

SCHEDULING INDEPENDENT JOBS WITH STOCHASTIC PROCESSING TIMES AND A COMMON DUE DATE ON PARALLEL AND IDENTICAL MACHINES

Erdal EREL * and Subhash C. SARIN

Department of Industrial Engineering and Operations Research,

Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061, U.S.A.

** Department of Management, Bilkent University, P.O. Box 8,06572, Maltepe, Ankara, Turkey*

Abstract

We propose a heuristic procedure that constructs a schedule for N jobs with stochastic processing times and a common due date on M parallel, identical machines. The criterion is the minimization of the total expected incompletion cost. A worst-case analysis for the ratio of the heuristic and optimal solutions is presented and a bound on the ratio is derived. The experimental results presented indicate that the heuristic procedure generates almost optimal solutions.

1. Introduction

In this paper we consider the problem of scheduling N jobs on M parallel, identical machines with jobs having a common due date and stochastic processing times. A job, if not completed within the due date, incurs a fixed incompletion cost to complete it at some other facility. The problem is to allocate and sequence the jobs on the machines with the objective to minimize the expected incompletion cost or the expected weighted number of tardy jobs. When the expected incompletion costs of the jobs are equal, then the criterion can be considered as minimizing the expected number of tardy jobs. This problem is a stochastic version of the M -parallel processor problem – a well known problem in the scheduling literature and practice. In an attempt to solve the assembly line balancing problem by the branch-and-bound type of procedure [6], the solution to this problem gives a bound value to effectively fathom the enumeration tree.

The problem of scheduling independent jobs on parallel, identical machines was first considered by McNaughton [14]. He developed rules for minimizing the total completion time where part of the same job may be assigned to more than one machine. Later, several researchers [3,5,8,9,12,17,18,19,22] examined the problem for different performance measures. For a review of sequencing and scheduling jobs on parallel machines, Graham, et al. [11] present an excellent survey. Lately, considerable research has been reported in the area of stochastic scheduling. For a review, the reader is referred to papers by Pinedo [15] and Pinedo and Weiss [16].

In view of the complexity of the problem, work has focused on developing heuristic procedures. One of the main measures of performance of a heuristic procedure for the minimization problem is the minimization of the ratio of the solution value obtained using the heuristic procedure to that of the optimal solution value with the least computational requirements. Sarin and Elmaghraby [20] proposed a heuristic procedure for the criterion of minimization of the total weighted completion times. They derived bounds on the worst-case performance of the procedure. Loulou [13] obtained upper bounds on the difference of the values of the heuristic solution and the optimal solution, whereas Graham [10] and Garey and Graham [7] obtained bounds on the ratio of the values of the heuristic solution and the optimal solution for the objective of minimizing the makespan. Coffman and Gilbert [2] obtained bounds on the ratio of the values of the heuristic solution and the optimal solution where the job performance times were chosen from a uniform distribution or an exponential distribution for the objective of minimizing the makespan. Bruno and Downey [1] determined an expression such that the probability of the ratio of the values of the heuristic solution and the optimal solution being less than the expression determined is greater than a prespecified value. The objective was to minimize the makespan, the job performance times were chosen from a uniform distribution. Eastman et al. [4] derived lower and upper bounds on the cost of an optimal schedule for the problem with the objective of minimizing total weighted flowtimes.

Our problem is different from those referred to above in the sense that an incompleteness cost is incurred for each job (proportional to its expected processing time) while the incompleteness probability depends on the position of the job in the sequence and its expected processing time. Job processing times are assumed to be random variables. We propose a heuristic and present a worst-case analysis on the ratio of the heuristic and optimal solutions. A finite bound on the ratio is derived. A bound on the ratio for an example problem is also defined and shown to be quite close to unity. In words, the heuristic proceeds as follows. The single machine sequencing rule presented by Sarin and Erel [21] gives a criterion to determine the relative order in which two jobs should appear in the optimal sequence of jobs on a single machine. This rule is applied to the problem on hand by transferring it to an equivalent single machine problem with its due date being equal to the product of the number of machines and the given due date of the problem. Once the jobs are sequenced on an equivalent single machine problem, they are then allocated in that order to the M machines. The jobs within each machine are resequenced according to the single machine sequencing rule, and the resultant schedule is the desired one.

In the sequel, we first present some notation and assumptions used for the problem discussed in this paper. Then the single machine sequencing rule is reviewed. The heuristic procedure of constructing a schedule on M machines is described and a property which translates the single machine sequence into a M -machine schedule is developed. Next, we present an analysis on the ratio of the

heuristic and optimal solutions. Finally, some computational experience on the performance of the procedure on some randomly generated problems is reported.

2. Notation and Definitions

Consider M parallel, identical machines with N jobs available at time zero. Assume that the machines are free at the moment and we wish to allocate and sequence the jobs among the machines. This problem can be considered as the generalization of the single machine sequencing problem for the objective of minimizing the expected incompletion cost to M parallel machines. The performance measure to be optimized is the expected incompletion cost. Let

f_i = flowtime of job i , for $i = 1, \dots, N$

d = common due date for all jobs

IC_i = incompletion cost of job i , for $i = 1, \dots, N$.

The performance measure can be expressed as:

$$\text{Min} \sum_{i=1}^N IC_i \times \Pr[f_i > d].$$

Each machine can process only one job at a time and, having once started a job, continues until its completion. Also, splitting of jobs among machines is not permitted. Job processing times are assumed to be normally distributed with known means. Wilhelm [23] has shown that the normal distribution is a legitimate distribution for representing processing times of jobs especially if the coefficient of variation is 0.3 or less. A job can be considered to be consisting of a number of elementary tasks. As the number of elementary tasks increases, it is shown that the normal distribution becomes a better approximation of their cumulative time. The mean and variance of the processing time of a job is the sum of the means and variances of the elementary task times. The greater the number of elementary tasks the greater will be the mean and variance of the job. Therefore, a large expected processing time of a job implies a greater number of elementary tasks and consequently a larger variance. Accordingly, let $\sigma_i^2 = a \times \mu_i$, $i = 1, \dots, n$, where a is a constant. Such a relationship between the mean and variance is atypical – consider, for example, the Poisson and binomial distributions which are approximated by the normal for certain parameter values. Also, the greater the expected processing time of a job, the longer it will take to complete it off-line thereby resulting in greater incompletion cost. Hence, assume $IC_i = k \times \mu_i$, $i = 1, \dots, n$, where k is a constant representing the cost of completion per unit time. In the above expression, as the sum of normal variates is normal, f_i is a normally distributed random variable for all i .

Let x denote the sum of the means of the processing times of the jobs assigned to a station, then the incompletion probability

$$p(x) = 1 - \Phi\left[\frac{d - xf}{\sqrt{ax}}\right] = \Phi\left[\frac{x - d}{\sqrt{ax}}\right].$$

The incompletion probability function, $p(x)$ is monotonically increasing and convex over the interval $0 \leq x \leq d'$ and monotonically increasing and concave for $x \geq d'$, for some $d' \leq d$ as shown in Sarin and Erel [21].

3. Review of the single machine sequencing rule

The single machine sequencing rule presented by Sarin and Erel [21] gives a criteria to determine the relative order in which two jobs should appear in the optimal sequence. The rule can be summarized as follows. If for any pair of jobs, the sum of the expected processing times of the remaining jobs is larger than d' , then the sequence which minimizes expected incompletion cost has two regions. The jobs are ordered in a nonincreasing order of their incompletion costs in the first region while they are ordered in a nondecreasing order of their incompletion costs in the second region. If the sum of the expected processing times of all the jobs is less than d' , then the problem is trivial to solve: Jobs are ordered in the descending order of their incompletion costs. This rule is implemented in a special enumeration scheme to generate the optimal sequence. It was observed that selecting the best node among the first 100 nodes generated by this enumeration scheme results in almost optimal sequences.

A property of this procedure which will be referred to later in this paper is as follows.

PROPERTY 1

If the sum of the expected processing times of all the jobs except the one with the smallest expected processing time is less than or equal to d' ($d' \leq d$), then the problem is trivial to solve, i.e., arranging jobs in the nondecreasing order of their incompletion costs gives the optimal solution.

4. Development of the heuristic procedure

Consider an arbitrary schedule R in which jobs i and j are assigned to machines s and t , respectively. Moreover, let's assume that $\xi_i \leq \xi_j$, where ξ_i is the sum of the expected processing times of the jobs, Z_i , preceding job i on the machine to which job i belongs. That is, $\xi_i = \sum_{k=1}^{Z_i} \mu_k$. In the schedule R' , the jobs i and j are interchanged in position. The situation is depicted in fig. 1. Let p_i and p_j be the incompletion probabilities of jobs i and j respectively in schedule

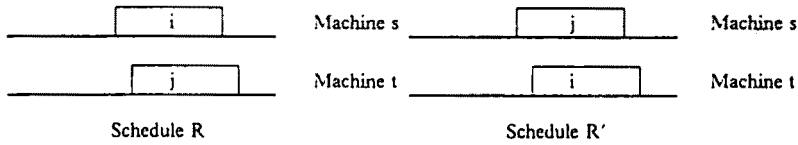


Fig. 1. Relative positions of jobs i and j in schedules R and R' .

R , and let p'_i and p'_j denote the incompletion probabilities of jobs i and j , respectively, in schedule R' .

The following theorem determines the relation between the increase in the incompletion probability of job i and the decrease in the incompletion probability of job j as a result of interchanging their positions, and states a relationship between the contributions of the two jobs to the total cost in schedules R and R' .

THEOREM 1

If $\xi_i + \mu_i \leq d'$, $\xi_j + \mu_j \leq d'$ ($d' \leq d$), $\mu_i \geq \mu_j$ and $\xi_i \leq \xi_j$, then $p'_i - p_i \geq p_j - p'_j$, and the contribution of the jobs i and j to the total cost in schedule R is less than or equal to that in schedule R' . If $\xi_i \geq d'$, $\mu_i \geq \mu_j$ and $\xi_i \leq \xi_j$, then $p'_i - p_i \leq p_j - p'_j$, and the contribution of the jobs i and j to the total cost in schedule R' is less than or equal to that in schedule R .

Proof

First, as shown in Sarin and Erel [21], the incompletion probability function for the normal distribution is such that for $x \leq d'$, it is monotonically increasing and convex, and for $x \geq d'$, it is monotonically increasing and concave. The first case, namely, $\xi_i + \mu_i \leq d'$ and $\xi_j + \mu_j \leq d'$ of the theorem belongs to the monotonically increasing and convex portion of the incompletion probability function and is shown in fig. 2. It depicts p_i , p'_i , p_j and p'_j . Since $\mu_i \geq \mu_j$ and $\xi_i \leq \xi_j$, $(p'_i - p_i)$ (designated by A) is clearly larger than $(p_j - p'_j)$ (designated by B) because both $(p'_i - p_i)$ and $(p_j - p'_j)$ represent change in probability over the same interval, namely $\xi_j - \xi_i$, the interval corresponding to $p'_i - p_i$ being placed to the right of that corresponding to $p_j - p'_j$. Let $\text{cost}(R)$ be the contribution of

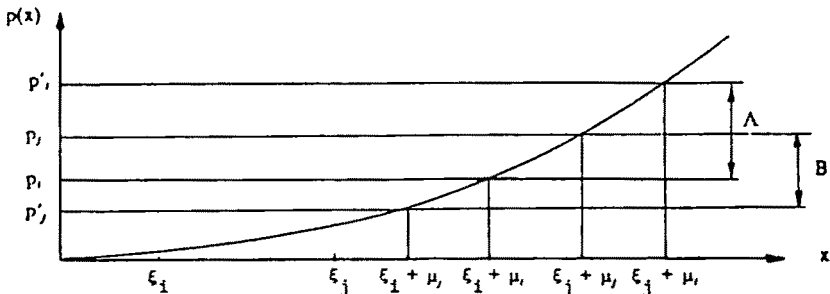


Fig. 2. Incompletion probabilities of tasks i and j in schedules R and R' .

the jobs i and j in schedule R . Since $IC_i \geq IC_j (\mu_i \geq \mu_j)$, it follows that $IC_i(p'_i - p_i) \geq IC_j(p_j - p'_j)$ or $IC_i p'_i + IC_j p'_j \geq IC_i p_i + IC_j p_j$, thereby implying that $\text{cost}(R') \geq \text{cost}(R)$.

The second case of the theorem belongs to the monotonically increasing and concave portion of the incompletion probability function and using arguments similar to those above it follows that $p'_i - p_i \leq p_j - p'_j$. Since $IC_i \geq IC_j$, it follows that $\text{cost}(R') \leq \text{cost}(R)$. Q.E.D.

In words, the above result implies that for a pair of jobs i and j assigned to different machines, if $\xi_i + \mu_i \leq d'$ and $\xi_j + \mu_j \leq d'$, then the job with larger incompletion cost should occupy the earlier position. If $\xi_i \geq d'$ and $\xi_j \geq d'$, then the job with smaller incompletion cost should occupy the earlier position.

4.1. PROCEDURE TO SCHEDULE N JOBS ON M MACHINES

The heuristic generates a M -machine schedule from the single machine sequence as follows:

Step 1. Set the due date equal to $M \times d$. Obtain the single machine sequence for this due date with the single machine sequencing rule.

Step 2. Allocate jobs to M machines sequentially in their order of appearance in the single machine sequence by assigning the next job to the machine that has the least sum of the expected processing times of the jobs already assigned to it. Continue until all the jobs are assigned.

Step 3. The jobs belonging to each machine are resequenced according to the single machine sequencing rule.

Step 2 of the heuristic follows from theorem 1. Step 3 is applied to the schedule generated in step 2, since the individual machine sequences obtained in step 2 may not satisfy the single machine sequencing rule.

5. Analysis on the ratio of the heuristic and optimal solutions

Consider the job i with an expected processing time of μ_i and partition it into h jobs. This is depicted in fig. 3. Note that the sum of the expected incompletion costs of the h jobs is equal to $\sum_{l=1}^h IC_l p_l$ and $p_h = p_i$.

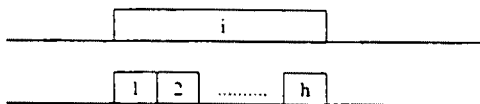


Fig. 3. Partitioning of job i into h jobs.

The following lemma determines a relationship between the expected incomple-
 tion cost of job i and the sum of the expected incomple-
 tion costs of the h jobs.

LEMMA 1

$$IC_i p_i > \sum_{l=1}^h IC_l p_l, \text{ for } h > 1.$$

Proof

By construction, $\mu_i = \sum_{l=1}^h \mu_l$ and $p_i > p_l$ for $l = 1, \dots, h - 1$, and $p_i = p_l$ for $l = h$, since $\xi_i + \mu_i \geq \xi_l + \mu_l$ for $l = 1, \dots, h - 1$, and $\xi_i + \mu_i = \xi_l + \mu_l$ for $l = h$. Therefore, $\mu_i p_i > \sum_{l=1}^h \mu_l p_l$ for $h > 1$. Since $IC_i = k \times \mu_i$ for all i , it follows that $IC_i p_i > \sum_{l=1}^h IC_l p_l$ for $h > 1$. Q.E.D.

Consider one of the h jobs and partition it into h' jobs ($h' > 1$). It follows from lemma 1 that the sum of the expected incomple-
 tion costs of the h' jobs is less than the expected incomple-
 tion cost of the original job. Thus, as the number of partitioned jobs increases, the expected incomple-
 tion cost of the sequence decreases. The following property summarizes the relation between the number of partitionings and the decrease in the expected incomple-
 tion cost of the sequence.

PROPERTY 2

In lemma 1, as the number of partitionings of a job increases, the difference between $IC_i p_i$ and $\sum_{l=1}^h IC_l p_l$ also increases.

To obtain desirable partitions of jobs, let

$$\mu_{\min} = \min_{j=1, \dots, N} \{ \mu_j \}$$

and

$$\mu_{\max} = \max_{j=1, \dots, N} \{ \mu_j \}.$$

Consider machine q with r jobs on it. Let T_q denote the sum of the expected processing times of the jobs on machine q . Replace the jobs on machine q by

$$f\left(\frac{T_q}{\mu_{\min}}\right)$$

jobs with expected processing times equal to μ_{\min} , where $f(x)$ denotes the largest integer less than or equal to x . Note that there will be

$$s = f\left(\frac{T_q}{\mu_{\min}}\right)$$

jobs with expected processing times equal to μ_{\min} and a fractional job with an expected processing time less than μ_{\min} as the last job on machine q . Let this

fractional job be denoted by v . The following lemma determines a relationship between the sum of the expected incompleteness costs of the jobs on machine q and

$$\sum_{i=1}^s IC_i p_i + IC_v p_v.$$

LEMMA 2

$\sum_{i=1}^s IC_i p_i + IC_v p_v$ is a lower bound on the sum of the expected incompleteness costs of the jobs on machine q .

The proof of this lemma follows from lemma 1 and property 2 and is omitted here for the sake of brevity.

Consider a schedule that has

$$B_M = f\left(\frac{1}{M\mu_{\min}} \sum_{i=1}^N \mu_i\right)$$

jobs with expected processing times equal to μ_{\min} on each machine. Let this schedule be denoted by $\Psi_{M,1}$. Similarly, let $\Psi_{M,2}$ denote the schedule that has

$$A_M = g\left(\frac{1}{M\mu_{\max}} \sum_{i=1}^N \mu_i\right) + 1$$

jobs with expected processing times equal to μ_{\max} on each machine, where $g(x)$ is the smallest integer larger than or equal to x . Note that $T_q = B_m \times \mu_{\min}$ and $T_q = A_M \times \mu_{\max}$ for all q in $\Psi_{M,1}$ and $\Psi_{M,2}$, respectively. Let $V(\Psi_{M,(\cdot)})$ denote the cost of the schedule $\Psi_{M,(\cdot)}$, and V_M denote the cost of any schedule of M machines of the N jobs; V_M^* denotes the cost of the optimal M -machine schedule. The following corollary determines a relationship between $V(\Psi_{M,1})$ and V_M .

COROLLARY 1

$V(\Psi_{M,1})$ constitutes a lower bound on V_M .

Proof

The proof of this corollary is by construction as follows. Consider any schedule of the N jobs on M machines. Replace the jobs on machine q by

$$f\left(\frac{T_q}{\mu_{\min}}\right)$$

jobs with expected processing times equal to μ_{\min} , for $q = 1, \dots, M$. Thus, there will be

$$f\left(\frac{T_q}{\mu_{\min}}\right)$$

jobs with expected processing times of μ_{\min} , and a job with an expected processing time less than μ_{\min} may remain as the last job on machine q , for $q = 1, \dots, M$. It follows from lemma 2 that the cost of this schedule is a lower bound on V_M . Let the fractional jobs on machines with large T_q 's be combined with the ones on machines with small T_q 's to form jobs with expected processing times equal to μ_{\min} . This process obviously decreases the cost of the schedule. The cost of the schedule can be further decreased by transferring jobs from machines with larger number of jobs to machines with smaller number of jobs. If the number of jobs on each machine cannot be made equal to each other, then the last jobs of the machines with higher number of jobs are pruned, so that each machine would have the same number of jobs. Obviously, the cost of this schedule is a lower bound on V_M . On the other hand, the resulting schedule is equivalent to $\Psi_{M,1}$ and it follows that $V(\Psi_{M,1})$ is a lower bound on V_M . Q.E.D.

The following corollary determines the relationship between $V(\Psi_{M,2})$ and the cost of any schedule in which the jobs are arranged such that they follow the conditions specified by Theorem 1.

COROLLARY 2

$V(\Psi_{M,2})$ constitutes an upper bound on the cost of any schedule of the N jobs that follow the conditions specified by theorem 1.

Proof

The proof of this corollary also follows by construction. Consider any schedule of the problem that satisfies theorem 1. Replace the jobs on machine q by

$$g\left(\frac{T_q}{\mu_{\max}}\right)$$

jobs with expected processing times equal to μ_{\max} , for $q = 1, \dots, M$. It follows from lemma 2 that the cost of this schedule is an upper bound on the cost of the

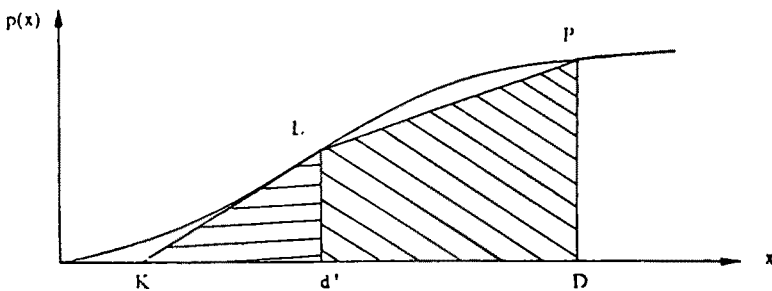


Fig. 4. Incompletion probability function, $p(x)$, and a lower bound on $\int p(x) \cdot dx$.

original schedule. Note that as the maximum difference between the number of jobs on the machines is one, since in any schedule that satisfies theorem 1,

$$\max_{i,j=1,\dots,N} \{T_i - T_j\} \leq \mu_{\max}.$$

The number of jobs on each machine can be made equal to each other by appending a job with an expected processing time of μ_{\max} to the machines with less number of jobs. Clearly, the cost of this schedule is also an upper bound on the cost of the original schedule. The resulting schedule has equal or less number of jobs on each machine than $\Psi_{M,2}$. Therefore, it follows that $V(\Psi_{M,2})$ is an upper bound on the cost of any schedule that satisfies theorem 1. Q.E.D.

The incompleteness probability function is shown in fig. 4. A lower bound on the integral of $p(x)$ from zero to D , for some $D > d'$, can be represented by the areas of the triangle KLd' and trapezoid $d'LPD$ as shown in fig. 4.

In order to compute these areas, note that

$$\text{slope of the line } KL = S1_{KL} = \frac{d}{dx} \left[1 - \Phi \left(\frac{d-x}{\sqrt{ax}} \right) \right]$$

where $x = d'$

$$\frac{d}{dx} \left[1 - \Phi \left(\frac{d-x}{\sqrt{ax}} \right) \right] = \frac{\exp \left(-\frac{(x-d)^2}{2ax} \right)}{\sqrt{2\pi}} \times \frac{(x+d)}{2\sqrt{a}x^{3/2}}.$$

Substituting d' instead of x yields the following expression for $S1_{KL}$:

$$S1_{KL} = \frac{\exp \left(-\frac{(d'-d)^2}{2ad'} \right)}{\sqrt{2\pi}} \times \frac{(d'+d)}{2\sqrt{a}d'^{3/2}}.$$

Therefore, area of the triangle

$$KLd' = \frac{[p(d')]^2 \sqrt{2\pi a} d'^{3/2}}{(d'+d) \exp \left(-\frac{(d'-d)^2}{2ad'} \right)}.$$

Area of the trapezoid

$$d'LPD = [D - d'] \left[\frac{p(d')}{2} + \frac{1}{2} \Phi \left(\frac{D-d}{\sqrt{aD}} \right) \right].$$

Hence,

$$\int_0^D p(x) \cdot dx > \frac{[p(d')]^2 \sqrt{2\pi a} d'^{3/2}}{(d'+d) \exp \left(-\frac{(d'-d)^2}{2ad'} \right)} + [D - d'] \left[\frac{p(d')}{2} + \frac{1}{2} \Phi \left(\frac{D-d}{\sqrt{aD}} \right) \right] \text{ for } D > d'. \tag{1}$$

Let $\Psi_{M,0}$ be the schedule generated by the heuristic, then the ratio

$$\frac{V(\Psi_{M,0})}{V_M^*}$$

is a measure of performance of the heuristic. Clearly,

$$\frac{V(\Psi_{M,0})}{V_M^*} \geq 1.$$

Theorem 2 states a worst-case upper bound on this ratio.

THEOREM 2

$$\frac{V(\Psi_{M,0})}{V_M^*} < \frac{2}{p(d') + 0.5} \left[1 + \frac{1.5\mu_{\max} + d' + \mu_{\min}}{\frac{1}{M} \sum_{i=1}^N \mu_i - d' - \mu_{\min}} \right],$$

for $\frac{1}{M} \sum_{i=1}^N \mu_i > d' + \mu_{\min}$.

Proof

From corollaries 1 and 2, it follows that:

$$\frac{V(\Psi_{M,0})}{V_M^*} \leq \frac{V(\Psi_{M,2})}{V(\Psi_{M,1})}. \tag{2}$$

We will first define a lower bound on $V(\Psi_{M,1})$ and an upper bound on $V(\Psi_{M,2})$.

$V(\Psi_{M,1})$ can be expressed as:

$$V(\Psi_{M,1}) = Mk\mu_{\min} \sum_{i=1}^{B_M} \Phi \left(\frac{i\mu_{\min} - d}{\sqrt{ia\mu_{\min}}} \right).$$

Note that all tasks have an incompleteness cost of $k\mu_{\min}$. A lower bound on $V(\Psi_{M,1})$ can be expressed as follows:

$$V(\Psi_{M,1}) = Mk\mu_{\min} \sum_{i=1}^{B_M} \Phi \left(\frac{i\mu_{\min} - d}{\sqrt{i - a\mu_{\min}}} \right) > Mk \int_0^{U_M} p(x) dx \tag{3}$$

where $U_M = \mu_{\min} B_M$. The relationship between

$$\mu_{\min} \sum_{i=1}^{B_M} \Phi \left(\frac{i\mu_{\min} - d}{\sqrt{ia\mu_{\min}}} \right)$$

and the area under the incompletion probability function from zero to U_M is depicted in fig. 5.

Following expression (1), expression (3) can be written as:

$$V(\Psi_{m,1}) > Mk \left[\frac{[p(d')]^2 \sqrt{2\pi a} d'^{3/2}}{(d' + d) \exp\left(-\frac{(d' - d)^2}{2ad'}\right)} + [U_M - d'] \left[\frac{p(d')}{2} + \frac{1}{2} \Phi\left(\frac{U_M - d}{\sqrt{aU_M}}\right) \right] \right], \quad U_M > d'.$$

Ignoring the first term and replacing

$$\Phi\left(\frac{U_M - d}{\sqrt{aU_M}}\right)$$

by its lower bound value of 0.5, we obtain:

$$V(\Psi_{M,1}) > Mk \left[\frac{p(d')}{2} + 0.25 \right] [U_M - d'], \quad U_M > d', \tag{4}$$

Since

$$U_M = \mu_{\min} f\left(\frac{1}{M\mu_{\min}} \sum_{i=1}^N \mu_i\right) \geq \mu_{\min} \left(\frac{1}{M\mu_{\min}} \sum_{i=1}^N \mu_i - 1\right),$$

expression (4) can be written as follows:

$$V(\Psi_{M,1}) > Mk \left[\frac{p(d')}{2} + 0.25 \right] \left[\frac{1}{M} \sum_{i=1}^N \mu_i - d' - \mu_{\min} \right],$$

$$\text{for } \frac{1}{M} \sum_{i=1}^N \mu_i > d' + \mu_{\min}. \tag{5}$$

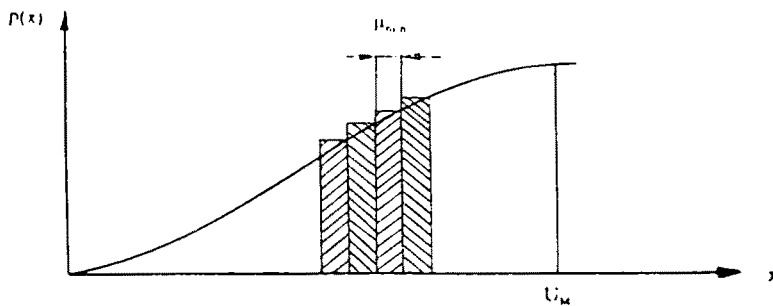


Fig. 5. Relation between $\mu_{\min} \sum_{i=1}^{B_M} \left(\frac{i\mu_{\min} - d}{\sqrt{ia\mu_{\min}}}\right)$ and the area under the incompletion probability function.

An upper bound on $V(\Psi_{M,2})$ can be derived using corollary 2 as follows. Each machine in $\Psi_{M,2}$ has A_M tasks that have expected processing times equal to μ_{\max} . Note that each task has an incompleteness cost of $k\mu_{\max}$. An upper bound on $V(\Psi_{M,2})$ can be written as:

$$V(\Psi_{M,2}) \leq Mk\mu_{\max} \sum_{i=1}^{A_M} \Phi\left(\frac{i\mu_{\max} - d}{\sqrt{ia}\mu_{\max}}\right) \tag{6}$$

Since $\Phi\left(\frac{1\mu_{\max} - d}{\sqrt{1a}\mu_{\max}}\right) \leq 0.5$ for $i = 1$,

and

$$\Phi\left(\frac{i\mu_{\max} - d}{\sqrt{ia}\mu_{\max}}\right) < 1.0 \text{ for } i = 2, \dots, A_M,$$

we can write the following expression:

$$\sum_{i=1}^{A_M} \Phi\left(\frac{i\mu_{\max} - d}{\sqrt{ia}\mu_{\max}}\right) < 0.5 + (A_M - 1) = A_M - 0.5. \tag{7}$$

Substituting expression (7) into expression (6) yields:

$$V(\Psi_{M,2}) < Mk\mu_{\max}(A_M - 0.5). \tag{8}$$

Since $A_M = g\left(\frac{1}{M\mu_{\max}} \sum_{i=1}^N \mu_i\right) + 1 \leq \frac{1}{M\mu_{\max}} \sum_{i=1}^N \mu_i + 2$,

Expression (8) can be written as follows:

$$V(\Psi_{M,2}) < Mk\mu_{\max}(A_M - 0.5) \leq Mk\mu_{\max} \left[\frac{1}{M\mu_{\max}} \sum_{i=1}^N \mu_i + 1.5 \right]. \tag{9}$$

By substituting the relationships given by expressions (5) and (9) into expression (2), we obtain,

$$\begin{aligned} \frac{V(\Psi_{M,0})}{V_M^*} &\leq \frac{V(\Psi_{M,2})}{V(\Psi_{M,1})} < \frac{Mk\mu_{\max} \left(\frac{1}{M\mu_{\max}} \sum_{i=1}^N \mu_i + 1.5 \right)}{Mk \left[\frac{p(d')}{2} + 0.25 \right] \left[\frac{1}{M} \sum_{i=1}^N \mu_i - d' - \mu_{\min} \right]} \\ &= \frac{2}{p(d') + 0.5} \left[1 + \frac{1.5\mu_{\max} + d' + \mu_{\min}}{\frac{1}{M} \sum_{i=1}^N \mu_i - d' - \mu_{\min}} \right]. \quad \text{Q.E.D.} \end{aligned}$$

Note that the bound derived in theorem 2 is defined for

$$\frac{1}{M} \sum_{i=1}^N \mu_i > d' + \mu_{\min}.$$

When

$$\frac{1}{M} \sum_{i=1}^N \mu_i \leq d' + \mu_{\min},$$

the problem is trivial to solve as indicated by property 1, because the optimal solution can be obtained simply by assigning the larger tasks to the earliest available position on any machine. Although the value of the bound increases as

$$\frac{1}{M} \sum_{i=1}^N \mu_i \text{ approaches } d' + \mu_{\min},$$

it remains at quite acceptable values for practical problem parameters. Also note that the value of the bound is an upper bound on the worst-case performance of the heuristic, since it is determined considering the two extreme cases simultaneously, namely, pertaining to the schedules $\Psi_{M,1}$ and $\Psi_{M,2}$. As can be easily seen, the value of the bound increases with increases in d' , M , μ_{\min} and μ_{\max} , but it decreases with an increase in $\sum_{i=1}^N \mu_i$. In other words, as the problem size increases, the value of the bound decreases. This is an attractive feature of the bound.

For sufficiently large d and small a values, it is shown in Sarin and Erel [21] that d' and d are quite close to each other. Assuming $d' = d$ and since $p(x = d) = 0.5$, then the value of the bound derived in theorem 2 reduces to the following expression:

$$\frac{V(\Psi_{M,0})}{V_M^*} < 2 \left[1 + \frac{1.5\mu_{\max} + d + \mu_{\min}}{\frac{1}{M} \sum_{i=1}^N \mu_i - d - \mu_{\min}} \right]$$

6. Performance of the heuristic

In this section, we illustrate the procedure of an example problem consisting of 6 jobs, and compare the solution of the heuristic to the optimal solution. Then,

Table 1
Parameters of the example problem

Job(i)	Mean(μ_i)	Variance(σ_i^2)	Incom. Cost(IC_i)
1	10.0	2.0	5.0
2	8.0	1.6	4.0
3	6.0	1.2	3.0
4	4.0	0.8	2.0
5	2.0	0.4	1.0
6	1.0	0.2	0.5

Table 2
Single-machine sequences and schedules for $M = 1, 2, 3$ and 4 for the example problem

M	Single-machine sequence	Incompletion cost of single-machine sequence	$\Psi_{M,0}$
1	2-6-5-4-3-1	11.087	Mach 1: 2-6-5-4-3-1
2	2-3-4-5-6-1	6.138	Mach 1: 2-5-6 Mach 2: 3-4-1
3	1-2-3-4-5-6	1.233	Mach 1: 1-6 Mach 2: 2-5 Mach 3: 3-4
4	1-2-3-4-5-6	0.000	Mach 1: 1 Mach 2: 2 Mach 3: 3-6 Mach 4: 4-5

we present an analysis on the performance of the heuristic by investigating the bound value on several randomly generated problems. The other relevant data are depicted in table 1.

The procedure is applied to the example problem for 1, 2, 3 and 4 machines. For $M = 1$, the problem becomes a single-machine sequencing problem, and the sequencing rule presented by Sarin and Erel [21] is applied for $d = 10$. The optimal sequence is obtained from the enumeration tree and is as follows: 2-6-5-4-3-1. The expected incompletion cost of this sequence is 11.09. For $M = 2$, the sequencing rule is applied for $d = 2 \times 10 = 20$, and the optimal single-machine sequence is as follows: 2-3-4-5-6-1. The expected incompletion cost of this sequence is 6.14. The jobs are allocated to the two machines sequentially in their order of appearance in the single-machine sequence as described in section 4.1. The 2-machine schedule, $\Psi_{2,0}$, generated and the associated total expected incompletion cost are depicted in tables 2 and 3, respectively. The single-machine sequences and the schedules generated for $M = 3$ and 4 are also depicted in table 2. $V(\Psi_{M,0})$ and V^* in table 3 represent heuristic and optimal solution values for different values of M .

The solutions of the heuristic procedure and the optimal solutions for 1, 2, 3 and 4 machines are depicted in table 3. As it is seen from the table, the heuristic

Table 3
Heuristic and optimal solutions of the example problem for 1, 2, 3 and 4 machines

M	$V(\Psi_{M,0})$	V^*
1	11.087	11.087
2	7.103	7.103
3	4.603	4.603
4	2.728	2.728

Table 4
Ratios of the heuristic solution value to that of the optimal solution value

# of jobs	# of mach	# of prob.	$d = 0.25 \times \sum_{i=1}^N \mu_i$			# of jobs	# of mach	# of prob.
			Value of the ratio					
			Ave.	Min.	Max.			
20	2	10	1.025	1.015	1.043	20	2	10
	3	10	1.049	1.036	1.072		3	10
	4	10	1.086	1.054	1.141		4	10
30	2	10	1.014	1.011	1.016	30	2	10
	3	10	1.026	1.021	1.033		3	10
	4	10	1.032	1.027	1.036		4	10
	5	10	1.061	1.045	1.080		5	10
	6	10	1.077	1.066	1.115		6	10
40	2	10	1.013	1.009	1.018	40	2	10
	3	10	1.020	1.018	1.024		3	10
	4	10	1.026	1.019	1.030		4	10
	5	10	1.032	1.024	1.043		5	10
	6	10	1.047	1.036	1.063		6	10
	7	10	1.067	1.053	1.082		7	10

procedure results in the optimal solution in each case. The optimal solution values are obtained by total enumeration.

The analysis presented in section 5 is a worst-case analysis on the ratio of the heuristic and optimal solutions as it was based on two extreme cases pertaining to $\Psi_{M,1}$ and $\Psi_{M,2}$. For a specific problem, a bound on the performance of the heuristic can be derived as follows.

$$\frac{V(\Psi_{M,0})}{V_M^*} \leq \frac{V(\Psi_{M,0})}{V(\Psi_{M,1})}$$

The bound given above is valid since $V(\Psi_{M,1})$ constitutes a lower bound on V_M^* , as shown in corollary 1.

To investigate the performance of the heuristic, the bound above was computed for several randomly generated problems. In the experimentation, we assume that $d' = d$, since the d values of all the problems are sufficiently large. Three sets of problems with 20, 30 and 40 tasks were created. Twenty-task problems were solved for 2, 3 and 4 machines, 30-task problems were solved for 2, 3, 4, 5 and 6 machines and 40-task problems were solved for 2, 3, 4, 5, 6 and 7 machines. For each number of machines, 10 problems were created. Due date was computed as $d = b \times [\sum_{i=1}^N \mu_i]$, and for each set three different values of b , namely, 0.25, 0.5 and 0.75 were used. Thus, a total of 420 problems were created and solved. In the test problems, $\mu_i \sim U[0; 20]$ with $\sigma_i^2 = a(\mu_i)$ and $IC_i = k(\mu_i)$ where $a \sim N[0.3; 0.067]$ and $k \sim N[0.05; 0.01]$. The bounds on the ratio of the

$d = 0.50 \times \sum_{i=1}^N \mu_i$			# of jobs	# of mach	# of prob.	$d = 0.75 \times \sum_{i=1}^N \mu_i$		
Value of the ratio						Value of the ratio		
Ave.	Min.	Max.				Ave.	Min.	Max.
1.044	1.014	1.055	20	2	10	1.094	1.066	1.123
1.075	1.044	1.106		3	10	1.172	1.118	1.230
1.117	1.054	1.178		4	10	1.236	1.177	1.302
1.028	1.023	1.033	30	2	10	1.055	1.034	1.062
1.045	1.030	1.056		3	10	1.097	1.054	1.121
1.066	1.053	1.079		4	10	1.134	1.065	1.171
1.091	1.072	1.104		5	10	1.185	1.147	1.210
1.117	1.080	1.145		6	10	1.245	1.149	1.313
1.020	1.014	1.029	40	2	10	1.039	1.028	1.054
1.030	1.026	1.035		3	10	1.062	1.042	1.081
1.042	1.034	1.049		4	10	1.087	1.054	1.111
1.059	1.044	1.076		5	10	1.111	1.087	1.146
1.065	1.044	1.091		6	10	1.153	1.096	1.201
1.094	1.075	1.116		7	10	1.186	1.162	1.226

values of the heuristic solution and the optimal solution for the problems solved are summarized in table 4. For each set of problems, the average, minimum and maximum bound values are given. As it is seen from table 4, the heuristic procedure generated almost optimal solution. Moreover, computationally, it is very easy to use.

7. Conclusions

In this paper we presented a heuristic procedure for the problem of scheduling N jobs with stochastic processing times on M machines with the objective of minimizing the expected incompleteness cost. The heuristic procedure constructs the M machine solution from a single machine sequence of the N jobs. Computational experience indicates that the heuristic procedure generates almost optimal solutions and it is very easy to use. An analysis of the worst-case value of the ratio of the heuristic and optimal solutions is presented and the worst-case value is shown to be finite. Moreover, the value of the ratio decreases as the problem size increases.

Acknowledgements

The authors are grateful to the referees for their useful comments which have helped to improve the presentation.

References

- [1] J.L. Bruno and P.J. Downey, Probabilistic bounds on the performance of list scheduling, *SIAM Journal on Computing* 15, No. 2 (1986) 409–417.
- [2] E.G. Coffman and E.N. Gilbert, On the expected relative performance of list scheduling, *Operations Research* 33, No. 3 (1985) 548–561.
- [3] A. Dogramaci and J. Surkis, Evaluation of a heuristic for scheduling independent jobs on parallel identical processors, *Management Science* 25, No. 12 (1979) 1208–1216.
- [4] W.L. Eastman, S. Even and I.M. Isaacs, Bounds for the optimal scheduling of n jobs on m processors, *Management Science* 11, No. 2 (1964) 268–279.
- [5] S.E. Elmaghraby and S.H. Park, Scheduling jobs on a number of identical machines, *AIIE Transactions* 6, No. 1 (1974) 113.
- [6] E. Erel and S.C. Sarin, A methodology to solve single-model, stochastic assembly line balancing problems, Working paper, 1987.
- [7] M.R. Garey and R.L. Graham, Bounds for multiprocessor scheduling with resource constraints, *SIAM Journal on Computing* 4, No. 2 (1975) 187–200.
- [8] M.R. Garey and D.S. Johnson, Two-processor scheduling with start-times and deadlines, *SIAM Journal on Computing* 6, No. 3 (1977) 416–426.
- [9] K.D. Glazebrook, Scheduling tasks with exponential service times on parallel processors, *Journal of Applied Probability* 16, No. 3 (1979) 685–689.
- [10] R.L. Graham, Bounds on multiprocessing timing anomalies, *SIAM Journal on Applied Mathematics* 17, No. 2 (1969) 416–429.
- [11] R.L. Graham, E.L. Lawler, J.K. Lenstra and A.H.G. Rinnoy Kan, Optimization and approximation in deterministic sequencing and scheduling: A survey (Stichting Mathematisch Centrum, Amsterdam, 1977).
- [12] J.N.D. Gupta and A.R. Maykut, Scheduling jobs on parallel processors with dynamic programming, *Decision Sciences* 4 (1973) 447–457.
- [13] R. Loulou, Tight bounds and probabilistic analysis of two heuristics for parallel processor scheduling, *Mathematics of Operations Research* 9, No. 1 (1984) 142–150.
- [14] R. McNaughton, Scheduling with deadlines and loss functions, *Management Science* 26 (1959) 1–12.
- [15] M. Pinedo, Stochastic scheduling with release dates and due dates, *Operations Research* 31, No. 3 (1983) 559–572.
- [16] M. Pinedo and G. Weiss, Scheduling jobs with exponentially distributed processing times and two machines with resource constraints, *Management Science* 30, No. 7 (1984) 883–889.
- [17] J.G. Root, Scheduling with deadlines and loss functions on k parallel machines, *Management Science* 11, No. 3 (1965) 460–475.
- [18] M.H. Rothkoph, Scheduling independent tasks on parallel processes, *Management Science*, 12, No. 5 (1966) 437–447.
- [19] S. Sahni, Preemptive scheduling with due dates, *Operations Research* 27, No. 5 (1979) 925–934.
- [20] S.C. Sarin and S.E. Elmaghraby, Bounds on the performance of a heuristic to schedule precedence-related jobs on parallel machines, *International Journal of Production Research* 22, No. 1 (1984) 17–30.
- [21] S.C. Sarin and E. Erel, Sequencing jobs on a single machine with a common due date and stochastic processing times, Working paper, Department of Industrial Engineering and Operations Research, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 1987.
- [22] R.R. Weber, Scheduling jobs with stochastic processing requirements on parallel machines to minimize makespan or flowtime, *Journal of Applied Probability* 19, No. 1 (1982) 167–182.
- [23] W.E. Wilhelm, On the normality of operation times in small-lot assembly systems: a technical note, *International Journal of Production Research* 25, No. 1 (1987) 145–149.