On-line Algorithms for Hybrid Flow Shop Scheduling

Jessen T. Havill
Department of Mathematics and Computer Science
Denison University
Granville, OH 43023
havill@denison.edu

Weizhen Mao
Department of Computer Science
The College of William and Mary
P. O. Box 8795
Williamsburg, Virginia 23187-8795
wm@cs.wm.edu

Abstract

We study a hybrid flow shop scheduling problem, in which the execution of each job must go through multiple stages in one specific order and at each stage there are parallel machines available to process the jobs that have entered the stage. The objective is to minimize the makespan. In this paper, we propose two on-line algorithms based on the greedy strategy. We also present analysis on the performance of the algorithms.

Keywords. Job scheduling, Hybrid flow shop, On-line algorithms.

1 Introduction

In the traditional flow shop scheduling problem, it is assumed that there is only one machine at each stage to execute passing jobs [6]. For example, when there are two stages (hence two machines with different functions) in the flow shop and the goal is to minimize the makespan, i.e., the time when all jobs are completed, the corresponding scheduling problem can be solved in $O(n \log n)$ [5]. However, when the execution of each job has to pass three or more stages, the problem becomes NP-complete [1].

With the development of hardware, software, and theory in parallel computing, the traditional model of flow shop scheduling is becoming somewhat unrealistic. Defined to capture the essence of parallel computing is the so-called hybrid flow shop model, in which each job has to go through multiple stages with parallel machines instead of a single machine. Results on hybrid flow shop scheduling to minimize makespan are limited. To list just a few, it has been proved that the problem is NP-complete even if there are two stages and there is only one machine at one of the two stages [3]; also, heuristics have been proposed for the case that there is one machine at stage one and parallel machines at stage two [4].

In this paper, we study the general hybrid flow shop scheduling problem with parallel machines at all stages to minimize the makespan. In Section 2, we propose a greedy on-line algorithm, GREEDY, for the m stage hybrid flow shop problem and give an analysis on the performance of the algorithm. In particular, we show that the makespan of the schedule produced by GREEDY is at most $\frac{m+n-1}{m} \cdot \frac{p_{max}}{p_{min}}$ times that of the optimal schedule for the same instance, where n is the number of jobs, m is the number of stages, and p_{max} and p_{min} are the maximum and minimum processing times respectively. We also show that this bound (also called performance ratio) is tight. In Section 3, we study a special case when there are two stages in the flow shop and at each stage, there are the same number of identical machines. Although, GREEDY can be applied to this special case achieving a tight performance ratio of $\frac{n+1}{2} \cdot \frac{p_{max}}{p_{min}}$, we present a better on-line algorithm based on the greedy list scheduling. We call this algorithm GREEDY-LIST. We prove that GREEDY-LIST achieves a constant performance ratio between 2 and 4.

2 Multiple Stage Hybrid Flow Shop Scheduling

Consider the m stage hybrid flow shop scheduling problem to minimize makespan. Assume that at stage i, there are m_i parallel machines and that the required processing time of job J_j at stage i is $p_j^{i,k}$ if it is assigned to the kth machine at stage i to execute.

In an on-line setting, jobs are given in a sequence such as $J_1, J_2, \ldots, J_n$ and an on-line algorithm must decide how to schedule each job for all stages without the knowledge of the jobs following it in the sequence. To be specific, suppose the algorithm is considering J_j at the moment. The algorithm knows $p_j^{i,k}$ for $i = 1, \ldots, m$ and $k = 1, \ldots, m_i$ (Note that there are a total of $\sum_{i=1}^{m} m_i$ positive numbers associated with each J_j.) and the schedules for $J_1, \ldots, J_{j-1}$. It then has to construct a schedule for J_j, i.e., which machine to use at each stage and when the execution starts on the chosen machine. Because on-line scheduling algorithms do not have the knowledge of the entire instance, they cannot produce optimal schedules and so are often used as heuristics.

In this section, we study an on-line algorithm for the m stage hybrid flow shop problem. It is called GREEDY since it is also a greedy algorithm. We define the algorithm as follows:

```
Algorithm GREEDY
For j from 1 to n
  For i from 1 to m
    Choose the machine at stage i with the
      smallest processing time for job Jj
    Schedule the job on that machine as
      early as possible
```

Algorithm GREEDY takes $n \cdot \sum_{i=1}^{m} m_i$ time steps to produce a schedule. It always chooses the fastest machine to use at each stage with the intention to make the makespan small. Next, we analyze the performance of GREEDY, i.e., how good/bad is the schedule constructed by GREEDY compared to the optimal schedule.

Theorem 1 *There exists an instance of the m stage hybrid flow shop scheduling problem for which the makespan of the schedule given by GREEDY is $\frac{m+n-1}{m} \cdot \frac{p_{max}}{p_{min}}$ times that of the optimal schedule, where n is the number of jobs, m is the number of stages, $p_{max} = \max_{i,j,k}\{p_j^{i,k}\}$, and $p_{min} = \min_{i,j,k}\{p_j^{i,k}\}$.*

Proof. Let $C_{max}(GREEDY)$ be the makespan of the schedule given by GREEDY and C^*_{max} be the optimal makespan for the same instance. We define our instance as follows: $m_i = n$ for all $i = 1, \ldots, m$ and $p_j^{i,k} = 1$ for all $j = 1, \ldots, n$, $i = 1, \ldots, m$, and $k = 1, \ldots, n$. Clearly, for this instance $p_{max} = p_{min} = 1$.

In the optimal schedule, all n jobs are executed in parallel by all n machines at each stage. Since for each job the execution at stage i will not start until the execution at stage $i-1$ is completed (which is the requirement of a flow shop) and there are m stages, we have

$$C^*_{max} = m \cdot 1 = m.$$

Since for fixed j and i, $p_j^{i,k} = 1$ for all $k = 1, \ldots, n$, it is possible for GREEDY to always pick the first machine at each stage to execute a job. So only the first machine at each stage will be used and all jobs are executed sequentially on the first machine at each stage. So we have,

$$\begin{aligned} C_{max}(GREEDY) &= (m+n-1) \cdot 1 \\ &= (m+n-1) \cdot \frac{p_{max}}{p_{min}} \\ &= \frac{m+n-1}{m} \cdot \frac{p_{max}}{p_{min}} \cdot C^*_{max}. \end{aligned}$$

Theorem 2 *For any instance of the m stage hybrid flow shop scheduling problem, the makespan of the schedule given by GREEDY is at most $\frac{m+n-1}{m} \cdot \frac{p_{max}}{p_{min}}$ times that of the optimal schedule.*

Proof. First we establish a lower bound to the optimal makespan, C^*_{max}. In the perfect scenario, all jobs are executed by the corresponding fastest machines in parallel at each stage. So

$$C^*_{max} \geq \max_{1 \leq j \leq n} \{ \sum_{i=1}^{m} \min_{k=1}^{m_i} \{p_j^{i,k}\}\} \geq m \cdot p_{min}.$$

Next we consider the schedule constructed by GREEDY. Let C_j^i be the completion time of J_j at stage i. Since J_1 is the first job considered by GREEDY, we have

$$\begin{aligned} C_1^1 &\leq p_{max} \\ C_1^2 &\leq C_1^1 + p_{max} \\ &\cdots\cdots \\ C_1^m &\leq C_1^{m-1} + p_{max}. \end{aligned}$$

For J_2, it may be scheduled on a different machine or the same machine as J_1 at each stage. Upper bounds can be established by hypothetically scheduling both J_1 and J_2 on one machine at each stage and using p_{max} for the execution time for both jobs. In addition, the upper bounds should also take into account the fact that the starting time of J_2 at stage i should never be earlier than the completion time of J_2 at stage $i-1$. So we have

$$\begin{aligned} C_2^1 &\leq C_1^1 + p_{max} \\ C_2^2 &\leq \max\{C_2^1, C_1^2\} + p_{max} \\ &\cdots\cdots \\ C_2^m &\leq \max\{C_2^{m-1}, C_1^m\} + p_{max}. \end{aligned}$$

We follow this reasoning for all jobs. Finally, we get the following upper bounds about J_n:

$$\begin{aligned} C_n^1 &\leq C_{n-1}^1 + p_{max} \\ C_n^2 &\leq \max\{C_n^1, C_{n-1}^2\} + p_{max} \\ &\cdots\cdots \\ C_n^m &\leq \max\{C_n^{m-1}, C_{n-1}^m\} + p_{max}. \end{aligned}$$

Obviously, J_n is the job scheduled and finished last. So $C_{max}(GREEDY) \leq C_n^m$. To estimate C_n^m, we start with the last inequality, which gives an upper bound to C_n^m. Whether it is C_n^{m-1} or C_{n-1}^m for the max function, C_n^m is reduced to some $C_{j_1}^{i_1}$ with $i_1 + j_1 = m+n-1$ plus p_{max}. We next use the upper bound to $C_{j_1}^{i_1}$, which is some $C_{j_2}^{i_2}$ with $i_2 + j_2 = m+n-2$ plus p_{max}. So the upper bound to C_n^m becomes $C_{j_2}^{i_2} + 2p_{max}$. We then notice the upper bound to $C_{j_2}^{i_2}$ is some $C_{j_3}^{i_3}$ with $i_3 + j_3 = m+n-3$ plus p_{max}. Therefore, C_n^m is bounded from above by $C_{j_3}^{i_3} + 3p_{max}$. If we continue this, we will eventually reach

C_1^1 and the bound to C_n^m is $C_1^1 + (m+n-2)p_{max}$. Using the very first inequality about C_1^1, we get $C_n^m \leq (m+n-1)p_{max}$. So we have

$$\begin{aligned} C_{max} & \quad (GREEDY) \\ \leq & \quad C_n^m \\ \leq & \quad (m+n-1)p_{max} \\ = & \quad \frac{m+n-1}{m} \cdot \frac{p_{max}}{p_{min}} \cdot (m \cdot p_{min}) \\ \leq & \quad \frac{m+n-1}{m} \cdot \frac{p_{max}}{p_{min}} \cdot C^*_{max}. \end{aligned}$$

3 Two Stage Flow Shop Scheduling

In this section we assume that a job's execution only has to go through two stages, i.e., $m = 2$ and that $p_j^{i,k} = p_j^i$ for all k, i.e., all machines at the same stage are identical. Since this scheduling problem remains to be NP-complete, we consider on-line heuristics. We can certainly use our algorithm GREEDY given in the previous section, yielding a performance ratio of $\frac{n+1}{2} \cdot \frac{p_{max}}{p_{min}}$. We are interested to know if there are some other on-line algorithms with better performance. The algorithm we shall propose and analyze is based on the greedy list scheduling algorithm [2] designed for the single stage multiple machine job scheduling problem. For this reason, we call our algorithm GREEDY-LIST.

```
Algorithm GREEDY-LIST
For j from 1 to n
  Stage 1: Schedule job Jj on the machine
           with the smallest current load
           (makespan)
  Stage 2: Schedule the job as early as
           possible
```

Algorithm GREEDY-LIST takes $n(m_1 + m_2)$ time steps to produce a schedule. At stage 1, it behaves exactly the same as Graham's List Scheduling algorithm. At stage 2, it is just a simple greedy algorithm. Clearly, the algorithm is on-line since it schedules the jobs one-by-one in the order of $J_1, \ldots, J_n$. Similar to our approach to GREEDY in the previous section, we will give two theorems on the performance of GREEDY-LIST.

Theorem 3 *There exists an instance of the 2 stage hybrid flow shop scheduling problem for which the makespan of the schedule given by GREEDY-LIST is 2 times that of the optimal schedule.*

Proof. We define the instance as follows: $m_1 = m_2$, $n = 2m_1$, $p_j^1 = 1$ and $p_j^2 = \epsilon$ for $j = 1, \ldots, \frac{n}{2}$, $p_j^1 = \epsilon$ and $p_j^2 = 1$ for $j = \frac{n}{2}+1, \ldots, n$, We assume that ϵ is an arbitrarily small positive number.

In the optimal schedule, $J_{\frac{n}{2}+1}, \ldots, J_n$ will be scheduled in parallel on all machines followed by $J_1, \ldots, J_{\frac{n}{2}}$ in parallel. Since at stage 2, there is an idle period of length ϵ at the beginning of the schedule, we have

$$C^*_{max} = 1 + 2\epsilon.$$

For the same instance, GREEDY-LIST schedules $J_1, \ldots, J_{\frac{n}{2}}$ in parallel and then $J_{\frac{n}{2}+1}, \ldots, J_n$ in parallel on all m_1 machines at stage 1. Therefore, the makespan for stage 1 is $1 + \epsilon$. At stage 2, the algorithm has to leave all machines idle until time 1 and then schedules $J_1, \ldots, J_{\frac{n}{2}}$ in parallel followed by $J_{\frac{n}{2}+1}, \ldots, J_n$ in parallel on all m_2 machines. Therefore, the makespan for stage 2 is $2 + \epsilon$. So,

$$C_{max}(GREEDY - LIST) = 2 + \epsilon = 2C^*_{max} - 3\epsilon.$$

Since ϵ is arbitrarily small, we have a ratio approaching 2.

Theorem 4 *For any instance of the 2 stage hybrid flow shop scheduling problem, the makespan of the schedule given by GREEDY-LIST is at most 4 times that of the optimal schedule.*

Proof. A lower bound to C^*_{max} can be easily established:

$$C^*_{max} \geq \max\{\max_j\{p_j^1 + p_j^2\}, \frac{1}{m_1}\sum_j p_j^1, \frac{1}{m_2}\sum_j p_j^2\}.$$

Since we will use this lower bound often in the proof, let us denote the inequality with $(*)$. To proceed, we observe that $C_{max}(GREEDY - LIST) = C^2_{max}$, the makespan for stage 2. To prove this, let J_{l1} and J_{l2} be the jobs that finish last at stages 1 and 2, respectively. (Note that they may be the same job.) Then, $C^1_{max} = C^1_{l1}$. So we have

$$\begin{aligned} C^2_{max} & \geq C^2_{l1} \\ & \geq C^1_{l1} + p^2_{l1} \\ & = C^1_{max} + p^2_{l1} \\ & \geq C^1_{max}. \end{aligned}$$

Therefore, we have $C_{max}(GREEDY - LIST) = \max\{C^1_{max}, C^2_{max}\} = C^2_{max}$.

Consider the schedule at stage 2 constructed by GREEDY-LIST. For J_{l2}, there are two possibilities corresponding to whether or not the starting time of J_{l2} at stage 2 is equal to the finishing time of J_{l2} at stage 1. We consider the first case, in which equality holds, i.e., $C^2_{l2} = C^1_{l2} + p^2_{l2}$. We have

$$\begin{aligned} C_{max} & \quad (GREEDY - LIST) \\ = & \quad C^2_{max} \\ = & \quad C^2_{l2} \\ = & \quad C^1_{l2} + p^2_{l2} \\ \leq & \quad (\frac{1}{m_1}\sum_j p_j^1 + p^1_{l2}) + p^2_{l2} \\ \leq & \quad 2C^*_{max} \quad \text{by } (*). \end{aligned}$$

Next we consider the second case, in which $C_{l2}^2 > C_{l2}^1 + p_{l2}^2$. Let Δ be the difference between the starting time of J_{l2} at stage 2 and the finishing time of J_{l2} at stage 1. If there is no idle time in Δ, then $\Delta \leq \frac{1}{m_2} \sum_j p_j^2$. So we get

$$\begin{aligned} C_{max} & \quad (GREEDY - LIST) \\ = & \quad C_{max}^2 \\ = & \quad C_{l2}^2 \\ = & \quad C_{l2}^1 + \Delta + p_{l2}^2 \\ \leq & \quad (\frac{1}{m_1} \sum_j p_j^1 + p_{l2}^1) + \frac{1}{m_2} \sum_j p_j^2 + p_{l2}^2 \\ \leq & \quad 3C_{max}^* \quad \text{by } (*). \end{aligned}$$

If there is idle time in Δ, then let J_h be the job scheduled right after the rightmost idle period overlapping Δ and let Δ' be the starting time of J_{l2} minus the starting time of J_h at stage 2. Note that Δ' might be negative. However, if it is positive, then $\Delta' \leq \frac{1}{m_2} \sum_j p_j^2$. Therefore, we have

$$\begin{aligned} C_{max} & \quad (GREEDY - LIST) \\ = & \quad C_{max}^2 \\ = & \quad C_{l2}^2 \\ \leq & \quad C_h^1 + |\Delta'| + p_{l2}^2 \\ \leq & \quad (\frac{1}{m_1} \sum_j p_j^1 + p_h^1) + \frac{1}{m_2} \sum_j p_j^2 + p_{l2}^2 \\ \leq & \quad 4C_{max}^* \quad \text{by } (*). \end{aligned}$$

4 Conclusions

In this paper, we proposed and analyzed two on-line algorithms of the hybrid flow shop scheduling problem. In particular, we proved a bound for our first algorithm, GREEDY. The bound is tight, it is however proportional to the number of jobs n, a parameter which may be arbitrarily large. Our second algorithm, GREEDY-LIST, is for the two stage special case. We showed that it achieves a small constant bound, although we are unable to eliminate the gap between the lower (2) and upper (4) bounds.

For future research, we will focus on proving a tight bound for GREEDY-LIST. We suspect it is 2. Also, we are interested in on-line algorithms with constant performance bounds for the multiple stage case. If there are no such algorithms, we wish to establish a lower bound proof using perhaps adversary arguments.

Acknowledgement

This research was supported in part by NSF grant NCR-9505963.

References

[1] M. R. Garey, D. S. Johnson, and R. Sethi, The complexity of flowshop and jobshop scheduling. *Mathematics of Operations Research*, **1**, 117–129 (1976).

[2] R. L. Graham, Bounds for multiprocessing timing anomalies, *SIAM J. Appl. Math.* **17**, 416–429 (1969).

[3] J. N. D. Gupta, Two-stage hybrid flowshop scheduling problem, *Journal of the Operational Research Society*, **39**, 359–364 (1988).

[4] J. N. D. Gupta and E. A Tunc, Schedules for a two-stage hybrid flowshop with parallel machines at the second stage, *International Journal of Production Research*, **29**, 1489–1502 (1991)

[5] S. M. Johnson, Optimal two- and three-stage production schedules with setup times included, *Naval Research Logistic Quarterly*, **1**, 61–68 (1954).

[6] E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan and D. B. Shmoys, Sequencing and scheduling: Algorithms and complexity, *Handbooks in Operations Research and Management Science, Volume 4: Logistics of Production and Inventory*, S. C. Graves, A. H. G. Rinnooy Kan and P. Zipkin, ed., North-Holland (1990).

Authors' Biography

Jessen T. Havill got his Ph.D. in computer science from the College of William and Mary in 1998. He is now Assistent Professor in the Department of Mathematics and Computer Science at Denison University. His research interests include on-line algorithms and network routing and scheduling.

Weizhen Mao received her Ph.D. in computer science from Princeton University in 1990. She then joined the faculty of the College of William and Mary and is now Associate Professor in the Department of Computer Science. Her research interests are in design and analysis of algorithms and combinatorial optimization. She has published papers in the areas of bin packing, job scheduling, network routing, and graph partitioning and mapping.