文章编号:1000-6788(2006)05-0099-07

一类带交货期约束的资源选择问题研究

李淑娟,李　言,曾志斌
(西安理工大学机械与精密仪器工程学院,陕西 西安 710048)

摘要:　研究生产计划和控制中带交货期约束且子任务之间具有先序关系的资源选择问题,证明了该问题是 NP 完全问题,目前该问题还没有多项式时间求解算法.建立了该问题的非线性整数规划模型,分析了模型中目标函数和约束函数的单调性,并根据该单调性构造了分支定界求解算法.与招投标算法比较,表明分支定界算法具有求解问题的规模大、运算速度快的优越性.

关键词:　资源选择;交货期;非线性整数规划;分支定界算法

中图分类号:　TP18　　　　**文献标识码**:　A

Research on Resource Selection with Due Date Constraint

LI Shu-juan, LI Yan, ZENG Zhi-bin
(School of mechanical & instrumental engineering, Xi'an University of technology, Xi'an 710048, China)

Abstract:　In this paper, we consider the resource selection problem with due date constraint in production planning and control where the sub-tasks form a precedence network. We prove that this problem is NP-complete. Thus this problem cannot have any polynomial time solution algorithm at present. We establish a nonlinear integer-programming model for this problem, and prove the monotonicity properties of the objective function and constraint function in the model. Basing on our observations, we construct a Branch and Bound method to solve the problem. The Branch and Bound method is superiority compared with bidding algrithm in the scale and calculation speed to resolve this problem.

Key words:　resource selection; due date; nonlinear integer programming; branch and bound

1　引言

在生产计划和控制过程中,通常需要把一个任务分解为若干个子任务,然后以招投标或其他的方式选择最好的资源来分别承担各个子任务,共同完成整个任务. 资源选择问题是生产调度过程中研究的热点问题之一[1,2].

在本文的研究方法中,考虑的影响资源选择的因素主要有任务的总费用、交货期、子任务之间的先序关系等. Talluri, S 提出了资源选择问题的两阶段数学规划方法,其考虑的因素是任务的总费用、工期[3]. 由于任务的各子任务之间往往有先序关系,整个任务构成一个活动网络图(PERT 图)[4],因此在研究资源选择问题时必须考虑子任务之间的先序关系. 文献[5]采用整数规划模型和遗传算法解决企业联盟中考虑任务总费用、完工期和子任务之间的先序关系下的伙伴优化选择问题.但文献[3,5]未证明问题的复杂性.文献[6]采用合同网中的招投标方式和基于 Mediator 的方法进行带交货期约束的资源选择问题,但采用该方法进行资源选择时,当问题的规模增加到一定程度时,算法的搜索速度大大减缓,甚至可能找不出最优解.

由于任务的交货期一般不允许拖延,本文以生产调度过程为对象,以生产成本为目标,以子任务间的先序关系和交货期为约束,获得一种调度过程中资源选择问题的整数规划及分支定界优化方法.把该问题

收稿日期:2005-04-22
资助项目:陕西省自然科学基金(2004E202),教育部春晖计划项目(Z2005-1-61004)

作者简介:李淑娟(1968 -),女(汉族),西安理工大学机械与精密仪器工程学院博士研究生,主要研究方向为:敏捷制造、智能控制等,E-mail: shujuanli @xaut.edu.cn;李言(1960 -),西安理工大学机械与精密仪器工程学院教授,博导;曾志斌(1970 -),西安理工大学博士研究生.

表示为一个非线性整数规划问题，并利用 0 - 1 背包问题证明了该问题是 NP 完全问题. 本文进一步分析了该问题的特性，并针对该特性构造求解该问题一个分支定界算法. 并与招投标算法进行了对比，证明是解决带交货期约束的资源选择问题的有效途径.

2 问题的模型和复杂性

假设整个任务由 n 个子任务组成，这些子任务由其先序关系构成一个活动网络 H. 若子任务 k 只能在子任务 i 完成之后开始，则有 $\langle i, k\rangle \in H$. 且不失一般性，假设 $i < k$，工程的最后一个子任务是 n. 如图 1 所示，图中 ⓘ代表子任务，→代表子任务间的先序关系约束，资源是按照各个子任务的工艺要求配置的，且对于一个子任务有多种可行的资源可以选择. 这就在子任务和资源之间形成了一种相关关系，在计算过程中通过数据库获取子任务和资源之间的联系.

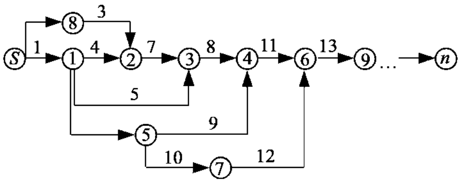

图 1 子任务先序关系下的活动网络图

假设任务的交货期是 D. 对于子任务 i，$i = 1, 2, \cdots, n$，设有 m_i 个资源 Agent 来应标，子任务 i 的候选 Agent j 的投标价格为 b_{ij}，施工周期为 q_{ij}. 假设每个子任务只能由一个资源 Agent 来承担. 该任务的目标是选择最好的资源 Agent，使得在整个任务不拖期的条件下，整个任务的总费用最小.

2.1 非线性整数规划模型

用向量 $x = (x_1, x_2, \cdots, x_n)^{\mathrm{T}}$，表示把第 i 个子任务承包给某资源 Agent x_i，$x_i \in D_i = \{1, 2, \cdots, m_i\}$，$i = 1, 2, \cdots, n$. 于是整个任务的总承包费用是 $\sum_{i=1}^{n} b_{ix_i}$，问题的解空间 $S = D_1 \times D_2 \times \cdots \times D_n$.

在解 x 下，令 $S_i(x)$ 和 $C_i(x)$ 分别表示子任务 i 的开工时间和完工时间，则 $C_n(x)$ 是整个任务的完工时间. 它们用如下的关键路径法求出.

算法 1 求各子任务的开工和完工时间.

Step 1 按如下公式计算子任务 i 的最早开工时间 S'_{ix_i} 和最早完工时间 C'_{ix_i}，$i = 1, 2, \cdots, n$.

$$S'_{ix_i} := \begin{cases} \max\{C'_{kx_k}, \forall \langle k, i\rangle \in H\}, & \text{若存在} \langle k, i\rangle \in H \\ 0, & \text{若不存在} \langle k, i\rangle \in H \end{cases},$$

$$C'_{ix_i} := S'_{ix_i} + q_{ix_i}.$$

Step 2 按如下公式计算各子任务的最晚开工时间 S_{ix_i} 和最晚完工时间 C_{ix_i}，$i = 1, 2, \cdots, n$.

$$C_{nx_n} := C'_{nx_n},$$

$$C_{ix_i} := \min\{S_{kx_k}, \quad \forall \langle i, k\rangle \in H\},$$

$$S_{ix_i} := C_{ix_i} - q_{ix_i}.$$

在算法 1 中，S_{ix_i} 和 C_{ix_i} 是子任务 i 的最晚开工时间和最晚完工时间. 若 $C_{nx_n} > D$，则表明 x 不是可行解.

设 x 是可行解，则当 $C_{ix_i} = C'_{ix_i}$ 时，子任务 i 是活动网络 H 的关键子任务. 记 S_c 为解 x 下所有关键子任务的集合. 若 $C_{ix_i} > C'_{ix_i}$，则子任务 i 有缓冲时间 $C_{ix_i} - C'_{ix_i}$.

于是，该资源选择问题可表示为：

$$(\mathrm{P}) \begin{cases} \min f(x) = \sum_{i=1}^{n} b_{ix_i} \\ \text{s.t.} \quad C_{nx_n} \le D \\ x = (x_1, x_2, \cdots, x_n)^{\mathrm{T}} \in S = D_1 \times \cdots \times D_n \end{cases} \tag{1}$$

其中 C_{nx_n} 由算法 1 求出.

显然,问题(P)是一个非线性整数规划问题,其中的约束 $C_{nx_n} \le D$ 无法显式表示,对一个解 x,只能由算法 1 确定该约束是否得到满足.

2.2　问题的复杂性

资源选择问题的判定形式如下:

实例 1　给定活动网络 H,子任务 i 的资源 Agent x_i, $x_i \in D_i = \{1, 2, \cdots, m_i\}$, x_i 的投标价格为有理数 b_{ix_i},工期为整数 q_{ix_i}, $i = 1, 2, \cdots, n$;工程的完工期限为整数 D. 有理数 B.

问:是否存在 $x_i \in D_i$, $i = 1, 2, \cdots, n$,使得 H 的关键路径长度 $C_{nx_n} \le D$,且(1)式中的 $f(x) \le B$?

定理 1　资源选择问题是 NP 完全问题.

证明　给定实例 1 的一个解 x,按照算法 1 可在多项式时间内求出活动网络 H 的关键路径并求出其长度 C_{nx_n},从而可在多项式时间内判定 C_{nx_n} 是否小于或等于 D,计算出 $f(x)$ 并判定其是否小于或等于 B. 故资源选择问题是 NP 问题. 下面采用限制法证明资源选择问题包含 0-1 背包问题作为其子问题.

设活动网络图 H 如图 2 所示.

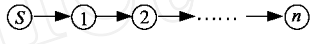

图 2　一个活动网络图 H

令 $m_1 = m_2 = \cdots = m_n = 2$,即每个子任务只有两个资源 Agent 投标,同时令 $q_{i1} = 0$, $b_{i1} = c_i$, $q_{i2} = t_i$, $b_{i2} = 0$,其中 c_i, t_i 均为整数, $i = 1, 2, \cdots, n$.

由于图 2 是线序的,故该活动网络 H 的关键路径就是从 s 到 n 的唯一一条路径. 因此此时的资源选择问题的判定问题是:在图 3 的有向图中是否存在一条从 s 到 n 的路径,使得该路径上的第一个数字之和小于或等于 D,同时第二个数字之和小于或等于 B?

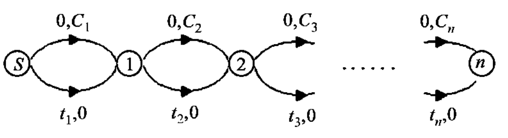

图 3　一个有向图

我们为上述特殊形式的资源选择问题建立数学模型. 设 $x_i = 0$ 表示选择图 3 的第 i 个项目的上一条边,设 $x_i = 1$ 表示选择图 3 的第 i 个项目的下一条边, $i = 1, 2, \cdots, n$. 于是图 3 的一条从 s 到 n 的路径的第一个数字之和是 $t_1 x_1 + t_2 x_2 + \cdots + t_n x_n$,第二个数字之和是

$$c_1(1 - x_1) + c_2(1 - x_2) + \cdots + c_n(1 - x_n) = \sum_{i=1}^{n} c_i - (c_1 x_1 + c_2 x_2 + \cdots + c_n x_n).$$

从而该特殊形式的资源选择问题的判定形式为:

实例 2　给定非负整数 t_i, c_i, $i = 1, 2, \cdots, n$,非负整数 D, B.

问:是否存在 $x_i \in \{0, 1\}$,使得 $t_1 x_1 + t_2 x_2 + \cdots + t_n x_n \le D$,且 $\sum_{i=1}^{n} c_i - (c_1 x_1 + c_2 x_2 + \cdots + c_n x_n) \le B$?

在实例 2 中,注意到 $\sum_{i=1}^{n} c_i - (c_1 x_1 + c_2 x_2 + \cdots + c_n x_n) \le B$ 等价于 $c_1 x_1 + c_2 x_2 + \cdots + c_n x_n \ge \sum_{i=1}^{n} c_i - B$. 于是实例 2 等价于:

实例 3　给定非负整数 t_i, c_i, $i = 1, 2, \cdots, n$,非负整数 D, B.

问:是否存在 $x_i \in \{0, 1\}$,使得: $t_1 x_1 + t_2 x_2 + \cdots + t_n x_n \le D$,且 $c_1 x_1 + c_2 x_2 + \cdots + c_n x_n \ge \sum_{i=1}^{n} c_i - B$? 此即 0-1 背包问题的判定形式.

综上,资源选择问题是 NP 完全问题. 证毕.

3　问题的性质与分支定界算法

由于资源选择问题是 NP 完全问题,而 NP 完全问题到目前为止还无法找到其多项式时间求解算法. 故只能采用分支定界算法或启发式算法如遗传算法等[7,8]求解. 但招标者一般是希望能找出最优的资源,即问题的最优解,因此下面我们构造求解该问题的分支定界算法.

在实际工程中,若子任务 i 的投标者(资源 Agent) j 和 k 满足 $b_{ik} \le b_{ij}$ 且 $q_{ik} < q_{ij}$,或 $b_{ik} < b_{ij}$ 且 $q_{ik} \le q_{ij}$,则投标者 j 肯定得不到子任务 i,理论上也容易证明问题(P)的最优解中不包含 j,从而可以在输入数据中把

j 剔除.

需要注意的是,当 $b_{ik} = b_{ij}$ 且 $q_{ik} = q_{ij}$ 时,投标者 j 和 k 可当作为一个投标者,而当问题(P)的最优解中包含了 j 时,则可在 j 和 k 中任选一个来承包子任务 i. 故不失一般性,假设

$$b_{i1} > b_{i2} > \cdots > b_{im_i},\ q_{i1} < q_{i2} < \cdots < q_{im_i},\quad i = 1,2,\cdots,n. \tag{2}$$

对问题(P)的解 x 和 y,令 $x \leq y$ 表示 $x_i \leq y_i$, $i = 1,2,\cdots,n$; $x < y$ 表示 $x_i \leq y_i$, $i = 1,2,\cdots,n$,且至少存在一个 j 使得 $x_j < y_j$.

定理 2 问题(P)中的 $f(x)$ 是严格单调下降函数, C_{nx_n} 是单调上升函数.

证明 对于问题(P)的解 x 和 y,若 $x < y$,即 $x_i \leq y_i$, $i = 1,2,\cdots,n$,且至少存在一个 j 使得 $x_j < y_j$,则由(2)式有

$$b_{ix_i} \geq b_{iy_i},\ i = 1,2,\cdots,n,\ i \neq j,\ b_{jx_j} \geq b_{jy_j}.$$

从而

$$f(x) = \sum_{i=1}^{n} b_{ix_i} > \sum_{i=1}^{n} b_{iy_i} = f(y).$$

故 $f(x)$ 是严格单调下降函数.

进一步,若 $x \leq y$,由于 $q_{ix_i} \leq q_{iy_y}$, $i = 1,2,\cdots,n$,故在算法 1 的计算过程中总有

$$S'_{ix_x} \leq S'_{iy_i},\ C'_{ix_x} \leq C'_{iy_i},\ i = 1,2,\cdots,n.$$

从而 $C_{nx_n} = C'_{nx_x} \leq C'_{ny_x} = C_{ny_n}$. 所以 C_{nx_n} 是 x 的单调上升函数. 证毕.

定义 1 设 $L = (l_1, l_2, \cdots, l_n)^{\mathrm{T}} \leq U = (u_1, u_2, \cdots, u_n)^{\mathrm{T}}$,且 l_i, u_i, $i = 1,2,\cdots,n$ 均为整数. 则称 $X = [L, U] = \{x \in R^n : l_i \leq x_i \leq u_i,\ i = 1,2,\cdots,n\}$ 为 R^n 中的一个箱.

于是问题(P)可改写为:

$$\begin{cases} \min f(x) = \sum_{i=1}^{n} b_{ix_i} \\ \text{s.t.}\ \ C_{nx_n} \leq D \\ x = (x_1, x_2, \cdots, x_n)^{\mathrm{T}} \in [E, M] \cap I^n \end{cases}, \tag{3}$$

其中 $E = (1,1,\cdots,1)^{\mathrm{T}}$, $M = (m_1, m_2, \cdots, m_n)^{\mathrm{T}}$, I^n 是 R^n 中整点的集合.

用分支定界法求解问题(3)时,需要把一个箱划分为两个子箱,采用的方法如下:

假设待划分的箱是 $X = [L, U]$. 初始时 $X = [E, M]$. 求 i 使得 $u_{i'} - l_{i'} = \max_{i = 1,2,\cdots,n}(u_i - l_i)$,并把 X 划分为两个子箱:

$$X_1 = \left\{ x \in X : x_{i'} \geq l_{i'} + \left[\frac{u_{i'} - l_{i'}}{2}\right] + 1 \right\},$$

$$X_2 = \left\{ x \in X : x_{i'} \leq l_{i'} + \left[\frac{u_{i'} - l_{i'}}{2}\right] \right\},$$

其中 $\left[\frac{u_{i'} - l_{i'}}{2}\right]$ 表示小于或等于 $\frac{u_{i'} - l_{i'}}{2}$ 的最大整数.

按照上述划分方法所得到的子箱 X_1 和 X_2 是 X 的子集,且包含了 X 中的所有整点.

定理 3 假设 $X = [L, U]$ 是分支定界法求解问题(3)过程中所得到的一个箱. 则 X 中不包含问题(3)的可行解当且仅当 L 不是可行解,即 $C_{nl_n} > D$.

证明 先证充分性. $\forall x \in X \cap I^n$,均有 $x \geq L$. 由定理 2, C_{nx_n} 是单调上升函数,故 $C_{nx_n} \geq C_{nl_n}$. 又 $C_{nl_n} > D$,从而 $C_{nx_n} > D$,即 X 中不包含问题(3)的可行解.

必要性显然成立. 证毕.

定理 4 假设 $X = [L, U]$ 是分支定界法求解问题(3)过程中所得到的一个箱. 则 $\forall x \in X \cap I^n$,若 $x \neq$

U,则有 $f(x) > f(U)$,且若 U 是可行解,则 U 是下述子问题的最优解,

$$\begin{cases} \min\ f(x) = \sum_{i=1}^{n} b_{ix_i} \\ \text{s.t.}\ C_{nx_n} \le D \\ x = (x_1, x_2, \cdots, x_n)^{\mathrm{T}} \in X \cap I^n \end{cases} \tag{4}$$

证明　$\forall x \in X \cap I^n$,均有 $x \le U$. 若 $x \ne U$,则 $x < U$. 由定理 2,$f(x)$ 是严格单调下降函数,故 $f(x) > f(U)$. 进一步,若 U 是可行解,则 U 显然是问题(4)的最优解. 证毕.

定理 5　假设 $X_1 = [L^1, U^1]$ 和 $X_2 = [L^2, U^2]$ 是分支定界法求解问题(3)过程中所得到的两个箱. 若 L^2 是可行解,且 $L^2 > U^1$,则 $\forall x \in X_1 \cap I^n$,有 $f(x) \ge f(U^1) > f(L^2)$,即 X_1 中不包含问题(3)的最优解.

证明　由 $L^2 > U^1$ 知 $\forall x \in X_1 \cap I^n$,均有 $x \le U^1 < L^2$. 又 L^2 是问题(3)的可行解,从而由 C_{nx_n} 是单调上升函数知 x 也是问题(3)的可行解. 进一步,由 $f(x)$ 是严格单调下降函数有 $f(x) \ge f(U^1) > f(L^2)$,从而 x 不是问题(3)的最优解. 故 X_1 中不包含问题(3)的最优解. 证毕.

对划分过程中得到的箱 $X = [L, U]$,若 L 是问题(3)的可行解,但 U 不是问题(3)的可行解,则在算法过程中要采用以下技巧对箱 X 进行紧缩. 令

$$a_i = \max\{a : L + ae_i \in X \cap I^n, \text{且 } L + ae_i \text{ 是可行解}\},$$

$$i = 1, 2, \cdots, n, \tag{5}$$

$$u_i = l_i + a_i,\ i = 1, 2, \cdots, n,$$

$$U = (u_1, u_2, \cdots, u_n)^{\mathrm{T}}, \tag{6}$$

其中 e_i 是第 i 个分量为 1,其余分量为 0 的 n 维向量.

由于 C_{nx_n} 是单调上升函数,(5)式的求解采用如下的二分法:

算法 2

Step 1. 若 $(l_1, \cdots, l_{i-1}, u_i, l_{i+1}, \cdots, l_n)^{\mathrm{T}}$ 是可行解,则停止计算,输出 $a_i = u_i - l_i$,否则转 Step 2.

Step 2. 令 $x_i := l_i + \left[\frac{u_i - l_i}{2}\right]$,$a := 0$,$\beta := u_i - l_i$.

Step 3. 若 $(l_1, \cdots, l_{i-1}, x_i, l_{i+1}, \cdots, l_n)^{\mathrm{T}}$ 是可行解,则令 $a := x_i - l_i$,$x_i := x_i + \left[\frac{u_i - x_i}{2}\right]$,转 Step 4;若 $(l_1, \cdots, l_{i-1}, x_i, l_{i+1}, \cdots, l_n)^{\mathrm{T}}$ 不是可行解,则令 $\beta := x_i - l_i$,$u_i := x_i$,$x_i := l_i + \left[\frac{u_i - l_i}{2}\right]$,转 Step 4.

Step 4. 若 $\beta - a = 1$,则停止计算,输出 $a_i = a$; 否则转 Step 3.

实际计算表明,按照此方法来紧缩箱 X 有利于加快分支定界算法的计算速度.

注 1　在算法 2 的 Step 3 中,判断 $(l_1, \cdots, l_{i-1}, x_i, l_{i+1}, \cdots, l_n)^{\mathrm{T}}$ 是否是可行解的方法是,在该解下用算法 1 求出活动网络的关键路径的长度,若不超过 D,则是可行解,否则不是可行解.

注 2　按照算法 2 对箱 X 紧缩不会删去问题(3)的可行解. 这是由于(3)式中的 C_{nx_n} 是单调上升函数. 事实上,由(5)式和(6)式知道,$L + a_i e_i$ 是可行点,但 $L + (a_i + 1) e_i$ 不是可行点. 从而由 C_{nx_n} 的单调性,对任意的 $x \in X \cap I^n$,若 $x \ge L + (a_i + 1) e_i$,则 $f(x) \ge f(L + (a_i + 1) e_i) > D$,即 x 不是可行点.

下面描述分支定界算法求解问题(3)的基本思想.

算法建立一个线性表 Q 来保存算法划分过程所得到的箱,Q 中的箱按照最大边长从大到小排序. 初始时 $Q = \{[E, M]\}$. 令 f^* 表示算法计算过程中所得到的问题(3)的当前最小值,x^* 是相应于 f^* 的当前最小解. 初始时置 $f^* := +\infty$. 接下来,从 Q 中取出一个具有最大边长的箱 X,并从 Q 中删除 X. 依照本文定义的划分方法把 X 对分为两个子箱 $X_1 = [L^1, U^1]$ 和 $X_2 = [L^2, U^2]$. 若 L^1 或 L^2 不是可行解,则由定理 3 知 X_1 或 X_2 中不包含问题 3 的可行解,从而删除 X_1 或 X_2. 然后用(5)式和(6)式的方法紧缩箱 X_1 和 X_2. 紧缩后的箱仍记为 $X_1 = [L^1, U^1]$ 和 $X_2 = [L^2, U^2]$.

在紧缩 X_1 或 X_2 中,若记

$$L^1 = (l_1^1, l_2^1, \cdots, l_n^1)^{\mathrm{T}},$$

$$U^1 = (u_1^1, u_2^1, \cdots, u_n^1)^{\mathrm{T}},$$

$$a_i^1 = \max\{a : L^1 + ae_i \in X \cap I^n, \text{且 } L^1 + ae_i \text{ 是可行解}\}, \quad i = 1, 2, \cdots, n, \tag{7}$$

$$a_i^2 = \max\{a : L^2 + ae_i \in X \cap I^n, \text{且 } L^2 + ae_i \text{ 是可行解}\}, \quad i = 1, 2, \cdots, n, \tag{8}$$

则对紧缩后的箱 X_1 和 X_2 有

$$u_i^1 = l_i^1 + a_i^1, \quad i = 1, 2, \cdots, n \tag{9}$$

$$u_i^2 = l_i^2 + a_i^2, \quad i = 1, 2, \cdots, n \tag{10}$$

而且由(5)式和(6)式可以知道,$L^1 + a_i^1 e_i$,$L^2 + a_i^2 e_i$,$i = 1, 2, \cdots, n$ 均是问题(3)的可行解. 于是可以令

$$f^* := \min\{f^*, f(L^1 + a_i^1 e_i), f(L^2 + a_i^2 e_i), \quad i = 1, 2, \cdots, n\}, \tag{11}$$

并令 x^* 为相应于该 f^* 的解. 由 $f(x)$ 的严格单调性,显然对于(9)式有 $f^* < f(L^1)$,且 $f^* < f(L^2)$.

进一步,若 U^1 是可行解,则由定理 4 知问题(4)在 X_1 上的最优解为 U^1. 从而删除 X_1,并且,若 $f(U^1) < f^*$,则令 $f^* = f(U^1)$,$x^* = U^1$. 对 U^2 进行同样讨论;

若 U^1 不是可行解,则由定理 4 知问题(4)在 X_1 上的最小值的一个下界为 $f(U^1)$. 于是若 $f(U^1) \geq f^*$,则表明 X_1 中不存在目标函数值比 f^* 小的解,故删除箱 X_1. 对 U^2 进行同样讨论;最后把上述操作中无法删除的箱加入线性表 Q 中. 以上步骤一直循环直至线性表 Q 空为止.

根据上述算法思想,求解问题(3)的分支定界算法描述如下:

算法 3　分支定界算法

Step 1. $Q := \{[E, M]\}$, $f^* := +\infty$.

Step 2. 令 Q 中具有最大边长的箱为 X,$Q := Q \setminus \{X\}$. 依照本文定义的划分方法把 X 对分为两个子箱 $X_1 = [L^1, U^1]$ 和 $X_2 = [L^2, U^2]$.

Step 3. 对 X_j, $j = 1, 2$,

Step 3.1. 若 L^j 不是可行解,删除 X_j.

Step 3.2. 用算法 2 求 $a_i^j = \max\{a : L^j + ae_i \in X \cap I^n$,且 $L^j + ae_i$ 是可行解$\}$,$i = 1, 2, \cdots, n$,并用(5)式和(6)式或(7)至(10)式的方法紧缩箱 X_1 和 X_2.

Step 3.3. $f^* := \min\{f^*, f(L^j + a_{ij} e_i), i = 1, 2, \cdots, n\}$.

Step 3.4. 若 U^j 是可行解,则删除 X_j,并且,若 $f(U^j) < f^*$,则令 $f^* := f(U^j)$,$x^* := U^j$.

Step 3.5. 若 $f(U^j) \geq f^*$,则删除 X_j.

Step 4. 把 Step 3 中无法删除的箱加入 Q 中.

Step 5. 若 Q 非空,则转 Step 2; 否则输出 f^* 和 x^* 作为问题(3)的最小值和最小解.

注:上述算法若输出 $f^* := \infty$,则表明问题(3)无可行解. 事实上,当问题(3)无可行解时,即$[E, M] \cap I^n$ 中无可行解,则由定理 3,E 不是问题(3)的可行解,从而在上述算法的 Step 3.1 即可把问题的整个箱删除,即在这种情况下算法迭代一次就结束. 由以上分析易知算法 3 具有如下性质:

定理 6　算法 3 在有限步内找到问题(3)的最优解或者判断其无解.

表 1　算法 3 与招投标算法的结果比较

Problems	N	m	Time1(秒)	Time2(秒)
1 - 10	10	10	1.83	3.35
11 - 20	20	10	12.65	150.23
21 - 30	30	10	34.63	626.37
31 - 40	40	10	265.7	4893.03
41 - 50	50	10	927.6	-
61 - 70	100	10	3628.1	-

4　算法测试与比较

我们随机产生一些实例,测试算法 3 的性能,并与文献[6]中的招投标算法进行比较. 算法 3 用 Free Pascal 编程,硬件环境是 PC Pentium Ⅳ 1.7GHz, Ram

256M DDR. 问题的测试实例的维数 n 从 10 到 100 共 6 个，其中每个子任务的投标者的个数 m 是 10 个. 当确定 n 和 m 后，实例的活动网络图是随机产生的，且对于活动网络图中的每个子任务，每个投标者的完工时间和费用从 1 到 100 的整数中按均匀分布随机选取. 对每对 n 和 m，共生成 10 个测试实例，我们计算算法 3 的平均计算时间 Time1. 文献[6]在 Jade 平台上处理同样问题，在与算法 3 相同的条件下得到的结果进行对比，采用合同网中招投标算法确定中标的资源所需的时间为最后一列 Time2，比较结果见表 1.

从对比结果可以看出，随着问题的规模的不断增大，分支定界算法在确定资源方面的优势不断增加. 当问题规模增加到一定程度时，采用招投标的算法由于资源 Agent 之间的频繁的数据传递导致速度很慢，甚至得不到结果，而采用分支定界算法可以在有限的时间内找到最优解.

参考文献:

[1] 刘海龙，吴铁军. 基于合同网的多 Agent 任务分配分布式优化算法[J]. 浙江大学学报(工学版)，2001，35(5)：550 - 554.
Liu H L，Wu T J. Distributed algorithm for task allocation in multi-agent system based on contract net[J]. Journal of Zhejiang University(Engineering Science)，2001，35(5)：550 - 554.

[2] 熊锐，吴澄. 车间调度问题的技术现状与发展趋势[J]. 清华大学学报，1998，38(10)：55 - 60.
Xiong R，WU C. Current status and developing trend of job shop scheduling research[J]. Journal of Tsinghua University(Science and Technology)，1998，38(10)：55 - 60.

[3] Talluri S，Baker R C. Quantitative framework for designing efficient business process alliance [A]. In: Proceedings of 1996 International Conference on Engineering and Technology Management[C]. Piscataway，1996，656 - 661.

[4] Elmaghraphy S E. Activity Networks-project Planning and Control by Network Models [M]. New York: Wiley，1977.

[5] Wang D，Yung KL，Ip W H. A heuristic genetic algorithm for subcontractor selection in a global manufacturing environment[J]. IEEE Transactions on Systems，Man and Cybernetics - Part C: Applications and Reviews，2001，31(2)：189 - 198.

[6] Li Y，Li S J，Liu Y，et al. Dynamic scheduling method based on combination of contract net with mediator[A]. Proceedings of 2005 International Conference on Machine Learning and Cybernetics[C]. Guangzhou，P. R. China，2005，339 - 344.

[7] Shehory O，Kraus S. Methods for task allocation via agent coalition formation[J]. Artificial Intelligence，1998，101(1)：165 - 200.

[8] Garey M R，Johnson D S. Computers and Intractability: A Guide to the Theory of NP-Completeness [M]. New York: WH Freeman and Company，1979.