文章编号: 1000-6788(2004)05-0061-05

并购后销售服务网络的优化问题

徐寅峰, 沈凤武
(西安交通大学管理学院管理科学系, 陕西 西安 710049)

摘要: 研究的是企业横向并购后的销售服务网络的优化集成问题, 文章以两个公司的横向并购为例, 构造了可变策略下的系统优化模型. 算法设计思想是: 对待合并后的销售服务网络, 采用"休克"式设计程序, 暂时对所有的销售服务网点"视而不见". 假设每个销售服务网点的运营能力相同, 那么, 所要研究的问题即转化为: 在现有客户条件下, 若一个销售服务中心可以为 k 个用户提供服务, 求, 如何确定客户销售服务中心的关系以使得销售服务中心总的效率最高. 基于这种策略, 文章给出了问题的优化模型及其求解算法, 同时分析了该算法的时间复杂性. 所得到结果与已有的研究结果具有较强的互补性.

关键词: 并购; 算法; 计算复杂性; k-子集拆分
中图分类号: F830　　**文献标识码**: A

Sales and Services Network Optimization after Merger and Acquisition

XU Yin-feng, SHEN Feng-wu
(Department of Management Science, Xi'an Jiaotong University, Xi'an 710049, China)

Abstract: The optimization problem of sales and services network after merger and acquisition is studied in this paper. Constructed optimized model in non-fixed strategy, all the sales centers could be changed and rearranged. The idea of our algorithm is: Using "shock" program to sales network after merger. We design our new network as if all the centers did not exist. We just optimize the new network according to the customer set. If each sales center is identical, the problem can be formulated as how to construct the sales center network to minimize the total cost as: if one sales center can serve k customers. The algorithm and its time complexity is discussed at the final part. The result is complementary with the existed research results.

Key words: merger; algorithm; complexity; k-partition

1 背景

两个公司发生横向并购(本文研究的是横向并购, 若非特别指出文中并购均指横向并购)后, 原来分别属于两个独立公司的销售服务网络中的销售服务点所对应的服务区域必然会发生交叉或重叠. 但在很多实施购并的企业中, 并没有运用科学的方法对销售服务网络加以优化, 多数企业采用的是简单的删减, 而且多数是被并企业的销售服务网点被裁减掉. 这种方法并不能保证被并企业原来的销售服务网点被裁减掉后, 销售服务网络的运营效率就变高了、成本就下降了.

本文研究的目的是对发生交叉、重叠区域所对应的销售服务网点进行优化评价, 删除冗余的销售服务网点, 使得两个公司合并后的销售服务分支机构的数量最少且效率极大的高于合并前任一公司的效率. 沈凤武与徐寅峰曾给出了关于横向并购初期固位策略下的销售服务网络结构优化构造方法, 其算法的计算复杂性为 $O((m+n)n^2\log n)$ [1]. 与固位策略相对应, 本文采用了可变策略对销售服务网络进行优化构造, 其含义是销售服务中心的位置在优化过程中可以变更或增添新的销售服务网点.

收稿日期: 2003-06-05
资助项目: 国家自然科学基金(10371094)
作者简介: 徐寅峰, 男, 教授, 博士生导师; 沈凤武(1968), 男, 博士研究生

2 理论基础

一个给定多元素集合的最小费用拆分问题以及出现很多应用领域, 关于这个问题文献[2- 5]的研究得出这个模型:

$$\min F(P) = \sum_{i=1}^{k} f(S_i)$$

$P = S_1 \cup S_2 \cup \cdots \cup S_k$ 是 n 元素的实数集合 $Z = \{z_1, \cdots, z_n\}$ 的一个拆分, $f: 2^z \to R$.

当 k 个任意值的时候, 我们简单的称上面的问题为拆分问题. 当 k 是个固定值时, 称这类问题为 k 拆分问题. Burkard, F. E 曾讨论了一种更特殊的情况, 不仅 k 是固定的, 而且, $M = \{|S_1|, |S_2|, \cdots, |S_k|\}$ 也是固定的, 并称这种问题为定形拆分(shape-partition) 问题[6]. 拆分后的任意两个子集 S_i 和 S_j 中元素不交叉即 $S_i \cap S_j = \varnothing$, 我们称这为有序拆分(ordered partition), 如果总存在一个最优拆分是规则的, 那么, 我们说 f 具有这样的性质:

性质 1 OP= OOP

最大权 k 子集拆分问题源于经济问题和工业问题, 某些组合优化的问题也可以化为这一模型, 在过去的研究中曾考虑过这样的问题, 如, 由 n 个工厂组成一些公司, 由于产品结构与加工工艺以及产业链的结构顺序的因素, 要求每个公司至多由 k 个工厂组成, 为了减少管理方面的费用及公司的运作效率, 要求公司的尽可能的少, 于是, 取公司的个数为 $\left\lceil \frac{n}{k} \right\rceil$, 其中 $\left\lceil \frac{n}{k} \right\rceil = \min\{x \mid kx \geq n, x$ 为一个整数$\}$. 记 N 为 n 个工厂的集合, N 的每个子集合称为公司, 对每个公司 S, 令 $w(S)$ 为此公司所产生的利润, $n = |N|$, 于是问题可表述如下:

$$\max \sum_{i=1}^{\lceil n/k \rceil} w(S_i)$$

满足:

$$\begin{cases} \bigcup_{i=1}^{\lceil n/k \rceil} S_i = N \\ S_i \cap S_j = \varnothing, \ i \neq j \\ |S_i| \leq k, i = 1, 2, \cdots, \left\lceil \frac{n}{k} \right\rceil \end{cases}$$

在文献[7]中研究了在油田的油井网络设计中存在的类似问题, 有 n 个油井需要在周围建立起一些原油提炼厂, 由于技术方面的原因, 一个提炼厂至多能够容纳 k 个油井. 提炼厂一般被称为 k 个油井的集中点 在油井和提炼厂之间有管线相连接, 每个油井仅属于一个提炼厂, 为减少将提炼厂的费用, 提炼厂的个数一般取为 $\left\lceil \frac{n}{k} \right\rceil$, 问题是如何确定油井和提炼厂的所属关系已使得中的效率最高成本最小.

3 模型的建立

现以两个公司的购并为例来介绍一下这类问题的优化过程.

公司 1 的销售服务中心集合用 $S_1 = \{a_1, a_2, \cdots, a_n\}$ 来表示;

公司 2 的销售服务中心的集合用 $S_2 = \{b_1, b_2, \cdots, b_m\}$ 来表示;

客户的集合用 $C = \{h_1, h_2, \cdots, h_{|C|}\}$ 来表示.

本文研究的策略是, 对待合并后的销售服务网络, 采用"休克"式设计程序, 暂时对所有的销售服务网点"视而不见", 好像他们并不存在一样, 但可以变更、迁移或添加新的网点(销售服务中心) 在此仅考虑现有客户群以及公司的销售服务能力, 基于这些重新构造新的销售服务网络.

假设每个销售服务网点的运营能力相同, 那么我们的问题及转化为, 在现有客户条件下, 若一个销售服务中心可以为 k 个用户提供服务, 求, 如何确定客户与销售服务中心的关系以使得销售服务中心总的效率最高.

模型 1 记 $r = |C|$(所有客户的数量), 销售服务中心的数量为 $\left\lceil \frac{r}{k} \right\rceil$, 令 $w(h_i)$ 为该销售服务中心的效率. 求:

$$\max \sum_{i=1}^{\lceil \frac{r}{k} \rceil} w(h_i),$$

满足:

$$\begin{cases} \bigcup_{i=1}^{\lceil \frac{r}{k} \rceil} h_i = C \\ h_i \cap h_j = \varnothing, i \neq j \\ |h_i| \leq k, i = 1, 2, \cdots, \lceil \frac{r}{k} \rceil \end{cases}$$

模型 2　对于子集 $h_i = \{c_{i1}, c_{i2}, \cdots, c_{ik}; q = 1, 2, \cdots, k\}$, 定义 $d(c_{ii}, c_{ij})$ 为点 c_{ii} 与 c_{ij} 之间的欧氏距离, $w(c_{iq})$ 为客户 c_{iq} 的权, 其含义是该点(客户)到其他点(客户)的最大欧式距离, 记 $w(c_{iq}) = \max d_{iq}$. 对每一个子集 h_i 求解 $c^* \in h_i$, 即销售服务中心的位置. 满足以下条件:

$$\min_{c^* \in h_i} \max_{c_{ii} \in h_i} d(c^*, c_{ii})$$

4　算法设计

4.1　模型 1 的算法及复杂性

4.1.1　记号与假设

对给定集合 C, 记 2^C 为 C 的所有子集的集合. $V \subseteq 2^C$ 称为 C 的一个 k-子集拆分(k-partition), 当且仅当:

$$\begin{cases} |V| = \lceil \frac{r}{k} \rceil \\ h_i \cap h_j = \varnothing, i \neq j, \text{ 任意 } h_i, h_j \in V \text{ 且 } h_i \neq h_j \\ |h_i| \leq k, h_i \in V \\ \bigcup_{h_i \in V} h_i = C \end{cases} \tag{4.1}$$

记 $w(h_i)$ 为满足 $|h_i| \leq k$ 的子集 h_i 的权, 于是, 最大权 k-子集拆分问题既为在 N 的所有 k- 子集拆分上求一个 k-子集拆分 V 使得 $\sum_{h_i \in V} w(h_i)$ 达到最大.

如果, r 恰为 k 的倍数, 那么, 4.1 式中的复杂表达则可简化为:

$$\begin{cases} |h_i| = k, h_i \in V \\ |V| = \frac{r}{k} \end{cases} \tag{4.2}$$

容易证明 n 不是 k 的倍数的情况可以转化为 n 是 k 的倍数的情形. 实际上, 假设 $r = mk + l$, 浦 $1 \leq l \leq k-1$, 在集合 C 中加入 $k - l$ 个新的元素可得到一个新的集合记为 C^*. 于是, $|C^*| = (m+1)k$. 对任意 $S \subset C^*$, 取 $w^*(h_i) = w(h_i \cap C)$ 为 h_i 的权, 则有:

$$\max_V \sum_{h_i \in V} w(h_i) = \max_{V^*} \sum_{h_i \in V^*} w^*(h_i) \tag{4.3}$$

其中 V, V^* 分别为 C 和 C^* 的 k-子集拆分. 在以下讨论中, 如无特殊性说明均假设 r 为 k 的倍数, 且将最大权 k-子集拆分问题简记为 kSP.

4.1.2　计算

1) 考虑 $k = 2$ 的情况

令 $C = \{h_1, h_2, \cdots, h_{|C|}; l = 1, 2, \cdots, r\}$, w_{ij} 为 N 的子集合 $\{x_i, x_j\}$ 的权, G 为以 $h_1, h_2, \cdots, h_{|C|}$ 为顶点的完全图, 点 x_i 与 x_j 之间的边的权为 w_{ij}. 注意到此时 $|C|$ 为偶数, 这显然有 C 的最大权 2-子集拆分恰好对于加权图 G 的最大权匹配. 由于一个完全图的最大权匹配可在 $O(n^4)$ 时间内求出[8], 其中 n 为图的顶点个数, 于是我们有如下结果:

定理 1　集合 C 的 $2SP$ 可在 $O(|C|^4)$ 时间内求出.

2) 考虑 $k \geq 3$ 的情况

下面将证明当 $k\geq 3$ 时 kSP 为NP 完全的. 首先看一下已被证明是NP 完全的三维匹配问题.

三维匹配问题(简记为3DM), 若存在一个集合 $M\subseteq W\times X\times Y$, 其中 W, X, Y 为具有相同元素个数的不相交集合, 问, 是否存在 M 的一个子集 $M'\subseteq M$, 满足 $|M'|=n$ 且对任意 (u,v,w) 与 (u',v',w') 属于 M' 且不相同的元素, 满足 $u\neq u', v\neq v', w\neq w'$.

为证明 3SP 为NP 完全, 我们想要证明如下引理.

引理 3DM 可多项式归约为 3SP

证明 给定一个 3DM 的例子, 可以构造一个 3SP 的例子如下:

令 $N=W\cup X\cup Y$, 对 N 中的任意一个含 3 个元素的子集 S 定义它的权为

$$w(S)=\begin{cases}2, & S\in M, \text{即} S \text{含} W, X, Y \text{中各一个元素}\\ 1, & \text{否则}\end{cases}$$

其中 $M\subseteq W\times X\times Y$. 显而易见, 算法可在多项式时间内完成. 接下来证明, M 包含一个三维匹配当且仅当 N 有一个 3-子集拆分满足:

$$\sum_{i=1}^{n} w(S_i)\geq 2n.$$

实际上, 如果 M' 为 M 的一个匹配, 那么 M' 恰含 n 个 M 的 3-元素不交子集. 于是, M' 为 N 的一个 3-子集拆分且 $\sum_{S\in M'} w(S)=2n$. 反之, 假设 N 有一个 3-子集拆分 V, 且满足 $\sum_{S\in V} w(S)\geq 2n$. 于是, V 必含 n 个 N 的 3-元素子集合, 且有 $w(S)=2, S\in V$. 由此可知对 $S\in V$ 有 $S\in M$. 即 V 为 M 的一个匹配. 证毕.

由于 3SP 可以用引理中类似的构造方法归约为 kSP, $k\geq 3$. 我们有如下推论.

推论 1 对 $k\geq 3$, 3DM 可多项式归约为 kSP.

由上述讨论可得如下结论:

定理 2 对 $k\geq 3$, kSP 为NP-完全的.

证明对于有限集合 N, 我们可以很容易地检验 N 的 n/k 个子集是否构成 N 的一个 3-子集拆分, 于是, kSP 为NP-完全的, 其中 $k\geq 3$.

3) 对于 $k\geq 3$ 情形的启发式算法

由定理 2 我们知道当 $k\geq 3$, kSP 为NP-完全的, 于是, 我们考虑求解 kSP 的启发式算法. 贪婪算法是一般的组合优化问题启发式算法最优先考虑的一个启发式算法. 由于 kSP 问题的特殊结构, 在后面给出的定理 3 中我们可以得到由贪婪算法求 kSP 问题所得到的解的一个比较好的性质. 首先我们给出用拟 Algol 语言所写的对应于 kSP 问题的贪婪算法:

```
begin
    T := {S | S ⊂ N, |S| = k};
    V := ∅;
    While |V| < n/k do
      begin
          choose S0 ∈ T such that w(S0) = max_{S∈T} w(S);
          V := V ∪ {S0}
          T := {S | S ∩ S0 = ∅, S ∈ T}
      end
    output V;
end
```

分析这个贪婪算法可得:

定理 3 贪婪算法求解 kSP 可在 $O(n^{k+1})$ 时间内完成, 且所得的节与最优解之比大于或等于 $1/k$, 其中 n 为输入集合 N 中元素的个数.

证明 令 $n=|N|$, 如上述贪婪算法中 T 至多含 $\binom{n}{k}$ 个元素, 由于算法计算 n/k 个具有局部最大权的 S_0, 所以总的运算次数之多为 $O(n^{k+1})$. 于是, 定理 3 中关于算法运算两部分的结论成立. 接下来考虑贪婪

算法所得到解的性质; 假设 $V=\{S_1, S_2, \cdots, S_m\}$ 为 N 的 kSP 的最优解, $V=\{S_1^*, S_2^*, \cdots, S_m^*\}$ 为贪婪算法的输出, 且不失一般性, 可以假设 $w(S_1^*)\ge w(S_2^*)\ge\cdots\ge w(S_m^*)$, 其中 $m=n/k$. 令

$$E_i=\{S \mid S\in V, S\cap S_i^*\ne\varnothing \text{ 且 } S\cap S_j^*=\varnothing, \text{对 } j<i \text{ 成立}\},\quad i=1,2,\cdots,m$$

于是, 有 E_i 包含至多 k 个元素, 且若 $S\in E_i$, 则显然有: $w(S)\le w(S_i^*)$, $i=1,2,\cdots,m$ 成立. 由下式 $\sum_{i=1}^{m} w(S_i)=\sum_{i=1}^{m}\sum_{S\in E_i} w(S)\le\sum_{i=1}^{m} kw(S_i^*)$ 得:

$$\frac{\sum_{i=1}^{m} w(S_i)}{\sum_{i=1}^{m} w(S^*)}\le k.$$

至此, 定理的第二部分的证.

故模型 1 的计算复杂性为 $O(n^{k+1})$.

4.2 模型 2 的算法及复杂性

4.2.1 算法

第一步　计算平面点集 h_i 的最远点 Voronoi 图, 记为 $V_F(h_i)$[9].

第二步　对 $\forall c_{ii}\in h_i$ 计算

$$w(c_{ii})=\max_{c_{ij}\in h_i}\{d(c_{ii}, c_{ij}), i\ne j=1,2,\cdots,n\}.$$

第三步　计算 $\min_{c_{ii}\in S}\{w(c_{ii})\}$ 让 $w(c^*)=\min_{c_{ii}\in h_i}\{w(c_{ii})\}$, $i=1,2,\cdots,n$.

4.2.2 计算复杂性

第一步计算的时间复杂性为 $O(n)$.

第二步计算的时间复杂性为 $O(n)$.

第三步计算的时间复杂性为 $O(n)$.

故模型 2 的整体计算的时间复杂性为 $O(n)$.

对于本文所需解决的问题, 由于, $O(n^{k+1})>O(n)$, 所以其算法的整体复杂性为 $O(n^{k+1})$.

5 结论

本文的优化思想是基于客户位置不变的基本假设来对并购后销售服务网络进行优化的, 这种假设在很多行业销售服务工作中是成立的, 具有较强的普遍性. 同时, 在优化策略上, 引入了灵活性较强的销售服务中心位置可变策略. $O(n^{k+1})$ 并不是一个最满意的结果, 由于通常企业的销售服务网点不会很大, 所以, 对本文所研究的问题而言, 这个结果是可行的. 关于并购后销售服务网络的后继研究我们将从 p-center 的角度对合并体进行优化.

参考文献:

[1] 沈凤武, 徐寅峰. 并购初期销售服务网络优化[J]. 中国管理科学, 2003, 5: 28-31.

[2] Barnes E E, Hoffman A J. Partition spectra and linear programming[A]. Proc Silver Jubilee Conf Combinatories [C]. Univ of Waterloo, Ontario, 1982.

[3] Chakravarty A K, Orlin J B, rothblum U G. Consecutive optimizers for a partitioning problem with applications to optimal inventory groupings for joint replenishment[J]. Operations Research, 1985, 33: 820-834.

[4] Hwang F K. Optimal partitions[J]. J Optimization Theory and Applications, 1981, 34: 1-10.

[5] Tanaev V S. Optimal subdivision of finite sets into subsets[J]. Akad Nank BSSR Doklady, 1979, 23: 26-28.

[6] Burkard F E. Constrained partition problems[J]. Disc Appl Math, 1990, 78: 21-31.

[7] 徐寅峰, 刘自成. 关于最大权 k 子集分拆问题[J]. 高校应用数学研究, 1994, 9(4): 453-457.

[8] Avis D. A survey of the weighted matching problem[J]. Network 1983, 13(4): 475-494.

[9] Nimrod Megiddo. Linear-time algorithms for linear programming in R^3 and related problems[J]. SIAM J COMPUT, 1983, 12(4): 758-774.