文章编号:1671-9352(2009)12-0036-05

工件有尺寸的单机批调度问题的在线算法

柏庆国,王忠志,张玉忠
(曲阜师范大学运筹与管理学院,山东 日照 276826)

摘要:将经典的批调度问题推广到考虑工件具有不同尺寸大小的单机在线批调度问题,当目标函数为工件的极大完工时间时,就所有工件在2个不同时刻到达的情形设计了一个竞争比不超过$\frac{161}{60}$的在线算法,并给出了此问题的一个下界。

关键词:批调度;在线算法;竞争比
中图分类号:O224 **文献标志码:**A

An algorithm for on-line batch scheduling with nonidentical job sizes

BAI Qing-guo, WANG Zhong-zhi, ZHANG Yu-zhong
(School of Operations Research and Management, Qufu Normal University, Rizhao 276826, Shandong, China)

Abstract: The classical on-line batch scheduling problem is generalized to the on-line case with non-identical job sizes. When the objective function is the maximum makespan, an on-line algorithm with the competitive ratio is not more than $\frac{161}{60}$ the case that all jobs have only two distinct arrival times, and a lower bound is also derived.

Key words: batch scheduling; on-line algorithm; the competitive ratio

0 引言

批调度问题是半导体生产过程的最后阶段提炼出来的一类重要调度问题。随着工业的发展,研究分批调度问题具有重要的实际意义。它可描述为:有固定的 n 个工件需要在一台或多台批处理机上加工,处理机一次最多能加工 B 个工件,工件加工过程中不允许被打断,也不允许移走或加入新工件。每一批的加工时间为属于该批的工件中最长工件的工时,开工时间应大于或等于该批中最晚到达的工件的到达时刻,且属于同一批的工件同时开工并同时完工。该问题的目标就是对工件应如何进行分批,并安排各批的次序,进而使相应的函数值达到最优。

一般批调度问题在对工件进行分批时,主要考虑每一批中工件的个数,但在实际中,有的工件尺寸大小不一,因此在分批时需要考虑工件尺寸的大小。对于工件不但有不同的加工时间而且有不同尺寸的批调度问题称为工件有尺寸的批调度问题。若此时目标函数为极大完工时间,则可用三元素表示法表示为 $1|B, s_j|C_{\max}$。假定机器的尺寸为1,工件 J_j 的尺寸为 $s_j(0<s_j<1)$,如果一个批中所含工件的尺寸之和等于1,称此批为满批,否则称为非满批。批 B_i 的加工时间是这批中所含工件的加工时间的最大者。若一个工件的尺寸大于0.5,则称此批工件为大工件,否则称为小工件。如果允许把工件的尺寸分为几个部分放在不同的批中加工,那么称工件是允许按尺寸拆分的。

收稿日期:2008-11-03
基金项目:国家自然科学基金资助项目(10671108;70971076);曲阜师范大学科研基金资助项目(XJZ200849;XJ0721)
作者简介:柏庆国(1979-),男,讲师,硕士研究生,研究方向为组合最优化,供应链管理.Email: qfnubaiqg@163.com

对于 $1|B,s_j|C_{\max}$,Uzsoy. R 证明了当所有工件的加工时间相同时，这个问题等同于装箱问题,它是 NP-完备的。他还提出了几个近似算法，但未给出他们的性能比[1]。Zhang.G 等对于 Uzsoy.R 提出的所有的近似算法进行了分析,并得到了相应的近似比,而且对于一致性排序的情形给出了最差性能比为 1.5 的离线算法，对一般情形下的离线批调度问题给出了一个最差性能比为 1.75 的算法[2]。张玉忠，柏庆国等则推广了经典的批调度问题,考虑了 $1|r_j\in\{0,r\},B,s_j|C_{\max}$ 的一致性批调度问题,即工件有不同的尺寸且分两次到达的批调度问题,他们给出了性能比不超过$\frac{33}{14}$的算法[3]。

本文在文献[2]的基础上考虑了工件分两次到达且有尺寸大小的在线批调度问题,首次对此问题设计了一个竞争比不超过$\frac{161}{60}$的在线算法。不妨假定最早的到达时间 $r_0=r$,另一个到达时间为 r。由于预先不知 r 的具体数据，因此将此问题称为工件有尺寸的在线批调度问题。用三元素表示法表示为 $1|r_j\in\{0,r\},B,s_j,\text{online}|C_{\max}$。

1　离线算法

1.1　$1|B,s_j|C_{\max}$ 问题的离线算法 MFFLPT

Zhang. G 等对于工件有尺寸的单机离线批调度问题的一般情形给出了一个最差性能比为 1.75 的 MF-FLPT 算法[2]。其中在此算法中若一批最先放入的是大工件，则称此批为 L 批。

1.1.1　算法 MFFLPT

步骤 1　按加工时间非增的顺序排列工件，若有工件加工时间相同，则把尺寸较大的工件排在前面。

步骤 2　对于当前的工件 J_j,分如下 2 种情况决定工件的安排:

(1) 若当前还不存在批或不存在(算法执行过程中生成的)活批，则为工件 J_j 开辟一个新的批。若放进工件 J_j 后此批仍不满，则使其成为一个活批。

(2) 存在一个活批 B_i，并且 B_i 有足够的空间放 J_j，则把当前的工件 J_j 放入批 B_i；若批 B_i 对于工件 J_j 没有足够的空间，分如下两种情况:

a 若 J_j 是一个大工件，则为工件 J_j 开一个新批;

b 若 J_j 是一个小工件，则拆分工件 J_j 把工件 J_j 的一部分放入 B_i,直到 B_i 成为满批，并把工件 J_j 的剩余部分排在还没有安排的工件前面。此时在批 B_i 之后的 L 批的个数记为 p：若 p 是奇数，则把最后一个 L 批作为活批；若 p 是偶数，不把任何批作为活批。在任何时刻只有一个活批且只有活批可以接受新工件。

步骤 3　把所有拆分的工件移走，对拆分的工件运用装箱问题的 FF(first-fit) 算法生成一些新批。

对于上述算法有如下结论:

引理 1[2]　对于 $1|B,s_j|C_{\max}$ 的离线情形，算法 MFFLPT 最差性能比是 1.75。

1.2　$1|B|C_{\max}$ 问题的离线算法 FBLPT

为得到 $1|B,s_j|C_{\max}$ 问题的最优解下界,本文首先介绍算法 FBLPT[4]，此算法是 Lee. C. Y.等首次提出的,且对于 $1|B|C_{\max}$ 的离线情形，此算法是多项式时间最优的。

1.2.1　算法 FBLPT

步骤 1　把工件按加工时间非增的次序编号,使 $p_1\geqslant p_2\geqslant\cdots\geqslant p_n$。

步骤 2　对于 $i=1,2,\cdots,\lfloor\frac{n}{B}\rfloor$，把编号为 $iB+1$ 到 $(i+1)B$ 的工件放在同一批中加工，其中 $\lfloor x\rfloor$ 表示小于或等于 x 的最大整数。

步骤 3　任意安排各批的次序。

2　$1|r_j\in\{0,r\},B,s_j,\text{online}|C_{\max}$ 问题的在线算法

把 MFFLPT 算法看作一个子算法可得到一个新的算法 BMFFLPT,用来考虑工件尺寸不同,而且所有工件

在两个不同时刻到达的在线情形。另外,在算法的设计过程中,需要求出问题最优值的下界,因此可以先将每一个工件拆分成若干个单位尺寸的工件,而加工时间不变,把问题转化为$1|B|C_{\max}$问题,然后利用FBLPT算法分批排序,即可得本文考虑的问题的最优值下界。用 J_1, J_2 分别表示 0, r 时刻到达的工件集,记 J 表示所有工件的集合,则 $J = J_1 \cup J_2$。用 T 和 T_1, T_2 分别表示 J 和 J_1, J_2 中工件运用算法 MFFLPT 得到的目标函数值。

2.1 算法 BMFFLPT

步骤 1 用算法 MFFLPT 对 J_1 分批,设得到 $k(k \geqslant 2)$批。

步骤 2 比较 p_1 与 $5C_1^*/6$ 的大小,若 $p_1 \leqslant 5C_1^*/6$ 转步骤 3;否则,转步骤 4。

步骤 3 将这 k 批按加工时间从大到小加工直至 $T_1 - p_k$,若 J_2 在 $T_1 - p_k$ 还没到达,转(1);否则,转(3)。

(1) 比较 $T_1 - p_k$ 与 $14C_1^*/15$ 的大小。若大于则用算法 MFFLPT 加工完 J_1,再加工 J_2;否则,转(2)。

(2) 从 $T_1 - p_k$ 空闲至 $14C_1^*/15$,此时,若 J_2 工件还没到达,则在 $14C_1^*/15$ 处加工第 k 批,再运用算法 MFFLPT 加工 J_2;否则,在 r 处用算法 MFFLPT 加工现有工件。

(3) 运用算法 MFFLPT 加工 J_1,直至 $7C_1^*/15$;若 J_2 在 $7C_1^*/15$ 后到达,则用算法 MFFLPT 加工完 J_1 再加工 J_2;否则,转步骤(4)。

(4) 若 J_2 在 $7C_1^*/15$ 前到达,则运用算法 MFFLPT 一直加工现有工件。

步骤 4 将这 k 批按加工时间从小到大加工直至 $T_1 - p_1$,若 J_2 在 $T_1 - p_1$ 还没到达,转(1);否则转(2)。

(1) 将工件从 $T_1 - p_1$ 空闲至 $14C_1^*/15$,此时,若 J_2 工件还没到达,则在 $14C_1^*/15$ 处加工第 1 批,再运用算法 MFFLPT 加工 J_2;否则,在 r 处用算法 MFFLPT 加工现有工件。

(2) 运用算法 MFFLPT 加工现有工件。

3 BMFFLPT 在线算法分析

本章对于 BMFFLPT 算法的竞争比进行分析,并与此问题的下界值进行了比较。

引理 2 对于工件有尺寸有两个到达时间的一般在线批调度问题,当最大批工件的加工时间 p_1 满足 $p_1 \leqslant \dfrac{5C_1^*}{6}$,则算法 BMFFLPT 的竞争比不超过$\dfrac{161}{60}$。

证明 显然 $C^* \geqslant \max\{r + C_2^*, C'^*\}$。又由结论知 $T_2 \leqslant 1.75C_2^*$, $T \leqslant 1.75C'^*$,因此 $C^* \geqslant \max\left\{r + \dfrac{4T_2}{7}, \dfrac{4T}{7}\right\}$。由于 $p_1 \leqslant \dfrac{5C_1^*}{6}$,根据算法,显然有 $T_1 - p_k \geqslant p_1$,分如下情形讨论:

(1) $r \geqslant T_1$。根据算法可知,$C_{\max} = r + T_2$,所以,$\dfrac{C_{\max}}{C^*} \leqslant \dfrac{r + T_2}{r + \dfrac{4T_2}{7}} = 1 + \dfrac{\dfrac{3T_2}{7}}{r + \dfrac{4T_2}{7}} \leqslant \dfrac{7}{4}$。

(2) $T_1 - p_k \leqslant r < T_1$。

① $T_1 - p_k > \dfrac{14C_1^*}{15}$,根据算法知,此时机器没有空闲,第 k 批的开工时间小于 r,则 J_1 和 J_2 分别加工,有 $C_{\max} = T_1 + T_2$,所以

$$\frac{C_{\max}}{C^*} \leqslant \frac{T_1 + T_2}{r + \dfrac{4T_2}{7}} = \frac{7}{4} + \frac{7T_1 - \dfrac{49r}{4}}{7r + 4T_2} \leqslant \frac{7}{4} + \frac{7p_k - \dfrac{21}{4}r}{7r + 4T_2} < \frac{7}{4} + \frac{7C_1^* - \dfrac{49}{10}C_1^*}{\dfrac{98}{15}C_1^*} = \frac{7}{4} + \frac{15}{14} - \frac{3}{4} < \frac{161}{60}。$$

② $T_1 - p_k \leqslant \dfrac{14C_1^*}{15}$。

(a) $r \leqslant \frac{14C_1^*}{15}$，此时 $C_{\max} = r + T' \leqslant \frac{14}{15}C_1^* + \frac{7}{4}C'^* \leqslant \frac{161}{60}C^*$；

(b) $r > \frac{14C_1^*}{15}$,此时有 $r > \frac{14C_1^*}{15} \geqslant T_1 - p_k$,根据算法有 $C_{\max} = \frac{14}{15}C_1^* + p_k + T_2$,又 $C^* \geqslant r + \frac{4T_2}{7}$,所以,

$$\frac{C_{\max}}{C^*} \leqslant \frac{\frac{14}{15}C_1^* + p_k + T_2}{r + \frac{4}{7}T_2} \leqslant \frac{7}{4} + \frac{p_k - \frac{21}{30}C_1^*}{\frac{14}{15}C_1^* + \frac{4}{7}T_2} \leqslant \frac{7}{4} + \frac{\frac{9}{30}C_1^*}{\frac{14}{15}C_1^*} = \frac{29}{14} < \frac{161}{60}。$$

(3) $r \leqslant T_1 - p_k$。

① $p_1 \leqslant r \leqslant T_1 - p_k$ 时分以下两种情况讨论:

(a) $r \geqslant \frac{7C_1^*}{15}$。此时 $C_{\max} = T_1 + T_2$。如果$\frac{T_2}{T_1} \leqslant \frac{8}{15}$,则$\frac{C_{\max}}{C^*} \leqslant \frac{T_1 + T_2}{\frac{4}{7}T_1} = \frac{7}{4} + \frac{7T_2}{4T_1} \leqslant \frac{161}{60}$;否则$\frac{C_{\max}}{C^*} \leqslant \frac{T_1 + T_2}{r + \frac{4}{7}T_1} \leqslant$

$$\frac{T_1 + T_2}{\frac{4}{15}T_1 + \frac{4}{7}T_2} = \frac{7}{4} + \frac{\frac{14}{15}}{\frac{7}{15} + \frac{T_2}{T_1}} \leqslant \frac{7}{4} + \frac{14}{15} = \frac{161}{60}。$$

(b) $r < \frac{7C_1^*}{15}$。此时必存在 j，使$\sum_{i=1}^{j} p_i < r \leqslant \sum_{i=1}^{j+1} p_i$,记 $T_0 = \sum_{i=1}^{j+1} p_i$, T' 为 T_0 以后运用算法 MFFLPT 加工工件得到的完工时间。可证明在这种情况下有 $C_{\max} \leqslant \frac{161}{60}C^*$,此时

$$C_{\max} = r + (T_0 - r) + T' \leqslant \frac{7}{15}C_1^* + (T_0 - r) + \frac{7}{4}C'^* < \frac{7}{15}C^* + r + \frac{7}{4}C^* < \frac{161}{60}C^*。$$

② $r < p_1$。显然有 $r < p_1 < T_1 - p_k$。若 $r \geqslant \frac{7C_1^*}{15}$,此时 $C_{\max} = T_1 + T_2$,类似于第 3 种情形①的证明可得结论;若 $r < \frac{7C_1^*}{15}$,根据算法知,$C_{\max} = p_1 + T' \leqslant C^* + \frac{7}{4}C^* < \frac{161}{60}C^*$。

综上所述,引理成立。

引理 3　对于工件有尺寸有两个到达时间的一般在线批调度问题，当最大批工件的加工时间 p_1 满足 $p_1 > \frac{5C_1^*}{6}$,则算法 BMFFLPT 的竞争比不超过$\frac{161}{60}$。

证明　利用引理 2 中最优值的下界分析,考虑如下情形证明结论成立。

(1) $r \geqslant T_1$。类似引理 2 中(1)的证明可知结论成立。

(2) $T_1 - p_1 \leqslant r < T_1$。易知当 $r > \frac{14C_1^*}{15}$时,根据算法有 $C_{\max} = \frac{14}{15}C_1^* + p_1 + T_2$,又因为 $C^* \geqslant r + \frac{4T_2}{7}$,所以

$$\frac{C_{\max}}{C^*} \leqslant \frac{\frac{14}{15}C_1^* + p_1 + T_2}{r + \frac{4T_2}{7}} \leqslant \frac{\frac{14}{15}C_1^* + p_1 + T_2}{\frac{14}{15}C_1^* + \frac{4}{7}T_2} = \frac{7}{4} + \frac{p_1 - \frac{21}{30}C_1^*}{\frac{14}{15}C_1^* + \frac{4}{7}T_2} \leqslant \frac{7}{4} + \frac{\frac{9}{30}C_1^*}{\frac{14}{15}C_1^*} \leqslant \frac{161}{60};$$

而当 $r \leqslant \frac{14C_1^*}{15}$时,根据算法有 $C_{\max} = r + T' \leqslant \frac{14}{15}C_1^* + \frac{7}{4}C^* < \frac{161}{60}C^*$。

(3) $r < T_1 - p_1$，则必存在 j，使$\sum_{i=j+1}^{k} p_i < r \leqslant \sum_{i=j}^{k} p_i$。记 $T_0 = \sum_{i=j}^{k} p_i$,此时有 $C_{\max} = T_0 + T' < (T_1 - p_1) + \frac{7}{4}C^* < \frac{14}{15}C_1^* + \frac{7}{4}C^* < \frac{161}{60}C^*$。

综上所述,引理成立。

结合引理 2 与引理 3,有如下结论:

定理 1　对于工件有尺寸有两个到达时间的一般在线批调度问题，算法 BMFFLPT 的竞争比不超过$\frac{161}{60}$。

对于上述问题,接下来分析所设计的在线算法的优劣性。由于 Zhang. G 等对于工件进行分批时只考虑工件个数且所有工件分两次到达的在线批调度问题提出了竞争比为 1.618 的算法[5],他们证明了此问题不存在竞争比小于 1.618 的算法,即他们所考虑问题的下界为 1.618。而本文的问题显然是 Zhang. G 等[5]所考虑问题的一种推广,因此 1.618 也可以看作本文所考虑问题的一个下界。

定理 2[5] 对于工件分批时只考虑工件个数,且所有工件分两次到达的在线批调度问题,不存在竞争比为 1.618 的算法。

尽管利用 Zhang. G 等的结果[5]得到本文所考虑问题的一个下界,但是不能确定此问题是否存在算法能够达到这个下界。虽然如此,通过证明本文提出算法的竞争比不超过$\frac{161}{60}$说明了在线算法 BMFFLPT 具有一定的有效性。

4 结语

本文考虑了工件既有不等的加工时间又有不同尺寸的批调度问题,当工件分两次到达时, 对此问题给出了一个在线算法,利用已有文献,本文还得到了所考虑问题的一个下界。而对于工件具有多个到达时间的在线批调度问题, 如何设计更好的的算法也是很有趣的问题。

参考文献:

[1] UZSOY R. Scheduling a single batch processing machine with non-idengtical job sizes[J]. International Journal of Production Research, 1994, 32:1615-1635.

[2] ZHANG G, CAI X Q, WONG C K. Minimizing makespan on a single batch processing machine with nonidentical job sizes[J]. Naval Research Logistics, 2001, 48:222-240.

[3] 张玉忠, 柏庆国, 徐健腾. 工件有尺寸且分两批到达的单机分批排序[J]. 运筹学学报, 2006, 10(4): 99-105.

[4] LEE C Y, UZSOY R. Minimizing makespan on a single batch Processing machine with dynamic job arrivals [J]. International Journal of production Research, 1999, 37: 219-236.

[5] ZHANG G, CAI X Q, WONG C K. On-line algorithms for minimizing makespan on batch processing machines[J]. Naval Research Logistics, 2001, 48: 241-259.

(编辑:孙培芹)