [J]. Journal of Southeast University (Natural Science Edition), 2017, 47(6): 1274-1278. (in Chinese)

(上接第 25 页)



胡军红,过秀成,陶涛,等.基于 k 最短路径的现代有轨电车线网优化[J].东南大学学报(自然科学版),2017,47(6):1274-1278.

- [19] 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) :17-29.
- [20] SAMUEL F, MOSHARA F, CHOWDHUR Y, et al.

PolyViNE: policy-based virtual network embedding across multiple domains[J]. Journal of Internet Services and Applications, 2013, 4(1) :6-10.

- [21] ONEILL R E, TALON J, PALESE P. The influenza virus NP(NS2 protein) mediates the nuclear export of viral ribonucleoproteins[J]. The EMBO Journal, 1998, 17(1) :288-296.

编辑 吴云芳