第34卷 第22期 计 算 机 工 程 2008年11月
Vol.34 No.22 Computer Engineering November 2008
·人工智能及识别技术· 文章编号：1000—3428(2008)22—0213—02 文献标识码：A 中图分类号：TP311

K-median问题贪心近似算法的分析与实验

肖进杰，谢青松，刘晓华
(山东工商学院信息与电子工程学院，烟台 264005)

摘 要：讨论 K-median 问题的贪心近似算法及其在实际计算中的表现。提出一个解 K-median 问题的贪心算法，证明该算法的近似度为 $O(\ln(n/k))$，通过实验证明该贪心算法在实际应用当中可以取得较好的效果，大约有 90%的客户能被距离其最近、次近和第三近的设备服务。
关键词：K-median 问题；贪心算法；近似算法

Analysis and Test of Approximation Greedy Algorithm for K-median Problems

XIAO Jin-jie, XIE Qing-song, LIU Xiao-hua
(School of Information and Electronic Engineering, Shandong Institute of Business and Technology, Yantai 264005)

【Abstract】This paper discusses approximation greedy algorithm for K-median and its property in actual computation. It presents a greedy algorithm for K-median and then proves that the approximation ratio of the greedy algorithm is at most $O(\ln(n/k))$. The test data shows that the greedy algorithm can get good results in actual computation and about 90% clients can be serviced by the facility whose distance to the client is the 1st, 2nd, 3rd.
【Key words】K-median; greedy algorithm; approximation algorithm

1 概述

K-median问题是一个有重要实用意义和研究意义的算法问题。假设有n个设备集合$F=\{F_1,F_2,\cdots,F_n\}$和m个客户集合$C=\{C_1,C_2,\cdots,C_m\}$，设备表示准备打开的服务中心，客户由设备为其提供服务，一个客户只需一个设备服务即可。客户和设备又分别视为空间中的点，因此取$V=F\cup C=\{v_1, v_2,\cdots,v_{m+n}\}$，且$v_j$到$v_i$的距离记为$c_{ij}$。若$v_j=c_j\in C$, $v_i=F_i\in F$，则c_{ij}表示客户c_j被设备F_i服务的代价。若选定设备子集$S\subseteq F$为所有客户服务，则服务成本为：$Cost(S)=\sum_{v_i\in C}c_{\sigma(v_i)v_i}$，其中，$\sigma(v_i)\in F$表示服务于客户$C_i$的设备。欲求一个设备集合$S\subseteq F$ ($|S|\leq k$表示集合S中设备的个数)，打开S中的设备为所有客户提供服务，使总的成本最小。若距离满足$c_{ij}=c_{ji}$且$c_{ij}+c_{jk}\geq c_{ik}$，其中，$v_i,v_j,v_k\in F\cup C$，则称该问题为公制空间K-median问题；若距离不满足上述的对称和三角不等式，则称该问题为非公制空间的K-median问题。

K-median问题在电网中心的确定、计算机网络中服务器的配置、物流中心的选择等方面均有重要的实用价值。Kariv和Hakimi[1]于 1979 年证明，即使在最大度为 3 的平面图上K-median问题也是NP-Complete问题；Papadimitriou[2]于 1981 年证明欧氏平面上的K-median问题是NP-Complete。现在也已经提出了很多解决K-median问题的近似算法，Charikar[3]等人于 1999 年利用线性规划技术设计了第 1 个K-median近似度$6\frac{2}{3}$的近似算法，Arya[4]等人于 2001 年用启发式局部搜索得到了 3+ε (其中$\varepsilon>0$)的近似算法。对于用贪心算法求解K-median问题，Marek Chrobak[5]等人于 2005 年用逆贪心算法得出K-median近似算法界于$\Omega(\log n/\log\log n)$与$O(\log n)$之间。

2 近似算法

公制空间 K-median 问题的形式化描述如下：

输入：设备集合$F=\{F_1,F_2,\cdots,F_n\}$，客户集合$C=\{C_1, C_2,\cdots,C_m\}$，任意 2 点$v_i,v_j\in F\cup C$，给定距离$c_{ij}=d(v_i,v_j)$。其中，$c_{ij}=c_{ji}$且$c_{ij}+c_{jk}\geq c_{ik}$；$v_i,v_j,v_k\in F\cup C$；$k$为正整数($k\leq n$)。

输出：找到一个设备子集$S\subseteq F$且$|S|\leq k$，使得$\min\{Cost(S)=\sum_{v_i\in C}c_{\sigma(v_i)v_i}\}$，其中，$\sigma(v_i)\in F$表示服务于客户$C_i$的设备。

最理想的解为每个客户都由距离其最近的设备服务。每一个设备可能服务零个、一个或多个客户，假设设备$F_i(1\leq i\leq n)$服务的客户的集合是$N(F_i)$，其中$N(F_1)\cup N(F_2)\cup\cdots\cup N(F_n)=C$，并且对于任意的$1\leq i,j\leq n$，都有$N(F_i)\cap N(F_j)=\phi$，如图 1 所示。

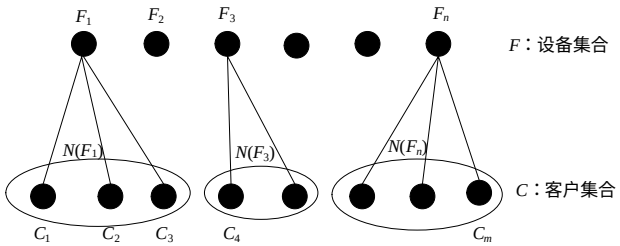

图 1 K-median 问题设备服务客户图示

下面近似算法的求解采用的是贪心算法。基本思想是把n个设备加入选中集合中，然后选择服务客户数量最少的设

作者简介：肖进杰(1979－)，男，讲师、硕士，主研方向：算法及其复杂性；谢青松、刘晓华，副教授
收稿日期：2008-03-03 **E-mail：**xiaojinjie@hotmail.com

备将其从集合中删除，把该设备服务的客户分配给剩余的其他设备，重复此过程直至选中集合中只剩下 k 个设备。

贪心算法形式化描述如下：

(1)初始状态。令 $S=\{F_1,F_2,F_3,\cdots,F_n\}$，即选中 n 个设备，每一个客户都由距离其最近的设备服务。

(2)对任意 $F_i\in S$，计算 $N(F_i)$，即计算 S 中每一个设备服务的客户的数量。

(3)取 $p_0=\{p\in S\mid \min N(p)\}$，$S=S-\{p_0\}$。

(4)对每一个 $c_i\in N(p_0)$，取 $f_0=\{f\in S\mid \min\{c_{c_i f}\}\}$，则 $N(f_0)=N(f_0)\cup\{c_i\}$。

(5)回到(2)直至 $|S|=k$，即 S 中只剩下 k 个设备。

假设设备集合的数量为n，客户集合的数量为m，那么客户和设备之间的距离可以用一个$m\times n$的二维数组来表示，可得上述算法总的时间复杂度为$O(m\times n^2)$。

3 算法近似度的证明

为了证明算法的近似度，给出下面的定义和引理。

定义 假设 n 个设备和 m 个客户的所有距离之中，最大的距离为 $d_{\max}$，最小的距离为 $d_{\min}$，且 $\alpha=\dfrac{d_{\max}}{d_{\min}}$。

引理 1 $1+\dfrac{1}{2}+\dfrac{1}{3}+\dfrac{1}{4}+\dfrac{1}{5}+\cdots+\dfrac{1}{n}=\ln n+R$ (R 为欧拉常数)。

引理 2 假设 I 是存在可行解的一个实例，$A(I)$ 为上述贪心算法求得的解对应的 I 的目标函数值，$Opt(I)$ 为其最优值。假设采用上述贪心算法从 n 个设备中选择 k 个所需的服务成本记为 $Cost_k(N)$，则 $Cost_n(N)\le Cost_{n-1}(N)\le\cdots\le Cost_k(N)$，并且满足 $A(I)=Cost_k(N)$，$Opt(I)\ge Cost_n(N)$。

证明：根据定义 $Cost_n(N)$ 表示n个设备都被选中，此时每一个客户都被其距离最短的设备服务，服务成本最小。根据贪心算法，下一步要从这n个设备中选择服务客户数量最少的设备，假设为p_0，对于客户 $c_i\in N(p_0)$ 此时只能分配给剩下的 n-1 个设备，且 c_i 与其距离肯定大于等于c_i与p_0的距离，所以 $Cost_n(N)\le Cost_{n-1}(N)$。同理可得 $Cost_{n-1}(N)\le Cost_{n-2}(N)$，$Cost_{n-2}(N)\le Cost_{n-3}(N)$，$\cdots$，$Cost_{k+1}(N)\le Cost_k(N)$，故可得 $Cost_n(N)\le Cost_{n-1}(N)\le\cdots\le Cost_k(N)$。

根据定义显然可得 $A(I)=Cost_k(N)$，$Opt(I)\ge Cost_n(N)$。

定理 1 设 I 是一存在可行解的 K-median 问题的实例，$A(I)$ 为上述贪心算法求得的解对应的 I 的目标函数值，$Opt(I)$ 为其最优值，则 $\dfrac{A(I)}{Opt(I)}\le 1+(\alpha-1)\ln\dfrac{n}{k}$。

证明：初始状态n个设备均被选中，此时假设服务客户数量最少的设备为p_0，则p_0至多服务$\lceil\dfrac{m}{n}\rceil$个客户，现在要将这$\lceil\dfrac{m}{n}\rceil$个客户分配给其他的设备，每一个客户重新分配后的距离比分配前最多相差 $(d_{\max}-d_{\min})$，从而可得

$$Cost_{n-1}(N)-Cost_n(N)\le\frac{m}{n}(d_{\max}-d_{\min})$$

同理可得

$$Cost_{n-2}(N)-Cost_{n-1}(N)\le\frac{m}{n-1}(d_{\max}-d_{\min})$$

$\cdots$

$$Cost_k(N)-Cost_{k+1}(N)\le\frac{m}{k+1}(d_{\max}-d_{\min})$$

以上各式左右两端分别相加可得

$$Cost_k(N)-Cost_n(N)\le\left(\frac{1}{n}+\frac{1}{n-1}+\frac{1}{n-2}+\cdots+\frac{1}{k+1}\right)\times m\times(d_{\max}-d_{\min})=\ln\frac{n}{k}\times m\times(d_{\max}-d_{\min})$$

根据引理 2 可得

$A(I)=Cost_k(N)$，$Opt(I)\ge Cost_n(N)$

从而

$$\frac{A(I)}{Opt(I)}\le\frac{Cost_k(N)}{Cost_n(N)}\le 1+\frac{\ln\frac{n}{k}\times m\times(d_{\max}-d_{\min})}{Cost_n(N)}\le 1+\frac{\ln\frac{n}{k}\times m\times(d_{\max}-d_{\min})}{m\times d_{\min}}=1+(\alpha-1)\ln\frac{n}{k}\le 1+\frac{\ln\frac{n}{k}\times m\times(d_{\max}-d_{\min})}{m\times d_{\min}}=1+(\alpha-1)\ln\frac{n}{k}$$

定理 2 上述贪心近似算法的复杂度至多为 $O(\ln(n/k))$。

通过定理 1 的证明显然可得该定理。

4 贪心算法实验数据

实验数据主要包括 2 部分：一部分是贪心算法得到的服务成本与最优解的服务成本之间的实际比值，这部分数据由于求最优解需要花费很长时间，故只对 20 个设备进行处理；另外一部分是测试每一个客户通过上述的贪心算法需要改变几次服务的设备，显然如果没有改变，则该客户由距离其最近的设备服务。如果改变一次则该客户由距离其次近的设备服务，如果改变 x 次则该客户由距离其第 x 近的设备服务。

4.1 实例的生成

公制空间的 K-median 问题客户和设备之间的距离要求是对称的并且满足三角不等式，对此采用字符串之间的海明距离来实现，具体做法是：随机产生 n 个和 m 个只包含 A, B 的字符串作为设备字符串和客户字符串，计算这些字符串之间的海明距离作为设备客户点之间的距离，很容易证明这样产生的距离是对称的并且满足三角不等式。

4.2 最优解的生成

全局最优解的求法采用的是穷举法，即采用组合的生成算法[6]，从n个设备集合中分别选出 1, 2, …, k个设备子集来服务整个客户集合，然后计算各自的服务成本，选出最小的那个设备子集作为全局最优解。

4.3 实验数据

本文中程序运行的计算机 CPU 为 P4-3.8 GHz，内存为 1 GB DDR，硬盘 160 GB。

4.3.1 近似性能比的平均值

这部分进行的试验中设备和客户之间的距离c_{ij}满足 $0\le c_{ij}\le 97$，设备数量n固定在 20。

此部分对 k=5,8,10,12,15,18 分别做了 50 次实验，分别求出用贪心算法得到的服务成本和用组合生成算法求得最优成本，算出其比值，然后取 50 次的平均值，数据如表 1 所示。

表 1 服务成本和全局最优的平均比值

m 取值	k=5	k=8	k=10	k=12	k=15	k=18
m=20	1.010 095	1.004 986	1.001 967	1.000 000	1.000 000	1
m=30	1.008 722	1.007 174	1.003 481	1.001 925	1.000 000	1
m=40	1.010 214	1.005 601	1.004 942	1.002 880	1.000 456	1
m=50	1.008 044	1.006 825	1.003 790	1.002 962	1.001 145	1

从表 1 可以看出通过上述贪心算法可以得到较好的近似性能比，特别是表中比值为 1 表示 50 次实验用贪心算法得到的服务成本和最优解每次都是相等的。 (下转第 217 页)