

多目的批量过程的受影响批次再调度算法

苏 生¹ 于海杰²

(电子科技大学计算机学院 成都 610054)¹ (电子科技大学管理学院 成都 610054)²

摘要 为解决多目的批量过程调度受延迟扰动的问题,提出了综合考虑新调度质量和稳定性的受影响批次再调度 ABR 算法,根据状态任务网 STN 所定义的工艺过程将受到延迟影响的所有批次向后移动最小可能值。仿真算例表明,ABR 再调度算法在完成时间指标和开始时间总延迟指标上均优于现有的右移再调度法 RSR。

关键词 多目的批量过程,再调度,受影响批次

中图分类号 F292,TP393.07 **文献标识码** A

Affected Batches Rescheduling Algorithm for Multipurpose Batch Process

SU Sheng¹ YU Hai-jie²

(School of Computer, University of Electronic Science and Technique of China, Chengdu 610054, China)¹

(School of Management, University of Electronic Science and Technique of China, Chengdu 610054, China)²

Abstract Affected Batch Rescheduling (ABR) Algorithm for multipurpose batch process was proposed in order to tackle delay of batch. The comprehension of quality and stability was used for the performance index of rescheduling solution. ABR algorithm delays all affected batches based on processes defined in state task network (STN). Simulation case illustrated that ABR algorithm is better than Right Shift Rescheduling (RSR) algorithm on makespan and the total delay time of start time of affected batches.

Keywords Multipurpose batch process, Rescheduling, Affected batch

生产是一个动态且充满不确定性的过程。作为安排生产的预先调度,始终面临着插入紧急订单、机器发生故障、批次重新处理、订单交货期变更以及原料延迟到达等多种因素的扰动。如果因为扰动而不能完成预先调度排定的任务,则需要对预先调度进行重排或者进行修复,即再调度。由于扰动经常发生,因此再调度在实际生产过程中具有重要意义。再调度除产生可行的新调度外,还需要尽量保持新调度与原调度的一致性,及调度的稳定性,以方便作业管理。多目的批量过程是一种按批量生产模式组织的复杂生产环境,常见于精细化工、食品和包装等行业,生产设备可以处理多种任务,每种任务可能有多个批次,每个批次的处理批量可能不同,这使得多目的批量过程的再调度问题更为复杂。

Cott 首先研究了批量过程生产环境下的再调度问题,考虑了处理时间的波动,并使用移动算法更改一个批次的开始时间^[1]。Kim 研究了基于右移再调度 RSR 规则的多目的批量过程再调度问题^[2],但是其获得的再调度延迟大。Honk-omp 针对处理时间的变化和机器故障提出了再调度框架,并通过组合确定性调度优化器和结合随机事件的模拟器处理再调度。基于离散时间混合整数线性规划 MILP(Mixed Integer Linear Programming)模型,作者考虑了一些再调度方法,其目标是 minimized 再调度结果与原始调度中的任务开始时间差^[3]。Vin 考虑了多产品批量过程生产环境下,发生机器故

障以及紧急订单到达时的再调度策略,并建立了连续时间 MILP 模型^[4]。文献[5-7]也建立了批量过程生产再调度的 MILP 模型。文献[8]研究了多产品类型半导体制造分批控制的再调度问题,提出了前向启发式方法。

根据上述文献,现有研究多集中在相对简单的多产品过程再调度。对于多目的批量过程再调度,现有研究提出的再调度策略或者综合性能较差(延迟多或稳定性差),或者比较耗时(基于 MILP 的方法)。为此,需要进一步研究新的多目的批量过程再调度方法,使其在满足再调度稳定性和优化性的同时可高效率实施。本文研究在不改变原调度中任务所分配的处理资源和处理顺序的条件下,如何实现有效再调度。

1 问题描述

在一个多目的批量过程环境中,处理工艺过程由状态任务网 STN(State Task Network)^[9] 表达,存在一个预先调度 S,调度中需要处理的批次集合为 B。由于扰动的存在,调度 S 中的某个批次出现延迟。现在需要找到一种再调度方法,产生一个新调度。与原调度相比,新调度的完成时间和延迟尽可能少,并尽可能保持原调度的稳定性。

2 算法的提出

再调度方法包括修补法和重生法。修补法按照一定规

到稿日期:2009-04-08 返修日期:2009-06-16 本文受国家自然科学基金(60904072,60702071),教育部新世纪人才基金(NCET-06-0811)资助。

苏 生(1978-),男,博士,讲师,主要研究方向为智能调度和优化算法等,E-mail:susheng@uestc.edu.cn;于海杰(1977-),女,博士,讲师,主要研究方向为生态工业、博弈论等。

则对现有调度进行修补以适应扰动,其特点是简单、快捷,但再调度结果不一定最优。重生法类似于调度,它重新生成整个调度,其特点与修补法相反,相对复杂、耗时,但优化效果相对较好。由于生产过程需要快速从扰动中恢复,再调度的效率非常重要,因此我们选择修补法在工厂内部进行再调度。

多目的批量过程是最复杂的一种批量生产过程,具有离散制造 Job Shop 环境的特征,但比 Job Shop 更为复杂。文献[10]提出了 Job Shop 再调度的启发式修补方法——受影响操作再调度方法 AOR (Affected Operations Rescheduling),其被证明在效用、稳定性和效率上均有较好的性能。本文将 AOR 方法扩展到针对多目的批量过程的启发式修补方法——受影响批次再调度方法 ABR (Affected Batches Rescheduling)。

为更好理解 ABR 对 AOR 的扩展,下面给出多目的批量过程生产调度问题和 Job Shop 调度问题的异同。

(1) 相同点:

- 具有多个作业(任务)都需要完成,作业的处理工艺过程预先确定;
- 每个作业(任务)需要多个工艺步骤(Job Shop 体现为操作,多目的批量过程体现为配方中的生产过程)才能完成;
- 不同作业(任务)的操作处理顺序可能不同。

(2) 不同点:

- Job Shop 中每个作业的每个操作的处理时间确定,只执行一次,而多目的批量过程中每个配方的每个任务根据处理量可能执行多次,每次执行为一个批次;
- Job Shop 中每个作业的每个操作的处理资源一般预先确定,且唯一,而多目的批量过程中每个配方的每个任务的处理资源是一个集合;
- Job Shop 中一个操作的前序操作和后序操作均唯一,而多目的批量过程的一个任务的前序任务和后序任务均可能为多个,且通过输入/输出物料与其它任务发生联系。

AOR 再调度方法的根本思想是以发生延迟扰动的操作为源点,从工艺约束下的紧后序操作和同一资源上的紧后序操作两个分支对受影响操作做最小程度的延迟(文献[10]将之表达为一个二叉树),并不改变每个资源上所有作业的处理顺序。图 1 给出了 AOR 的示意图。

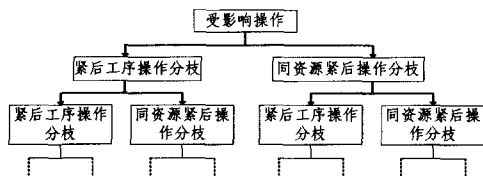


图 1 受影响操作再调度

根据这种思想,实现多目的批量过程再调度的 ABR 方法更为复杂,而多目的批量过程与 Job Shop 的不同点是导致其复杂化的主要原因。例如,由于第一个和第三个不同点,多目的批量生产的受影响批次不为一个二叉树,而是一个普通树(或多叉树);一个批次延迟所影响的紧后批次不是预先确定的,而需要通过其处理任务的输出物料(如状态任务网 STN 中一个任务的输出状态)的现有量的变化对其它批次的影响来获得。图 2 用一个预定要多次注入水和输出水的水容器的变化过程给出了某个批次延迟后其某个输出物料对原调度中的其他批次的影响示意图(水容器中的水即为这种输出物料)。

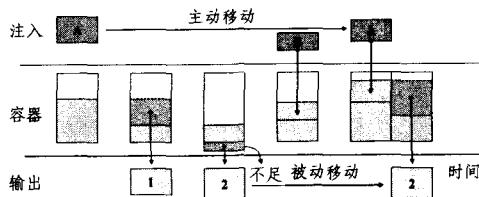


图 2 容器中水的变化过程

从图 2 可以看出,当预定要注入水容器的 A 发生延迟移动后,容器中的水比原计划要少 A 的量,使得容器中的现有量仅能满足输出 1,但不能满足输出 2。为此将具有最少延迟的 2 延迟到被延迟的 A 之后输出。在 A 的延迟期间,如果还有其它注入来源(如图 2 中的 B),则此来源仅对它之后的输出需求有影响,对它之前的输出需求没有影响,如注入 B 不影响输出 1 和 2。

3 算法描述

算法 受影响批次再调度法 ABR

输入:工厂内的原调度,受影响(延迟)批次 b ,延迟时间 dev ,完成时间 $makespan$

输出:处理延迟扰动后的新调度及其再调度成本

- (1) 初始化 $i=0, O=b$,当前批次 $t=b, startTime_t = startTime_t + dev$
- (2) While($O \neq \emptyset$) Do
- (3) 从集合 O 中随机选择一个元素作为当前批次 t
- (4) For $j=1$ to $|OS_t|$ Do
- (5) 令在 $endTime_t \sim endTime_t + dev$ 期间输出 OS_t 的第 j 个状态的批次集合为 OB_j ,输入 OS_t 的第 j 个状态的批次集合为 IB_j ,并按 $endTime$ 和 $startTime$ 分别排列 OB 和 IB 中的批次, $av_{j,time}$ 为 OS_t 的第 j 个状态在 $time$ 时刻的可用量
- (6) For $k=1$ to $|OB|$ Do
- (7) 令集合 $\Theta = \{x \mid x \text{ 的 } startTime_x < OB_k \text{ 的 } endTime_k, \text{ 且 } x \in IB\}$
- (8) For $x = |\Theta|$ to 1 Do
- (9) If $av_{j,startTime_x} < n_x * ip_{jx}$ Then
- (10) 将批次 x 延迟 $endTime_t - startTime_x$,即 x 的开始时间 $startTime_x = endTime_t$,结束时间 $endTime_x = startTime_x + procTime_x$
- (11) 更新 $makespan = \text{Max}(makespan, startTime_x)$, $devSt = devSt + (endTime_t - startTime_x)$
- (12) $O = O \cup x$
- (13) Else
- (14) $av_{j,startTime_x} = av_{j,startTime_x} - n_x * ip_{jx}$
- (15) End If
- (16) End For
- (17) $av_{j,endTime_k} = av_{j,endTime_k} + n_k * op_{jk}$
- (18) End For
- (19) End For
- (20) End While

从 ABR 算法可以看出,一个批次受到延迟后,不仅需要检查紧位于此批次之后的所有批次是否也会延迟,且需要用已延迟批次的完成时间更新调度完成时间 $makespan$ 以及总延迟时间,如语句(11)所示。

4 实验与分析

图 3 给出了分别用 ABR 和 RSR 方法对同一个原调度进行再调度的实例。图中的线段代表批次,线段左右两端的数字分别代表批次的开始时间和结束时间,线段上标注的数字

代表处理的任务和任务的处理量,如 0(22)表示此批次处理任务 0,处理量为 22。图 3 中的任务 0~5 之间的关系由图 4 所示的 STN 表达(注意这里的序号都减 1)。再调度要求为在原调度中插入序号为 3 的任务,其订单处理量为 30。从图 4 可以看出,需要插入的 1 和 4 两个任务(批次),在图中体现为 0 和 3。

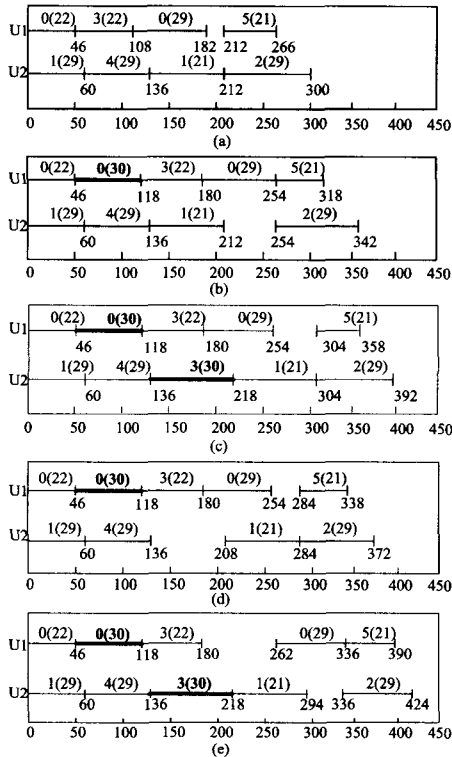


图 3 ABR 和 RSR 调度实例

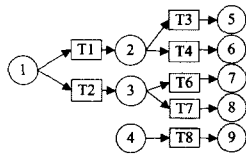


图 4 STN 表达的多目的批量过程

在图 3 中,(a)为原调度,(b)和(c)为应用 ABR 法分别插入第一个任务和插入两个任务后获得的再调度结果,(d)和(e)为应用 RSR 法分别插入第一个任务和插入两个任务后获得的再调度结果。从图 3 可以看出,应用 ABR 再调度法,插入第一个任务后其新调度的完成时间为 342,开始时间为 228,插入二个任务后其完成时间为 392,开始时间延迟为 410。应用 RSR 再调度法,插入第一个任务后其新调度的完成时间为 372,开始时间为 380,插入二个任务后其完成时间

为 424,开始时间延迟为 552。可以看出,不论在完成时间指标上,还是在开始时间延迟指标上,ABR 法均优于 RSR 法。

结束语 本文研究了复杂多目的批量过程在受到延迟扰动时的再调度问题。在深入分析多目的批量过程调度问题和 Job Shop 调度问题异同的基础上,借鉴针对 Job Shop 的受影响操作再调度 AOR 算法,提出了多目的批量过程的受影响批次再调度 ABR 算法。仿真算例结果显示,提出的 ABR 算法在新调度完成时间和原调度开始时间延迟两种评价指标上均优于现有的右移再调度算法。后续工作将研究在更改作业处理顺序和资源条件下如何提高再调度在新调度质量和原调度稳定性上的综合性能。

参考文献

- [1] Cott B J, Macchietto S. Minimizing the Effects of Batch Process Variability Using Online Schedule Modification[J]. Computers and Chemical Engineering, 1989, 13: 105-113
- [2] Kim M, Lee I B. Rule-based Reactive Rescheduling System for Multi-purpose Batch Processes[J]. Computers and Chemical Engineering, 1997, 21: 1197-1201
- [3] Honkomp S J, Mockus L, Reklaitis G V. A Framework for Schedule Evaluation with Processing Uncertainty[J]. Computers and Chemical Engineering, 1999, 23: 595-609
- [4] Vin J P, Ierapetritou M G. A New Approach for Efficient Rescheduling of Multiproduct Batch Plants[J]. Industrial and Engineering Chemistry Research, 2000, 39: 4228-4238
- [5] Roslof J, Harjunkoski I, Bjorkqvist J, et al. An MILP-based Reordering Algorithm for Complex Industrial Scheduling and Rescheduling[J]. Computers and Chemical Engineering, 2001, 25: 821-828
- [6] Mendez C A, Cerda J. Dynamic Scheduling in Multiproduct Batch Plants[J]. Computers and Chemical Engineering, 2003, 27: 1247-1259
- [7] Mendez C A, Cerda J. An MILP Framework for Batch Reactive Scheduling with Limited Discrete Resources[J]. Computers and Chemical Engineering, 2004, 28: 1059-1068
- [8] Abdullah C, Amarnath B. Cycle time reduction in batch processing by upstream rescheduling[C]//IIE Annual Conference and Expo 2007-Industrial Engineering's Critical Role in a Flat World. 2007: 1471-1476
- [9] Floudas C A, Lin X. Continuous-time versus Discrete-time Approaches for Scheduling of Chemical Processes: A Review[J]. Computers and Chemical Engineering, 2004, 28: 2109-2129
- [10] Abumaizar R J, Svestka J A. Rescheduling job shops under disruptions[J]. International Journal of Production Research, 1997, 35: 2065-2082

(上接第 214 页)

- [2] Surdeanu M, Harabagiu S, Williams J, et al. Using Predicate-Argument Structures for Information Extraction[C]//Proceedings of ACL. 2003: 8-15
- [3] Surdeanu M, Harabagiu S. Infrastructure for open-domain information extraction[C]//Proceedings of the Human Language Technology Conference, 2002: 325-330
- [4] Chieu Hai Leong, Ng Hwee Tou. A Maximum entropy Approach to Information Extraction from Semi-Structured and Free Text[C]//Proceedings of the 18th National Conference on Artificial Intelligence. 2002: 786-791
- [5] Ahn D. The Stages of Event Extraction[C]//Proceedings of the

- Workshop on Annotations and Reasoning about Time and Events. 2006: 1-8
- [6] 赵妍妍, 秦兵, 车万翔, 等. 中文事件抽取技术研究[J]. 中文信息学报, 2008, 22(1): 3-8
- [7] Ding C, He Xiaofeng. Cluster Merging and Splitting in Hierarchical Clustering Algorithms[A]//Proceedings of the 2002 IEEE International Conference on Data Mining[C]. Maebashi City, Japan: Maebashi TERRSA, 2002: 139-146
- [8] Ding C, He X, Zha H, et al. A Min-Max Cut Algorithm for Graph Partitioning and Data Clustering[A]//Proceedings of the IEEE International Conference[C]. San Jose, California, USA: Data Mining, 2001: 107-114