

文章编号:1007-5321(2019)01-0109-05

DOI:10.13190/j.jbupt.2018-036

一种适用于 $(p+m)$ -中点问题的服务设施放置算法

焦计平, 逯海, 洪学敏, 石江宏

(厦门大学信息科学与技术学院, 厦门 261005)

摘要: 针对雾计算应用中服务设施放置问题, 将其建模成 $(p+m)$ -中点问题, 提出了一种基于贪婪策略与禁忌搜索策略相结合的启发式服务设施放置算法. 提出的算法适用于一般拓扑、任意需求分布的网络. 性能分析结果表明, 提出的算法是多项式时间的, 在当扩展服务节点数和请求节点数相等时能够达到性能上的最优. 仿真结果验证了新算法的有效性.

关键词: 放置问题; 禁忌搜索; 搜索域; 图状网络

中图分类号: TN929.53

文献标志码: A

A Service Facility Placement Algorithm for the $(p+m)$ -Median Problem

JIAO Ji-ping, LU Hai, HONG Xue-min, SHI Jiang-hong

(School of Information Science and Engineering, Xiamen University, Xiamen 361005, China)

Abstract: This service facility placement problem is investigated, which appears representatively in fog computing for the cost minimization and the optimal resource utilization. After the problem being modelled as a $(p+m)$ -median problem, a novel heuristic placement algorithm is proposed that combines the greedy and tabu-search strategies. The proposed algorithm can be used in networks with arbitrary topology and random demand distribution. Analysis results show that it is polynomial in time complexity and can reach the optimal performance in the case that the number of the extended service nodes in the network is equal to that of the request nodes. Finally, simulations verify the advantages above.

Key words: placement problem; tabu-search; search domain; graph network

在网络边缘节点上部署服务设施能够有效降低响应时延与链路带宽需求, 在雾计算和第5代移动通信系统(5G)中有着广泛的应用^[1-2]. 这时, 一个很自然的问题是, 在哪些节点上部署数目有限的服务设施能够使性能达到最优, 这也被称作服务设施放置问题, 其典型数学模型是使整体性能或平均性能达到最优的 p -中点问题(p -median problem). 已经证明, 在一般网络拓扑中, p -中点问题是 NP-Hard 的^[3-4]. 而针对这一问题的求解算法主要为以下3类: 特殊拓扑下的放置算法^[5]、有效的启发式算

法^[6]以及各种近似算法^[7]. 例如, Lorena 等^[8]提出了一种基于遗传算法的启发式方法. 笔者研究了 p -中点问题的一个扩展版本, 即 $(p+m)$ -中点问题, 是指在不改变网络中已有 m 个预置服务节点位置的前提下如何向网络中继续引入 p 个扩展服务节点, 并使新引入服务节点的总效用达到最大. 在此基础上, 提出了一种基于贪婪策略与禁忌搜索策略相结合启发式服务设施放置算法. 与现有结果相比, 提出的算法适用一般拓扑网络, 在性能与复杂度方面也具有充分优势.

收稿日期: 2018-02-07

基金项目: 国家自然科学基金项目(61571378)

作者简介: 焦计平(1985—), 男, 博士生, E-mail: jjp@xmu.edu.cn; 石江宏(1968—), 男, 教授, 博士生导师.

1 问题描述

令 $G = (V, E)$ 表示所研究的图状网络. 其中, V 为网络节点的集合, E 为边的集合. 不失一般性, 假设 G 是一个简单联通图. 进一步, $(p+m)$ -中点问题涉及 2 种类型的服务节点: m 个位置不可变更的预置服务节点以及 p 个位置可自由改变的扩展服务节点. 称与请求用户(或设备)直接相连的网络节点为请求节点, 称每个请求节点所连接的用户数目为该节点的需求. 针对每个用户的服务响应, 假设不同链路所消耗的通信成本是无差别的. 基于最小化总的成本方面的考虑, 假设任意节点都将选择节点间链路跳数最小的路径进行通信, 并称这一跳数为节点对间的距离. 当节点间存在多条最短路径时, 从中随机选取一条作为通信链路, 将不会影响得到的性能结果.

在此基础上, 总的通信成本被定义为经需求加权后请求节点与响应节点之间的距离之和, 引入扩展服务节点的目的在于降低该总成本, 并称这种成本上的降低为服务设施产生的效用. 经推导可得, 每个扩展服务设施的效用可以表示为

$$r_j = \sum_i a_i (\tilde{d}_i - d_{ij}) x_{ij}$$

其中: i 表示请求节点, j 表示服务节点, d_{ij} 表示节点 i, j 间的距离, \tilde{d}_i 表示在只存在预置服务节点情况下请求节点 i 到指派给自身的服务节点间的距离, x_{ij} 指示节点 i, j 间的指派关系. 更具体地, $x_{ij} = 1$ 表示服务节点 i 被指派给请求节点 j ; 而 $x_{ij} = 0$ 表示服务节点 i 未指派给请求节点 j . 加总所有扩展服务节点的效用, 可得总效用为

$$R = \sum_j r_j$$

这时, 成本最小化问题将被转换为如下效用最大化问题

$$\begin{aligned} & \max R \\ \text{s. t. } & \begin{cases} \sum x_{ij} = 1 \\ x_{ij} \leq y_j \\ \sum y_j = p + m \\ x_{ij}, y_j \in \{0, 1\} \end{cases} \end{aligned}$$

其中: y_j 表示节点 j 的服务设施配置状态, 是问题的决策变量. $y_j = 1$ 表示节点 j 是一个服务节点, 相反, $y_j = 0$ 表示该节点不是一个服务节点.

2 服务设施放置算法

所提出的基于禁忌搜索策略的启发式服务设施放置算法以迭代方式来增添网络中的服务节点数目, 每次迭代将进一步被分解成 2 个阶段: (扩展) 服务节点添加阶段以及放置模式调整阶段. 前一阶段是指向网络中引入一个新的服务设施, 并将其放置于在不改变所有现有服务节点位置前提下使总效用 R 达到最大的节点上, 这种方式显然是基于贪婪式策略的. 放置模式调整阶段依然以迭代方式来实现, 借助禁忌搜索策略的放置模式进行微调, 以进一步提高扩展服务节点所产生的总效用.

在所提出的算法中, 禁忌搜索策略的实现建立在“搜索域(search domain)”这一概念之上的, 而搜索域则是指以某个服务节点为中心的节点子集. 令 S_j 表示以服务节点 j 为中心的搜索域, 那么它将是

$$1) j \in S_j$$

$$2) \text{ 当 } i \neq j \text{ 时, } i \in S_j \text{ 当且仅当}$$

$$\begin{cases} y_i = 0 \\ d_{ij} \leq r \\ i, j \text{ 之间最短路径上不存在除 } j \text{ 之外的服务节点} \end{cases}$$

这里, 条件 2) 中的 r 被称作搜索半径, 是一取值为正的常量. 条件 2) 中第 3 个分支所排除的节点被称为由其他服务节点所阻塞的节点, 而这种现象被称作是非中心服务节点的阻塞效应. 图 1 显示以服务节点 1 为中心不同搜索半径下的搜索域, 用于对搜索域这一概念进行简要解释. 当 $r = 2$ 时, 搜索域为 $\{1, 2, 3, 4, 6, 7, 8, 9, 10, 11\}$. 其中, 节点 5 和 13 是异于搜索域中心的其他服务节点, 节点 12 被服务节点 5 所阻塞. 类似地, $r = 3$ 时的搜索域为 $\{1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 15, 16, 17\}$.

基于上述论述, 放置模式调整阶段内的每次迭代可以描述如下.

1) 对于每个扩展服务节点, 在以自身为中心的搜索域内为中心节点上的服务设施找到一个最佳候选置换位置, 使得在不改变其他扩展服务节点位置的前提下, 将搜索域中心节点上的服务设施移动到被选定的最佳候选节点位置时的总效用大于将该服务设施放置在搜索域内的其他节点位置时的总效用.

2) 如果步骤 1) 中所有服务设施在放置位置上的移动都不能使总效用获得一个正向增量, 那么退

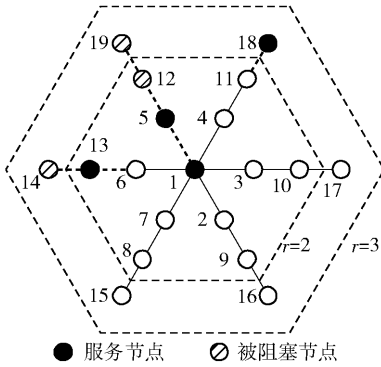


图 1 以服务节点 1 为中心的搜索域

出放置模式调整阶段。否则,选出那个能使总效用增量达到最大的扩展服务节点以及相关联的放置位置调整方式。

3) 执行由步骤 2) 所确定出的关于服务设施在放置位置的调整。

4) 返回步骤 1), 重复执行上述步骤。

最终,表 1 列举了算法伪码描述以及性能分析中将会用到的符号和表示,算法 1 正式描述了所提出的算法。

表 1 算法描述与分析中的符号说明

符号	含义
V_R	全体服务节点集合
$V_{R-fixed}$	预置服务节点集合
V_{R-free}	扩展服务节点集合
V_Q	全体请求节点集合
$V_{Q,j}$	指派给服务节点 j 的请求节点集合
S_j	以服务节点 j 为中心的搜索域
$N = \ V\ $	网络中节点总数
γ	网络中的扩展服务节点数目
$p = \ V_Q\ $	网络中请求节点总数
γ	服务设施 γ 的放置位置
i	指派给请求节点 i 的服务节点
$R(\gamma \mapsto j)$	不改变现有服务节点位置,将网络中引入的第 γ 服务设施放置在结点 j 位置时的总效用
$R(j_1 \mapsto j_2)$	不改变其他服务节点位置,将服务节点 j_1 上的服务设施移动到节点 j_2 位置时的总效用

算法 1 基于禁忌搜索策略服务设施放置算法
输入:

1 网络拓扑,需求配置,预置服务节点

2 $\gamma \leftarrow 0$

服务节点添加阶段:

3 $\gamma \leftarrow \gamma + 1$

4 $\tilde{R}^{(0)} \leftarrow \max_{j \in V} R(\gamma \mapsto j)$ (计算以贪婪策略引入一个新的服务设施时的总效用)

5 $\gamma \leftarrow \arg \max_{j \in V} R(\gamma \mapsto j)$ (返回能达到这一总效用服务设施的放置位置)

6 $y_{\gamma} \leftarrow 1$

7 $m \leftarrow 0$

放置模式调整阶段:

8 $m \leftarrow m + 1$

9 for all $j \in V_{R-free}$ do

10 $y_j \leftarrow 0$

11 $\tilde{R}_j^{(m)} \leftarrow \max_{s \in S_j} R(j \mapsto s)$

12 $i \leftarrow \arg \max_{s \in S_j} R(j \mapsto s)$ (寻找与指定服务节点相对应的置换节点)

13 $y_{ij} \leftarrow 1$

14 $\tilde{R}^{(m)} \leftarrow \max_k \tilde{R}_k^{(m)}$

15 if $\tilde{R}^{(m-1)} < \tilde{R}^{(m)}$ then

16 $j \leftarrow \arg \max_k \tilde{R}_k^{(m)}$ (确定最优置换节点)

17 $y_j \leftarrow 0$

18 $y_{ij} \leftarrow 1$

19 for all $j \in V_{R-free}$ do

20 if $R_j = 0$ then (重新放置处理过程中出现的空闲扩展服务设施)

21 $y_j \leftarrow 0$

22 $\gamma \leftarrow \gamma - 1$

23 else

24 go to Line 8

输出:

25 $R(\gamma) \leftarrow \tilde{R}^{(m-1)}$

26 go to Line 3 until $\gamma = p$

3 性能分析

首先给出算法 1 的一个性质,接下来对算法复杂度进行分析。

定理 假设网络规模有限。当 $\gamma = p$ 时,执行算法 1 能够使扩展服务设施全部被放置在请求节点上,此时的服务设施放置模式是理论最优的。

证明

1) 对任意 $j \in V_{R-free}$, 当 $|V_{Q,j}| = 1$ 时,考虑在从 j 到 $V_{Q,j}$ 中唯一请求节点之间的最短路径上的所有节点。假设这些节点中的每一个都有可能成为一个单

独服务节点,那么距离请求节点越近,服务节点的效用也就越大.按照这一原理,模式调整阶段将会使位于节点 j 上的服务设施最终移动到 $V_{Q,j}$ 中唯一的请求节点位置上,这一请求节点将成为能为自身服务的节点.

2) 由算法的第 19 ~ 22 行可知, V_{R-free} 不包含效用为 0 的服务节点,意味着对任意 $j \in V_{R-free}$, $V_{Q,j}$ 非空. 因此,当 $p = |V_Q|$ 时,每个 $V_{Q,j}$ 只能包含一个请求节点.按照(1)的结论,这些请求节点同时也将成为服务节点.

3) 当每个请求节点上的需求都能被其自身的服务设施服务响应时,针对所有需求的通信成本都是最小的,这时总的通信成本将达到理论上的最优,且没必要再增添更多的扩展服务设施 \square

对算法 1 的时间复杂度分析相对复杂,需要对放置模式调整阶段的内在处理机制做更深入了解.首先,改变一个服务设施的放置位置可能引发 2 种后果:

- 1) 请求节点子集 $V_{Q,j}, j \in V_{R-free}$ 均保持不变;
- 2) 上述部分请求节点子集发生改变.

对第 1 种情况的分析相对简单,此时每个服务设施都是基于贪婪策略在范围有限的搜索域内进行放置位置上的调整.然而,对第 2 种情况的分析要复杂得多,此时某些服务设施在放置位置上的改变将会增加/减少相邻服务节点所服务的需求.如图 2 所示,这一现象是由 2 种因素所引起的,分别称之为排斥效应和吸引效应.进一步,排斥效应是指对于一个请求节点而言,原本指派给它的服务节点上的服务设施被移动到一个与该请求节点距离较远的位置,使得这个请求节点到某个其他服务节点的距离更近,导致这个距离更近的服务节点被指派给该请求节点.类似地,吸引效应是服务设施放置位置发生改变之后,对于某些请求节点而言,被更新后的服务节点与该请求节点间的距离小于该请求节点与其原始服务节点之间的距离,最终导致更新后的服务节点被指派给该请求节点,从而使部分 $V_{Q,j}$ 中的元素发生改变.

无论排斥效应还是吸引效应都有可能引发连锁反应,导致服务设施放置模式调整阶段内的迭代过程持续进行.在上述迭代过程中,尽管不能保证所有服务节点的效用都是不变或者递增的,但算法第 15 行确保每次迭代后的网络总效用都是递增的,同时这一增量不会小于如下门限值:

$$\delta = \min \{ |a_{i_1} - a_{i_2}| \mid a_{i_1} \neq a_{i_2}, \forall i_1, i_2 \in V_Q \}$$

上述分析推导出了算法最坏情况下的时间复杂度,进一步考虑 $p = |V_Q|$ 这一极限情况,用 R_{max} 表示这时的总效用.那么,算法最多经历 R_{max}/δ 次服务设施放置位置上的移动.当所有需求均为有限值时,这一结果可以表示为 $O(N)$. 进一步,每次服务设施放置位置上的移动最多涉及 p 次候选位置的搜索,每次搜索都将在一个节点数目不超过 N 的范围内不断尝试.每次尝试最多改变 $|V_Q|$ 个请求-响应指派关系,而这一过程的时间复杂度将不超过 $O(Np)$. 综上所述,整个算法的时间复杂度不超过 $O(N^3p^2)$.

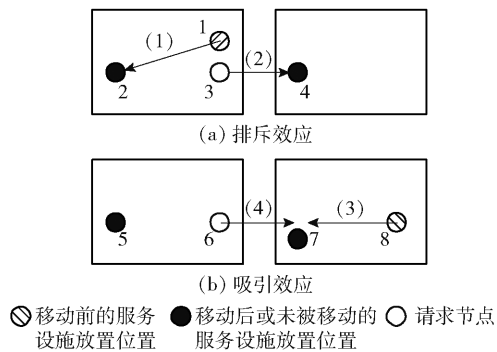


图 2 模式调整阶段的排斥效应和吸引效应

4 仿真结果

通过一个代表性的数值仿真来评估算法 1 (PA, proposed algorithm) 的性能. 仿真中, 搜索半径被设置为 2. 出于比较目的, 仿真同时考虑了另外 2 种经典算法: 穷举算法 (EA, exhaustive algorithm) 和贪婪算法 (GA, greedy algorithm) [7]. 穷举算法是指以暴力搜索的方式遍历整个解空间, 通过逐次比较确定问题的最优解, 而贪婪算法是指仅使用贪婪策略方式来依次确定出每个服务设施的放置位置, 即在不改变现有服务节点位置的前提下为新增服务设施寻找能使总效用达到最大的节点位置. 出于一般性考虑, 仿真中的网络拓扑是节点总数以及请求节点数目 (每个请求节点配置单位需求) 给定时的随机图, 其中任意 2 个节点相邻的概率被设置为 0.3. 主要仿真项目包括算法的归一化程序运行时间以及归一化平均性能差距, 结果如表 2 和表 3 所示. 出于拓扑上的随机性, 表中的每个数据都是 100 次独立实验的平均结果.

根据表 2 可以看出, EA 运行时间确实是网络规模 N 的指数级函数. 另外, 该运行时间也受参数 p

的影响,即 N 和 p 同时增加时运行时间的变化幅度大于只有 N 增加时的情况,这一结论对另外 2 种算法依旧适用. 比较显示,PA、EA 运行时间随网络规模递增幅度要明显小于 EA. 尽管 PA 的运行时间略高于 GA,但相对于 EA,这一差距是可接受的. 表 3 展示了 PA 和 GA 相对于由 EA 所确定的理论最优值之间的归一化的平均性能差距. 可以看出,相对于 GA-EA 间的性能差距,PA-EA 之间的差距更小. 因此,PA 在性能与算法复杂度之间展示了一种新的折中关系,这对比较关注性能的应用来说具有重要意义.

表 2 仿真 3 中不同算法的归一化程序运行时间

N	p	PA	GA	EA
12	6	1	0.264 4	34.019
14	8	1.780 5	0.445 3	240.520
16	8	2.288 9	0.555 2	837.310
18	10	3.772 0	0.810 1	5 358.800
20	10	4.454 7	0.922 1	17 725.000
22	12	6.798 6	1.295 3	109 889.000

表 3 仿真 3 中不同算法的归一化性能差距

N	p	PA 与 EA 性能差距/ 10^{-3}	GA 与 EA 性能差距/ 10^{-3}
12	6	1.407	10.490
14	8	1.848	8.726
16	8	3.151	14.480
18	10	1.802	7.777
20	10	2.029	12.890
22	12	1.618	9.726

5 结束语

研究了移动边缘计算中的服务设施放置问题,将其建模成一个 $(p+m)$ -中点问题,进而提出了一种基于禁忌搜索策略的启发式放置算法. 提出的算法有着广泛的适用性,能够用于任意拓扑和需求分布的数据网络. 相对于贪婪算法,提出的算法在性

能和复杂度之间实现了更好的折中:提出的算法有着更接近于理论最优的性能,但仍是多项式时间的. 当所引入的服务设施数目等于请求节点数目时,执行该算法能够得到最优放置模式. 为了使提出的算法更具有实用价值,下一步将研究对不同链路上的通信成本予以区分,关注此链路权值在成本计算方面的作用,这将有助于与一些更实用的性能评估指标,如 QoS、可用信道带宽以及内容获取时延等建立更为深入地联系.

参考文献:

[1] 田辉, 范绍帅, 吕昕晨, 等. 面向 5G 需求的移动边缘计算[J]. 北京邮电大学学报, 2017, 40(2): 1-10.
Tian H, Fan S, Lv X, et al. Mobile edge computing for 5G requirements [J]. Journal of Beijing University of Posts and Telecommunications, 2017, 40(2): 1-10.

[2] Gao L, Luan T H, Yu S, et al. FogRoute: DTN-based data dissemination model in fog computing [J]. IEEE Internet of Things Journal, 2017, 4(1): 225-235.

[3] Kariv O, Hakimi S L. An algorithmic approach to network location problems. I: the p -centers[J]. Siam Journal on Applied Mathematics, 1979, 37(3): 513-538.

[4] Kariv O, Hakimi S L. An algorithmic approach to network location problems. II: the p -medians [J]. Siam Journal on Applied Mathematics, 1979, 37(3): 539-560.

[5] Tamir A. An $O(pn^2)$ algorithm for the p -median and related problems on tree graphs[J]. Operations Research Letters, 1996, 19(2): 59-64.

[6] Rolland E, Schilling D A, Current J R. An efficient tabu search procedure for the p -median problem[J]. European Journal of Operational Research, 1997, 96(2): 329-342.

[7] Charikar M, Guha S, Shmoys D B. A constant-factor approximation algorithm for the k -median problem [J]. Journal of Algorithms, 2003, 65(1): 129-149.

[8] Lorena L A N, Furtado J C. Constructive genetic algorithm for clustering problems[J]. Evolutionary Computation, 2001, 9(3): 309-327.