文章编号:1671-9352(2007)04-0063-04

工件尺寸不同的并行机批调度问题

杨振光[1],李曙光[2,3],王秀红[1]

(1. 鲁东大学　数学与信息学院, 山东　烟台　264025;
2. 山东大学　数学与系统科学学院, 山东　济南　250100;
3. 烟台大学　数学与信息科学学院, 山东　烟台　264005)

摘要:考虑并行批加工机上不同尺寸工件的调度问题;目标是极小化最大完工时间.给出了一个$(2+\varepsilon)$-近似算法,$\varepsilon>0$可以任意小.

关键词:近似算法; 调度理论;批加工;最大完工时间
中图分类号:O224;TP301　　**文献标识码:**A

Parallel-machine batch scheduling with non-identical job sizes

YANG Zhen-guang[1], LI Shu-guang[2,3] and WANG Xiu-hong[1]

(1. School of Math. and Info., Ludong Univ., Yantai 264025, Shandong, China;
2. School of Math. and System Sci., Shandong Univ., Jinan 250100, Shandong, China;
3. School of Math. and Info. Sci., Yantai Univ., Yantai 264005, Shandong, China)

Abstract: The problem of scheduling jobs with non-identical sizes on parallel batching machines is considered; the objective is to minimize the maximum completion time (makespan). A $(2+\varepsilon)$-approximation algorithm is presented, where $\varepsilon>0$ can be made arbitrarily small.

Key words: approximation algorithms; scheduling theory; batch processing; makespan

0　引言

分批调度(batch scheduling)是集成电路芯片制造过程中提出的一类新型调度问题[1].本文考虑并行机上不同尺寸工件的分批调度问题.给定工件的集合 $J=\{1,2,\cdots n\}$和 m 台可以成批加工的机器.假定所有工件同时到达.工件 j 的尺寸为 $s_j\in(0,1]$,加工时间为 p_j.每台机器可以同时对若干个工件进行成批加工,只要这些工件的尺寸之和不超过 1.一个批次的加工时间是该批次中所有工件中最长的加工时间.同一批次被加工的工件有相同的开工时间(所在批次的开工时间)和完工时间(所在批次的完工时间,即所在批次的开工时间加上加工时间).问题的目标是找到一个调度,使得最大完工时间(makespan),即最后一个被加工工件的完工时间,最短.

调度理论中相当多的问题是 NP-困难的.对于 NP-困难问题,研究的重点在于设计好的近似算法.给定一个极小化问题,若对于该问题的任意实例,算法 A 都能在多项式时间之内找到一个目标值不超过 ρ 倍最优值的可行解,则称算法 A 是该极小化问题的一个 ρ-近似算法.ρ 称为算法 A 的近似比.一族算法$\{A_\varepsilon\}$称为一个多项式时间近似方案,如果对任意给定的正数 ε,算法 A_ε 是一个$(1+\varepsilon)$-近似算法.(请参阅文献[2].)

收稿日期:2006-03-20
基金项目:国家自然科学基金资助项目(60373025)
作者简介:杨振光(1963-　),男,副教授,研究方向:算法理论.

对于分批调度问题的研究,基本局限于所有工件具有相同尺寸$\frac{1}{b}$(整数 b 称为批容量)的情形.文献[3]详尽讨论了单机($m=1$)各种目标函数的此类问题,并证明了对于极小化最大完工时间问题,当所有工件同时到达时是多项式时间可解的,而对于工件到达非同时的一般情况,该问题是 NP-困难的.对于后者,文献[4]得到了多项式时间近似方案.极小化最大完工时间的并行机批调度问题是 NP-困难的,因为此时即使当 $b=1$ 并且所有工件同时到达时仍然是 NP-困难的[5](因此我们的问题也是 NP-困难的).当工件具有释放时间(不同时到达)时,文献[6]给出了多项式时间近似方案.

当工件具有不同尺寸时,对于极小化最大完工时间的分批调度问题,已有结果均只局限于讨论单机的情形[7~9].文献[7,8]研究了所有工件同时到达的情形:[7]给出了若干启发式算法,[8]给出了一个$\frac{7}{4}$-近似算法并分析了[7]中所有的算法,指出其中最好的一个其近似比为2.文献[9]研究了工件具有释放时间的情形,给出了$(2+\varepsilon)$-近似算法,ε 是任意小的正数.

据我们所知,目前尚未有并行机上不同尺寸工件分批调度问题的研究结果.本文研究这一问题,假定所有工件同时到达.结合文献[8~10]的技巧,给出了$(2+\varepsilon)$-近似算法,ε 是任意小的正数.

在第1节我们假设工件允许裂开,给出了多项式时间近似方案.第2节考虑工件不允许裂开的情形,给出了$(2+\varepsilon)$-近似算法,ε 是任意小的正数.第3节是结语,指出了可供进一步研究的问题.

1 工件允许裂开的情形

在一个调度中,若工件 j 被分成两个工件 j_1 和 j_2,分别在不同的批次中加工,并且满足 $p_{j_1}=p_{j_2}=p_j$,$s_{j_1}+s_{j_2}=s_j$,则称工件 j 是裂开的.本节假设所有工件允许裂开.设此问题的最优值为 sopt.设 ε 是任意小的正数.不失一般性,设 $\varepsilon\le 1$.我们将得到一个调度,其目标值不超过$(1+\varepsilon)\cdot \mathrm{sopt}$.称该调度为$(1+\varepsilon)$-近似调度.

算法分为五步:最优分批;合并小批次;对各批次的加工时间取整;枚举所有可能的调度,找出可行调度中的最优者;还原得到原问题的一个调度.

首先对工件进行分批:按加工时间从大到小的顺序,依次将工件放入第一批 B_1,直到该批中工件的尺寸之和恰好为1(放入该批中的最后一个工件可能要裂开).同样地,将剩余工件(裂开的工件优先)依次放入第二批 B_2、第三批 B_3、…,直到没有剩余工件为止.

设所得到的分批为 Ψ.引理1说明 Ψ 可以导出最优调度.

引理1 存在一个最优调度,该调度中工件的分批为 Ψ.

证明 考虑一个最优调度.取该调度中加工时间最大的那个批次,与其他批次之间进行工件的交换,将该批次转换为 B_1.易见所得调度仍为最优.同样地,取剩余批次中加工时间最大的批次,将其转换为 B_2.以此类推,最后得到的最优调度,其分批恰为 $B_1,B_2,B_3\cdots$,即 Ψ.

设所得批次最大的加工时间为 $p_{\max}$,所有批次的加工时间之和为 D.

引理2 $\max\{p_{\max},\frac{D}{m}\}\le \mathrm{sopt}\le p_{\max}+\frac{D}{m}$.

证明 显然有 $\mathrm{sopt}\ge p_{\max}$.假设 $\mathrm{sopt}<\frac{D}{m}$,则任一机器的完工时间(在该机器上加工的最后一个批次的完工时间)小于$\frac{D}{m}$,故 m 台机器上加工的所有批次的加工时间之和小于$\frac{D}{m}\cdot m=D$,矛盾.故有 $\mathrm{sopt}\ge\max\{p_{\max},\frac{D}{m}\}$.

对所得批次应用列表调度算法(List Scheduling Algorithm)[11]:任一机器闲置时,任取一个未被加工的批次在该机器上加工.设所得调度中,最后一个被加工的批次的开工与完工时间分别为 S 和 T.根据列表调度算法,在时刻 S 之前,任一机器不会闲置,这说明 $S\le\frac{D}{m}$.故有:$T\le S+p_{\max}\le p_{\max}+\frac{D}{m}$.这样我们就得到了一个

可行调度，其目标值不超过 $p_{\max}+\frac{D}{m}$. 因此有 $\mathrm{sopt}\le p_{\max}+\frac{D}{m}$.

令 $d=\frac{\varepsilon}{4}\cdot\max\{p_{\max},\frac{D}{m}\}$. 由引理 2 可知 $d\le\frac{\varepsilon}{4}\cdot\mathrm{sopt}$. 加工时间小于 d 的批次称为小的，否则称为大的. Ψ 中若有两个小批次，则我们将其合并，看作一个批次，其加工时间为原来两个批次的加工时间之和. 持续应用这一方法，最终得到的所有批次中，至多有一个是小的. 这个小批次可以忽略，因为我们在得到其它批次的 $(1+\varepsilon)$- 近似调度之后，总可以把它安排在完工时间最小的机器上加工. 由于完工时间最小的机器的完工时间必不超过 sopt，我们就得到了所有批次的 $(1+\varepsilon)$- 近似调度. 因此不妨认为合并之后，所有的批次都是大的.

设合并后的分批为 Ψ'. 引理 3 说明分批 Ψ' 所对应的最优调度为 $(1+\frac{\varepsilon}{2})$- 近似调度.

引理 3　分批 Ψ' 所对应的最优调度的目标函数值不超过 $\mathrm{sopt}+2d$.

证明　为叙述方便，称由合并小批次所得到的大批次为合并批次，其它的大批次为原大批次.

考虑分批 Ψ 所对应的一个最优调度. 对每台机器上的批次适当交换加工次序之后，可认为小批次在每台机器上都是连续加工的. 即：任一机器上，任两个小批次之间，没有大批次被加工. 保留原大批次，用合并批次替换小批次，使得每台机器上合并批次的加工时间之和恰好超过小批次的加工时间之和（或没有剩余的合并批次为止）. 易见所有的合并批次均被安排到机器上. 这样就得到分批 Ψ' 所对应的一个可行调度. 注意到任一合并批次的加工时间小于 $2d$，因此所得调度的最大完工时间不超过 $\mathrm{sopt}+2d$，为 $(1+\frac{\varepsilon}{2})$- 近似调度.

记所有批次为 $E_1,E_2\cdots$，相应的加工时间为 $p(E_1),p(E_2),\cdots$. 对批次的加工时间应用“取整”（rounding）的技巧[11]：任一批次 E_i 的加工时间取为不小于 $p(E_i)$ 的 $\frac{\varepsilon}{3}\cdot d$ 的最大整数倍. 即：$p'(E_i)=\left\lceil\frac{p(E_i)}{\frac{\varepsilon}{3}\cdot d}\right\rceil\cdot\frac{\varepsilon}{3}\cdot d$. 令 k 表示所有不同加工时间的数目. 注意到 $d\le p(E_i)\le\frac{4}{\varepsilon}\cdot d$，故有：$k<\left\lfloor\frac{12}{\varepsilon^2}\right\rfloor$.

设对批次加工时间取整之后，Ψ' 转化为 Ψ''.

引理 4　分批 Ψ'' 所对应的最优调度为 $(1+\varepsilon)$- 近似调度.

证明　引理 3 证明中所描述的 $(1+\frac{\varepsilon}{2})$- 近似调度，在对批次加工时间取整之后，目标值不超过 $(1+\frac{\varepsilon}{2})(1+\frac{\varepsilon}{2})\cdot\mathrm{sopt}\le(1+\varepsilon)\cdot\mathrm{sopt}$，因此成为一个 $(1+\varepsilon)$- 近似调度.

由于 k 具有常数上界 $\left\lfloor\frac{12}{\varepsilon^2}\right\rfloor$，我们可以用如下的方法在多项式时间之内找出这个 $(1+\varepsilon)$- 近似调度.

由引理 2 可知 $\mathrm{sopt}\le\frac{8}{\varepsilon}\cdot d$. 由于所有批次都是大的，故引理 3 证明中所描述的 $(1+\frac{\varepsilon}{2})$- 近似调度中，任一机器上安排加工的批次数目不超过 $\left\lfloor\frac{(1+\frac{\varepsilon}{2})\cdot\mathrm{sopt}}{\mathrm{d}}\right\rfloor\le\left\lfloor\frac{(1+\frac{\varepsilon}{2})\cdot\frac{8}{\varepsilon}\cdot d}{\mathrm{d}}\right\rfloor=\left\lfloor\frac{8}{\varepsilon}\right\rfloor+4$. 设一台机器上加工批次的不同可能性有 Γ 种，分别标记为 $1,2,\cdots\Gamma$. 用 m_i 表示取第 i 种可能性的机器数，则一个调度可用数组 $(m_1,m_2,\cdots m_\Gamma)$ 来描述. 由前面讨论，可限制 $\Gamma\le1+k+k^2+\cdots+k^{\lfloor\frac{8}{\varepsilon}\rfloor+4}\le2\cdot k^{\lfloor\frac{8}{\varepsilon}\rfloor+4}$. 数组 $(m_1,m_2,\cdots m_\Gamma)$ 的不同可能性至多为 $(m+1)^\Gamma$，为 m 的多项式. 枚举之后，取所得可行调度中目标值最小者，经过还原（合并批次还原为相应的小批次，批次的加工时间还原），我们就得到了 $(1+\varepsilon)$- 近似调度. 这样我们有如下定理.

定理 1　设所有工件同时到达，且工件允许裂开. 当目标函数是极小化最大完工时间时，并行机上不同尺寸工件的分批调度问题有多项式时间近似方案.

2　工件不允许裂开的情形

上一节在工件允许裂开的假设下，我们得到了 $(1+\varepsilon)$- 近似调度. 这一调度未必满足工件不允许裂开的

实际要求.本节我们考虑工件不允许裂开的情形.设此问题的最优值为 opt.

设工件允许裂开时,所得到的最优分批是 $B_1, B_2, B_3, \cdots$(引理 1).设 j 是该分批中的一个裂开工件,裂开 j 所得到的两部分分别在批次 B_i 和 B_{i+1} 中.批次 B_i 和 B_{i+1} 分别称为工件 j 的前批和后批.

算法 Unsplit

第 1 步:调用上节的算法,得到工件允许裂开时的一个$(1+\frac{\varepsilon}{2})$-近似调度 π_1.

第 2 步:任一裂开工件从 π_1 中取出,重新安排在该工件前批所在的机器上,作为一个批次进行加工.

第 3 步:任一机器上所安排的批次以任意次序连续进行加工.得到调度 π_2.

定理 2 设所有工件同时到达,且工件不允许裂开.当目标函数是极小化最大完工时间时,算法 Unsplit 给出了一个$(2+\varepsilon)$-近似调度.

证明 调度 π_2 中,所有工件都是不裂开的,因此是一个可行调度.设 π_1 和 π_2 的目标值分别为 T_1 和 T_2.注意到在 π_1 中,任一批次至多是一个裂开工件的前批,并且该批次的加工时间不小于该裂开工件的加工时间.这说明 $T_2 \leq 2T_1$.由 $T_1 \leq (1+\frac{\varepsilon}{2}) \cdot \mathrm{sopt}$, $\mathrm{sopt} \leq \mathrm{opt}$,得到:$T_2 \leq (2+\varepsilon) \cdot \mathrm{opt}$.故 π_2 是一个$(2+\varepsilon)$-近似调度.

3 结语

本文考虑并行批加工机上不同尺寸工件的调度问题,假设所有工件同时到达.对于目标是极小化最大完工时间的问题,给出了$(2+\varepsilon)$-近似算法,ε 是任意小的正数.一个很自然的问题是:能否找到近似比更好的算法? 另外,工件具有不同释放时间的问题也值得研究.

参考文献:

[1] C Y Lee, R Uzsoy, L A Martin Vega. Efficient algorithms for scheduling semiconductor burn-in operations[J]. Operations Research, 1992, 40:764~775.

[2] C H Papadimitriou, K Steiglitz. Combinatorial optimization: algorithms and complexity[M]. New Jersey: Prentice-Hall, Englewood Cliffs, 1982.

[3] P Brucker, A Gladky, H Hoogeveen, et al. Scheduling a batching machine[J]. Journal of Scheduling, 1998, 1:31~54.

[4] X Deng, C K Poon, Y Zhang. Approximation algorithms in batch processing[J]. Journal of Combinatorial Optimization, 2003, 7(3): 247~257.

[5] M R Garey, D S Johnson. Computers and intractability: A guide to the theory of NP-completeness[M]. New York: W H Freeman and Company, 1979.

[6] S Li, G Li, S Zhang. Minimizing makespan with release times on identical parallel batching machines[J]. Discrete Applied Mathematics, 2005, 148(1):127~134.

[7] R Uzsoy. A single batch processing machine with non-identical job sizes[J]. International Journal of Production Research, 1994, 32: 1 615~1 635.

[8] G Zhang, X Cai, C Y Lee, et al. Minimizing makespan on a single batch processing machine with non-identical job sizes[J]. Naval Research Logistics, 2001, 48:226~240.

[9] S Li, G Li, X Wang, et al. Minimizing makespan on a single batching machine with release times and non-identical job sizes[J]. Operations Research Letters, 2005, 33:157~164.

[10] L A Hall, D B Shmoys. Approximation schemes for constrained scheduling problems[A]. Proceedings of the 30th annual IEEE Symposium on Foundations of Computer Science[C]. North Carolina: IEEE Press, 1989. 134~139.

[11] R L Graham. Bounds for certain multiprocessor anomalies[J]. Bell System Technical Journal, 1966, 45:1 563~1 581.

（编辑:李晓红）