Scheduling the Assembly Process with Uncertain Material Arrivals

By

CHEUNG Chit-Cheung, Gavin

Department of Systems Engineering
& Engineering Management

The Chinese University of Hong Kong

A dissertation submitted in partial fulfillment of requirements
for the degree of Master of Philosophy

1998

Abstract

In designing the assembly process, the significance of uncertain arrival of materials has not reasonably addressed in the literature despite its significant impact on the on-time delivery performance of the end products. In this dissertation, we study a problem of process re-engineering with the concern for the effect of stochastic arrival of components on the assembly line. The objective is to minimize the overall impacts of uncertainty to the on-time delivery performance of the end product. It turns out that the problem can be formulated as a sequential optimization problem. To the best of our knowledge, no previous study has dealt with the similar problem. We show that the structure of the associated mathematical model is very similar to the single-machine problem, which aims at minimizing the total cost with respect to the completion-time of each job. It is unlikely to solve the problem optimally in polynomial time, in general. However, we are able to obtain solutions to some special cases. These solutions provide some insights in developing efficient heuristic algorithms. The performance of the heuristics is tested through solving several sets of problems. Satisfactory results of the heuristics are obtained in our experiment.

摘要

在安排產品的裝配過程時，設計者往往沒有考慮到原料的不確定到達時間對產品的準時交貨所帶來的影響。這篇論文主要就是要針對原料的不確定到達這個因素，從而研究如何能夠通過流程重新設計來減底它對產品準時交貨的影響。這其實是一個順序優化問題，據我們所知，到現時爲止並未有相類似的研究發表過。然而，當把問題轉化成數學模型後，我們發現它在數學結構上與一個NP-完備的單機器問題很相似，那個單機器問題的目標是把由各個工序的完成時間所決定的成本減至最低。由於那個單機器問題已被證明是NP-完備，我們相信這裡所提出的問題也未能找到任何多項式算法可以解決。然而，有某些特例是可以很快找到解決方案的，這些特例的解決方案更給我們提供了一些提示，讓我們可以建立一些有效的啓發式算法。我們通過電腦程式去測試所建立的啓發式算法的效能，實驗結果顯示我們的啓發式算法有滿意的表現。

Acknowledgment

I would like to thank many people who have helped and guided me through my research work. In particular, I should express my deepest gratitude to Professor H. Yan, my research supervisor, for his invaluable guidance and encouragement. I would like to thank my colleagues who made my study much memorable and joyful.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Scheduling is concerned with the allocation of resources over time to perform a collection of tasks that exists in most manufacturing and production environments. The sequencing problem is a specialized scheduling problem in which the ordering of jobs completely determines the schedule concerned. Moreover, the sequencing problem is one concerning a single resource, or machine [1]. As simple as it is, however, the single-machine scheduling problem is still very important for several reasons. It illustrates a variety of scheduling topics in a tractable model. It provides a context in which we may investigate many different performance measures and several solution techniques. It is also a building block for the development of a comprehensive understanding of scheduling concepts, an understanding that should ultimately facilitate the modeling of complicated systems. In order to understand completely the behavior of a complex system, it is vital to understand the working of its components and quite often the single-machine problem appears as an elementary component in a large scheduling problem. Sometimes it may even be possible to solve the embedded single-machine problem in dependently and then to incorporate the result into a larger problem. For example, in multiple-operation processes there is often a bottleneck stage, and the treatment of the bottleneck itself with single-machine analysis may determine the properties of the entire schedule. At other times, the level at which decisions must be made may dictate that the processing facility should be treated in the aggregate, as a single resource.

This dissertation investigates a sequencing problem of scheduling the assembly process of a set of components, which aims at maximizing the chance of on-time delivery of the end product subject to a fixed due date and uncertain release times of the components. This study is motivated by the design of assembly process in the assembly lines of the manufacturer of electronic devices in China, where the on-time delivery of the end product is sometimes suffered from the late arrival of raw components, such as electronic components and sub-assemblies.

1.1 Motivation

The assembly lines are one of the essential parts in the manufacturing process. Semi-finished products are transferred from the upstream to the assembly lines for accomplishment by adding various components. On-time delivery of the end products in the assembly lines in real life are subject to many sources of uncertainty. Among the sources with major impact are unstable release of the components and sub-assemblies.

Having visited a manufacturer of electronic devices in China, we learned from their experience that the on-time delivery of end product is adversely affected by the late arrival of the components from the overseas. Moreover, in the electronic industry, some components are commonly needed by many electronic devices, which are highly demanded throughout the year. Therefore, the transportation uncertainty and supply shortage of the components make the release times unstable.

A good schedule avoiding or reducing the probability of late delivery of the end products is highly desirable. Any lateness of delivery of the end product can be very costly because extra money should be paid for the expensive air transportation to cover the missing of the shipping schedule. Considering the major resources of uncertainty, one may add more machines to cope with the uncertainty arising from machine breakdowns. On the other hand, to tackle the uncertain arrival of the components, one may build up high safety stocks. However, higher safety stocks incur higher inventory cost and take higher risk of obsolescence due to short product life cycles of electronic components.

We are interested in investigating whether the effect of the uncertain arrival of the components may be reduced through process re-engineering of the current assembly process. Basically, first-come-first-serve (FCFS) policy seems to be a reasonable policy to process the components to the semi-finished products. However, the fact is that there exists precedence relations in the operations of the components due to the physical structure of the semi-finished products and the design of the assembly lines. In addition, the sequence of operations of the components is always decided in advance. This practice may obviously lead to higher chance of late delivery of the end product. For example, without considering the distribution of component arrivals, a component with high arrival uncertainty may be scheduled to process first. This means even though the chance of late release of this component is very high, the operations of other ready components still have to wait upon the completion of this component. We are strongly motivated to study the scheduling of the assembly process with respect to the arrival distribution of components.

Another example similar to the assembly process problem is the loading of goods to a ship. Goods are first transported by trucks or trains from various locations to ports for shipment. The arrival time of the goods varies due to transportation uncertainty. On the other hand, since the goods are shipped to different countries, for the convenience of unloading, those goods to be unloaded earlier should be placed on top of others in the ship. Therefore, the placement of goods in the ship as well as the order of loading of goods are decided in advance. However, lots of time may be taken for waiting the late goods if the goods are loaded in an order that is scheduled without considering the uncertain arrival of the goods. We hope to find a sequence of loading of goods such that the chance of waiting for any late goods is minimized. The problem of our study is generalized as below.

1.2 Problem Description

A series of operations are needed to produce a product. For each operation, one specific material (component) should be added to the semi-finished product. The processing time of each operation is fixed while the arrival time of material is a random variable which may be described by a probability distribution. Given a due date for the delivery of the end product, we should schedule the operations sequence such that the probability of on-time delivery is maximized.

In our case, we do not need to consider the precedence of the operations because we assumed that once the optimal schedule of the operations is obtained, the setting of the semi-finished product and the assembly can be adjusted according to the sequence of the operations. Moreover, without loss of generality, we assume that the expected (mean) arrival time of each material is the same. Furthermore, the distribution of any material arrival and the processing time of any operation are known.

The fitness of a schedule is evaluated by the overall probability of on-time delivery of the end product. This is determined by the product of a series of individual probabilities which represent the chance that a material is available before its originally scheduled production time. In this dissertation, the statement "overall probability of a sequence" is equivalent to the fitness of a sequence (or schedule).

1.3 Contributions

In this dissertation, we study a sequencing problem of assembly process where the objective is to minimize the overall impact of the uncertain arrival of components on the on-time delivery of the end product. We also demonstrate that the assembly process scheduling problem is very similar to the single-machine scheduling problem which aims at minimizing the total cost associated with the completion time of each job. Moreover, we construct an effective heuristic algorithm to find the solution of the general problem. In addition, some approaches are suggested to determine the upper-bound of the optimal solution, which is useful for evaluating the performance of heuristic algorithms. We hope this study will be a step towards more studies of the evaluation of the significance of uncertain supplies in assembly process scheduling.

1.4 Thesis Organization

This dissertation is organized as follows. A general introduction is presented in Chapter 1. In Chapter 2 we mathematically formulate the problem and illustrate the difficulties in solving it, which is followed by a literature review. Chapter 3 provides the solutions to some special cases of our problem. In addition, the approaches to finding the upper-bound of the optimal solution are discussed in this chapter as well. Chapter 4 presents a heuristic algorithm to solve the problem. Moreover, the experimental and analytical results of the heuristic algorithm are given in Chapter 5.

Finally, concluding remarks are given in Chapter 6. We summarize the results and provide further research directions in Chapter 6.

Chapter 2

Problem Formulation and Solution Approaches

In this chapter, we firstly formulate the problem mathematically. Second, we compare our problem with a known NP-complete single-machine problem. Finally, the difficulties of solving the problem are discussed.

2.1 Mathematical Modeling

A series of operations are required to produce the end product. Each operation has a deterministic processing time and is non-preemptive. One specific material with uncertain arrival time is required to carry out the operation. In other words, an operation can only be started after the appropriate material is available. We are trying to find a sequence such that, with respect to the uncertain material arrivals, the probability of on-time delivery of the end product is maximized. This problem is mathematically defined as follows:

Notations

$\mathcal{N}$ = set of the operation index, $\{1, 2, \cdots, n\}$

i = index of operation, $i = 1, 2, \cdots, n$

t_i = processing time of operation i, which is a constant,

$P_i(t)$ = distribution function of the material arrival for operation i,

$p_i(t)$ = probability density function of the material arrival for operation i at time t,

μ_i, σ_i^2 = mean and variance of the distribution of the material arrival for operation i respectively,

$\mathcal{S}$ = set of all permutations of n jobs,

s = a sequence in $\mathcal{S}$,

D = due date of the end product,

T_0 = starting time of a sequence,

$T_i(s)$ = starting time of operation i in the sequence s,

$\pi(s)$ = probability of on-time delivery of the end product associated by sequence s.

The objective is to find an optimal sequence s^* such that

$$\pi(s^*) = \max_{s \in \mathcal{S}} \{ \prod_{i=1}^{n} P_i(T_i(s)) \} \tag{2.1}$$

where we assume that all distribution functions are independent.

Initial Time of A Sequence

Lemma 2.1. *The initial time of a sequence T_0 should be delayed as much as possible,*

$$T_0 = D - \sum_{i=1}^{n} t_i \tag{2.2}$$

which is a constant without depending on the order of the operations.

Proof. Suppose an optimal sequence s^* is initialized at T' which is earlier than T_0, i.e. $T' < T_0$. The appropriate probability of s^* is,

$$\pi_{T'}(s^*) = \prod_{i=1}^{n} P_{[i]}(T' + \sum_{j=1}^{i-1} t_{[j]}) \tag{2.3}$$

where $[i]$ is the operation in the i-th position of sequence s. If we shift the initial time from T' to T_0, we get another probability as

$$\pi_{T_0}(s^*) = \prod_{i=1}^{n} P_{[i]}(T_0 + \sum_{j=1}^{i-1} t_{[j]}) \tag{2.4}$$

Since $P_i(t)$ is non-decreasing function and $T_0 > T'$, from (2.3) and (2.4), we have

$$\pi_{T_0}(s^*) > \pi_{T'}(s^*)$$

On the other hand, we should not start the sequence after T_0; otherwise, the delivery of the end product will be late. i.e.

If the initial time $T' > T_0$, then

$$\begin{aligned} T' &> D - \sum_{i=1}^{n} t_i \\ T' + \sum_{i=1}^{n} t_i &> D \end{aligned}$$

which implies the existence of lateness. □

Above Lemma also implies that the any idle time in the operation sequence does not improve the overall probability of on-time delivery of the end product.

2.2 Transformation of Problem

To the best of our knowledge, no study has dealt with the similar problem. The most relevant literature appears to be the single-machine problem. In the single-machine problem, the objective is to minimize the total cost with respect to the completion time of each job. By using our notations, the objective of the single-machine problem can be expressed as

$$\min_{s \in \mathcal{S}} \{ \sum_{i=1}^{n} f_i(T_i(s) + t_i) \} \tag{2.5}$$

where $f_i(t)$ for $t \geq 0$ is the individual cost function of job i, which is assumed to be non-decreasing and differentiable. Therefore, instead of *maximizing* the *product* of a series of non-decreasing functions used in our problem, the single-machine problem aims at *minimizing* the *sum* of a series of non-decreasing functions.

Now, by taking logarithm, our objective function in (2.1) becomes

$$\ln \prod_{i=1}^{n} [P_i(T_i(s))] = \sum_{i=1}^{n} \ln[P_i(T_i(s))] \tag{2.6}$$

Since $\ln(x)$ is monotonic increasing for all real x, we have:

if a sequence $s^* \in S$ *maximizes* $\ln[\pi(s)]$, *it also maximizes* $\pi(s)$, $\forall\, s \in \mathcal{S}$.

Furthermore, by substituting $\ln[P_i(t)]$ in (2.6) by $f_i(t)$, we express our objective function as:

$$\max_{s \in \mathcal{S}} \{\sum_{i=1}^{n} f_i(T_i(s))\}$$

Now, we see that the objective function of our problem is very similar to that of the singe machine problem. In the literature, however, it has been proved that the above single-machine problem is NP-complete [3]. In other words, it is unlikely for an algorithm to solve the single-machine problem optimally in a reasonable time. Bearing the similarity of our problem and the single-machine problem in mind, we strongly believe that our problem is also NP-complete. Thus, in our study, we tackle the problem by developing some heuristics to find near-optimal solutions. Before constructing our heuristic, we have to study some special structures of our problem for more insights into developing a better heuristic algorithm.

2.3 Problem Analysis

In this section, we discuss the difficulties we encountered in the investigation of a solution methodology. The first part is difficulty finding the optimality criteria. The second difficulty is the construction the effective heuristic.

2.3.1 Optimality Criteria

In our problem, when we try to find the optimality criteria of a set of operations, one of the major difficulties comes from the correlation of the associated parameters of each operation. This is just like the common difficulty faced in some NP-complete scheduling problems which is explained as follows.

We consider two operations i and j in a sequences. Let $s' = \{A, i, j, B\}$ be a sequence in which operation j follows operation i immediately, where A and B are the sets of operations scheduled before operation i and after operation j respectively in s'. Let $s'' = \{A, j, i, B\}$ be the sequence obtained by switching position of the operation i and j in s'. We have,

$$\frac{\pi(s')}{\pi(s'')} = \frac{P_i(T)P_j(T+t_i)}{P_j(T)P_i(T+t_j)} \tag{2.7}$$

where $T = \sum_{k \in A} t_k$.

From (2.7), we also have

$$\pi(s') \geq \pi(s'')$$

if and only if

$$\frac{P_j(T+t_i)}{P_j(T)} \geq \frac{P_i(T+t_j)}{P_i(T)} \tag{2.8}$$

We cannot use (2.8) as the global optimality criteria for the two operations because the quantities in the left- and right-hand sides of the inequality in (2.8) depend on both the processing times of operations i and j, and the current state T. This means, given two operations, we can only decide the optimal order of i and j at the instant T. However, we cannot guarantee that this order remains at other instants. This

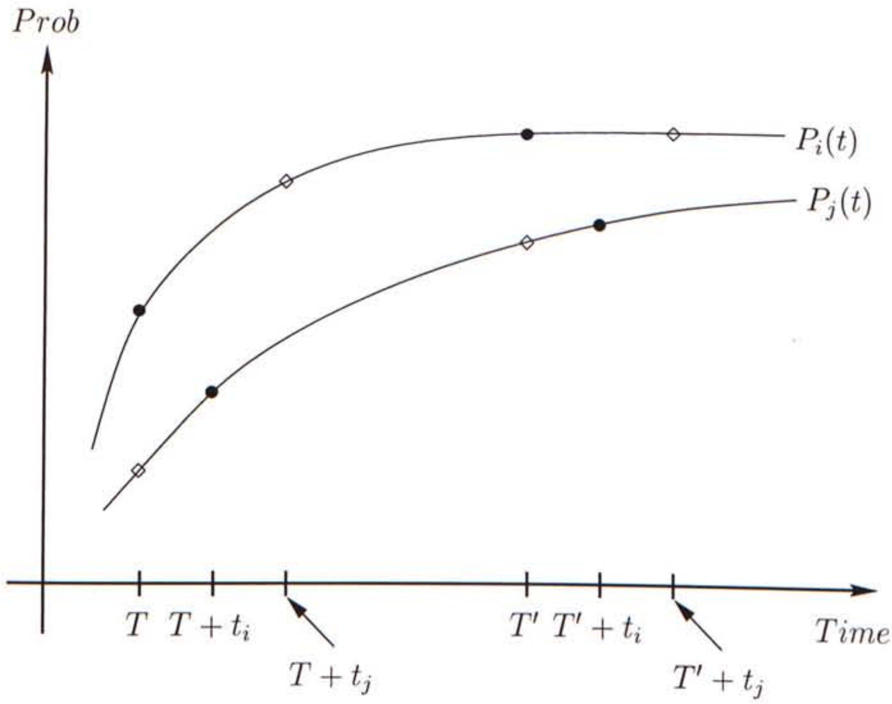

Figure 2.1: Optimal Orders in Two Instants

scenario is depicted in Figure 2.1 in which the curves are the distributions $P_i(t)$ and $P_j(t)$. Since

$$P_i(T)P_j(T+t_i) < P_j(T)P_i(T+t_j)$$

and,

$$P_i(T')P_j(T'+t_i) > P_j(T')P_i(T'+t_j)$$

the orders $\{j,i\}$ is better than $\{i,j\}$ at instant T, however $\{i,j\}$ is better than $\{j,i\}$

at instant T'. Therefore, in our problem, we can only find the local optimal criteria of the two operations.

2.3.2 Heuristic Solutions

The effective heuristic procedures are needed to solve difficult problems. Some heuristics attempt to solve a dual problem of the original problem to obtain a near-optimal solution. For example, to solve the previous single-machine problem, the heuristics usually transform the cost functions to the approximate linear functions and the optimal solution associated by the linear cost functions can be achieved by a simple rule [2]. Although the quality of the approximate solution may be affected by a poor choice of the approximate functions, this approach at least acts as the basis for generating the heuristics.

In our problem, however, it is not easy to find such a relaxation. The difficulties of finding the relaxation are:

1. Our objective function is the product of a series of distribution functions each of which has a fixed form.

2. Some distribution functions do not have closed forms and some of them are mathematically intractable.

Objective Function

By considering the *single-machine scheduling problem* which consists of only two jobs, i and j. Let $s' = \{i, j\}$ and $s'' = \{j, i\}$ we have

$$Z(s') - Z(s'') = [f_i(T_0) + f_j(T_0 + t_i)] - [f_j(T_0) + f_i(T_0 + t_j)] \tag{2.9}$$

where $Z(s)$ is the total cost associated by the jobs sequence s.

Now, let the cost function $f(t)$ in the single-machine problem be linear, i.e. for $k = i, j$,

$$f_k(t) = \alpha_k t + \beta_k \tag{2.10}$$

where α_k and β_k are constant. Then we have

$$\begin{aligned} Z(s') - Z(s'') &= [(\alpha_j(T_0 + t_i) + \beta_j) - (\alpha_j T_0 + \beta_j)] \\ &\quad -[(\alpha_i(T_0 + t_j) + \beta_i) - (\alpha_i T_0 + \beta_i)] \\ &= \alpha_j t_i - \alpha_i t_j \end{aligned}$$

Thus,

$$Z(s') \geq Z(s'') \quad \Leftrightarrow \quad \frac{\alpha_j}{t_j} \geq \frac{\alpha_i}{t_i} \tag{2.11}$$

Therefore, when the cost functions are linear and the objective is to minimize $Z(s)$, then the optimal solution of the single-machine problem can be obtained by sequencing the jobs in descending order of the ration α_k/t_k, where k is the job index.

On the other hand, for our problem, suppose we consider the two sequence s' and s'' too, we have

$$\pi(s') - \pi(s'') = P_i(T_0)P_j(T_0 + t_i) - P_j(T_0)P_i(T_0 + t_j) \tag{2.12}$$

When the distribution functions are linear, i.e. the distribution $P_i(t)$ is uniform, we have

$$\begin{aligned} P_i(t) &= \frac{t - a_i}{b_i - a_i} \quad \text{for } a_i \leq t \leq b_i \\ &= \alpha_i t + \beta_i \end{aligned} \tag{2.13}$$

where $\alpha_i = 1/(b_i - a_i)$ and $\beta_i = -a_i/(b_i - a_i)$.

From (2.12), if

$$\begin{aligned} & \pi(s') \geq \pi(s'') \\ \Rightarrow \quad & (\alpha_i T_0 + \beta_i)(\alpha_j(T_0 + t_i) + \beta_j) \geq (\alpha_j T_0 + \beta_j)(\alpha_i(T_0 + t_j) + \beta_i) \end{aligned} \tag{2.14}$$

From the above inequality, we see that the distribution functions lose the linearity after the production in the objective function. So that we cannot obtain any benefits from the linearity of the uniform distribution. Conclusively, the structure (production) of the objective function has added extra difficulty in solving the problem.

Distribution Functions

Mathematically, when $P_i(t) = e^{\alpha_i t + \beta_i}$, when we take logarithm to the objective function as in (2.6), we can transform the objective function as follows,

$$\ln[\pi(s)] = \sum_{i=1}^{n} (\alpha_i T_i(s) + \beta_i) \tag{2.15}$$

which is equivalent to the single-machine problem with linear cost functions. However, we cannot find such a distribution which can be approximately expressed as $e^{\alpha t+\beta}$. Since $P_i(t)$ can only be expressed by various distributions which have fixed forms or even no closed forms, the characters of distribution functions sometimes limit us to investigate the optimality of the solution. For example, normal distribution which is a commonly used distribution function. However, note that the cumulative density function (cdf) of normal is expressed as the integral of the density function which is hard to investigate. Considering another distribution, the exponential, which is a member of the large Exponential family. Let $P_i(t) = (1 - e^{-\lambda_i t})$, the quantity of the objective function associated by a series of operations is

$$\pi(s) = \prod_{i=1}^{n}(1 - e^{-\lambda_i T_i(s)})$$

The computation of above expression is very complicated. As the reasons, the characters of the distribution functions also put us extra difficulties to investigate the problem.

2.4 Literatures Review on Single-Machine Scheduling

Research on minimizing the general non-linear problem has primarily concentrated on techniques to find an optimal solution. The solution methodology used by researchers is either dynamic programming [3]-[6] or branch and bound [7]-[14].

Heuristics procedures for special cases (e.g. minimizing the total tardiness) are considered by Wilkerson and Irwin [15] which utilize an Adjacent Pairwise Interchange (API) methodology incorporating the dominance properties developed by Emmons [16]. The API algorithm chooses a sequence of jobs as a basis, for example $(1, 2, \ldots, n)$, it considers every pair of adjacent jobs and switches them if the total cost is lowered as the result of this switch. This method gives a locally optimal solution. One of the difficulties of the API algorithm is finding a sequence as a starting solution, see [1] and [17]. Fry *et al.* [17] develop a heuristic procedure based on Wilkerson and Irwin's method for the mean tardiness problem. They use three different sequences as the starting basis. Another heuristic procedure, called the modified due date (MDD) algorithm, for mean tardiness is given by Baker and Bertrand [18]. The MDD chooses a job at each iteration based on the smallest due date or completion time of a job whichever is the minimum.

The heuristic methods for the general non-linear problem is very limited. Fisher and Kreiger [19] theoretically analyze a heuristic solution based on the ratio rule of Smith [2]. They consider approximating the sum of non-linear concave profit functions and provide an algorithm which always obtains at least 2/3 of the optimal profit. This bound is only valid for maximizing total profit and does not hold for minimizing total cost which is the objective of the single-machine problem. However, Fisher and Kreiger's algorithm is easily applicable to the total cost problem. Alidaee recently presents several techniques for general non-linear cost functions, see [20] and [21]. In [20] Alidaee proposes the Dynamic Algorithm (DA) through utilizing the differentials of the cost functions. The algorithm is empirically compared with the linearized algorithm by Fisher and Kreiger in [19]. In [21] Alidaee proposes two algorithms. One is the evolution of DA and the other is based on a linear least square approximation of the cost functions. These algorithms are also compared empirically with Fisher and Kreiger's algorithm and the DA.

Although the algorithms suggested in the literatures cannot be used to solve our problem, they provide us some insights in constructing any feasible heuristics. For example, the API algorithm is adopted as a part of our two level heuristic in this dissertation.

Chapter 3

Discussion of Some Special Cases

In last chapter, we have demonstrated that our problem is very similar to the single-machine scheduling problem which is NP-Complete. Therefore, finding an polynomial time algorithm to solve our problem to the optimality is impractical at this stage. However, in this chapter, we will show that under some special problem structures, the optimal scheduling policies of our problems can be achieved. In the first section, we discuss the two operations problem. From the result of this problem, we construct the *Smallest Rate of Probability Increasing Potential First* (PIPF) rule, which is one of the bases of our heuristics. In the second section, the problem with identical distributions is discussed. We establish the *Largest Processing Time First* (LPTF) rule to solve this sort of problems. In addition, the error bound of using LPTF rule to solve the general problem is also discussed. In the last section, the problems with large initial time and special processing times structure are introduced. To solve this type of problems, the *Smallest Variance First* (SVF) rule is developed. Moreover, error bound of SVF rule is also discussed. Lastly, we discuss the error bound of the heuristics.

3.1 Two Operations

We consider a sequence which consists only two operations i and j. Obviously, the possible schedules are

$$s' = \{i, j\},$$

and

$$s'' = \{j, i\}.$$

With reference to Figure 3.1, we can express the overall probabilities of these two sequences as

$$\pi(s') = P_i(T_0)P_j(T_0 + t_i), \tag{3.1}$$

and

$$\pi(s'') = P_j(T_0)P_j(T_0 + t_j). \tag{3.2}$$

Note that $T_0 = D - t_i - t_j$, we have

$$T_0 + t_i = D - t_j,$$

and

$$T_0 + t_j = D - t_i.$$

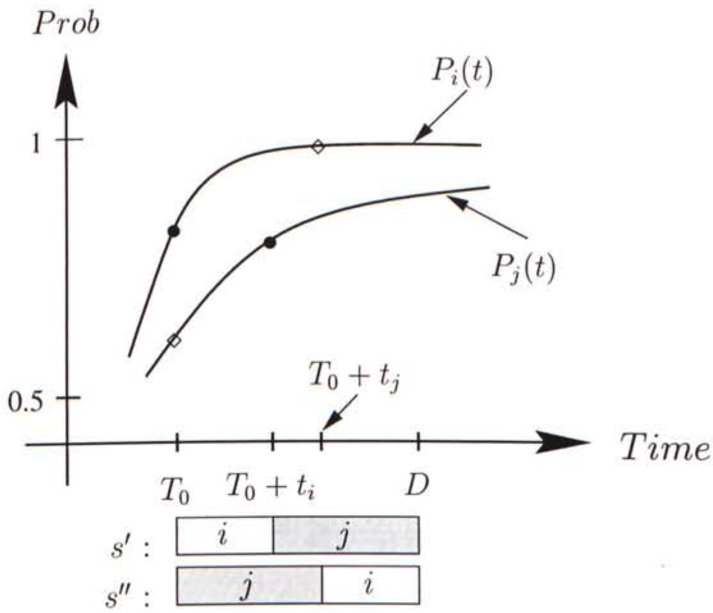

Figure 3.1: Distributions of Two Operations

Thus, we express (3.1) and (3.2), respectively, as

$$\pi(s') = P_i(T_0)P_j(D - t_j), \tag{3.3}$$

and

$$\pi(s'') = P_j(T_0)P_i(D - t_i). \tag{3.4}$$

Note that if i is scheduled at the end of the sequence, the starting time of i should be $D - t_i$, and the appropriate probability $P_i(D - t_i)$ should be the maximum value that $P_i(t)$ can attain in any sequence. Since the probability of i at time T_0 is $P_i(T_0)$, the ratio

$$\frac{P_i(D - t_i)}{P_i(T_0)} \tag{3.5}$$

can be interpreted as the rate of probability increasing potential of i at time T_0. Whereas, $\frac{P_j(D-t_j)}{P_j(T_0)}$ is the rate of probability increasing potential of j at time T_0.

Now, by (3.3) and (3.4), if

$$\pi(s') \geq \pi(s'')$$

then

$$\begin{aligned} & P_i(T_0)P_j(D-t_j) \geq P_j(T_0)P_i(D-t_i) \\ \Rightarrow \quad & \frac{P_j(D-t_j)}{P_j(T_0)} \geq \frac{P_i(D-t_i)}{P_i(T_0)} \end{aligned} \tag{3.6}$$

This directly implies that the operation with a smaller rate of probability increasing potential should be processed first. We summarize above discussions as a proposition as follows.

Proposition 3.1. *The sequence generated by the Smallest Rate of Probability Increasing Potential First (PIPF) rule is optimal.*

Proof. See (3.1) to (3.6). □

3.2 Identical Distributions

For the second special problem structure, we consider a set of operations $\mathcal{N} = \{1, 2, \cdots, n\}$. For any operation $i, j \in \mathcal{N}$, we have

$$P_i(t) = P_j(t) = P(t), \quad \forall t > 0. \tag{3.7}$$

where $P(t)$ is a cumulative density function.

We depict $P(t)$ in Figure 3.2, where the initial time of the sequence is

$$T_0 = D - \sum_{i=1}^{n} t_i.$$

Note that $P(t)$ is non-decreasing function, i.e. for $\Delta \geq 0$,

$$P(t + \Delta) \geq P(t). \tag{3.8}$$

In this case, the optimal sequence can be obtained according to Lemma 3.2 as given

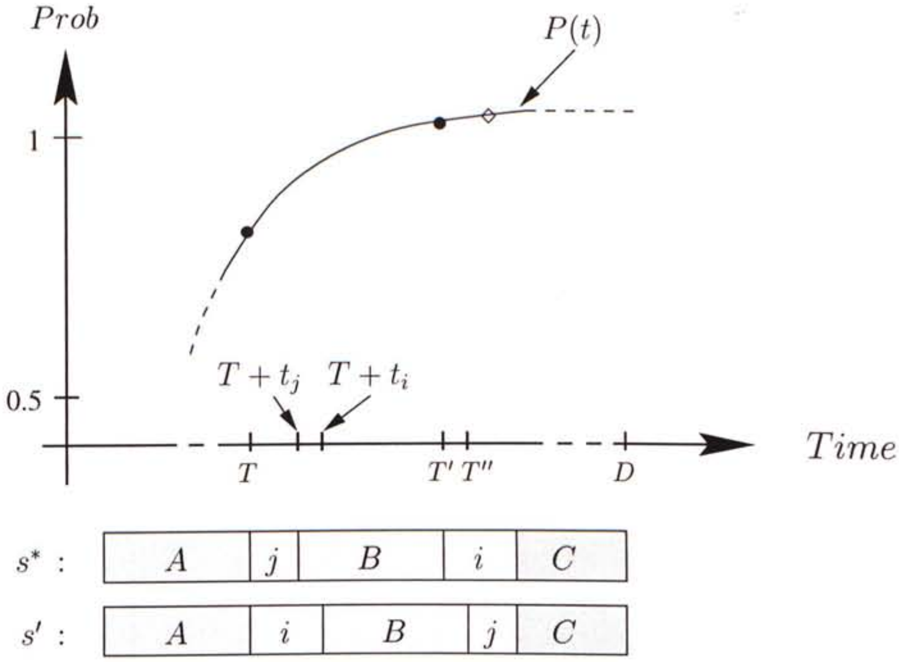

Figure 3.2: Identical Distributions

below.

Lemma 3.2. *When all distributions are identical, the operations in the optimal sequence should follow the Largest Processing Time First rule, i.e. the optimal sequence* *is*

$$s^* = \{1, 2, \cdots, n\}$$

if and only if

$$t_n \geq t_{n-1} \geq \cdots \geq t_1$$

Proof. (By Contradiction)

With reference to Figure 3.2, suppose that there exists an optimal sequence s^* which is not scheduled by the *Largest Processing Time First* (LPTF) rule. Therefore, the optimal sequence is

$$s^* = \{\underbrace{\ldots}_{A}, j, \underbrace{\ldots}_{B}, i, \underbrace{\ldots}_{C}\}$$

when $t_i > t_j$. The overall probability associated with s^* is

$$\begin{aligned} \pi(s^*) &= P_A P_j(T) P_B P_i(T') P_C \\ &= P_A P_B P_C P(T) P(T') \end{aligned}$$

where P_A, P_B and P_C are the probabilities contributed by the subsequences A, B and C respectively, and T and T' are the starting times of j and i respectively. Now, by interchanging the position of i and j in s^*, we obtain another sequence

$$s' = \{\underbrace{\ldots}_{A}, i, \underbrace{\ldots}_{B}, j, \underbrace{\ldots}_{C}\}$$

In this new sequence, the starting time of the subsequence B is $T + t_i$ instead of $T + t_j$ in s^*. Therefore, the probability contributed by the subsequence B now is P'_B which is obviously larger than P_B as $T + t_i > T + t_j$. Now, the appropriate probability of the alternative sequence is

$$\begin{aligned}
\pi(s') &= P_A P_i(T) P'_B P_j(T'') P_C \\
&= P_A P'_B P_C P(T) P(T'') \\
&> P_A P_B P_C P(T) P(T'') \\
&> P_A P_B P_C P(T) P(T') \quad (\because T'' > T') \\
&= \pi(s^*)
\end{aligned}$$

$\Rightarrow$ Contradiction!

$\square$

Therefore, the sequence obtained by LPTF rule is optimal when the distributions are identical.

3.2.1 Error Bound of LPTF – Maximum Distribution Approach

Given a problem with non-identical distributions, the sequence of operations can still be obtained by LPTF rule. However, the overall probability of the sequence must not be as good as the optimal solution and is bounded in certain percentage of the optimal solution. This error bound largely depends on the variation of each distribution which is discussed in the following.

Suppose we are given a set of operations $\mathcal{N}$, where neither the processing times nor the variance is identical. However, we assume that the means of all distributions are equal. Note that the assumption of equal means is applicable to all our discussions except that when the distribution $P_i(t)$ are exponential. Let s^* and $\bar{s}$ be the global optimal sequence and the sequence obtained by LPTF rule, respectively. We first define the following function

$$P_{max}(t) = \max_{i \in \mathcal{N}}\{P_i(t)\} \quad \text{for } T_0 \le t \le D$$

which is just the distribution function with the smallest variance, and the difference between $P_i(t)$ and $P_{max}(t)$ in the time interval $[T_0, D]$ as

$$\delta_i(t) = P_{max}(t) - P_i(t) \quad \text{for } T_0 \le t \le D$$

By the principle of optimality, we have

$$\pi(\bar{s}) \le \pi(s^*) \le \pi(s^*_{max}) \le \pi(\bar{s}_{max})$$

where s^*_{max} and $\bar{s}_{max}$ are the sequences identical to s^* and $\bar{s}$ respectively, but the distribution $P_i(t)$ is replaced by $P_{max}(t)$ in s^*_{max} and $\bar{s}_{max}$. Thus, we have

$$\frac{\pi(\bar{s})}{\pi(\bar{s}_{max})} \le \frac{\pi(\bar{s})}{\pi(s^*)}$$

which is the error bound of the solution associated by the LPTF rule in solving the general problems.

By expressing the error bound in terms of $\delta_i(t)$, we have

$$\frac{\pi(\bar{s})}{\pi(s^*)} \ge \prod_{i=1}^{n}[1 - \frac{\delta_i(T_i(\bar{s}))}{P_{max}(T_i(\bar{s}))}] \tag{3.9}$$

We see from (3.9) that the error associated by LPTF rule in solving the general problem increases as $\delta_i(t)$ increases. For example, consider the 5-operation problem in which the distributions are not identical. Suppose $\delta_i(T_i(\bar{s})) = 0.01$ and $P_{max}(T_i(\bar{s})) = 0.8$ for all i. The error bound of the solution associated by LPTF rule is determined by $(1 - 0.01/0.8)^5 = 0.94$. Obviously, if the distributions are identical such that $\delta_i(T_i(\bar{s})) = 0$ for all i, the solution associated by LPTF is the optimal.

Remarks

Finding a reasonable upper bound of the optimal solution such as $\pi(\bar{s}_{max})$ is very important for testing the performance of any heuristic algorithm in solving the general problems. For example, let S^* is the sequence generated by the heuristics. Since we cannot determine $\pi(s^*)$, the error bound of the heuristic solution can only be determined by $\pi(S^*)/\pi(\bar{s}_{max})$.

3.3 Large Initial Time and Special Processing Times Structure

The last special problem structure is that, for a set of operations $\mathcal{N} = \{1, 2, \ldots, n\}$ which have equal mean of the material arrival distributions, the special structure of the processing times is specified as

$$t_1 \geq t_2 \geq \cdots \geq t_n \tag{3.10}$$

for $\sigma_1 \leq \sigma_2 \leq \cdots \leq \sigma_n$. Note that $t_1 = t_2 = \cdots = t_n$ is a special case of this structure.

For easy exposition, we assume that the material arrivals are normally distributed such that each distribution is distinguished by the appropriate variance. Note that the assumption of distributions does not affect the generality of our discussion about the optimal scheduling policy to this sort of problem structure.

Assume $\sigma_j \leq \sigma_{j+1}$ for $j = 1, \ldots, n-1$, such that we have $t_j \geq t_{j+1}$. Now suppose i and j be any two operations in $\mathcal{N}$ where $\sigma_i \leq \sigma_j$ and $t_i \geq t_j$. The distributions and densities of the material arrivals of i and j are graphically shown in Figure 3.3, where μ is the common mean of the distributions. According to the properties of the normal density functions, there always exists a point $m_{(i,j)} > \mu$ such that the density $p_i(m_{(i,j)}) = p_j(m_{(i,j)})$. Geometrically, we can say that $P_i(t)$ and $P_j(t)$ have the same slope at $m_{(i,j)}$. For the normal distributions, $m_{(i,j)}$ can be determined by the following expression,

$$m_{(i,j)} = \mu + \sigma_i \sigma_j \sqrt{\frac{2\ln(\sigma_j/\sigma_i)}{\sigma_j^2 - \sigma_i^2}} \tag{3.11}$$

Moreover, when σ_j is approaching to σ_i, the limit of $m_{(i,j)}$ is

$$\lim_{\sigma_j \to \sigma_i} m_{(i,j)} = \mu + \sigma_i \tag{3.12}$$

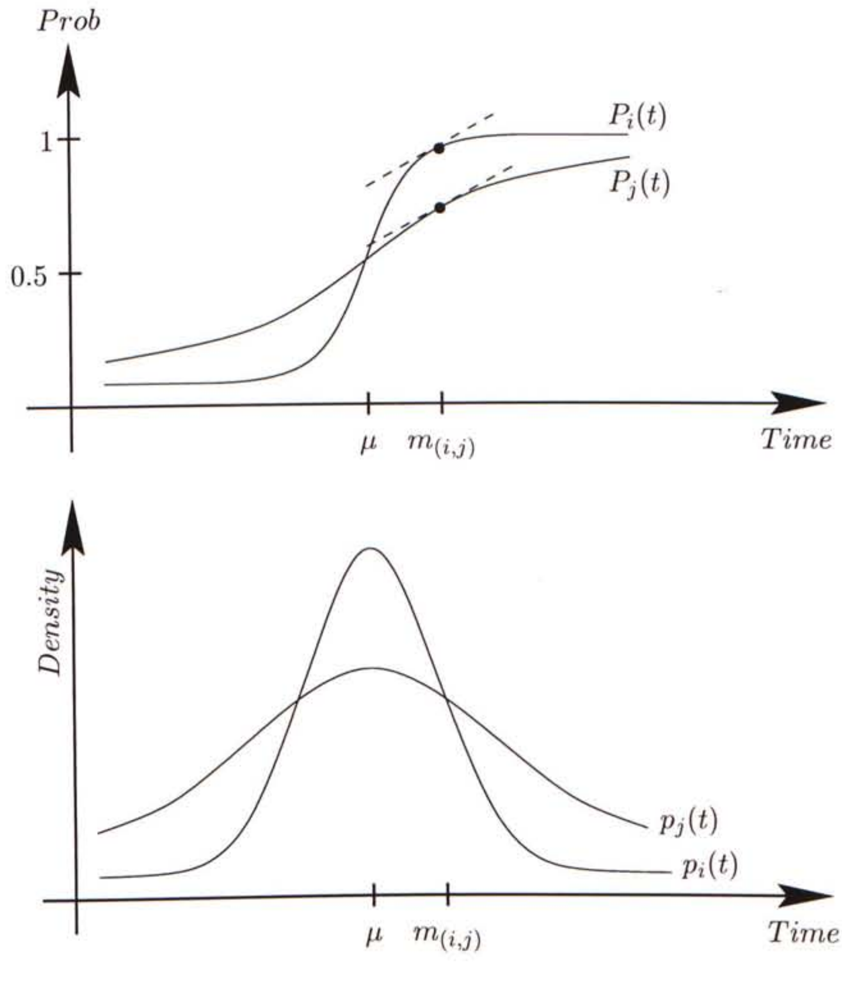

Figure 3.3: Two Operations with Equal Means and Equal Processing Times

The proof are given in Appendix A. Note that

$$p_j(t) \geq p_i(t) \quad \text{for} \quad t \geq m_{(i,j)}. \tag{3.13}$$

Therefore, for $t \geq m_{(i,j)}$, the distribution function which has larger variance such as $P_j(t)$ always increases faster than $P_i(t)$ which has smaller variance. In general, we can find a point $m^* \geq \mu$ such that, for all $i, j \in \mathcal{N}$,

$$m^* = \max_{i,j \in \mathcal{N}} \{m_{(i,j)}\}$$

Now considering above two operations i and j again, suppose the sequences s' is defined as $\{\cdots, i, \cdots, j, \cdots\}$. By interchanging the position of i and j in s', we obtain another sequence $s'' = \{\cdots, j, \cdots, i, \cdots\}$. These sequences are shown in Figure 3.4. Let T be the common starting time of i and j in s' and s'', respectively and, T' and T'' are the starting times of i in s'' and j in s' respectively. We have

$$\begin{aligned}
& P_i(T') - P_i(T) \leq P_j(T'') - P_j(T) \\
\Rightarrow \quad & \frac{P_i(T') - P_i(T)}{P_i(T)} \leq \frac{P_j(T'') - P_j(T)}{P_j(T)} \quad (\because P_i(T) > P_j(T)) \\
\Rightarrow \quad & \frac{P_i(T')}{P_i(T)} \leq \frac{P_j(T'')}{P_j(T)} \qquad\qquad (3.14) \\
& P_i(T)P_j(T'') \geq P_j(T)P_i(T') \\
\Rightarrow \quad & \pi(s') \geq \pi(s'')
\end{aligned}$$

which implies that i should always precedes to j whenever the starting time time $T > m^*$. Thus, if $T_0 \geq m^*$, we can obtain the optimal sequence by *firstly* assigning the operation which has the *smallest variance and largest processing time.* To this end, we can give a Lemma as follows.

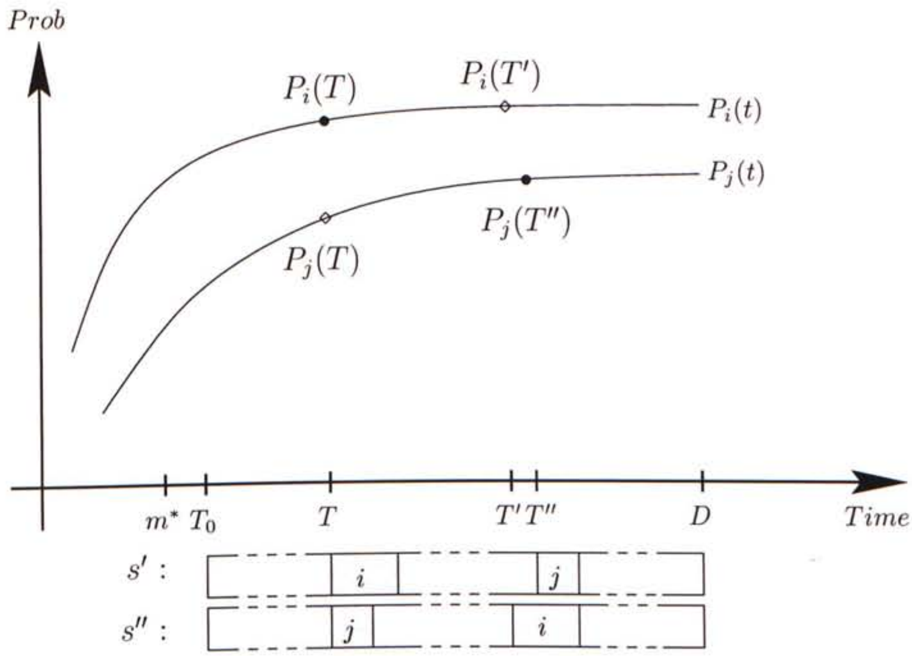

Figure 3.4: Combinations of i and j

Lemma 3.3. *If the initial time of a sequence is not less than m^* and the processing time structure is as specified in (3.10), then the optimal sequence should be obtained by using the Smallest Variance First (SVF) rule.*

i.e. For $T_0 = D - \sum_{i \in \mathcal{N}} t_i \geq m^*$, the optimal sequence is

$$s^* = \{1, 2, \cdots, n\}$$

if and only if

$$\sigma_1^2 \leq \sigma_2^2 \leq \cdots \leq \sigma_n^2.$$

Note that, in our example, normal distributions are used and the order of variance are $\sigma_i^2 < \sigma_{i+1}^2$ for $i = 1$ to $n-1$. From Figure 3.5, we observe that the maximum

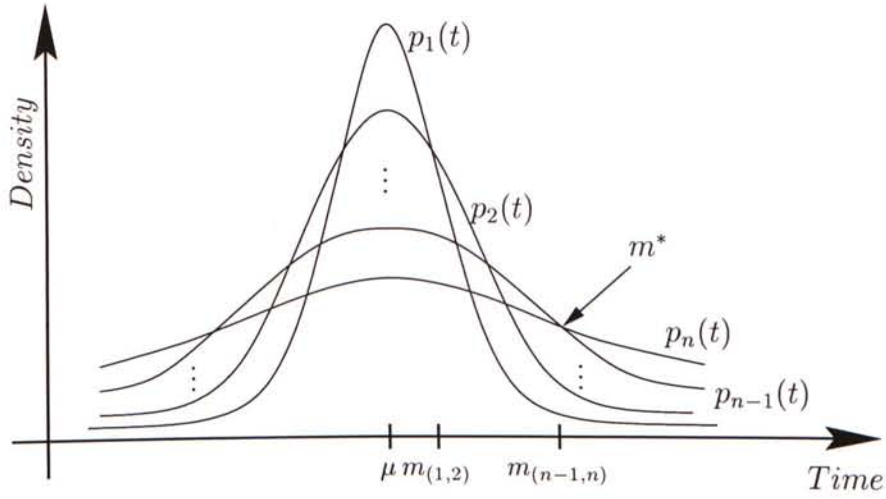

Figure 3.5: Largest Crossing Point m^*

point m^* should be the crossing point of $p_n(t)$ and $p_{n-1}(t)$, i.e. $m^* = m_{(n-1,n)}$. See Appendix A for a formal proof.

3.3.1 Application of SVF to Exponential Distribution

It is very restricted to assume that the initial time of the sequence is not less than m^* when we consider normal or uniform as the material arrival distributions. However, if we consider exponential distribution as the material arrivals, then SVF rule can be applied with less restriction. Now, suppose we are given a set of operations $\mathcal{N}$ in which the processing times satisfy the special processing time structure in (3.10), and the material arrival of the operations are exponentially distributed. Note that the mean and variance of a exponential distribution, $P(t) = (1 - e^{-\lambda t})$, are $1/\lambda$ and $1/\lambda^2$ respectively. If the problem with exponential distributions satisfies the assumption of the special processing times structure, we have

$$t_1 \geq t_2 \geq \cdots \geq t_n \quad \text{and} \quad \frac{1}{\lambda_1} \leq \frac{1}{\lambda_2} \leq \cdots \leq \frac{1}{\lambda_n}$$

With reference to Figure 3.6, we give lemma below.

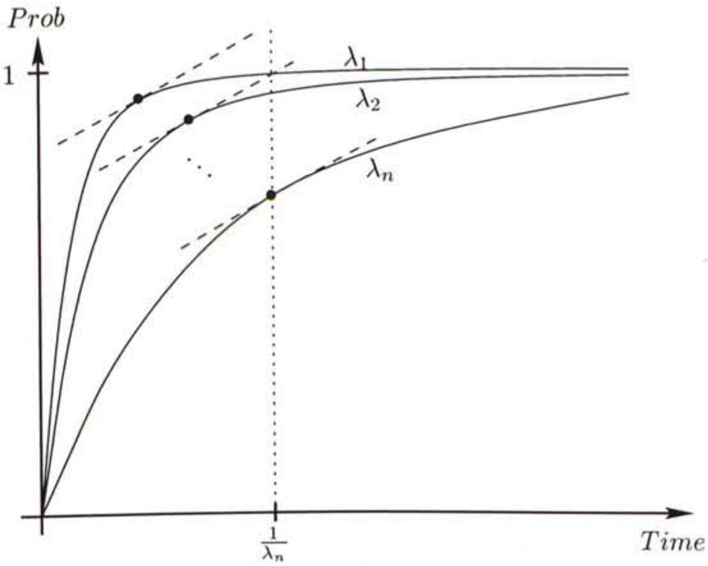

Figure 3.6: Exponential Distributions

Lemma 3.4. *For $t > 1/\lambda_n$, the order of the density functions (or slope) of $P_i(t)$ should be,*

$$p_1(t) \leq p_2(t) \leq \cdots \leq p_n(t)$$

Proof. Let $\frac{1}{\lambda_i} \leq \frac{1}{\lambda_j}$, for $t \geq \frac{1}{\lambda_j}$

$$\begin{aligned}
\frac{p_j(t)}{p_i(t)} &= \frac{\lambda_j\, e^{-\lambda_j t}}{\lambda_i\, e^{-\lambda_i t}} \\
&= \frac{\lambda_j}{\lambda_i} e^{(\lambda_i - \lambda_j)t} \\
\Rightarrow \quad \ln \frac{p_j(t)}{p_i(t)} &= (\lambda_i - \lambda_j)t - \ln \frac{\lambda_i}{\lambda_j} \\
&\geq \frac{\lambda_i}{\lambda_j} - 1 - \ln \frac{\lambda_i}{\lambda_j} \quad (\because\ t \geq \frac{1}{\lambda_j})
\end{aligned}$$

Assume $\lambda_i = \alpha \lambda_j$, where $\alpha \geq 1$.

Let

$$\begin{aligned}
f(\alpha) &= \alpha - 1 - \ln \alpha \\
&\leq \ln \frac{p_j(t)}{p_i(t)}
\end{aligned}$$

Since

$$f'(\alpha) = 1 - \frac{1}{\alpha} \geq 0$$

and, at $\alpha = 1$,

$$f(1) = 1 - 1 - \ln 1 = 0$$

Thus, $f(\alpha) \geq 0$ for $\alpha \geq 1$. Moreover, we can conclude that, for $t \geq \frac{1}{\lambda_j}$,

$$\begin{aligned}
\frac{p_j(t)}{p_i(t)} &\geq 1 \\
\Rightarrow \quad p_j(t) &\geq p_i(t)
\end{aligned}$$

□

Obviously, $1/\lambda_n$ in this scenario can be interpreted as m^* in the SVF rule. Moreover, as we assume the initial time of the sequence being not less than the mean of any arrival distributions, i.e. $T_0 \geq 1/\lambda_i$, $\forall\, i \in \mathcal{N}$, we have $T_0 \geq m^*$. So that SVF rule can be applied to solve the problem of exponential distributions.

3.3.2 Error Bound of SVF – Switching Processing Times Approach

When SVF is applied to solve the problem without assuming any special structure of the processing times, in the case of $T_0 \geq m^*$, the associated solution will not be as good as the optimal solution. In what follows, we propose an approach to determine the error bound of the solution associated with the SVF rule, with respect to the optimal solution. Since we cannot find the optimal solution directly, we can only use some approximate upper bound of the optimal solution to determine the error bound. The question is how to find a reasonable approximation of the optimal solution. Obviously, we can use the *Maximum Distribution Approach* which has been proposed in Sec. 3.2, to determine the upper bound of the optimal solution. However, we can expect that the appropriate upper bound will not be very close to the optimal solution when the deviation of the variance of the distributions are large. Therefore, we suggest an effective approach to find the bound through *switching the processing times*. For instance, let (P_i, t_i) be the operation associated with the arrival distribution $P_i(t)$ and the processing time t_i. For two operations (P_i, t_i) and (P_j, t_j), the appropriate operations obtained by switching the processing times are (P_i, t_j) and (P_j, t_i).

Suppose we are now allowed to freely switch the processing times between the operations. Obviously, the new optimal solution under this relaxation should be at least as good as the original optimal solution. Moreover, we can get the new optimal solution by the following lemma.

Lemma 3.5. *If $T_0 \geq m^*$, and*

$$\sigma_1 \leq \sigma_2 \leq \cdots \leq \sigma_n,$$

then the processing times $t_{[i]}$ should be assigned to operation with the distribution $P_i(t)$ such that the order of $t_{[i]}$ is

$$t_{[1]} \geq t_{[2]} \geq \cdots \geq t_{[n]}.$$

And, according to the SVF rule, the new optimal sequence is

$$\check{S} = \{(P_1, t_{[1]}), (P_2, t_{[2]}), \cdots, (P_n, t_{[n]})\},$$

where $(P_i, t_{[i]})$ is the newly defined operation which is associated with the material arrival $P_i(t)$ and processing time $t_{[i]}$, such that

$$\pi(\check{S}) \geq \pi(s^*)$$

Proof. (By Contradiction)

By considering the two distributions $P_i(t)$ and $P_j(t)$ and the two processing times t_k

and t_l. Suppose $\sigma_i \leq \sigma_j$ and $t_k \leq t_l$. If the new optimal sequence $\check{S}$ is not scheduled according to Lemma 3.5, then there exists one of the following three scenario:

1. $\check{S}_1 = \{\mathcal{A}, (P_i, t_k), \mathcal{B}, (P_j, t_l), \mathcal{C}\}$
2. $\check{S}_2 = \{\mathcal{A}, (P_j, t_k), \mathcal{B}, (P_i, t_l), \mathcal{C}\}$
3. $\check{S}_3 = \{\mathcal{A}, (P_j, t_l), \mathcal{B}, (P_i, t_k), \mathcal{C}\}$

where $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$ are three sub-sequences of the remaining operations. Let

T is the completion time of the sub-sequence $\mathcal{A}$,

T' is the starting time of the sub-sequence $\mathcal{C}$,

K is the probability contributed by both $\mathcal{A}$ and $\mathcal{C}$,

$P_{\mathcal{B}}$ and $P'_{\mathcal{B}}$ are the probability contributed by the sub-sequence $\mathcal{B}$ with the starting times at $(T + t_k)$ and $(T + t_l)$ respectively. Note that $P_{\mathcal{B}} \leq P'_{\mathcal{B}}$ as $t_l \geq t_k$.

case 1 :

$$\begin{aligned}\pi(\check{S}_1) &= K\, P_i(T)\, P_{\mathcal{B}}\, P_j(T' - t_l) \\ &= K\, P_{\mathcal{B}}\, P_i(T)\, P_j(T' - t_l)\end{aligned}$$

Now, by switching the position of the processing times t_k and t_l, we can obtain another sequence

$$S = \{\mathcal{A}, (P_i, t_j), \mathcal{B}, (P_j, t_i), \mathcal{C}\}$$

So that

$$\begin{aligned} \pi(S) &= K\, P'_{\mathcal{B}}\, P_i(T)\, P_j(T' - t_k) \\ &\geq \pi(\check{S}_1) \end{aligned}$$

$$\Rightarrow \quad \text{Contradiction}$$

case 2 : Let S is defined as in Case 1. According to Lemma 3.3, we have

$$\pi(S) \geq \pi(\check{S}_2)$$

$$\Rightarrow \quad \text{Contradiction}$$

Case 3 :

$$\pi(\check{S}_3) = K\, P_j(T)\, P'_{\mathcal{B}}\, P_i(T' - t_k)$$

Now, by switching the position of $P_i(t)$ and $P_j(t)$ in $\check{S}_3$ we have the new sequence S as the one defined in Case 1.

Since

$$\begin{aligned} \frac{\pi(S)}{\pi(\check{S}_3)} &= \frac{P_i(T)\, P_j(T' - t_k)}{P_j(T)\, P_i(T' - t_k)} \\ &= \frac{P_i(T)}{P_i(T' - t_k)} \frac{P_j(T' - t_k)}{P_j(T)} \\ &\geq \frac{P_i(T)}{P_i(T' - t_k)} \frac{P_i(T' - t_k)}{P_i(T)} \\ &= 1 \\ \Rightarrow \quad \pi(S) &\geq \pi(\check{S}_3) \end{aligned}$$

$$\Rightarrow \quad \text{Contradiction}$$

Therefore, the sequence scheduled according to Lemma 3.5 is optimal under the relaxation of processing times. □

If $\breve{s}$ is the sequence obtained by SVF rule, then by the principle of optimality we have

$$\pi(\breve{s}) \leq \pi(s^*) \leq \pi(\breve{S})$$

$$\Rightarrow \quad \frac{\pi(\breve{s})}{\pi(\breve{S})} \leq \frac{\pi(\breve{s})}{\pi(s^*)}$$

Therefore, the error bound of applying SVF rule to solve the problem without assuming the special structure of the processing times can be determined by $\pi(\breve{s})/\pi(\breve{S})$.

3.3.3 Extended Error Bound Analysis

Finding a closer upper bound to the optimal solution is very important for us to evaluate the performance of any heuristic algorithm. In the previous section, we have described an approach to determine the upper bound of the optimal solution under the assumption of $T_0 \geq m^*$. However, this assumption is still very restrictive when the distributions are normal or uniform. Therefore, in this section, we extend the *Switching Processing Times Approach* to the problem in which $\mu < T_0 < m^*$.

The idea of our approach is replacing $P_i(t)$ by an approximate function $H_i(t)$ for all i, such that

1. $H_i(t) \geq P_i(t)$ and

2. the order of $H_i'(t)$, the first derivative of $H_i(t)$, can be expressed as

$$H_i'(t) \leq H_j'(t), \tag{3.15}$$

if $\sigma_i \leq \sigma_j$, for $t > \mu$.

Since $H_i(t) \geq P_i(t)$, the optimal solution associated with $H_i(t)$ should be at least as good as the optimal solution of the original problem. Moreover, by the property of $H_i'(t)$ in (3.15), we can apply Lemma 3.5 to find the upper bound of the new optimal solution associated with $H_i(t)$. Moreover, this upper bound is also used as the upper bound of the original optimal solution. In the following, we give the methodology to construct the appropriate approximate function $H_i(t)$ for $P_i(t)$ in normal and uniform distributions.

Approximation of Normal Distribution

When the distributions are normal, there exists a maximum crossing point m^* which can be determined by

$$m^* = \mu + \sigma_{n-1}\sigma_n\sqrt{\frac{2\ln(\sigma_n/\sigma_{n-1})}{\sigma_n^2 - \sigma_{n-1}^2}},$$

where the order of the variance are $\sigma_1^2 \leq \sigma_2^2 \leq \cdots \leq \sigma_n^2$.

At $t = m^*$, the slope of $P_n(t)$ is

$$l^* = p_n(m^*) = \frac{1}{\sqrt{2\pi}\sigma_n} e^{-(m^*-\mu)^2/2\sigma_n^2}.$$

Let $m_i^* > \mu$ is the time at which the slope of $P_i(t)$ is equal to l^*, i.e.

$$\begin{aligned} p_i(m_i^*) &= l^* \\ \Rightarrow \quad \frac{1}{\sqrt{2\pi}\sigma_i} e^{-(m_i^*-\mu)^2/2\sigma_i^2} &= l^* \\ m_i^* &= \mu + \sqrt{2}\,\sigma_i\,[\ln \frac{1}{\sqrt{2\pi}\sigma_i\, l^*}]^{\frac{1}{2}}. \end{aligned}$$

Note that $m_1^* \leq m_2^* \leq \cdots \leq m_n^*$ as $\sigma_1 \leq \sigma_2 \leq \cdots \leq \sigma_n^*$.

Now, the approximate function for normal distribution can be constructed as following

$$H_i(t) = \begin{cases} P_i(m_i^*) - l^*(m_i^* - t) & , \text{ for } \mu < t < m_i^*; \\ \\ P_i(t) & , \text{ for } t \geq m_i^*. \end{cases} \tag{3.16}$$

Let $h_i(t)$ be the first derivative (or slope) of $H_i(t)$, we obtain

$$h_i(t) = \begin{cases} l^* & , \text{ for } \mu < t < m_i^*; \\ \\ p_i(t) & , \text{ for } t \geq m_i^*. \end{cases} \tag{3.17}$$

Since $h_1(t) \leq h_2(t) \leq \cdots \leq h_n(t)$, for $t > \mu$, Lemma 3.5 is applicable for generating a new optimal sequence S_H such that $\pi(S_H) \geq \pi(s^*)$. Then $\pi(S_H)$ acts as the upper bound of the optimal solution of the original problem.

Approximation of Uniform Distribution

The idea of constructing the approximate function $H_i(t)$ for the uniform distribution $P_i(t)$ is similar to the normal distribution case. Note that the distributions have the common mean μ. Thus, if the order of variance is $\sigma_1^2 \leq \sigma_2^2 \leq \cdots \leq \sigma_n^2$, then the order of the upper bounds of the uniform distributions $U(a_i, b_i)$ should be

$$b_1 \leq b_2 \leq \cdots \leq b_n.$$

To construct the approximate function $H_i(t)$, b_i can be adopted as m_i^* as defined in the normal distribution case. Moreover, the corresponding slope l^* can be determined by the uniform density of the distribution with the largest variance, i.e.

$$l^* = \frac{1}{2(b_n - \mu)}.$$

Now, the approximate function can be expressed as

$$H_i(t) = \begin{cases} 1 - l^*(b_i - t) & , \text{ for } \mu < t < b_i; \\ 1 & , \text{ for } t \geq b_i. \end{cases} \tag{3.18}$$

The first derivative (or slope) of $H_i(t)$ is

$$h_i(t) = \begin{cases} l^* & , \text{ for } \mu < t < b_i; \\ 0 & , \text{ for } t \geq b_i. \end{cases} \tag{3.19}$$

From the definition of $h_i(t)$ we have, for $t > \mu$,

$$h_1(t) \leq h_2(t) \leq \cdots \leq h_n(t).$$

Thus, we have constructed the approximate functions of $P_i(t)$. We can apply Lemma 3.5 to generate the new optimal sequence S_H. And $\pi(S_H)$ acts as the upper bound of the optimal solution of the original problem.

In summary, suppose that the order of the variance are $\sigma_1^2 \leq \sigma_2^2 \leq \cdots \leq \sigma_n^2$, and the order of the processing times are $t_{[1]} \geq t_{[2]} \geq \cdots \geq t_{[n]}$. In Tables 3.1 to 3.3, we list above three approaches for determining the appropriate upper bound for the problem with different distributions.

Exponential Distributions	
Approximate Function, $H_i(t)$	$P_i(t)$, for all t
Optimal Sequence, S_H	$\{(H_1, t_{[1]}), (H_2, t_{[2]}), \cdots, (H_n, t_{[n]})\}$

Table 3.1: Finding Upper Bound for Exponential Distributions

Normal Distributions	
Approximate Function, $H_i(t)$	$\begin{cases} P_i(m_i^*) - l^*(m_i^* - t) & , \text{ for } \mu < t < m_i^* \\ P_i(t) & , \text{ for } t \geq m_i^* \end{cases}$
Crossing Point, m_i^*	$\mu + \sqrt{2}\,\sigma_i\,[\ln \frac{1}{\sqrt{2\pi}\sigma_i\, l^*}]^{\frac{1}{2}}$
Maximum Slope, l^*	$\frac{1}{\sqrt{2\pi}\sigma_n}\, e^{-(m^*-\mu)^2/2\sigma_n^2}$
Optimal Sequence, S_H	$\{(H_1, t_{[1]}), (H_2, t_{[2]}), \cdots, (H_n, t_{[n]})\}$

Table 3.2: Finding Upper Bound for Normal Distributions

Concluding Remarks

According to above results, we have some intuitions regarding scheduling the assembly process of the components with uncertain release times. In some circumstances, the

Uniform Distributions	
Approximate Function, $H_i(t)$	$\begin{cases} 1 - l^*(b_i - t) & , \text{for } \mu < t < b_i \\ 1 & , \text{for } t \geq b_i \end{cases}$
Crossing Point, m_i^*	b_i
Maximum Slope, l^*	$\frac{1}{2(b_n - \mu)}$
Optimal Sequence, S_H	$\{(H_1, t_{[1]}), (H_2, t_{[2]}), \cdots, (H_n, t_{[n]})\}$

Table 3.3: Finding Upper Bound for Uniform Distributions

sequence of operations is important for scheduling the assembly process that are summarized as follows.

1. when the distributions are identical and the difference between the processing times are large, the operation with the smallest processing time should not scheduled first.

2. When the initial time of a sequence is large relative to the mean of the distributions and, the processing times are close, whereas the distributions are variant, we should avoid scheduling the operation associated with the distribution with large variance first.

3. When the distributions are exponential and the processing times are close, we should also avoid scheduling the operation associated with the distribution with large variance first.

Chapter 4

Heuristics to Solve the General Problems

In this chapter, we present a heuristic algorithm to solve general problems. The idea of the heuristics is based on the results of the three special problem structures which have been discussed in Chapter 3. The heuristic algorithm basically consists of two levels. Level 1 combines both PIPF and LPTF rules to achieve a preliminary sequence. The reason is that using either PIPF or LPTF individually to find a preliminary sequence may not be effective. This is because PIPF rule overlooks the importance of the processing times while LPTF rule does not consider the significance of the probability increasing potential. In Level 2, we perform *Adjacent Pairwise Interchanging* (API) to improve the preliminary sequence. Finally, the computational complexity of the heuristic procedures is also discussed. We first discuss the heuristics in detail.

4.1 Level 1 – PIPF and LPTF Rules

In the first level of the heuristics, we apply both the PIPF and LPTF rules to achieve a preliminary sequence. We have discussed PIPF and LPTF rules in Chapter 3. The idea of PIPF is that we should process a operation first, if it has less probability increasing potential of material arrival than others. On the other hand, we have used LPTF rule to solve the problem with identical distributions. The intuition of LPTF is that if we process the operation with larger processing time first, then the other operations should have more time to wait the non-arrival materials such that the overall probability of the operations sequence will be higher. The details of Level 1 are further explained as follows.

Suppose we are given a set of operations $\mathcal{N}$, where t_i and σ_i^2 are the processing time and the variance of the material arrival of operation i respectively. Now, assume that we have scheduled some operations and formed an immediate sequence, which are represented by $J \subset \mathcal{N}$, and J^C is the set of the remaining unscheduled operations. Let T is the total completion time of the J. Note that T is also the starting time of the next operation following J. According to the definition of the initial time of a sequence, T can be determined by

$$T_0 + \sum_{i \in J} \{t_i\} \quad \text{or} \quad D - \sum_{i \in J^C} \{t_i\}.$$

Since, for any operation $i \in J^C$, the maximum probability that i can attain is $P_i(D -$

t_i), the probability increasing potential of i at T is

$$g_i(T) = P_i(D - t_i) - P_i(T), \tag{4.1}$$

and the associated rate of probability increasing potential is

$$\begin{aligned} r_i(T) &= \frac{g_i(T)}{P_i(T)} && (4.2) \\ &= \frac{P_i(D - t_i)}{P_i(T)} - 1. && (4.3) \end{aligned}$$

Suppose that j is another operation in J^C, where

$$\begin{aligned} r_j(t) &\geq r_i(T) \\ \Rightarrow \quad \frac{P_j(D - t_j)}{P_j(T)} &\geq \frac{P_i(D - t_i)}{P_i(T)} \\ P_i(T)P_j(D - t_j) &\geq P_j(T)P_i(D - t_i) \end{aligned}$$

which implies that the operation with larger rate of probability increasing potential should be processed later. So that scheduling a operation with larger $r_i(T)$ latter can utilize larger potential of probability of material arrival of the operation i than that of other operations.

Apart from the rate of probability increasing potential, the length of processing times should be also considered as the criteria of the scheduling. Especially when the rates of probability increasing potential of two operations are equal or very close. If we apply LPTF as another criteria to schedule the operations, then we can utilize larger probability increasing potential during processing the operation with longer processing time. However, when we apply both PIPF and LPTF rules at the same time, there may exist conflicts between these two rules at some instants. To tackle the conflict, we simply compute the product of $r_i(t)$ and $1/t_i$, and then use the numerical value of $r_i(t)/t_i$ to determine the sequencing criteria of the operations. In such way, we can combine both PIPF and LPTF to form the scheduling criteria of the preliminary sequence in Level 1. The criteria is defined as follows:

Select the operation i from J^C such that

$$\frac{r_i(T)}{t_i} = \min_{j \in J^C} \{ \frac{r_j(T)}{t_j} \},$$

then add i to the rear of J.

The algorithm of Level 1 is given below:

Algorithm 1. *(Level 1 - PIPF and LPTF)*

Step 1 : *Initialize* $J^C = \{1, 2, \ldots, n\}$,

$J = \emptyset$,

$T = D - \sum_{i=1}^{n} \{t_i\}$.

Step 2 : *(a) Choose* $i \in J^C$ *such that*

$\frac{r_i(T)}{t_i} = \min_{j \in J^C} \{ \frac{r_j(T)}{t_j} \}$.

If there is a tie choose the least index.

(b) Add i to the rear of J,

Delete i from J^C,

Set $T = T + t_i$.

Step 3 : *If $J^C = \emptyset$ then stop and J is the sequence chosen be the algorithm* *of Level 1. Otherwise go to Step 2.*

4.2 Level 2 – Adjacent Pairwise Interchange

Having obtained the preliminary sequence J by the Algorithm 1, we try to improve J in Level 2 by switching any pair of adjacent operations if the switch brings up the overall probability. This approach is referred as *adjacent pairwise interchange* (API). The idea of API approach comes from the criteria of local optimality of two adjacent operations and, this method gives us a local optimal solution. One of difficulties of applying API algorithm usually is that the sequence should be chosen as the starting base since a good starting base of the API algorithm may lead to the better solution, even a global optimal solution. This is the reason for us to construct Level 1 to generate a preliminary sequence J. The API approach is discussed in details below.

Let $J = \{\ldots, [i], [i+1], \ldots\}$, and the starting time of $[i]$ in J be T, where $[i]$ is the i-th operation in J. The idea of API is that if

$$P_{[i]}(T)P_{[i+1]}(T + t_{[i]}) \leq P_{[i+1]}(T)P_{[i]}(T + t_{[i+1]}) \tag{4.4}$$

we can switch the positions of $[i]$ and $[i+1]$ to achieve a better schedule. Therefore, we can perform API recursively to achieve a local optimal sequence according to the criteria in (4.4). However, a disadvantage of API is that polynomial computational time of API is not guaranteed. Thus, to avoid this, our approach is designed as follows:

We consider n iterations. For each iteration, API is performed to the operations $[i]$ and $[i+1]$ for $i = 1$ to $n-1$ provided that the condition in (4.4) is satisfied.

Thus, there are at most $(n-1)$ interchanges in each iterations. And the maximum number of pairwise interchanges is $n(n-1)$. Now we give the algorithm of Level 2 as below:

Algorithm 2. *(Adjacent Pairwise Interchange)*

Step 1 : Obtain J from Level 1,

Initialize $T = D - \sum_{i=1}^{n}\{t_i\}$,

Set Count=0.

Step 2 : Flag = 1,

For $i = 1$ *to* $n-1$, *do the following:*

If $P_{[i]}(T)P_{[i+1]}(T + t_{[i]}) \leq P_{[i+1]}(T)P_{[i]}(T + t_{[i+1]})$

then temp$=[i+1]$,

$[i+1] = [i]$,

$[i] = temp$,

$T = T + t_{[i]}$,

$Flag = 0$.

where $[i]$ *is the i-th operation in the sequence J.*

Count=Count+1.

Step 3 : If Flag=1 or Count=n then stop and J is the final sequence of our heuristics.

Otherwise go to Step 2.

We have discussed the approaches of finding the upper-bound of optimal solution for various problems in Chapter 3. Note that for different type of problems, we can generate the sequence S_H such that $\pi(S_H) \geq \pi(s^*)$, where s^* is the optimal sequence. To determine the error bound of S^*, the sequence associated by above heuristics, we can compute the ratio $\pi(S^*)/\pi(S_H)$. In such way, we can evaluate the effectiveness of the heuristics. This is further discussed in Chapter 5.

4.3 Computational Complexity

The complexity of the proposed heuristics is polynomial, $O(n^2)$ which is demonstrated as follows. To achieve the preliminary sequence J in Level 1, we have to perform at most $(n-1)$ comparisons to select the minimum $r_j(T)/t_j$ for n iterations. So that the computational complexity of Algorithm 1 is bounded by $O(n^2)$. On the other hand, for Algorithm 2, we have to perform at most $(n-1)$ pairwise interchanging for each iteration. And we have limited the number of iterations by n. Thus, the complexity of Algorithm 2 is also $O(n^2)$. Therefore, the overall computational complexity of our heuristics is $O(n^2)$.

Chapter 5

Experimental Results

The two level heuristic is implemented. In this chapter, we present the experimental results of the heuristic in solving the problems using three types of material arrival distributions, including normal, exponential and uniform. Moreover, the performance of the heuristic in tackling each distribution is evaluated through comparing the results to the upper bound of the appropriate optimal sequences. Before giving the numerical results, we firstly introduce the design of our experiment. And we also describe the approaches used to evaluate the heuristic in solving each type of problems.

5.1 Design of Experiments

Basically, when the experiment is designed, we have to consider two aspects, i.e., (i) generating the representative samples, and (ii) evaluating the performance of the heuristic accurately. For each of these two aspects, we go through the details as follows.

5.1.1 Design of Problem Parameters

In the experiment, for each type of distributions, we consider six different problem sizes including n = 6, 8, 10, 15, 20, 30 and 50. For the problems size n = 6, 8 and 10, we can evaluate our heuristic by comparing the heuristic result with the optimal solution, which is obtained by the enumeration. However, for the problem sizes which are greater than 10 operations, it will be difficult to obtain the global optimal solution. Thus, we only evaluate the heuristic by comparing the results with the approximate upper bound of the optimal solution for the large problems.

As we have assumed that the means are equal for all material arrival distributions, we can fix the mean μ for all sample problems except those using exponential distribution, for which the means are equal to the appropriate standard deviation. To generate a problem, without loss of generality the standard deviation σ_i as well as the processing time t_i of each operation are uniformly generated from a given ranges.

Furthermore, the starting time of the sequence are also uniformly generated from the range $[\mu+\sigma_{min}, \mu+\sigma_{max}]$ for the problems using normal or uniform distributions, where σ_{min} and σ_{max} are the minimum and maximum standard deviations respectively of the distributions in the samples. For exponential distribution, since the starting time is assumed to be greater than the means, T_0 is uniformly generated from the range $[\sigma_{max} + \sigma_{min}, 2\,\sigma_{max}]$. We prefer to generate the starting time in such range because it leads to the *reasonable* overall probability of the sequence. For example, if the initial time is less than $\mu+\sigma_{min}$, then the magnitude of most individual probability $P_i(t)$ will be in the range 0.5 to 0.9. If the number of operations in the problem is 10, and the probabilities in an optimal sequence are, says 0.8 for each, then the overall probability is $(0.8)^{10} = 0.1074$ for which the chance of on-time delivery of the end product is too small and unreasonable. On the other hand, if the initial time are very large, the probability of most material arrivals is close to 1 so that the overall probability is very large. In such a case, the importance of finding a optimal sequence will not be very significant. Therefore, to emphasis the significance of the optimal sequence in a problem, we chose above range to generate the initial time of the sequence. Having generated T_0, the due date D can be determined by $T_0 + \sum_{i=1}^{n} t_i$.

Conclusively, to generate a sample problem, we should give the following information:

1. Type of distributions,
2. Problem size,
3. Mean of distribution functions (for normal and uniform distribution function),
4. Range of standard deviation of the material arrival distributions, and
5. Range of processing time of the operations.

Finally, 50 samples for each type of problems are generated. In the next section we will describe the approach of the evaluation of heuristic.

5.1.2 Evaluation Methods

We evaluate the heuristic by two approaches. The first one is performed through comparing the results of the heuristic with the global optimal solutions. This method can only be used when the problem sizes are small, i.e. $n = 6$, 8 and 10. For $n > 10$, it is impractical to find the global optimal solution by the global search methods. So that other attempts should be used. Note that, in the last chapter, some approaches are suggested to find the upper bound of the optimal solutions for normal, exponential and uniform distribution functions. Thus, according to the upper bounds obtained by those approaches, we can evaluate the results of our heuristic through determine the ratio of our results to the associated upper bounds. However, under some problem structures, when the upper bound is not close to the real optimal solution, the evaluation of the heuristic will not be very accurate. Therefore, we want to find some upper bounds which are close to the real optimal solutions. Note that, for the problem of normal or uniform distribution, the approximate function $H_i(t)$ is close to the original distribution function $P_i(t)$, if the deviation of the variance of distribution functions is narrow. On the other hand, when the deviation of the processing times is narrow, the optimal solution associated by the switching processing times approach will approximate to the original optimal solution since the difference between the switched processing time and the original processing time in not significant. Therefore, a better approximation of the upper bound can be achieved by assigning the processing times and standard deviations with narrow deviations. In the following we will describe how to assign the problem parameters of the sets of general problem as well as the sets of additional problems such that the better approximate upper bounds can be generated.

Exponential

When the type of distributions in a problem is exponential, we can obtain the upper bound of the optimal solution by using Lemma 3.5. With reference to Lemma 3.5, we see that the difference between the upper bound and the optimal solution decreases as the deviation of the processing times of the operations decreases. Therefore, other than the set of general problem parameters, we also introduce another set of parameters in which the range of the processing times is 1/3 of the general parameters. The set of general problem parameters and the additional problem parameters are presented in Table 5.1.

Normal and Uniform

For the problems of normal and uniform distribution functions, we use the *approximate function* $H_i(t)$, as defined in (3.16) and (3.18) respectively to replace the original distributions $P_i(t)$, and apply Lemma 3.5 to find the upper bound of the optimal so-

	General Problems	Additional Problems
problem size n	6, 8, 10, 15, 20, 30, 50	
mean μ_i	σ_i	
range of σ_i	[10, 50]	[10, 50]
range of t_i	[3, 9]	[5, 7]

Table 5.1: Summary of Problems Parameters for Exponential

lution. However, the approximation is suffered from the large deviations of both the processing times and the standard deviation of the distributions. Therefore, for the problems using these two distributions, we introduce two additional sets of problem parameters to improve the accuracy of the upper bounds. In the first set, the range of the standard deviations is 3/40 of the general problems. And, in the second set, the range of the processing times is 1/3 of the general problems. The parameters of the general problem and the two additional problems are summarized in Table 5.2.

5.2 Results Analysis

To test the effectiveness of the proposed heuristic in finding the optimal or near-optimal schedules, we use the heuristic to solve the sets of problems with the parameters presented as in Table 5.1 and 5.2. For the problem sizes n = 6, 8 and 10, the optimal schedules can be found by a global search method. Thus, to evaluate

	General Problems	1st Additional Probs	2nd Additional Probs.
problem size n	6, 8, 10, 15, 20, 30, 50		
mean μ_i	100		
range of σ_i	[10, 50]	[30, 33]	[10, 50]
range of t_i	[3, 9]	[3, 9]	[5, 7]

Table 5.2: Summary of Problems Parameters for Normal and Uniform

the heuristic, neither upper bounds of the optimal solution nor any additional sets of problems are needed. For other problem sizes $n > 10$, we need to solve both the general and additional sets of problems for each type of distributions. The evaluations are given below.

5.2.1 Evaluation for Problems with Small Size

For the problems with the sizes $n = 6$, 8 and 10, we test the effectiveness of the heuristic by comparing the results with the global optimal schedule. A set of 50 problems was generated from each set of problem parameters. For each problem, suppose S^* is the schedule generated by our heuristic and s^* is the optimal schedule obtained by global searching method. The effectiveness of the heuristic Q_g is defined as,

$$Q_g = \frac{\pi(S^*)}{\pi(s^*)}$$

Therefore, if Q_g is equal to 1, the probability associated by S^* equals the probability of the optimal sequence. Table 5.3 presents the number of optimal solution (OPT-NUM), the minimum (MIN), average (AVR) and maximum (MAX) values of Q_g for each set of problems.

Distribution	Exponential			Normal			Uniform		
Prob Size	$n = 6$	$n = 8$	$n = 10$	$n = 6$	$n = 8$	$n = 10$	$n = 6$	$n = 8$	$n = 10$
OPT-NUM	50	50	50	49	48	43	38	39	39
AVR	1.0	1.0	1.0	0.9997	0.9992	0.9967	0.9898	0.9877	0.9814
MIN	1.0	1.0	1.0	0.9869	0.9747	0.9515	0.8593	0.8839	0.7118
MAX	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

Table 5.3: Comparative Evaluation of heuristic with Global Optimal

Considering Table 5.3 our heuristic gives optimal solutions to all problems of exponential distribution. For the problems of normal distribution, the heuristic generated 49 and 48 optimal solutions out of 50 in the problems with sizes 6 and 8 respectively. While 43 optimal solutions are generated for the problems with size 10. And the associated average values of Q_g are all greater than 0.995. Moreover the worst cases are not less than 0.95 for all problem sizes. Although the results for the problems of uniform distribution are not as good as the others, the heuristic still generated 4/5 of the optimal solutions for the problems with sizes 6, 8 and 10. Moreover, the average values of Q_g in uniform problems are not less than 0.98. Therefore, we have presented that, for each distribution, the heuristic generally give satisfactory results in solving the problems with small sizes.

With reference to Table 5.3, we see that the uniform results are the worst, whereas the exponential results are the best. This is because the PIPF rule used in the heuristic can determine the fitness of a job in a sequence accurately only when the rate of increasing of the appropriate distribution is continuous and gentle. For uniform distribution, the rate of increasing is not continuous and sometimes very sharp. So that the results of uniform problems are not as good as others. This can be illustrated by the following example.

Consider a problem of uniform distribution. Let i and j be any two operations out of n operations, where $t_i = t_j = t$, and $P_i(D-t) = P_j(D-t) = 1$. At instant T, where $T > T_0$, let

$$P_i(T) = 0.8,\ P_i(T+t) = 1 \text{ and } P_j(T) = 0.7,\ P_j(T+t) = 0.8$$

Since

$$r_i(T) = \frac{1-0.8}{0.8} = \frac{1}{4}$$

and,

$$r_j(T) = \frac{1-0.7}{0.7} = \frac{3}{7} > r_i(T)$$

According to the criteria of the heuristic, i is scheduled before j. And the associated probability is $0.8 \times 0.8 = 0.64$. However, by switching the order of the i and j, the associated probability is $0.7 \times 1 = 0.7$. Thus, the selection of the heuristic is wrong.

The value of $r_i(t)$ only reflects the ultimate increasing potential of a distribution. However, when we apply $r_i(t)$ as a part of the criteria to select the operations, we assume that the probability is increasing gradually and the heuristic refer $r_i(t)$ as the indicator of the increasing potential throughout the time interval. This is incorrect if the slope of distribution jump at some instants. This is why the results of uniform are not as satisfactory as that of normal and exponential.

5.2.2 Evaluation for Problems with Large Size

For $n > 10$, we test the effectiveness of the heuristic by comparing the results with the upper bounds of the appropriate optimal solutions. A set of 50 problems was generated from each set of the parameters including the additional sets of problems. For each problem, suppose S^* is still the schedule obtained by the heuristic and S_H is the new optimal schedule associated by the approximate function $H_i(t)$ and Theorem 3.5. The effectiveness of the heuristic Q_H is defined as,

$$Q_H = \frac{\pi(S^*)}{\pi(S_H)}$$

Therefore, the closer Q_H to 1, the better result is obtained by the heuristic. Tables 5.4 to 5.6 present the average (AVR), minimum (MIN) and maximum (MAX) values of Q_H in both the general and additional problems with the sizes n = 15, 20, 30 and 50.

Considering Table 5.4 the average values of Q_H for the general problems with the

	General Problems $\sigma_i = [10, 50]$, $t_i = [3, 9]$			Additional Problems $\sigma_i = [10, 50]$, $t_i = [5, 7]$		
n	AVR	MIN	MAX	AVR	MIN	MAX
15	0.9486	0.8611	0.9940	0.9810	0.9223	0.9968
20	0.9514	0.8666	0.9923	0.9754	0.9413	0.9931
30	0.9484	0.8380	0.9903	0.9788	0.9089	0.9961
50	0.9812	0.9508	0.9980	0.9899	0.9596	0.9974

Table 5.4: Comparative Evaluation of heuristic with Upper Bounds – Exponential

sizes $n = 15$, 20 and 30, and $n = 50$ are about 0.95 and 0.98 respectively. Although the average of Q_H for the additional problems are not improved significantly, the minimum values of Q_H in the additional problems are much better than that in the general problem. The reason is that the design of the additional problems have avoided to generate the problems which might lead to extremely inaccurate upper bounds of the optimal solution.

Considering Table 5.5 the average values of Q_H for the general problems increase as the problem sizes increase. However, for the first additional problems in which the range of σ_i are narrow, the average as well as the minimum values of Q_H are very stable for all problem sizes, where the average and minimum values are greater than 0.97 and 0.94 respectively. Moreover, for the sizes $n = 15$, 20 and 30, the heuristic gives a better results in the the 2nd additional problems than the general problems.

	General Problems $\sigma_i = [10, 50]$, $t_i = [3, 9]$			1st Additional Problems $\sigma_i = [30, 33]$, $t_i = [3, 9]$			2nd Additional Problems $\sigma_i = [10, 50]$, $t_i = [5, 7]$		
n	AVR	MIN	MAX	AVR	MIN	MAX	AVR	MIN	MAX
15	0.9006	0.7145	0.9956	0.9746	0.9560	0.9906	0.9456	0.7819	0.9983
20	0.9267	0.5912	0.9998	0.9713	0.9443	0.9886	0.9729	0.8626	0.9994
30	0.9749	0.7506	0.9999	0.9738	0.9423	0.9914	0.9890	0.8890	0.9999
50	0.9874	0.9136	0.9999	0.9759	0.9568	0.9941	0.9803	0.8503	0.9999

Table 5.5: Comparative Evaluation of heuristic with Upper Bounds – Normal

But, for $n = 50$, the results of general problems are better.

Considering Table 5.6 the average values of Q_H are about 0.95 for $n = 15$ and 20, and 0.99 for $n = 30$ and 50. In the 1st additional problems, the average and minimum values of Q_H are greater than 0.97 and 0.94 respectively. Similar to the results in the problems of normal distribution, the result of the 2nd additional problems are not significantly better than that of the general problems.

With reference to the overall results in Table 5.4 to 5.6, we can conclude that the results of the heuristic are generally better for the problems with the largest size 50. The reason is that if the size of a problem is large, then the effect of any non-approximate function $H_i(t)$ on the accuracy of the upper bound of the optimal solution will be diminished. Combining the results of the heuristic in solving the problems with small and large sizes, we have demonstrated that the two level heuristic generate

	General Problems $\sigma_i = [10, 50]$, $t_i = [3, 9]$			1st Additional Problems $\sigma_i = [30, 33]$, $t_i = [3, 9]$			2nd Additional Problems $\sigma_i = [10, 50]$, $t_i = [5, 7]$		
n	AVR	MIN	MAX	AVR	MIN	MAX	AVR	MIN	MAX
15	0.9619	0.7001	1.0	0.9743	0.9457	0.9942	0.9509	0.4547	1.0
20	0.9463	0.5967	1.0	0.9710	0.9444	0.9956	0.9907	0.7978	1.0
30	0.9935	0.8755	1.0	0.9759	0.9573	0.9960	0.9954	0.9103	1.0
50	0.9967	0.9271	1.0	0.9812	0.9582	0.9963	0.9978	0.9566	1.0

Table 5.6: Comparative Evaluation of heuristic with Upper Bounds – Uniform

satisfactory results for the general problems.

Chapter 6

Conclusion

6.1 Summary

This thesis studies an assembly process scheduling problem. We have formulated the problem to the mathematical model. Moreover, an NP-Complete single-machine problem is illustrated and compared with the problem. We have demonstrated their similarity in the mathematical structure.

To solve the problem, we firstly investigate some special cases of the problem. Afterwards, according to the results of these special cases, we construct a heuristic algorithm to solve the general problem. We also develop some approaches to determine the upper-bound of the optimal solutions of the problem using various types of distribution. The upper-bounds are used to evaluate the performance of the heuristics in solving the general problems. Finally, a Java program is written to implement the heuristic algorithm and the experimental results show that our heuristics generate satisfactory solutions to the problems with Exponential, Normal and Uniform Distributions.

6.2 Future Extension

In this study, we consider that the components are needed by a single product only. However, in practice, some components are shared by various products. Moreover the due dates of different products are also varying. Therefore, studying the problem of multiple products with different due dates is definitely worthy of pursuit.

On the other hand, the current objective of our problem is maximizing the overall probability of on-time delivery of the end product. We may also consider another objective of minimizing the deviation of material arrival probabilities in a schedule. We believe that the result obtained by this objective should also give a satisfactory overall probability of on-time delivery of the end product. This is illustrated by $0.7 \times 0.9 = 0.63$ and $0.8 \times 0.8 = 0.64$. We see that the product of the numbers with less deviation is larger than the other, even though the sums of both pairs of numbers are equal. We believe that the new objective can lead to a new objective function which provides us another direction to investigate the problem. So that an effective solution might be obtainable.

Appendix A

Crossing Point of Normal Density Functions

Let σ_i^2 and σ_j^2 are the variance of the two Normal Distributions with the cdf $P_i(\cdot)$ and $P_j(\cdot)$ respectively, where $\sigma_i \leq \sigma_j$, and μ is the common mean of these two distributions. Suppose x is the crossing point of the density functions $p_i(\cdot)$ and $p_j(\cdot)$, where $x > \mu$. We have

$$\begin{aligned}
p_i(x) &= p_j(x) \\
\frac{1}{\sqrt{2\pi}\sigma_i} e^{-(x-\mu)^2/2\sigma_i^2} &= \frac{1}{\sqrt{2\pi}\sigma_j} e^{-(x-\mu)^2/2\sigma_j^2} \\
\frac{\sigma_j}{\sigma_i} &= e^{\frac{(x-\mu)^2}{2}(\frac{\sigma_j^2-\sigma_i^2}{\sigma_j^2\sigma_i^2})} \\
(x-\mu)^2 &= 2\frac{\sigma_j^2\sigma_i^2}{\sigma_j^2-\sigma_i^2}\ln(\frac{\sigma_j}{\sigma_i}) \qquad \text{(A.1)}
\end{aligned}$$

Thus, the value of the crossing point can be obtained by

$$x - \mu = \sqrt{\frac{2\sigma_i^2\sigma_j^2 \ln(\sigma_j/\sigma_i)}{\sigma_j^2 - \sigma_i^2}}, \tag{A.2}$$

$$\text{or} \quad x = \mu + \sqrt{\frac{2\sigma_i^2\sigma_j^2 \ln(\sigma_j/\sigma_i)}{\sigma_j^2 - \sigma_i^2}}. \tag{A.3}$$

Now, let

$$y = (x-\mu)^2. \tag{A.4}$$

By (A.1),

$$\frac{dy}{d\sigma_j} = \frac{4\,{\sigma_i}^2\sigma_j \ln(\frac{\sigma_j}{\sigma_i})}{({\sigma_j}^2 - {\sigma_i}^2)} + 2\,\frac{{\sigma_i}^2\sigma_j}{{\sigma_j}^2 - {\sigma_i}^2} - 4\,\frac{{\sigma_i}^2{\sigma_j}^3 \ln(\frac{\sigma_j}{\sigma_i})}{({\sigma_j}^2 - {\sigma_i}^2)^2}. \tag{A.5}$$

Note that, by (A.4), we have

$$\begin{aligned} \frac{dy}{d\sigma_j} &= \frac{d(x-\mu)^2}{d\sigma_j} \\ &= 2(x-\mu)\frac{dx}{d\sigma_j} \\ \Rightarrow \quad \frac{dx}{d\sigma_j} &= \frac{1}{2(x-\mu)}\frac{dy}{d\sigma_j}. \end{aligned} \tag{A.6}$$

By (A.2) and (A.5), we have

$$\begin{aligned} \frac{dx}{d\sigma_j} &= \frac{{\sigma_i}^2\sigma_j\left({\sigma_j}^2 - {\sigma_i}^2(1 + 2\,\ln(\frac{\sigma_j}{\sigma_i}))\right)}{\sqrt{2{\sigma_i}^2{\sigma_j}^2 \ln(\frac{\sigma_j}{\sigma_i})\,({\sigma_j}^2 - {\sigma_i}^2)^{-1}}\,({\sigma_j}^2 - {\sigma_i}^2)^2} \\ &= Kf(\sigma_j), \end{aligned} \tag{A.7}$$

where

$$K = \frac{\sigma_i^2\sigma_j}{2\sqrt{\sigma_i^2\sigma_j^2 \ln(\frac{\sigma_j}{\sigma_i})\,({\sigma_j}^2 - {\sigma_i}^2)^{-1}}({\sigma_j}^2 - {\sigma_i}^2)^2},$$

and

$$f(\sigma_j) = {\sigma_j}^2 - {\sigma_i}^2(1 + 2\ \ln(\frac{\sigma_j}{\sigma_i}))\,.$$

Since

$$\begin{aligned}\frac{d\,f(\sigma_j)}{d\,\sigma_j} &= 2\sigma_j - 2\frac{\sigma_i^2}{\sigma_j} \\ &= \frac{2}{\sigma_j}(\sigma_j + \sigma_i)(\sigma_j - \sigma_i) \\ &> 0\,,\end{aligned}$$

and when $\sigma_j = \sigma_i$,

$$f(\sigma_i) = 0.$$

We have, for $\sigma_j > \sigma_i$,

$$f(\sigma_j) > 0$$

which further implies, by (A.7),

$$\frac{dx}{d\sigma_j} > 0\,.$$

Thus, we can say that x is monotonically increasing as σ_j increases.

On the other hand, by taking the limit to (A.1)

$$\begin{aligned}
\lim_{\sigma_j \to \sigma_i} (x-\mu)^2 &= \lim_{\sigma_j \to \sigma_i} 2\frac{\sigma_j^2 \sigma_i^2 \ln(\sigma_j/\sigma_i)}{\sigma_j^2 - \sigma_i^2} \\
&= 2\sigma_i^2 \lim_{\sigma_j \to \sigma_i} \frac{\sigma_j^2 \ln(\sigma_j/\sigma_i)}{\sigma_j^2 - \sigma_i^2} \\
&= 2\sigma_i^2 \lim_{\sigma_j \to \sigma_i} \frac{2\sigma_j \ln(\sigma_j/\sigma_i) + \sigma_j^2(\frac{1}{\sigma_j})}{2\sigma_j} \\
&= 2\sigma_i^2 (\frac{0+\sigma_i}{2\sigma_i}) \\
&= \sigma_i^2 \\
\Rightarrow \quad \lim_{\sigma_j \to \sigma_i} x &= \mu + \sigma_i \qquad \text{(A.8)}
\end{aligned}$$

Thus, the limit of the crossing point x, as σ_j approaching to σ_i, is $(\sigma_i + \mu)$.

Therefore, we can conclude that for any two Normal Distributions, which are specified by the common mean μ and variance σ_j^2 and σ_i^2 with the order $\sigma_i \leq \sigma_j$, if x is the crossing point of the the appropriate density functions, where $x > \mu$, then

$$x = \mu + \sqrt{\frac{2\sigma_i^2 \sigma_j^2 \ln(\sigma_j/\sigma_i)}{\sigma_j^2 - \sigma_i^2}}$$

which has the lower bound at $\mu + \sigma_i$. Moreover, if σ_i is fixed then x increases as σ_j increases.

Appendix B

Probaiblity Distributions

B.1 Uniform Distribution

$$\text{Density :} \quad f(x) = \begin{cases} \frac{1}{b-a} & \text{, for } a \le x \le b\,; \\ 0 & \text{, elsewhere.} \end{cases}$$

$$\text{Mean :} \quad \mu = \frac{a+b}{2}$$

$$\text{Variance :} \quad \sigma^2 = \frac{1}{12}(b-a)^2$$

B.2 Exponential Distribution

$$\text{Density}: \quad f(x) = \begin{cases} \frac{1}{\beta} e^{-x/\beta} & \text{, for } x > 0, \beta > 0\,; \\ 0 & \text{, elsewhere.} \end{cases}$$

$$\text{Mean}: \quad \mu = \beta$$

$$\text{Variance}: \quad \sigma^2 = \beta^2$$

B.3 Normal Distribution

$$\text{Density}: \quad f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(x-\mu)^2/2\sigma^2} \quad \text{, for } -\infty < x < \infty\,.$$

$$\text{Mean}: \quad \mu$$

$$\text{Variance}: \quad \sigma^2$$

Bibliography

[1] K. R. Baker (1974) *Introduction to Sequencing and Scheduling.* Wiley, New York.

[2] W. E. Smith (1956) Various optimizers for single-stage production. *Naval Res. Logist. Q.* **3**, 59-66.

[3] J. K. Lenstra, A. H. G. Rinnooy Kan and P. Brucker (1977) Complexity of machine scheduling problems. *Ann. Discrete Math.* **1**, 343-362.

[4] K. R. Baker and L. Sharage (1978) Finding an optimal sequence by dynamic programming: an extension to precedence related tasks. *Opns Res.* **26**, 111-119

[5] M. Held and N. Karp (1962) A dynamic programming approach to sequencing problems. *J. Soc. Industrial and Appl. Math.* **10**, 196-209.

[6] C. N. Potts and L. N. Van Wassenhove (1982) A decomposition algorithm for the single machine total tardiness problem. *Opns Res. Lett.* **1**, 177-181.

[7] U. Bagchi, Y. L. Chang and R. S. Sullivan (1987) Minimizing absolute and squared deviations of completion times with different earliness and tardiness penalties and a common due date. *Naval Res. Logist.* **34**, 739-751.

[8] U. Bagchi, R. S. Sullivan and Y. L. Chang (1987) Minimizing mean squared deviations of completion times about a common due date. *Mgmt Sci.* **33**, 897-906.

[9] P. C. Bagga and K. R. Karla (1980) A note elimination procedure for Townsend's algorithm for solving the single machine quadratic penalty function scheduling

problem. *Mgmt Sci.* **26**, 633-636.

[10] M. L. Fisher (1976) A dual algorithm for the one-machine scheduling problem. *Math. Prog.* **11**, 229-251.

[11] S. K. Gupta and T. Sen (1984) On the single machine scheduling problem with quadratic penalty function of completion times: an improved branching procedure. *Mgmt Sci.* **30**, 644-647.

[12] A. Schild and I. J. Fredman (1961) On scheduling tasks with associated linear loss functions. *Mgmt Sci.* **7**, 280-285.

[13] T. Sen and P. Dileepan (1990) Minimizing a generalized quadratic penalty function of job completion times: an improved branch-and-bound approach. *Engineering Cost and Production Economics.* **18**, 197-202.

[14] W. Townsen (1978) The single machine problem with quadratic penalty function of completion times: a branch-and-bound solution. *Mgmt Sci.* **24**, 530-534.

[15] J. Wilkerson and J. D. Irwin (1971) An improved algorithm for scheduling independent tasks. *AIIE Trans.* **3**, 333-342.

[16] H. Emmons (1969) One machine sequencing to minimize certain function of job tardiness. *Opns Res.* **17**, 701-715.

[17] T. D. Fry, L. Vince, K. Macleod and S. Fernandez (1989) A heuristic solution procedure to minimize $\bar{T}$ on a single machine. *J. Opl Res. Soc.* **40**, 293-297.

[18] K. R. Baker and J. W. M. Bertrand (1982) A dynamic priority rule for scheduling against due-dates. *J. Opns. Mgmt.* **8**, 37-42.

[19] M. L. Fisher and M. A. Kreiger (1984) Analysis of linearization heuristics for single-machine scheduling to minimize profit. *Math. Prog.* **28**, 218-225.

[20] B. Alidaee (1991) Single machine scheduling with nonlinear cost functions. *Comps and Opns Res.* **18**, 317-322.

[21] B. Alidaee (1993) Numerical methods for single machine scheduling with non-linear cost functions to minimize total cost. *J. Opl Res. Soc.* **44**, 125-132.