Scheduling Hybrid Flowshop with Parallel Batching Machines and Task Compatibilities

Adrien Bellanger [*]   Ammar Oulamara [†]

1 Introduction

We consider a two-stage hybrid flowshop scheduling problem where each of n tasks is to be processed first at stage 1 and then at stage 2. The first stage contains m_1 identical parallel discrete machines and the second stage contains m_2 identical parallel batching machines. Each discrete machine can process no more than one task at time and each batching machine can process up to k ($k < n$) tasks simultaneously in batch. The processing time of task j on any machine of stage one requires p_j time units and on any machine of stage two requiers q_j time units, which is given by the interval $[a_j, b_j]$. On the second stage the tasks are processed in batch and all tasks of the same batch start and finish together with the additional constraint that the tasks of the same batch have to be compatible. A compatibility is a symmetric binary relation in which a pair (i, j) of tasks is compatible if they share a similar processing time on the second machine (i.e, $[a_i, b_i] \cap [a_j, b_j] \neq \emptyset$). The batch processing time on the batching machine is determined as the maximum initial value a_j of compatible tasks. The objective is to find a schedule such that the completion time of the latest batch is minimized. This problem is motivated by scheduling in tire manufacturing industry [2]. A typical tire is built in a two-stage process. In the first stage (tire building), all components (sidewalls and tread) are assembled and radial tires are built on a round drum that is part of the tire building machine. The end result is called a *green tire* or *uncured tire*. In the second stage (tire curing), curing occurs through a series of chemical reactions. Tire curing is a high-temperature and high-pressure batch operation in which a pair of uncured tires is placed into a mold at a specified temperature. Each type of tire must be cured for a total duration in an interval. Two kinds of tires can be cured together if they share a same value of total curing duration. After the mold is closed it cannot be opened until the curing reaction is completed for both green tires on the same mold.

Although extensive research has been carried out on batch scheduling problems [1], [3], [4], [5], [6], to the best of our knowledge, the problem of hybrid flowshop involving batching machines and task compatibilities has not been considered before.

[*]`Adreien.Bellanger@loria.fr`. Loria - Inria Lorraine, Ecole des Mines de Nancy, Parc de Saurupt, 54042, Nancy, France

[†]`Ammar.Oulamara@loria.fr`. Loria - Inria Lorraine, Ecole des Mines de Nancy, Parc de Saurupt, 54042, Nancy, France

Since this problem is NP-hard, we focus on developing a heuristic algorithms with a performance guarantee. First we consider the general case with m_1 machines at stage one and m_2 machines at stage two, abbreviated in the following as $FH2(m_1, m_2)$. We propose three heuristics H_1, H_2 and H_3. The heuristic H_1 schedules tasks at the first stage following LPT rule (i.e. sort the tasks in non-increasing order of their processing time at the first stage, and assign each task not yet scheduled to a machine with minimum load), and on the second stage after constructing a list L of batches following the FBCLPT rule [2] (i.e. sort the tasks in non-increasing order of their initial interval value a_j, and let L be the list of tasks. At iteration i, construct a new batch B_i and add to B_i the first task j not yet placed ; starting with j put into B_i the next $k - 1$ tasks not yet placed that are compatible with j, remove from L the tasks of B_i and go to the next iteration until L become empty), each batch is scheduled as soon as possible (i.e when all tasks that compose a batch are available for processing on second stage) on batching machine with minimum load. H_1 gives a schedule in $O(nlogn)$ with a performance guarantee of $\frac{8}{3} - \frac{2}{3m}$, where $m = \max\{m_1, m_2\}$ and this bound is tight. Heuristic H_2 constructs a list L of batches using the FBCLPT rule, then for each batch of L not yet placed on the second stage, schedules its tasks on the first stage according to LPT rule, and on the second stage the batch is scheduled on a machine with minimum load. H_2 gives a schedule in $O(nlogn)$ with a performance guanrantee of $\frac{10}{3} - \frac{4}{3m}$. Heuristic H_3 starts as H_2 by constucting a list L of batches using FBCLPT rule, then the obtained batches are scheduled following johnson's algorithm by considering each stage as one machine (i.e. the processing time of each batch B_i is given as $p_1(B_i) = \sum_{j \in B_i} p_{1,j}/m_1$ and $p_2(B_i) = a_i(B_i)/m_1$ at first and scond stage, respectively). H_3 gives a schedule in $O(nlogn)$ with performance guarantee of $4 - \frac{2}{m}$. We conduct an extensive conputational experiments to compare these heuristics to lower bounds of problem, we observe that even the heuristic H_1 is the best one from the theorical point of view, the experiments show the superiority of heuristics H_2 and H_3.

We consider the special cases $FH2(1, m)$ and $FH2(m, 1)$ in which a stage 1 consists of one discrete machine and a stage 2 consists of one batching machine, respectively. For the case $FH2(1, m)$, heuristics H_2 and H_3 give performance guarantee of $\frac{7}{3} - \frac{1}{3m}$ and $3 - \frac{1}{m}$, respectiveley. For the case $FH2(m, 1)$, H_1 and H_3 give performance guarantee of $\frac{7}{3} - \frac{1}{3m}$ and $3 - \frac{1}{m}$, respectiveley.

References

[1] P. Brucker and A. Gladki and H. Hoogeveen and M.Y. Kovalyov and C. Potts and T. Tautenhahn and S.L. Van De Velde. (1998). *Scheduling a batching machine,Journal of Scheduling. Vol. 1, pp. 31–54.*

[2] A. Kamgaing Kuiteing, A. Oulamara, G. Finke. (2006). *Flowshop Scheduling Problem With Batching Machines And Task Compatibilities, Incom06, pp 39-44, Elsevier, 2006.*

[3] B.M.T. Lin and T.C.E. Cheng. (2001). *batch scheduling in the no-wait two-machine flowshop to minimize the makespan. Computers & Operations Research. Vol. 28. pp. 613–624.*

[4] A. OULAMARA AND M.Y. KOVALYOV AND G. FINKE. (2005). *Scheduling a no-wait flow-shop with unbounded batching machines. IIE Transactions on Scheduling and Logistics. Vol. 37 (8). pp. 685-696.*

[5] C.N. POTTS AND L.N. VAN WASSENHOVE. (1991). *Integrating scheduling with batching and lot-sizing: a review of algorithms and complexity. Journal of the Operational Research Society. Vol. 46. pp. 395–406.*

[6] C.N. POTTS AND V.A. STRUSEVICH AND T. TAUTENHAHN. (2001). *Scheduling batches with simultaneous job processing for two-machine shop problems. Journal of Scheduling. vol. 4. pp. 25–51.*