运筹学与控制论

DOI:10.11721/cqnuj20220607

两个代理商松弛工期指派与位置有关负荷资源约束单机排序问题*

罗 成 新

(广东科技学院 通识教育学院,广东 东莞 523073)

摘要:【目的】研究两个代理商松弛工期指派资源约束单机排序问题。【方法】代理商通过竞争在同一台处理机上处理各自任务集合,各有一定数量的资源可以分派给任务。任务有待定的松弛工期,处理时间与位置有关且是所获资源量的凸函数。目标是求出任务的处理顺序、工期和资源分配方案,使得乙代理商任务中最大费用不超过给定值,且甲代理商任务最大费用取最小值。将问题转化为凸规划问题,先求出任务资源数量;再通过求解指派问题确定任务的处理顺序,进而求得工期。【结果】给出了多项式时间的最优算法,提供算例说明算法的有效性。【结论】分析表明算法的计算时间复杂度为 $O(n^3)$,其中 n 为两个代理商任务数中较大的一个。

关键词:排序;双代理商;松弛工期;位置负荷;资源分配

中图分类号:O223 **文献标志码**:A **文章编号**:1672-6693(2022)06-0001-08

排序问题在企业管理和生产加工等领域中有着较大的应用。“准时”原则在排序问题中得到广泛使用,这一原则要求赋予每个任务一个工期,违背工期的任务需要付出一定费用。工期要在决策过程中确定,是决策变量,决策者需要综合平衡供应商与消费者的利益来确定工期。松弛工期即任务的工期定义为处理时间加上一个待定常数(称为公共流)[1-5]。

常见的供需环境是在市场上同时存在多个代理商,他们为消费者提供相同产品,因此代理商之间具有竞争关系。自然地,消费者要选择对自己有利的代理商。这类问题被称为多代理商排序问题[6-10]。显然,小的工期(即工期较早)有利于消费者,大的工期(即工期较晚)有利于生产商,却降低了竞争力。这就是说,工期对于代理商提高竞争力起了很大作用。因此研究多代理商工期指派排序问题有重要意义。

近些年关于任务处理时间与位置关系的研究,反映了现代排序理论的一个发展趋势。在与位置有关的排序中,学者更倾向于任务具有可变处理时间,这更符合实际情况。其中一种情况是管理者可以通过增加资源来缩短任务的处理时间。文献中称这类问题为资源分配(可控处理时间)排序问题。资源分配问题在实际中应用广泛,得到深入研究,取得了丰富研究成果。文献[11]列举了许多关于可控处理时间的研究成果。将工期指派与可控处理时间结合起来进行研究有广泛的应用意义,受到研究者高度关注。已经取得了大量的成果[12-16]。

最小化的目标函数通常被定义为各种费用总和。近些年来一些学者将目标函数取作所有任务费用中最大费用,决策者需要确定决策变量使最大费用取最小值。这类问题的研究有着重要理论意义和应用背景。例如:可以用来评估生产系统的效率、控制管理过程中最坏情形规模及以相同方式平等对待所有代理商[17-22]。其中文献[22]研究两个竞争代理商松弛工期指派最大费用最小化单机排序问题,目的是确定任务工期、处理顺序,使得最大费用最小,得到了多项式时间的最优算法,

本文研究两个竞争代理商松弛工期指派资源约束最大费用最小化排序问题。代理商各有任务集,需要通过竞争在同一台处理机上处理。代理商各具有一定数量的资源可以分配给任务来减少处理时间。目的是在可用资源总量有限的条件下,寻找任务处理顺序和工期以及资源分配方案,使得其中一个代理商最大费用不超过给定值,且另一个代理商最大费用取最小值。给出了多项式时间的最优算法,将文献[22]结论推广到具有位置有关负荷及资源约束情形。

* **收稿日期**:2021-09-07 **修回日期**:2021-10-12 **网络出版时间**:2022-12-08 14:31

资助项目:国家自然科学基金(No. 11171050);广东科技学院创新强校工程项目(No. GKY-2019CQYJ-16)

第一作者简介:罗成新,男,教授,研究方向为运筹学与组合最优化,E-mail:luochengxin@163.com

网络出版地址:https://kns.cnki.net/kcms/detail//50.1165.N.20221207.1733.015.html

1 问题描述

有A、B两个代理商,分别有 n_K 个独立的任务 $J^K=\{J_1^K,J_2^K,\cdots,J_{n_K}^K\}(K\in\{A,B\})$,需要通过竞争在同一台处理机上完成处理,处理机一次只能处理一个任务,且处理过程不可以中断。用 $p_{jr}(u_j)$($p_{il}(v_i)$)表示代理商A(B)的任务 J_j^A(J_i^B)排在位置 $r(l)$ 处理时的处理时间,分别由下式确定:

$$p_{jr}(u_j)=\left(\frac{w_{jr}}{u_j}\right)^k,u_j>0,j,r=1,2,\cdots,n_A, \tag{1}$$

$$p_{il}(v_i)=\left(\frac{s_{il}}{v_i}\right)^k,v_i>0,i,l=1,2,\cdots,n_B。 \tag{2}$$

这里 $w_{jr}(s_{il})$ 表示任务 $J_j^A(J_i^B)$ 排在位置 $r(l)$ 处理时负荷,$u_j(v_i)$ 是分配给 $J_j^A(J_i^B)$ 的资源数量,$k>0$ 为给定常数。

在松弛工期意义下,每个任务 $J_j^K(j=1,\cdots,n_K)$ 具有公共流决定的工期,即代理商 K 任务 J_j^K 工期 d_j^K 定义为处理时间与公共流 q_K 之和:

$$d_j^K=p_j^K+q_K,j=1,\cdots,n_K,K\in\{A,B\}, \tag{3}$$

其中:$q_K\geqslant 0$ 是决策变量,习惯是称为代理商 K 的公共流;当 $K=A$ 时,$p_j^A=p_{jr}(u_j)$ 由(1)式给出,当 $K=B$ 时,$p_i^B=p_{il}(v_i)$ 由(2)式给出。在给定任务的资源分配方案 $\boldsymbol{u}=(u_1,u_2,\cdots,u_{n_A})$ 和 $\boldsymbol{v}=(v_1,v_2,\cdots,v_{n_B})$ 后,用 $\pi=\pi(\boldsymbol{u},\boldsymbol{v})$ 表示相应的任务处理顺序,$C_j^K=C_j^K(\pi)$ 表示任务 $J_j^K(j=1,\cdots,n_K)$ 的完工时间(必要时用 $J_{(j)}^K$ 表示代理商 K 排在位置 j 的任务,$S_{(j)}^K(\pi)$,$C_{(j)}^K(\pi)$ 分别表示任务 $J_{(j)}^K$ 的开始处理时间和完工时间)。在工期 d_j^K 确定后,J_j^K 完工时间 C_j^K 具有关于工期的提前或延误量,分别为 $E_j^K=\max\{d_j^K-C_j^K,0\}$ 和 $T_j^K=\max\{C_j^K-d_j^K,0\}$。

在给定处理顺序后,任务可能提前或延误,或者按时完工(这时没有费用)。可以看出,较小的公共流值产生较早的工期,对消费者更有吸引力;反之,较大的公共流值产生较晚的工期,有利于代理商,但可能减少代理商对消费者的吸引力。因此在任务费用中需要考虑流值大小。故任务费用由3部分组成:提前、延误、公共流值(影响工期开始时间),其中公共流费用的大小用来度量生产商的竞争力。于是任务 J_j^K 费用定义为:

$$Z_j^K=\max\{\alpha_K E_j^K+\gamma_K q_K,\beta_K T_j^K+\gamma_K q_K\},j=1,2,\cdots,n_K,K\in\{A,B\}, \tag{4}$$

其中:J_j^K 单位时间提前和延误费用分别为 α_K、β_K,公共流值 q_K 单位费用为 γ_K,都是正常数。代理商 K 总费用定义为 n_K 个费用中最大一个:

$$Z_K=\max_{1\leqslant j\leqslant n_K}\{Z_j^K\}=\max_{1\leqslant j\leqslant n_K}\{\max\{\alpha_K E_j^K+\gamma_K q_K,\beta_K T_j^K+\gamma_K q_K\}\}, \tag{5}$$

考虑如下问题(简记为 P_0):代理商A(B)可使用的资源总量 $U(V)$ 给定,即资源要满足约束条件 $\sum_{j=1}^{n_A}u_j\leqslant U\left(\sum_{i=1}^{n_B}v_i\leqslant V\right)$,需要确定资源分配方案 $\boldsymbol{u}=(u_1,u_2,\cdots,u_{n_A})$ 和 $\boldsymbol{v}=(v_1,v_2,\cdots,v_{n_B})$、任务的处理顺序 π 和工期 $d_j^K(j=1,2,\cdots,n_K)$,在代理商B的费用满足 $Z_B\leqslant Q$ 条件下,使得代理商 A 的费用 Z_B 取最小值,其中 Q 是代理商B最大费用的上限,为给定常数。用三参数表示法 P_0 可记作:

$$P_0:1\mid \mathrm{SLK},\mathrm{conv},\sum_{j=1}^{n_A}u_j\leqslant U,\sum_{i=1}^{n_B}v_i\leqslant V\mid Z_A:Z_B。$$

其中:SLK表示任务具有松弛工期,conv表示任务处理时间是由(1)、(2)式给出的资源的凸函数。

2 问题 P_0 的最优解

现在逐步求出问题 P_0 的最优解。一个明显的事实是,在最优排序中首个任务开始处理时间为零时刻,且两个相邻任务之间没有时间空闲。给出最优解之前,需要如下结论。

引理1[22] 最优任务处理顺序是块状结构,即代理商A、B的任务分别集中在一起排列形成两个快,然后两块相邻排列。因此最优排序候选者仅有两种:$\pi_1=\sigma(\pi_A,\pi_B)$ 和 $\pi_2=\sigma(\pi_B,\pi_A)$,其中 π_K 表示代理商 K 的任务处理顺序($K\in\{A,B\}$)。

引理 2[22]　给定任务处理顺序 π 和资源分配方案后，代理商 K 的最优流值及相应的费用分别由下式确定：

$$q_K(\pi)=\begin{cases}0,\gamma_K>\beta_K\\ \dfrac{\alpha_K S_1^K+\beta_K S_{n_K}^K}{\alpha_K+\beta_K},\gamma_K\leqslant\beta_K\end{cases},\tag{6}$$

$$Z_K(\pi)=\begin{cases}\beta_K S_{n_K}^K,\gamma_K>\beta_K\\ \dfrac{\beta_K(\alpha_K+\gamma_K)S_{n_K}^K-\alpha_K(\beta_K-\gamma_K)S_1^K}{\alpha_K+\beta_K},\gamma_K\leqslant\beta_K\end{cases}。\tag{7}$$

$K\in\{\mathrm{A},\mathrm{B}\}$，其中为了简化记号，用 $S_{n_K}^K\equiv S_{(n_K)}^K$，$S_1^K\equiv S_{(1)}^K$ 分别表示代理商 K 第 n_K 个和第 1 个任务的开始处理时间。

注意到，(6)、(7)式中，$S_{n_K}^K$，S_1^K 都与任务处理顺序 π 有关。由(7)式可以看出，为使得 $Z_K(\pi)$取最小值，应选择任务处理顺序 π，使得 $S_{n_K}^K$ 取最小值($\gamma_K>\beta_K$)；或使得 $S_{n_K}^K$ 取最小值，同时使 S_1^K 取最大值($\gamma_K\leqslant\beta_K$)。由于 $S_{n_K}^K$、S_1^K 与任务所在位置和获得的资源数量有关，因此为了给出具体表达式，需要引入 0-1 变量$\{X_{jr}:j,r=1,2,\cdots,n_{\mathrm{A}}\}$及$\{Y_{il}:i,l=1,2,\cdots,n_{\mathrm{B}}\}$，其中 $X_{jr}=1$，如果任务 J_j^{A} 排在位置 r 处理；否则 $X_{jr}=0$；$Y_{il}=1$，如果任务 J_i^{B} 排在位置 l 处理，否则 $Y_{il}=0$。事实上，给定上述 0-1 变量后，等价于给定了 A，B 各自任务的处理顺序。于是由(1)，(2)式可得，在任务处理顺序 π_1 下 S_1^K、$S_{n_K}^K$ 的表达式为：

$$S_1^{\mathrm{A}}(\pi_1)=0,S_{n_A}^{\mathrm{A}}(\pi_1)=\sum_{j=1}^{n_{\mathrm{A}}}\sum_{r=1}^{n_{\mathrm{A}}-1}p_{jr}(u_j)X_{jr}=\sum_{j=1}^{n_{\mathrm{A}}}\sum_{r=1}^{n_{\mathrm{A}}-1}\left(\frac{w_{jr}}{u_j}\right)^k X_{jr}。\tag{8}$$

由(7)式可得：

$$Z_{\mathrm{A}}(\pi_1)=\rho_{\mathrm{A}}\sum_{j=1}^{n_{\mathrm{A}}}\sum_{r=1}^{n_{\mathrm{A}}-1}\left(\frac{w_{jr}}{u_j}\right)^k X_{jr},\rho_{\mathrm{A}}=\min\left\{\beta_{\mathrm{A}},\frac{\beta_{\mathrm{A}}(\alpha_{\mathrm{A}}+\gamma_{\mathrm{A}})}{\alpha_{\mathrm{A}}+\beta_{\mathrm{A}}}\right\},\tag{9}$$

$$S_1^{\mathrm{B}}(\pi_1)=C_{n_{\mathrm{A}}}^{\mathrm{A}}(\pi_1)=\sum_{j=1}^{n_{\mathrm{A}}}\sum_{r=1}^{n_{\mathrm{A}}}\left(\frac{w_{jr}}{u_j}\right)^k X_{jr},\tag{10}$$

$$S_{n_B}^{\mathrm{B}}(\pi_1)=\sum_{j=1}^{n_{\mathrm{A}}}\sum_{r=1}^{n_{\mathrm{A}}}\left(\frac{w_{jr}}{u_j}\right)^k X_{jr}+\sum_{i=1}^{n_{\mathrm{B}}}\sum_{l=1}^{n_{\mathrm{B}}-1}\left(\frac{s_{il}}{v_i}\right)^k Y_{il},\tag{11}$$

$$Z_{\mathrm{B}}(\pi_1)=\begin{cases}\beta_{\mathrm{B}}\left[\sum\limits_{j=1}^{n_{\mathrm{A}}}\sum\limits_{r=1}^{n_{\mathrm{A}}}\left(\frac{w_{jr}}{u_j}\right)^k X_{jr}+\sum\limits_{i=1}^{n_{\mathrm{B}}}\sum\limits_{l=1}^{n_{\mathrm{B}}-1}\left(\frac{s_{il}}{v_i}\right)^k Y_{il}\right],\gamma_{\mathrm{B}}>\beta_{\mathrm{B}}\\ \gamma_{\mathrm{B}}\sum\limits_{j=1}^{n_{\mathrm{A}}}\sum\limits_{r=1}^{n_{\mathrm{A}}}\left(\frac{w_{jr}}{u_j}\right)^k X_{jr}+\frac{\beta_{\mathrm{B}}(\alpha_{\mathrm{B}}+\gamma_{\mathrm{B}})}{\alpha_{\mathrm{B}}+\beta_{\mathrm{B}}}\sum\limits_{i=1}^{n_{\mathrm{B}}}\sum\limits_{l=1}^{n_{\mathrm{B}}-1}\left(\frac{s_{il}}{v_i}\right)^k Y_{il},\gamma_{\mathrm{B}}\leqslant\beta_{\mathrm{B}}\end{cases}。\tag{12}$$

类似的，在 π_2 下 S_1^K、$S_{n_K}^K$ 的表达式如下：

$$S_1^{\mathrm{B}}(\pi_2)=0,S_{n_{\mathrm{B}}}^{\mathrm{B}}(\pi_2)=\sum_{i=1}^{n_{\mathrm{B}}}\sum_{l=1}^{n_{\mathrm{B}}-1}\left(\frac{s_{il}}{v_i}\right)^k Y_{il},\tag{13}$$

$$S_1^{\mathrm{A}}(\pi_2)=\sum_{i=1}^{n_{\mathrm{B}}}\sum_{l=1}^{n_{\mathrm{B}}}\left(\frac{s_{il}}{v_i}\right)^k Y_{il},S_{n_{\mathrm{A}}}^{\mathrm{A}}(\pi_2)=\sum_{i=1}^{n_{\mathrm{B}}}\sum_{l=1}^{n_{\mathrm{B}}}\left(\frac{s_{il}}{v_i}\right)^k Y_{il}+\sum_{j=1}^{n_{\mathrm{A}}}\sum_{r=1}^{n_{\mathrm{A}}-1}\left(\frac{w_{jr}}{u_j}\right)^k X_{jr},\tag{14}$$

再由(7)式可得：

$$Z_{\mathrm{B}}(\pi_2)=\rho_{\mathrm{B}}\sum_{i=1}^{n_{\mathrm{B}}}\sum_{l=1}^{n_{\mathrm{B}}-1}\left(\frac{s_{il}}{v_i}\right)^k Y_{il},\rho_{\mathrm{B}}=\min\left\{\beta_{\mathrm{B}},\frac{\beta_{\mathrm{B}}(\alpha_{\mathrm{B}}+\gamma_{\mathrm{B}})}{\alpha_{\mathrm{B}}+\beta_{\mathrm{B}}}\right\},\tag{15}$$

$$Z_{\mathrm{A}}(\pi_2)=\begin{cases}\beta_{\mathrm{A}}\left\{\sum\limits_{i=1}^{n_{\mathrm{B}}}\sum\limits_{l=1}^{n_{\mathrm{B}}}\left(\frac{s_{il}}{v_i}\right)^k Y_{il}+\sum\limits_{j=1}^{n_{\mathrm{A}}}\sum\limits_{r=1}^{n_{\mathrm{A}}-1}\left(\frac{w_{jr}}{u_j}\right)^k X_{jr}\right\},\gamma_{\mathrm{A}}>\beta_{\mathrm{A}}\\ \gamma_{\mathrm{A}}\sum\limits_{i=1}^{n_{\mathrm{B}}}\sum\limits_{l=1}^{n_{\mathrm{B}}}\left(\frac{s_{il}}{v_i}\right)^k Y_{il}+\frac{\beta_{\mathrm{A}}(\alpha_{\mathrm{A}}+\gamma_{\mathrm{A}})}{\alpha_{\mathrm{A}}+\beta_{\mathrm{A}}}\sum\limits_{j=1}^{n_{\mathrm{A}}}\sum\limits_{r=1}^{n_{\mathrm{A}}-1}\left(\frac{w_{jr}}{u_j}\right)^k X_{jr},\gamma_{\mathrm{A}}\leqslant\beta_{\mathrm{A}}\end{cases},\tag{16}$$

现在分析如何使(9)式中的 $Z_{\mathrm{A}}(\pi_1)$取最小值。注意到 ρ_{A} 是与任务顺序和资源分配无关正常数，由(9)式可以看出，可以通过求解下述优化问题使 $Z_{\mathrm{A}}(\pi_1)$取最小值：

$$\begin{cases}\min Z_1=\sum\limits_{j=1}^{n_A}\sum\limits_{r=1}^{n_A}\left(\dfrac{w_{jr}}{u_j}\right)^k X_{jr}\\ \text{s.t.}\begin{cases}\sum\limits_{j=1}^{n_A}u_j\leqslant U,u_j>0,j=1,2,\cdots,n_A\\ \sum\limits_{j=1}^{n_A}X_{jr}=1,r=1,2,\cdots,n_A-1\\ \sum\limits_{r=1}^{n_A}X_{jr}\leqslant 1,j=1,2,\cdots,n_A\\ X_{jr}=0,1,j,r=1,2,\cdots,n_A\end{cases}\end{cases}。\tag{17}$$

求解优化问题(17)的过程分为两步:1) 在假设代理商A任务排序已知条件下,计算最优资源分配数量(依赖于任务顺序)。2) 求出任务最优排序,进一步由(6)、(7)式计算最优松弛变量 q_K 和相应的最优目标函数值。其中第一步计算最优资源分配数量可以通过求解下述非线性凸规划问题(NLP)实现:

$$\begin{cases}\min Z_1=\sum\limits_{j=1}^{n_A}\sum\limits_{r=1}^{n_A}\left(\dfrac{w_{jr}}{u_j}\right)^k X_{jr}\\ \text{s.t.}\begin{cases}\sum\limits_{j=1}^{n_A}u_j\leqslant U\\ u_j>0,j=1,2,\cdots,n_A\end{cases}\end{cases}。\tag{18}$$

第二步是将第一步得到的最优资源分配方案代入(18)式中目标函数,此时的目标函数仅仅是$\{X_{jr},j,r=1,2,\cdots,n_A\}$的函数,可以利用指派问题求出任务的最优处理顺序。

下面的引理3给出了第一步的具体解法。

引理3 给定任务处理顺序 π_1、P_0 的最优资源分配方案 $\boldsymbol{u}^*=\boldsymbol{u}(\pi_1)=(u_1^*,\cdots,u_{n_A}^*)$ 和 $\boldsymbol{v}^*=\boldsymbol{v}(\pi_1)=(v_1^*,\cdots,v_{n_B}^*)$及相应目标函数值 $Z_K(\pi_1)$分别为:

$$u_j^*=\frac{\left(\sum\limits_{r=1}^{n_A}w_{jr}^kX_{jr}\right)^{\frac{1}{(k+1)}}}{\sum\limits_{j=1}^{n_A}\left(\sum\limits_{r=1}^{n_A}w_{jr}^kX_{jr}\right)^{\frac{1}{(k+1)}}}U,\tag{19}$$

$$v_i^*=\frac{\left(\sum\limits_{l=1}^{n_B}s_{il}^kY_{il}\right)^{\frac{1}{(k+1)}}}{\sum\limits_{i=1}^{n_B}\left(\sum\limits_{l=1}^{n_B}s_{il}^kY_{il}\right)^{\frac{1}{(k+1)}}}V,\tag{20}$$

$$Z_A(\pi_1)=\frac{\rho_A}{U^k}\left(\sum_{j=1}^{n}\left(\sum_{r=1}^{n}w_{jr}^kX_{jr}\right)^{\frac{1}{(k+1)}}\right)^{k+1},\tag{21}$$

$$Z_B(\pi_1)=\begin{cases}\beta_B\left\{\dfrac{1}{U^k}\left(\sum\limits_{j=1}^{n_A}\left(\sum\limits_{r=1}^{n_A}w_{jr}^kX_{jr}\right)^{\frac{1}{(k+1)}}\right)^{k+1}+\dfrac{1}{V^k}\left(\sum\limits_{i=1}^{n_B}\left(\sum\limits_{l=1}^{n_B}s_{il}^kY_{il}\right)^{\frac{1}{(k+1)}}\right)^{k+1}\right\},\gamma_B>\beta_B\\ \dfrac{\gamma_B}{U^k}\left(\sum\limits_{j=1}^{n_A}\left(\sum\limits_{r=1}^{n_A}w_{jr}^kX_{jr}\right)^{\frac{1}{(k+1)}}\right)^{k+1}+\dfrac{\beta_B(\alpha_B+\gamma_B)}{(\alpha_B+\beta_B)V^k}\left(\sum\limits_{j=1}^{n_B}\left(\sum\limits_{r=1}^{n_B}s_{il}^kY_{il}\right)^{\frac{1}{(k+1)}}\right)^{k+1},\gamma_B\leqslant\beta_B\end{cases}。\tag{22}$$

证明 以情形1为例,情形2的证明类似。

由问题NLP中目标函数可以看出,Z_1 与 $u_j>0(j=1,2,\cdots,n_A)$成反比。若有资源剩余,则可以把剩余资源分配给任意一个任务,这样可以使目标函数值变小,因此最优资源分配方案应该把资源用完,即满足 $\sum\limits_{j=1}^{n_A}u_j=U$。这样问题NLP转化为在等式约束条件下求最小值问题。采用拉格朗日(Lagrange)乘子法求解问题NLP。Lagrange函数取为:

$$L(\boldsymbol{u},\lambda)=\sum_{j=1}^{n_A}\sum_{r=1}^{n_A}\left(\frac{w_{jr}}{u_j}\right)^kX_{jr}+\lambda\left(\sum_{j=1}^{n_A}u_j-U\right),\tag{23}$$

这里 λ 是 Lagrange 乘子(目标函数的影子价格)。注意到 $L(\boldsymbol{u},\lambda)$ 是关于变量 $(\boldsymbol{u},\lambda)$ 的凸函数，因此给出最优资源所满足的充分必要条件：

$$\frac{\partial L(\boldsymbol{u},\lambda)}{\partial \lambda}=\sum_{j=1}^{n_A}u_j-U=0, \tag{24}$$

$$\frac{\partial L(\boldsymbol{u},\lambda)}{\partial u_j}=-k\sum_{r=1}^{n_A}\frac{w_{jr}^k}{u_j^{k+1}}X_{jr}+\lambda=0,j=1,2,\cdots,n_A。 \tag{25}$$

利用(25)式得：

$$\sum_{r=1}^{n_A}\frac{w_{1r}^k}{u_1^{k+1}}X_{1r}=\sum_{r=1}^{n_A}\frac{w_{jr}^k}{u_j^{k+1}}X_{jr},j=2,\cdots,n_A。 \tag{26}$$

由(26)式进一步得到：

$$u_j=u_1\left(\frac{\sum\limits_{r=1}^{n_A}w_{jr}^kX_{jr}}{\sum\limits_{r=1}^{n_A}w_{1r}^kX_{1r}}\right)^{\frac{1}{k+1}},j=2,\cdots,n_A。 \tag{27}$$

结合(27)、(24)式可得：

$$u_1=\frac{\left(\sum\limits_{r=1}^{n_A}w_{1r}^kX_{1r}\right)^{\frac{1}{k+1}}}{\sum\limits_{j=1}^{n_A}\left(\sum\limits_{r=1}^{n_A}w_{jr}^kX_{jr}\right)^{\frac{1}{k+1}}}U。 \tag{28}$$

(28)式代入(27)式得(19)式，再将(19)式代入(18)式中 Z_1 表达式，可以得到(21)式。对于(20)、(22)式，需要注意到在(12)式中，变量 $\{X_{jr}\}$ 与 $\{Y_{il}\}$ 是各自独立和，并且系数为正的常数，与上述方法类似可得。　　证毕

下面引理给出了第二步的具体解法。

引理 4　分别求解下述的指派问题可以得到问题 P_0 中代理商 A、B 任务的最优处理顺序 π_A、π_B：

$$\text{LAP1:}\begin{cases}\min\sum\limits_{j=1}^{n_A}\sum\limits_{r=1}^{n_A}w_{jr}^{\frac{k}{(k+1)}}X_{jr}\\ \text{s.t.}\begin{cases}\sum\limits_{j=1}^{n_A}X_{jr}=1,r=1,\cdots,n_A-1\\ \sum\limits_{r=1}^{n_A}X_{jr}\leqslant 1,j=1,2,\cdots,n_A\\ X_{jr}\in\{0,1\},j,r=1,2,\cdots,n_A\end{cases}\end{cases},\text{LAP2:}\begin{cases}\min\sum\limits_{i=1}^{n_B}\sum\limits_{l=1}^{n_B}s_{il}^{\frac{k}{(k+1)}}Y_{il}\\ \text{s.t.}\begin{cases}\sum\limits_{i=1}^{n_B}Y_{il}=1,l=1,\cdots,n_B-1\\ \sum\limits_{l=1}^{n_B}Y_{il}\leqslant 1,i=1,2,\cdots,n_B\\ Y_{il}\in\{0,1\},i,l=1,2,\cdots,n_B\end{cases}\end{cases}。$$

证明　以代理商 A 为例，代理商 B 的证明类似。

由 $\{X_{jr}\}$ 的定义及(21)式得：

$$Z_A(\pi_1)=\frac{\rho_A}{U^k}\left(\sum_{j=1}^{n}\left(\sum_{r=1}^{n}w_{jr}^kX_{jr}\right)^{\frac{1}{k+1}}\right)^{k+1}=\frac{\rho_A}{U^k}\left(\sum_{j=1}^{n}\sum_{r=1}^{n}w_{jr}^{\frac{k}{k+1}}X_{jr}\right)^{k+1}。 \tag{29}$$

注意到 U、k、ρ_A 都是正的常数，从(29)式可以看出，为了求 $Z_A(\pi_1)$ 的最小值，应求函数 $\sum\limits_{j=1}^{n_A}\left(\sum\limits_{r=1}^{n_A}w_{jr}^{\frac{k}{k+1}}X_{jr}\right)$ 在决策变量 $\{X_{jr}:j,r=1,2,\cdots,n_A\}$ 满足适当约束条件下的最小值。因此通过解指派问题 LAP1 可以得到代理商 A 的任务的最优处理顺序 π_A，其中第一组等式约束保证了前 (n_A-1) 位置上各有一个任务；第二组不等式约束保证了每个任务至多指派到一个位置；有一个任务没有指派到任何位置。LAP1 的作用是从 n_A 个任务中选出 n_A-1 个排在前 n_A-1 个位置处理，代理商 A 将剩余一个排在最后位置处理。

同理，求解 LAP2 可得代理商 B 的任务的最优处理顺序 π_B。合并两个部分排序后得到可能的最优务处理顺序 $\pi_1=(\pi_A,\pi_B)$。　　证毕

以上分析表明，利用引理 4 可以求出处理顺序 π_1 中代理商 A、B 中任务的最优处理顺序。再由引理 3 中(19)、(20)式可计算最优资源数量 u^*、v^*，如果满足 $Z_B^*(\pi_1)\leqslant Q$，则已经求出最优解了。反之，如果 $Z_B^*(\pi_1)>Q$，则 π_1 不是可行排序。这时要考虑 $\pi_2=(\pi_B^*,\pi_A^*)$ 的可行性。如果可行则可求出最优解；如果不可行则问题没有

可行解。对 π_2 处理方法与 π_1 完全类似,只需互换A、B所有数据,在前面具体方法中作适当调整即可,在此不做详细论述,可以参看后面算例2。

总结上述的分析,可以给出求解 P_0 的算法。

算法1 输入初始数据 α_K、β_K、γ_K、k、U、V、Q、n_K,$K\in\{A,B\}$;与位置有关负荷矩阵 $\boldsymbol{W}=(w_{jr})_{n_A\times n_A}$,$\boldsymbol{S}=(s_{il})_{n_B\times n_B}$。

步骤1(检验 π_1 的可行性),分别求解指派问题LAP1、LAP2,得到A、B中任务处理顺序。带入(21)、(22)式得 $Z_A^*(\pi_1)$,$Z_B^*(\pi_1)$。如果 $Z_B^*(\pi_1)\leqslant Q$,则 π_1 是最优排序,转步骤2;否则(即 $Z_B^*(\pi_1)>Q$),转步骤4。

步骤2,由(19)、(20)式计算出最优资源数量 $\boldsymbol{u}^*=(u_1^*,u_2^*,\cdots,u_{n_A}^*)$,$\boldsymbol{v}^*=(v_1^*,v_2^*,\cdots,v_{n_B}^*)$。

步骤3,利用(1)、(2)式计算 $p_j^A(\pi_1)$、$p_i^B(\pi_1)$,由(6)式计算最优松弛变量 $q_A^*=q_A^*(\pi_1)$,$q_B^*=q_B^*(\pi_1)$,由(3)式计算相应任务工期 $d_j^A(\pi_1)$、$d_i^B(\pi_1)$。

步骤4(检验 π_2 的可行性),此时,LAP1、LAP2中最优解和最优目标函数值不变。由(21)式做适当调整后计算 $Z_B^*(\pi_2)$。如果 $Z_B^*(\pi_2)\leqslant Q$,则 π_2 是最优排序,转步骤5;否则(即 $Z_B^*(\pi_2)>Q$),转步骤6。

步骤5,输出最优排序 $\pi_2^*=(\pi_B^*,\pi_A^*)$:按照(22)式适当调整后计算最优值 $Z_A^*(\pi_2)$,以下与步骤2、步骤3相同。

步骤6,算法结束,输出结论:问题 P_0 无可行解,从而没有最优解。

定理1 算法1是多项式时间算法,并且可以在时间 $O(n^3)$ 内求出问题 P_0 的最优解,其中 $n=\max\{n_A,n_B\}$。

证明 前面一系列的分析说明由算法1可以求出问题 P_0 的最优解,或者判断无可行解。其中步骤1中指派问题的时间复杂性分别是 $O(n_{n_A}^3)$ 和 $O(n_{n_B}^3)$。其余步骤用时不超过 $O(n_A)$ 或 $O(n_B)$,或是常数时间。因此总的时间复杂性是 $O(n^3)$。 证毕

下面的算例展示了算法1的实际求解过程。

算例1 取数 $k=1$,代理商A数据为 $U=100$,$n_A=8$,$\alpha_A=5$,$\beta_A=15$,$\gamma_A=9$,与位置有关的负荷矩阵在表1中给出;代理商B数据为 $V=50$,$Q=135$,$n_B=7$,$\alpha_B=3$,$\beta_B=7$,$\gamma_B=5$,在表1中去掉最后一行、一列后得到代理商B位置负荷矩阵。其中行表示位置,列表示任务。

表1 代理商A负荷矩阵(代理商B的负荷矩阵去掉第8行、第8列)

Tab.1 The workload matrix of agent A (The one of Agent B is obtained by deleting the 8th row and the 8th column)

$j/r(i/l)$	1	2	3	4	5	6	7	8
1	49	36	44	78	39	27	20	81
2	31	35	100	34	6	64	69	66
3	41	16	49	12	64	55	25	5
4	70	37	58	73	25	57	49	71
5	2	43	11	74	5	4	61	20
6	81	52	36	49	36	40	2	16
7	18	95	32	64	73	55	11	36
8	1	76	44	22	25	71	17	12

由于费用参数满足 $\beta_A=15>9=\gamma_A$,$\beta_B=7>5=\gamma_B$,故代理商A与B都属于情形2。指派问题LAP1中的系数矩阵在表2中给出,LAP2的系数矩阵是表2中去掉第8行、第8列得到的7阶方阵。

首先用LINGO求解指派问题LAP1(迭代12次,用时不足1秒),得到最优解如下(对应表2中黑体部分):$X_{12}=X_{25}=X_{34}=X_{56}=X_{67}=X_{73}=X_{81}=1$,其余 $X_{jr}=0$。显然任务 J_4^A 是没有被选中的,因此将它排在最后一个位置(对应表2中星号标记),即A的最优任务排序为 $\pi_A^*=[J_8^A,J_1^A,J_7^A,J_3^A,J_2^A,J_5^A,J_6^A,J_4^A]$。

再用LINGO求解LAP2(迭代13次),得到最优解如下(对应表2中括号部分黑体部分):$Y_{12}=Y_{25}=Y_{34}=Y_{56}=Y_{63}=X_{71}=1$,其余 $Y_{il}=0$。由于任务 J_4^B 没有被选中,因此将它排在最后一个位置(对应表2中双星号标记),即B的最优任务排序为 $\pi_B^*=[J_7^B,J_1^B,J_6^B,J_3^B,J_2^B,J_5^B,J_4^B]$。

现在判别 $\pi_1=(\pi_A^*,\pi_B^*)$ 是否可行。首先由(19)、(20)式得:

$$\boldsymbol{u}^*=(u_8^*,u_1^*,u_7^*,u_3^*,u_2^*,u_5^*,u_6^*,u_4^*)=$$

$$(3.288\,4,19.730\,4,18.612\,3,11.377\,8,8.056\,6,6.576\,8,4.636\,6,27.721\,1)$$

$$\boldsymbol{v}^*=(v_7^*,v_1^*,v_6^*,v_3^*,v_2^*,v_5^*,v_4^*)=(6.805\,8,9.630\,8,9.630\,8,5.553\,7,3.932\,6,3.210\,3,11.235\,9)。$$

由(1)、(2)式求出任务处理时间分别为:

$$(p_{81}^A,p_{12}^A,p_{73}^A,p_{34}^A,p_{25}^A,p_{56}^A,p_{67}^A,p_{48}^A)=$$

$$(0.304\,1,1.824\,6,1.719\,3,1.054\,7,0.744\,7,0.608\,2,0.431\,4,2.566\,1),$$

$(p_{71}^{B},p_{12}^{B},p_{63}^{B},p_{34}^{B},p_{25}^{B},p_{56}^{B},p_{47}^{B})=(2.644\ 9,3.738\ 0,3.738\ 0,2.160\ 7,1.525\ 8,1.244\ 9,4.361\ 0)$。

表 2　代理商 A 指派问题系数矩阵(代理商 B 指派问题系数矩阵去掉第 8 行、第 8 列)

Tab.2　The ALP cost matrix of agent A (The one of agent B is obtained by deleting the 8th row and 8th column)

$j/r(i/l)$	1	2	3	4	5	6	7	8
1	7.0	**(6.0)**	6.63	8.83	6.25	5.2	4.47	9.0
2	5.57	5.92	10.0	5.83	**(2.45)**	8.0	8.31	8.12
3	6.4	4.0	7.0	**(3.46)**	8.0	7.42	5.0	2.24
4	8.37	6.08	7.62	8.54	5.0	7.55	7.0**	8.43*
5	1.41	6.56	3.32	8.60	2.24	**(2.0)**	7.81	4.47
6	9.0	7.21	**(6.0)**	7.0	6.0	6.32	**1.41**	4.0
7	**(4.24)**	9.75	**5.66**	8.0	8.54	7.42	3.32	6.0
8	**1.0**	8.72	6.63	4.7	5.0	8.43	4.12	3.46

显然 $S_1^A(\pi_1)=0$，$S_8^A(\pi_1)=6.687\ 0$，$S_1^B(\pi_1)=9.248\ 1$，$S_7^B(\pi_1)=24.300\ 4$。由(7)式或(21)、(22)式可得 $Z_A^*(\pi_1)=70.213\ 5$，$Z_B^*(\pi_1)=130.533\ 8<135=Q$，故 $\pi_1=(\pi_A^*,\pi_B^*)$可行，且是最优任务处理顺序。由(6)式计算出 $q_A^*=5.012\ 5$，$q_B^*=19.784\ 7$。由(3)式求出任务最优工期分别为(精确到小数点后 4 位)：

$(d_8^A,d_1^A,d_7^A,d_3^A,d_2^A,d_5^A,d_6^A,d_4^A)=(5.316\ 6,6.837\ 1,6.731\ 8,6.069\ 9,5.757\ 2,5.620\ 7,5.443\ 9,7.578\ 6)$，

$(d_7^B,d_1^B,d_6^B,d_3^B,d_2^B,d_5^B,d_4^B)=(22.429\ 6,23.522\ 7,23.522\ 7,21.945\ 4,21.310\ 5,21.029\ 6,24.145\ 7)$。

算例 2　调整后的数据为 $\alpha_A=1,\beta_A=9,\gamma_A=19,Q=130,\alpha_B=3,\beta_B=17,\gamma_B=7$，其余数据同算例 1。

由于费用参数满足 $\beta_A<\gamma_A$，$\beta_B>\gamma_B$，故代理商 A 属于情形 1，B 属于情形 2。指派问题系数矩阵与费用参数无关，所有指派问题最优解不变，与例 1 相同。最优资源分配方案也不变。如果 B 任务排在最后，则由(12)式得 $Z_B^*(\pi_1)=285.162\ 3>130=Q$。因此 B 应排在 A 前。于是考虑 $\pi_2=(\pi_B^*,\pi_A^*)$情况下，类似地得到 $Z_B^*(\pi_2)=127.944\ 6<130=Q$，故 $\pi_2=(\pi_B^*,\pi_A^*)$可行，是最优任务处理顺序。由(7)式得 $Z_A^*(\pi_2)=224.902\ 7$。由(6)式计算出：$q_A^*=0$，$q_B^*=12.794\ 5$。由(3)式求出任务最优工期分别为：

$(d_8^A,d_1^A,d_7^A,d_3^A,d_2^A,d_5^A,d_6^A,d_4^A)=(0.304\ 1,1.824\ 6,1.719\ 3,1.054\ 7,0.744\ 7,0.608\ 2,0.431\ 4,2.566\ 1)$，

$(d_7^B,d_1^B,d_6^B,d_3^B,d_2^B,d_5^B,d_4^B)=(15.439\ 4,16.532\ 5,16.532\ 5,14.955\ 2,14.320\ 3,14.039\ 4,17.155\ 5)$。

参考文献：

[1] ADAMOPOULOS G I, PAPPIS C P. Single machine scheduling with flow-allowances[J]. Journal of Operational Research Society, 1996, 47(10): 1280-1285.

[2] LI G, LUO M L, ZHANG W J, et al. Single-machine due-window assignment scheduling based on common flow allowance, learning effect and resource allocation[J]. International Journal of Production Research, 2015, 53(4): 1228-1241.

[3] MOR B, MOSHEIOV G. Minmax scheduling problems with common flow-allowance[J]. Journal of the Operational Research Society, 2012, 63(9): 1284-1293.

[4] MOR B, MOSHEIOV G. Parallel machine scheduling problems with common flow-allowance[J]. Intetnational Journal of Production Economics, 2012, 139(2): 623-633.

[5] WANG X Y, WANG M Z. Single machine common flow allowance scheduling with a rate-modifying activity[J]. Computers & Industrial Engineering, 2010, 59(4): 898-902.

[6] AGNETIS A, BILLAUT J C, GAWIEJNOWICZ S, et al. Multiagent scheduling[M]. Berlin: Springer, 2014.

[7] AGNETIS A, PACCIARELLI D, PACIFICI A. Multi-agent single machine scheduling[J]. Annals of Operations Research, 2007, 150(1): 3-15.

[8] CHENG T C E, CHENG S R, WU W H, et al. A two-agent single-machine scheduling problem with truncated sum-of-processing-times based learning considerations[J]. Computers and Industrial Engineering, 2011, 60(4): 534-541.

[9] GERSTL E, MOSHEIOV G. Scheduling problems with two competing agents to minimize weighted earliness-tardiness[J]. Computers & Operations Research, 2013, 40(1): 109-116.

[10] MOR B, MOSHEIOV G. Scheduling problems with two competing agents to minimize minmax and minsum earliness measures [J]. European Journal of Operational Research, 2010, 206(3): 540-546.

[11] SHABTAY D, STEINER G. A survey of scheduling with controllable processing times[J]. Discrete Applied Mathematics, 2007, 155(13): 1643-1666.

[12] 罗成新，翟雯瑾. 具有一般截断因子和资源约束单机工期窗口排序问题[J]. 重庆师范大学学报(自然科学版)，2017，34(4)：1-7.
LUO C X, ZHAI W J. Single machine due-window assignment and scheduling problem with general job-dependent truncated learning effect under convex resource constraints[J]. Journal of Chongqing Normal University (Natural Science), 2017, 34(4): 1-7.

[13] 王吉波，梁茜茜，张博. 带有学习与恶化效应的共同工期指派问题[J]. 重庆师范大学学报(自然科学版)，2019，36(3)：1-6.
WANG J B, LIANG X X, ZHANG B. Common due date assignment problem with learning and deterioration effects[J]. Journal of Chongqing Normal University (Natural Science), 2019, 36(3): 1-6.

[14] LU Y Y, LI G, WU Y B, et al. Optimal due-date assignment problem with learning effect and resource-dependent processing times[J]. Optimization Letters, 2014, 8(1): 113-127.

[15] ZOU D X, JIANG C, LIU W W. Note on the time complexity of resource constrained scheduling with general truncated job-dependent learning effect[J]. Journal of Combinatorial Optimization, 2020, 40(4): 861-868.

[16] HE H Y, LIU M Q, WANG J B. Resource constrained scheduling with general truncated job-dependent learning effect[J]. Journal of Combinatorial Optimization, 2017, 33(2): 626-644.

[17] MOSHEIOV G, SARIG A. Minmax scheduling problems with a common due-window[J]. Computers and Operations Research, 2009, 36(6): 1886-1892.

[18] MOR B, MOSHEIOV G. Minmax scheduling problems with common flow-allowance[J]. Journal of the Operational Research Society, 2012, 63(9): 1284-1293.

[19] MOR B, MOSHEIOV G. Minimizing maximum cost on a single machine with two competing agents and job rejection[J]. Journal of the Operational Research Society, 2016, 67(12): 1524-1531.

[20] MOR B. Minmax common due-window assignment and scheduling on a single machine with two competing agents[J]. Journal of the Operational Research Society, 2018, 69(4): 589-602.

[21] 罗成新，张庚. 工期窗口指派可控处理时间资源约束最大费用最小化排序问题[J]. 重庆师范大学学报(自然科学版)，2020，37(1)：52-59.
LUO C X, ZHANG G. Minmax scheduling problem with common due-date and controlled processing times and resource restriction[J]. Journal of Chongqing Normal University (Natural Science), 2020, 37(1): 52-59.

[22] MOR B, MOSHEIOV G. A two-agent single machine scheduling problem with due-window assignment and a common flow-alloance[J]. Journal of Combinatorial Optimization, 2017, 33(4): 1454-1468.

Operations Research and Cybernetics

A Two-Agent Slack Due-Date Assignment Single Machine Scheduling Problem with Position-Dependent Workload and Resource Constraint

LUO Chengxin

(College of General Education, Guangdong University of Science and Technology, Dongguan Guangdong 523073, China)

Abstract: [**Purposes**] A two-agent slack due-date assignment single machine scheduling problem with position-dependent workload and resource constraint is studied. [**Methods**] Agents compete on the use of a common processor to process their own set of jobs. Each agent has some resource to allocate to its jobs. Each job has a slack due-date to be decided. The processing time of a job is position-dependent and is convex in resource amount allocated. The goal is to find the joint job sequence and due-dates, resource amount allocated to each job that minimizes the maximal cost among all jobs of the one agent, subject to an upper bound on the maximal cost of the second agent. The problem is converted to a convex programming problem. By solving it the resource allocation policy is obtained. The optimal job sequence is obtained by solving two assignment problems. Then the due-dates are obtained. [**Findings**] An optimal algorithm is presented in polynomial time. Two examples are provided to show the efficiency of the algorithm. [**Conclusions**] The analysis shows that the time complexity of our algorithm is $O(n^3)$, where n is the bigger number of jobs between the two agents.

Keywords: scheduling; two-agent; slack due-date; position-dependent workload; resource allocation

(责任编辑 黄 颖)