OPTIMIZATION ALGORITHMS FOR RE-ENTRANT FLOW SHOP SCHEDULING HEURISTIC USING BOTTLENECK APPROACH

SH SALLEH BIN SH AHMAD
ISSTELLA MENGAN ANAK GANI

A project report submitted in completion of
the University Short Term Grant awarded by
Universiti Tun Hussein Onn Malaysia

Faculty of Mechanical and Manufacturing Engineering
Universiti Tun Hussein Onn Malaysia

May 2013

ABSTRACT

This project is an extension to the study of Bottleneck Adjacent Matching (BAM) in 2008, a research area related to re-entrant flow shop job scheduling with the aim of finding the shortest makespan of entire process. Sample operation took place at Cyber Manufacturing Center (CMC), UTHM with process routing of M1-M2-M3-M4-M3-M4. It is observed that the CMC operation exhibits bottleneck characteristic at two points of its operation, M1 and M4-M3-M4. This study exploits the bottleneck characteristic in developing new heuristic that works on minimizing the total process time as well as improving the result at medium P1 Dominance level. Bottleneck dominance level is evaluated initially prior to applying appropriate algorithm to select the right job to be placed at the right position. The heuristic of the new method which is called the Floating P1 Dominance Level (FP1DL) is developed and tested using Macro Programming in Microsoft Excel and a total of 3000 simulations were conducted upon random generated data and the final result of this simulation was compared with the results obtained from actual iteration, NEH and NH. FP1DL heuristic performance was intended for six jobs problem and data measurement was divided into weak, medium, and strong P1 Dominance. The result evaluation shows that FP1DL computation is still reliable to produce schedule makespan but somehow it was unable to serve as an excellent solving method since it only poses an overall accuracy of 51.30%. Optimization of the method traces the problem to originate from P1 Dominance Level evaluation prior to job selection. This matter is validated with a staggering result improvement after a modification of the job selection method.

ABSTRAK

Projek ini merupakan lanjutan daripada kajian *Bottleneck Adjacent Matching (BAM)* pada 2008, iaitu bidang penyelidikan yang berkaitan dengan *re-entrant flow shop scheduling* dengan matlamat untuk mencari tempoh tersingkat bagi keseluruhan proses tersebut. Sampel operasi dalam kajian ini merujuk kepada proses yang terdapat di *Cyber Manufacturing Center* (CMC), UTHM dengan aliran proses M1-M2-M3-M4-M3-M4. Didapati bahawa operasi CMC tersebut menunjukkan ciri-ciri *bottleneck* pada dua pusat operasi tersebut, iaitu M1 dan M4-M3-M4. Kajian ini mengeksploitasi ciri-ciri *bottleneck* ini dalam membentuk heuristic baru yang berfungsi untuk mencari masa tersingkat sekali gus memperbaiki keputusan pada P1 Dominan peringkat sederhana. Kecenderungan dominan ini dinilai terlebih dahulu sebelum mengaplikasikan algoritma yang berpatutan untuk memilih bahan kerja yang betul untuk diletakkan pada kedudukan yang tepat. Heuristic yang terbaru ini dinamakan *Floating P1 Dominance Level* (FP1DL) dibina dan diuji menggunakan *Macro Programming in Microsoft Excel* dan sejumlah 3000 simulasi dijalankan atas kumpulan data yang dihasilkan secara rawak dan keputusan akhir daripada simulasi tersebut dibandingkan dengan keputusan sebenar dan keputusan yang diperolehi menggunakan kaedah NEH dan NH. Heuristik FP1DL diuji pada enam bahan kerja dan data ini dibahagikan kepada P1 Dominan peringkat lemah, sederhana dan tinggi. Keputusan penilaian menunjukkan pengiraan FP1DL dapat menghasilkan jadual kerja tetapi gagal untuk menghasilkan susunan yang effisien memandangkan ianya hanya mampu mencapai ketepatan sebanyak 51.30% secara keseluruhannya. Masalah ini dikenal pasti berpunca daripada penilaian P1 Dominan sebelum pemilihan bahan kerja dan perkara ini disahkan dengan keputusan cemerlang yang dihasilkan setelah pengubah-suaian dilakukan ke atas masalah ini.

CONTENTS

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Most of successful developments in manufacturing sector from all around the world today are determined by efficiency of the management in assembly system and operation line itself. An operation is described as the processing of a job i on the machine M_j by which the processing times are all given in advance (Danneberg *et al.*, 1999), whereas an assembly line is defined as a "dedicated type manufacturing" in which workstations are arranged sequentially and work is performed on products as they move from one station to the other (Khan *et al.*, 2002).The performance evaluation of manufacturing operation is often associated to the problem exhibited in the process structure within in. Most heavy industries are known to apply flow shop system in their assembly system of which it utilizes specialized resources and these jobs will be done through a series of work path in completing the production line. This process is somehow less flexible than a job shop, as it requires a certain way to permutate the job schedule in order to achieve the shortest makespan of entire process. Johnson had proposed optimal solution for two and three stage production in 1954 but slight generalization to this problem had already lead to NP-hard problems (Lenstra *et al.*, 1977). Since then this matter had held attention of many researchers for decades.

One of the subclass of flow shop which is quite eminent in industries is the re-entrant flow shop. It differs from the ordinary flow shop in such way that the job routing may return to any facility within the production line once or more before completing the whole process. Re-entrant flow shop is usually implemented in high-tech industry such as fabrication of semiconductor (El-Khouly *et al.*, 2009), printed circuit board (PCB) (Che *et al.*, 2012), and thin film transistor-liquid crystal display (TFT-LCD) (Choi *et al.*, 2011). The re-entering of the job will cause bottleneck condition to occur in between certain facilities. Since each machine can handle only one job at a time and pre-emption of an operation is not permitted, the next proceeding job on that machine would have to queue for its turn, thus causing idle time that may lags the entire operation.

This project provides an opportunity to explore and investigate an internet-based collaborative design and manufacturing process scheduling which resembles a four machine permutation re-entrant flow shop. The study emphasizes on optimization algorithms for re-entrant flow shop scheduling heuristic using bottleneck approach and this computation is specifically intended for the Cyber Manufacturing Centre (CMC) at Universiti Tun Hussein Onn Malaysia (UTHM).

1.1 Background of Study

In general, the whole research is about incorporating optimization algorithm heuristic into resolving problems related to scheduling *n* jobs at four machines with re-entrant characteristic at machines (*M3-M4-M3-M4*). The research is basically an extension to the previous research related to Absolute Bottleneck Adjacent Matching (ABAM) heuristics done by Bareduan (2009), for which on the same basis, would apply the same algorithms as used in previous study. Permutation of re-entrant flow shop of four machines in the project is similar to Cyber Manufacturing Centre (CMC) at Universiti Tun Hussein Onn Malaysia (UTHM) in such way that design and manufacturing activities would go through six stages of operation with re-entrant characteristic at two of the resources for merely similar process at each places (Bareduan *et al.*, 2008). The resources utilized in the system are the CAD system (P22), CAM system (P23), CNC postprocessor (P24), and CNC machine (P25). The process of generating CNC program for prototyping (T3) and CNC program for customer (T5) are executed on the same CNC postprocessor and similarly the process of prototype machining (T4) and parts machining (T6) are executed on the same CNC machine. The operation flow line of CMC is represented in Petri Net modelling as in **Figure 1.1**

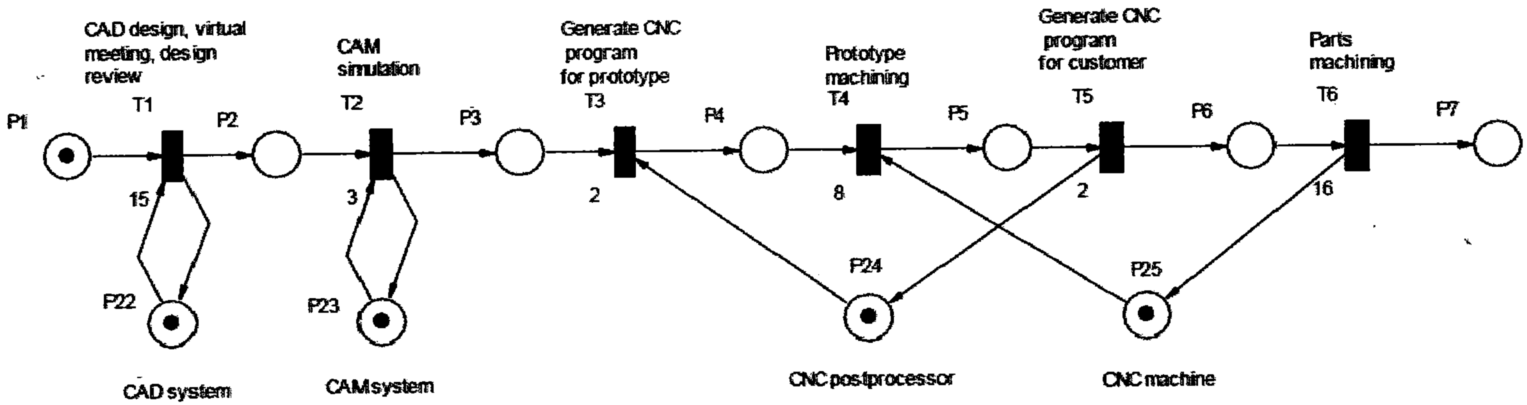

Figure 1.1: Petri Net Model of CMC activities (Bareduan, 2009)

The algorithm was developed based on bottleneck characteristic, defined by the part that posed the longest execution period, which often occurred in re-entrant flow shop problem. Behaviour of bottleneck process is explained via series of mathematical properties and conditions. By introducing a method called P1 bottleneck dominance level measurement, the dominant machine is initially identified prior to determination of appropriate scheduling procedure. The research had resulted in development of four new and effective scheduling algorithm-based heuristic which were called BAM (Bottleneck Adjacent Matching) 1, BAM 2, BAM 3, and BAM 4. All these heuristics were designed to minimize discontinuity between the bottleneck machine of the current job and its subsequent processes. As in experimental result, implementation of each heuristic achieved different performance within specific range of P1 dominance level and number of jobs.

Now that the solving criteria had been established, the next step which is the main focus in the current research is to manipulate these algorithms by combining some of the algorithm subjected to a certain permutation rule derived from the recurring problem in order to achieve better result. Method proposed in this research still maintain P1 bottleneck dominance level measurement as a part of determining which algorithm is to be used in the job selection, in this case, a combination of ABAM 1 and ABAM 2. The rational behind incorporation of these two algorithms is that P1 dominance level may interchange upon each stage of job selection that it is necessary to consider suitable computational method depending on the dominancy level presented in the problem condition. ABAM 1 was actually derived from BAM 3 index which was developed in conjunction with dominancy of CAD process (P1) while ABAM 2 was derived from BAM 1 index that suitable for dominant CNC (P4,P5,P6) processes. In order to identify the shortest makespan to the entire operation, algorithms related to job sequencing with respect to their processing time are applied to the system using certain computerized programming generated with Microsoft Excel.

1.2 Problem Statement

As an assumption, there are n jobs (j_1, j_2, j_3,...,j_{n-1}, j_n) to be processed at four machines (M1, M2, M3, M4) with processing sequence of M1-M2-M3-M4-M3-M4. From the sequence, it is identified that after the jobs are done at M4, they would return to M3 and next to M4 once again before completing the process. Due to preceding job re-entrant at M3, the proceeding job M2 would have to wait for its turn to enter M3. This waiting process would cause idle that slows down the entire operation by increment of total processing time. As time plays an important role as critical constraint, slow operation is an ultimate taboo in manufacturing industry for which it would affect the company in terms of production cost, competitiveness and reliability.

These sorts of issues would raise questions regarding the effective ways to handle the scheduling involving re-entrant flow shop. How to minimize the makespan? How is it possible to reduce the time discontinuity between the bottleneck process and the proceeding job without affecting process time of other subsequent job? How to achieve a good heuristic that able to rearrange the job sequence without having to do numerous, and time-consuming enumeration? If a certain heuristic is developed, can it be applied to different sets of job with different processing time? Would this heuristic be able to solve the problem if the bottleneck point of the process shifted to other point within the system? Can this mathematical heuristic be converted into a programming language that easy to understand and applied in industry?

The problems arising gave an overview to focus point of the research. One knows re-entrant flow shop would induce bottleneck point at the first point of job entrance and another at the re-entrant part of the process. In order to achieve the shortest makespan, especially when it deals with a re-entrant flow-shop, it is more convenient to construct a scheduling heuristic based on bottleneck approach rather than conducting random sequential enumeration of all jobs which appears to be a meticulous process. This process is independent to number of job and can be programmed easily provided a mathematical properties and condition are given along with the solution. To assess the performance of the new heuristic, the experimental result of the current method should be compared with result from enumeration of the same job set obtained from previous research.

1.3 Objectives of Study

i. To produce a new Bottleneck Adjacent Matching (BAM) algorithm-based heuristic complement to evaluation of floating P1 dominance level at the beginning of each job selection, that can be used to improve the performance of bottleneck-based scheduling heuristic for re-entrant flow shop.

ii. To assess the performance of the new optimization algorithms heuristic against the previous heuristic method.

1.4 Scope of Study

i. The project focuses on MIM2M3M4M3M4 flow shop with dominant machine at M1 and M3-M4-M3.

ii. Develop new heuristic from combination of by evaluating the dominancy level after each job assignment.

iii. Develop computer program for evaluation of performance of the new heuristic using Microsoft Excel and Visual Basic for Application.

iv. Compare the performance of new heuristic against the results of previous research that are the Nawaz-Enscore-Ham (NEH) and New Heuristic (NH)

1.5 Significant of Research

This project intends to improve the result at the medium P1 dominance level by introducing a new approach known as floating dominance level measurement. The dominancy level evaluation will determine the job selection by applying either ABAM 1 or ABAM 2 indexes. Programming and simulations are carried out using Microsoft Excel and Visual Basic for Appilcation. A series of testing will be conducted using random data and the results of computation will be compared with NEH and previous NH results.

CHAPTER 2

LITERATURE REVIEW

Scheduling is the process of decision making that crucially applied in area of activities that involve optimization of resources in their process flow. Hence, it is good to assume that there is no specific definition of scheduling as it depends on each area affiliated to the usage of scheduling itself. For example, scheduling in business refers to assigning an appropriate number of workers to the job during each day of work; while in computer science, scheduling is considered as a method to access threads, process or data flows to system resources. As in manufacturing industry, which is also the focus of this project, scheduling involves the process of allocating jobs to processing centres or machines. All these scheduling pose the same characteristic, which is known as the process of deciding how to commit resources from a variety of possible tasks with presence of constraints such as duration, predecessor activities, predecessor relationships, resources availability and due dates.

Scheduling has a major impact on manufacturing industry, as it plays an important role in optimizing production process to achieve maximum efficiency. Basic model of scheduling theory assume that all machines are continuously available for processing throughout the planning horizon (Schmidt, 2000). This assumption is true in some justified cases. However, it is no longer applicable when the machines are no longer applicable for processing due to maintenance requirements, breakdown, or other constraints that can be found in many areas of production. The dynamism of real world resource planning problem induces prescheduled that may result to limited machine availability. An example of occurring problem happens at the operational level of production. As job processing normally fixed in terms of starting and finishing time and machine assignment, the newly released jobs to the shop floor will have to be processed within the remaining free processing intervals since there are already jobs assigned to the time intervals. The problem can be severely affected when the readily processed jobs are required to re-enter the facilities for similar process, by which it will further extend the time horizon as well as extending the idle time of new jobs. This phenomenon can be found in multi-stage manufacturing production such as assembly of semiconductor (Kumar *et al.*, 2006) of which wafers need to revisit the same machines several times to produce several layers that constitute each circuit (Jing *et al.*, 2011), and production of integrated circuit (Pearn *et al.*, 2004). Such cases are known as re-entrant flow shop, and many recent publications on re-entrant flow shop scheduling problems addressed the objective of minimizing the makespan.

2.1 Scheduling criteria

Production scheduling tools had proven to be greatly outperforming the older manual scheduling methods. For instance, the simplest, yet readily available scheduling tool in industry is Microsoft Excel. This tool had virtually helped production scheduler with powerful graphical interfaces which can be used to optimize real-time work loads in various stage of production. Programming of tools triggers pattern recognition that allows the software to automatically create scheduling opportunities which might not be apparent without data review.

Still, in determining the scheduling pattern, production schedulers would have to contribute some manual works in providing computerized program to be incorporated into the software. This manual works refers to the scheduling process of allocating job entries sequence and machine assignment to each job. To produce the right yet optimum schedule depends on the volume of orders, nature of operations and overall complexity of jobs. Scheduling has often implemented with the objective to achieve criteria below.

i. Minimization of completion time

Most scheduling researches are dedicated to achieve this criterion. Numerous variation of permutation of job sequence are studied in determining the average completion time per job.

ii. Maximize utilization of facilities

This criterion is evaluated by determining the percentage of facilities utilization at each stage of job processing. Higher percentage of facilities utilization reflects lesser idle time of job processing, and thus contributes to maximum efficiency of the whole system.

iii. Minimize work-in-process (WIP) inventory

Number of jobs present in the production line is highly related to WIP inventory. In afford of reducing number of WIP, facilities need to be utilized to the maximum. The other way is to set a certain target on numbers of job completion in production. This can often be seen at manufacturing industry whereby scheduler would set total number of jobs to be completed each day in production line. Abundance of WIP is wasteful in terms of production, time and cost; which is a complete taboo in manufacturing industry.

iv. Minimize customer waiting time

Evaluation of average number of late days gives out information on how long does the customers have to wait for their product, and thus related to the effort to reduce idle time experienced by the machines.

There are also important parameters have to be considered in building up an optimum schedule. As the manufacturing process spans over a timely planned horizon, it is obvious that job processing time is an important factor in scheduling.

Note that in this project, the job processing time has been given earlier and machine setup time is not included in the total processing time.

The next important factor is the machine availability. To understand machine availability, one should refer to, the process structure of the manufacturing operation. In a classical shop scheduling problem, it is assumed that job visits any machine at most once, while in this project, the process structure to be deal with involve re-entrant of job to the system more than once. When re-entrant occurs, it is more likely that certain machine would not be available for a certain interval of time. Hence right sequence of job is required so that not much time is spent on waiting predecessor job to complete.

2.2 Flow shop scheduling

Manufacturing operations normally involve certain type of process flow structure and it is either job shop or is flow shop. Job shop is rather easy to be scheduled as it utilizes general purpose resources and it is highly flexible. Flow shop is less flexible from job shop in such way that it uses specialized resources, and the operation works follow fixed path. Each job has exactly one operation for every machine and all jobs would go through all the machines in the same order. Scheduling of flow shop is never an easy task and the problem has held attention of many researchers in last 30 years. Most of the problems concern the objective of minimizing makespan. Makespan refers to the time between the beginning of execution of the first job on the first machine and the completion of execution of the last job on the last machine.

A flow shop scheduling consists of n jobs (j = 1, 2, 3,.., n) on Mi machines (i = 1, 2, 3,..., n). A job would have to undergo consequence processes Pi (i = 1, 2, 3,..., n) at machines Mi. Hence a job can only starts the process on one particular machine only when the predecessor job has finished its process on that machine. Consider an m-machine flow shop with P stages in series, and one or more machines exist at each stage. Each job has to be processed in each of the m stages in the same order which is each job has to be processed in first stage 1, then proceed to stage 2 and so on. Operation times for each job in different stages might be different. Allahverdi *et al.* (2008) had classified flow shop problems as below.

i. Flow shop

There is only one machine at each stage. This is the area of focus in this project.

ii. No- wait flow shop

A succeeding operation begins immediately after the preceding job completed.

iii. Flexible (hybrid) flow shop

More than one machine exists in at least one stage.

iv. Assembly flow shop

Each job consists of m-1 specific operations, each of which has to be performed on a predetermined machine of the first stage, and assembly operation to be performed on the second stage machine.

The first proposed a polynomial time algorithm to solve two-machine had been introduced in 1954 (Johnson, 1954). However, slight generalizations of this problem had lead to NP-hard problems (Lenstra *et al.*, 1977). Several branch and bound algorithms and heuristics have been developed for problem *PF//C*, for example, approximation solutions to the n-job, m-machine sequencing problem where no passing is considered and (Campbell *et al.*, 1970) and heuristic algorithm for the m-machine, n-job flow shop sequencing problem (Nawaz *et al.*, 1983). *PF//C* is used to indicate that permutation flow shop problem is being considered (Graham *et al.*, 1979). In case of considering criterion of minimizing the sum of completion times, only few results are available.

2.3 Re-entrant flow shop

The basic characteristic of re-entrant flow shop is that some jobs visit machines more than once (Wang *et al.*, 1997). The study focuses on two machine re-entrant flow shop scheduling problem with the objective of minimizing makespan. Jobs are composed of six operations done on four machines, in which two of the operations require all the jobs to re-enter the facilities at two machines before completing the whole process. The process flow can be shortened as M1-M2-M3-M4-M3-M4. A few assumptions can be made towards this study:

i. Jobs are available at time zero which is at the beginning of the scheduling horizon, there is given a set of jobs to be scheduled during the horizon.

ii. As the process runs, no job can be pre-empted.

iii. No breakdown is to occur upon each machine.

iv. Setup time of each job has been included at each job processing time. The significant of setup time has been explained in earlier part of this chapter.

Finding an optimal schedule to minimize the makespan in re-entrant flow shop is always a difficult task (Hekmatfar *et al.*, 2011). In fact, it is already known that the sequencing problem in a flow shop scheduling in which *n* jobs have to be processed on *m* machines is proved to be NP-hard. Many recent publications on re-entrant flow shop scheduling problems addressed the objective of minimizing makespan. Dominance properties and lower bounds and heuristics had been incorporated in developing a branch and bound algorithm in two machines problem with job re-entrant route M1-M2-M1-M2 (Choi *et al.*, 2008). Similar problem had been considered with multi family jobs and machine setup times between processing jobs from different families (Yang *et al.*, 2008). The problem is proved to be NP-hard, and researchers have been working on developing heuristic algorithm in order to solve the problem effectively (Yang, *et al.*, 2008). A three extended mixed BIP (binary integer programming) models and six intended effective heuristics had also been proposed in investigating a re-entrant permutation flow shop (Pan *et al.*, 2003).

2.4 Bottleneck-based heuristic

Bottleneck is a phenomenon of which it is frequently occurs in manufacturing industry. This phenomenon is an analogous to observation at the neck of the bottle, of which it poses a queue of an amount of volume from the larger part as it enters the smaller part. The illustration of the analogy is shown in Figure 2.1 (a). Red circle is to indicate predecessor job on process and blue circle indicate current job to be processed. In manufacturing production, bottleneck normally happens for the fact that number of machines is usually limited for processing large amount of jobs. Concentrating at re-entrant flow shop, bottleneck would occur at machines that involve with re-entrant part of the process. As in Figure 2.1 (b), as the predecessor job returns to the facility, the current job would have to queue to enter the machine. In other words, number of job waiting to be processed on the machine had increase, as well as the waiting time of current job. As for now, the objective is to find the optimum job sequence so that the waiting time of current job can be lessen, as well as utilizing machines efficiently and thus minimizing the makespan of entire process

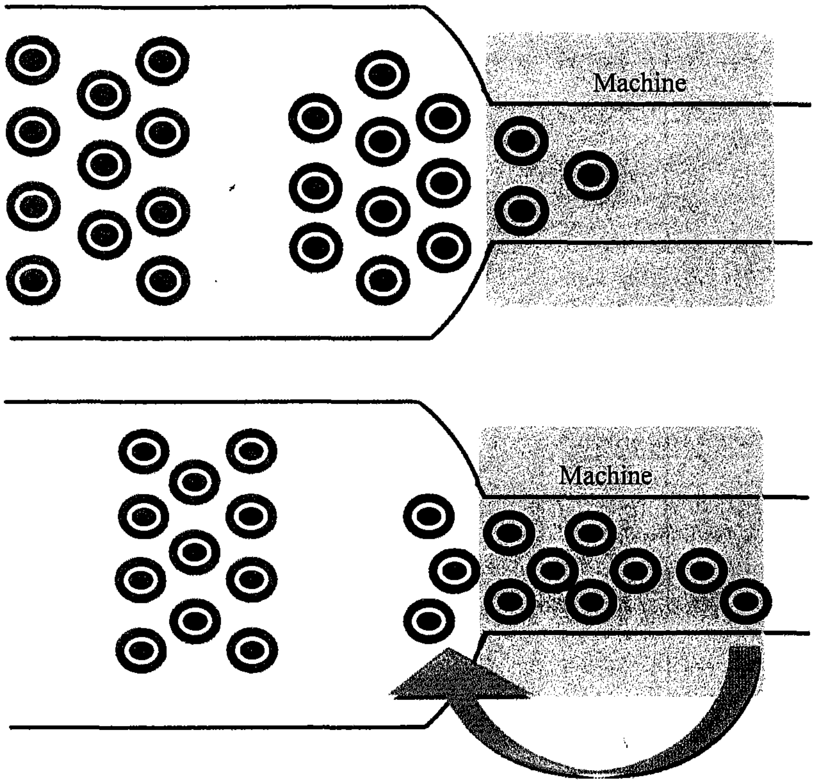

Figure 2.1 (a): Bottleneck analogous to manufacturing process in industry; 2.1(b): Bottleneck analogous to re-entrant flow shop

Bottleneck management is very important but somehow not much progress is reported on bottleneck approach in solving re-entrant flow shop problem. In scheduling literature, bottleneck heuristic approach is known to be among the most successful methods in solving shop scheduling problem. This is because researchers had focused their attention on the origin of the bottleneck instead of simply analysing the whole process which can lead to waste of time. Among the research conducted subjected to bottleneck approach is shifting bottleneck heuristic, where an approximation method is described to solve the minimum makespan of job shop scheduling (Adams *et al.*, 1988). The bottleneck identification and the local re-optimization procedures are based on repeatedly solving certain one-machine scheduling problems. The research has been restudied by means of identifying a new type of precedence relationship that may exist in an OMSP between the predecessor of an operation and the successor of another (Mukherjee *et al.*, 2007). A distributed version of modified shifting bottleneck heuristic has also been made by considering the job shop environment that contains parallel batching machines, machines with sequence-dependent setup times and re-entrant process flows (Mönch *et al.*, 2005).

The study has even brought to an extend of incorporating more sophisticated sub problem solution procedures such as genetic algorithm (Mönch *et al.*, 2007).

Among the few research that emphasized on bottleneck approach on flow shop involved development of a specific version of shifting bottleneck heuristic to solve the re-entrant flow shop sequence problem (Demirkol *et al.*, 2000). Another research that applies the bottleneck-based heuristic is the implementation of Absolute Bottleneck Adjacent Matching in re-entrant flow shop (Bareduan, 2009) which lead to the study of this project. As the problem solving is intended for CMC in University Tun Hussein Onn Malaysia, the process scheduling resembles a four machine permutation re-entrant flow shop with process route of M1-M2-M3-M4-M3-M4. Process at M1 and M4-M3-M4 pose high potential of being dominant machine of which the bottleneck condition is expected to occur. The heuristic performance is evaluated for small, medium, and large job number using random data. For purpose of comparison and testing, the results of FP1DL is compared with NEH heuristic, which is known as the best heuristic for scheduling (Nawaz, *et al.*, 1983) and New Heuristic (NH) method.

CHAPTER 3

METHODOLOGY

In this chapter, the evaluation method and working procedures in completing the project is discussed in detail. This section describes every stage of work progress for completing the new scheduling heuristic for re-entrant flow shop. In other words, it acts as a guideline to develop a new bottleneck based heuristic for 4 machine re-entrant flow shop scheduling as well as performance evaluation of the heuristic. Even though at some parts the method is similar to previous research, a few modifications are applied in certain parts especially in area that related to evaluation of floating dominance of the job available, which is also one of the main focuses of this study. The main goal is to develop an improved version of Bottleneck Based Heuristic for solving problem of M1-M2-M3-M4-M3-M4 re-entrant flow shop with potential dominancy at M1 and M3-M4-M3.

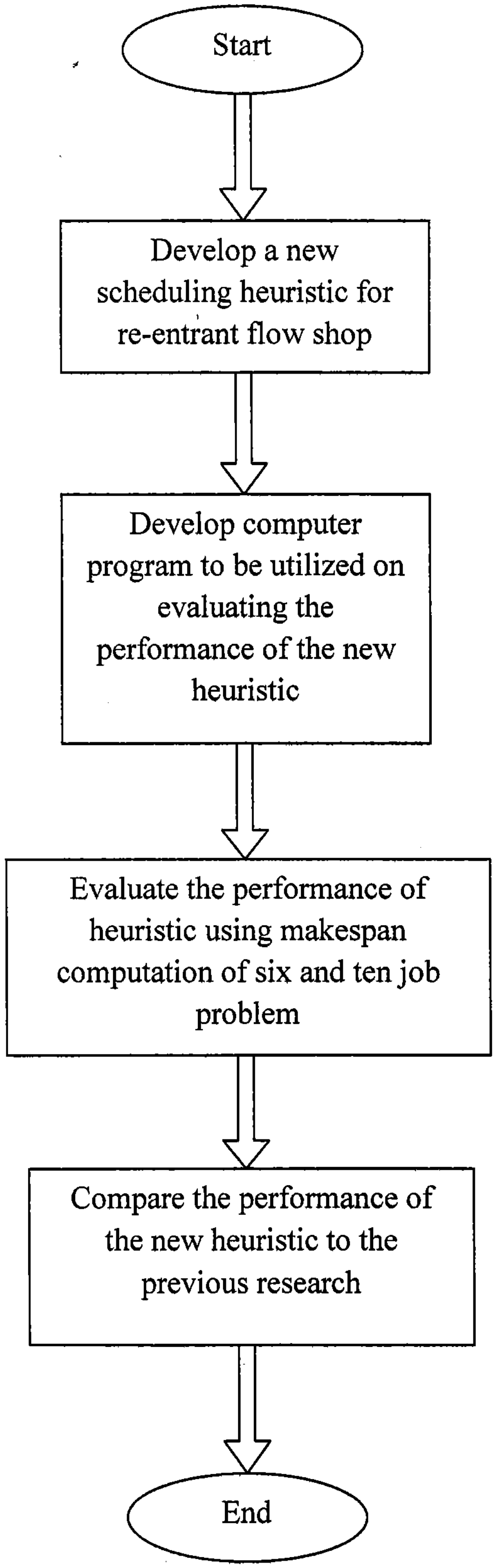

Figure 3.1: Methodology flow chart

The whole process actually starts from finding and gathering information about bottleneck based heuristic. Literature reviews regarding the study of bottleneck based heuristic are as stated in Chapter 2. It is then the knowledge of bottleneck based heuristic being optimized as well as utilized in solving scheduling of *n* job problem, in which in this case is six jobs problem. Re-entrant and permutation nature of the process routing would cause actual bottleneck condition to occur at M1 and combination of M4+M3+M4. As different approach from previous research, a new heuristic is constructed by evaluating floating dominance level with respect to current job available at every stage of job selection. Further explanation regarding floating dominance level evaluation is discussed in Chapter 4. After the heuristic is optimized and a new solution has been constructed, the heuristic is be written in computer programming language to be used as foundation in finding the optimum job schedule.

The next step is to evaluate the performance of the new heuristic using computer program. In this project, Microsoft Excel and Visual Basic for Application are used to develop computer programming of the heuristic. Microsoft Excel is chosen for this study as it provides the best foundation for it is readily equipped with useful add-in, the Visual Basic for Application which is useful for enumeration of random data. On the other hand, Microsoft Excel is readily available in every computer which made it easy to be used for domestic and industrial purposes, and thus does not involve major cost in terms of software licensing. The performance evaluation for this heuristic is tested for makespan computation of six and ten job problems. A similar test would be conducted to for comparison purposes with NEH method and the initial research method, which is the Absolute Bottleneck Adjacent Matching (ABAM). The average makespan ratios are computed from ratio of floating dominance heuristic to the minimum makespan from complete enumeration. The processing time for each process is randomly generated using uniform distribution pattern on realistic data range obtained from previous literature. A total of 3000 simulation are conducted by using randomly data in order to evaluate the heuristic accuracy.

The final result obtained would be compared to previous result of ABAM heuristic research. The goal is to have a better optimizing result at medium P1 dominance level.

3.1 Gathering information about Bottleneck Based Heuristic

It was stated before that this research is an extension to the existing study on bottleneck based re-entrant flow shop. The improved version proposed in this study, which is the bottleneck heuristic made based on evaluation of floating dominance level at each stage of job scheduling is directed to solving the similar problem faced in previous research. Heuristic that utilizes bottleneck approach is known to be among the most successful method, since it focuses on solving scheduling problem that is caused by bottleneck condition at re-entrant section of the operation flow line. A few researchers had involved in the study of bottleneck based heuristic, namely Dermirkol & Uzsoy (2000).

The Petri Net modelling of CMC activities in Figure 1.1 could be use to relate the CMC operation line to this study of bottleneck based heuristic. Note that in studying the CMC re-entrant flow shop, these assumptions are adopted into the system:

i. Pre-emption of operation is not allowed,
ii. Each machine can handle only one job at a time,
iii. All jobs are to be processed on each machine in the same order,
iv. All setup times are included in job processing times,
v. All machines are continuously available, that is no occurrence of machine breakdown in the middle of on going process, and
vi. There is a given set of jobs to be scheduled at the beginning of the scheduling horizon, in which jobs are available at time zero (all jobs that arrive in the middle of horizon would be considered at the beginning of the next scheduling horizon)

3.2 Developing new scheduling heuristic for re-entrant flow shop

With the objective of developing an improved version of existing heuristic, the same algorithm would be applied in optimizing the new heuristic. Besides, after evaluating of floating dominance level the process continues with job selection which requires the same algorithm. The process of floating dominance level evaluation and implementation of ABAM algorithm are described as in the figure 3.2. The blue arrows indicate steps to be done when P1 is dominant while vice versa are denoted by red arrows. Note that green arrows are present after selection of first and last job. This indicates the process of re-evaluating P1 dominance level of remaining jobs. The process is cyclic in nature and would stop after scheduling of last remaining jobs. Elaborations of ABAM 1 and ABAM 2 indexes are shown as below:

i. In case when P1, {P(1,j)+P(2,j)+P(3,j)} is dominant, the scheduling is done by selecting the last job to the first job in operation. Since the last job is selected by choosing the job with smallest {P(2,j)+P(3,j)+P(4,j)+P(5,j)+P(6,j)} value, the consequent job (second last job) would be chosen using ABAM 1 index.

ABAM 1 index:

$$Max\left[\{P(2,j-1)-P(1,j)\},\left\{\sum_{i=2}^{5}P(i,j-1)-\sum_{i=1}^{2}P(i,j)\right\},\left\{\sum_{i=2}^{6}P(i,j-1)-\sum_{i=1}^{3}P(i,j)\right\}\right]$$

ii. In cases when P1 is not dominant, the scheduling is done by selecting job in ascending manner. Job with smallest {P(1,j)+P(2,j)+P(3,j)} value is selected as first job, and scheduling of consequent jobs (second job) would be implementing ABAM 2 index.

ABAM 2 index:

$$Max\left[\{P(3,j)-P(6,j-1)\},\sum_{i=1}^{3}P(i,j)-\sum_{i=2}^{6}P(i,j-1),\sum_{i=2}^{3}P(i,j)-\sum_{i=3}^{6}P(i,j-1)\right]$$

The parameters of both indexes are shown below:

i	=	process sequence of the jobs in CMC
	=	1, 2, 3, 4, 5, 6; each represents M1, M1, M3, M4, M5, M6 respectively
j	=	job number according to scheduling sequence (j=1, 2, 3,..., n)
$P(i,j)$	=	processing time of j_{th} job at i_{th} process sequence

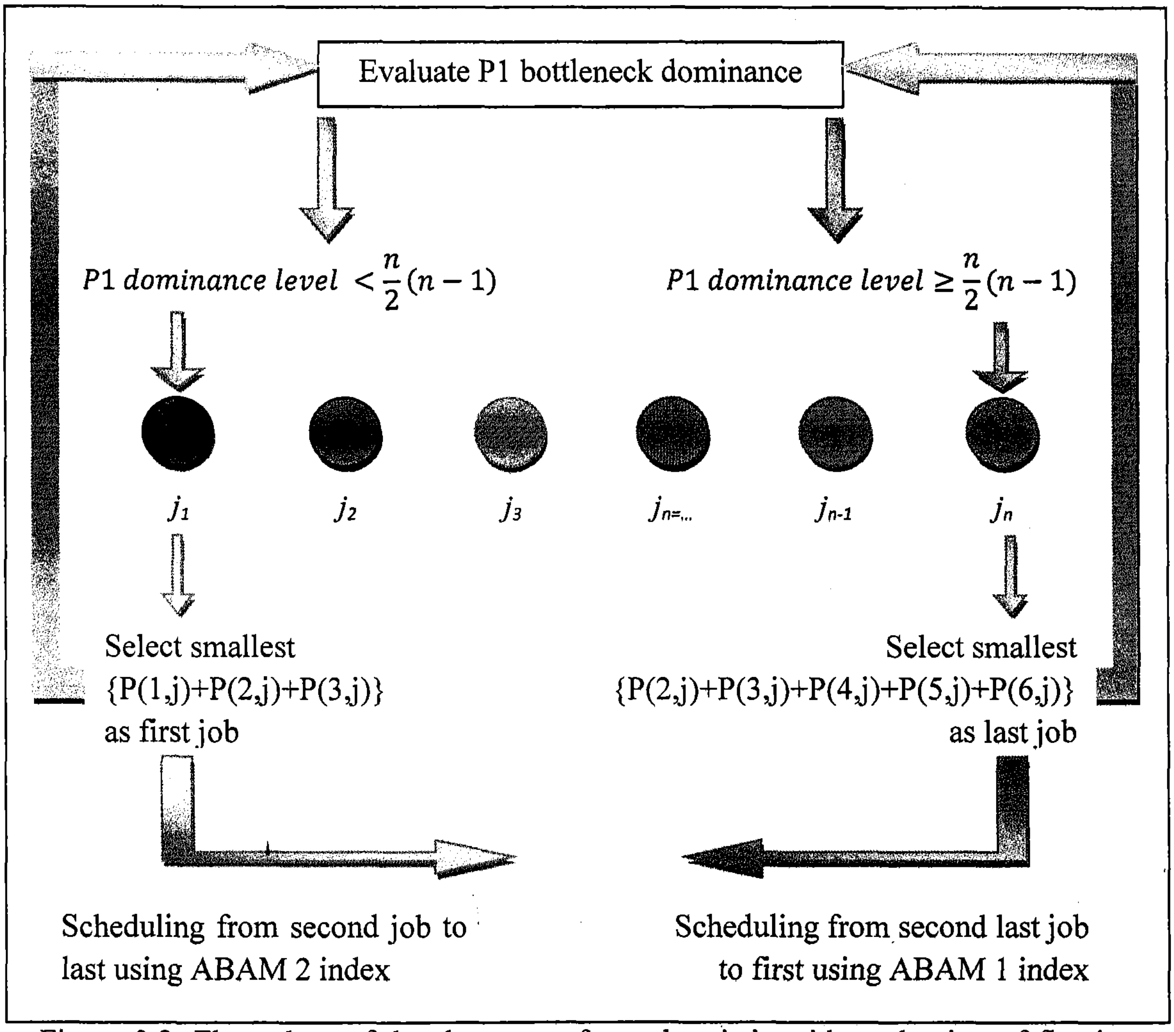

Figure 3.2: Flow chart of development of new heuristic with evaluation of floating dominance level

3.3 Developing computer program

Program input is done at the earlier stage of the process and it is required for researchers to develop a certain computerized programming in order to execute the program's instruction in the central processor and hence, enabling development of heuristic foundation for data simulation. Computer programs are majorly used in simulation of data and performance measurement of tested heuristic. In this research, Microsoft Excel with built-in Visual Basic for Application is used to generate random data for purpose of testing, as well as evaluating the performance of the heuristic.

3.4 Performance evaluation

The heuristic performance evaluation is done upon makespan computation of six job problems by computer simulation. In six jobs problem, the best schedule arrangement comes from complete enumerations that provide minimum makespan value. For comparison purposes, a similar test is conducted using NEH heuristic, which is known as the best heuristic for flow shop scheduling in predicting the job sequence that produces optimum makespan for CMC. The NEH heuristic procedure is obtained from previous research. The results are also compared with previous NH results in order to evaluate the improvement on performance of medium P1 dominance level.

A total of 3000 simulation are conducted to six job problem using the new heuristic (heuristic with floating dominance evaluation). The results from the FP1DL (current method), New Heuristic (NH) and NEH heuristic are compared with the optimum makespan from complete enumeration.

During each simulation, makespan from heuristic and optimum makespan from complete enumeration are recorded. The ratio between the heuristic makespan and the optimum makespan from enumeration is then computed for performance measurement. The percentage of occurrences is also calculated based on equation below:

$$\text{FP1DL makespan ratio} = \frac{FP1DL\ makespan}{Optimum\ Makespan}$$

(Equation 3.1)

$$\text{NEH makespan ratio} = \frac{NEH\ Makespan}{Optimum\ makespan}$$

(Equation 3.2)

$$\text{NH makespan ratio} = \frac{NH\ Makespan}{Optimum\ makespan}$$

(Equation 3.3)

$$\text{Percentage of FP1DL optimum result} = \frac{No.of\ occurences\ FP1DL\ makespan\ ratio=1}{Number\ of\ FP1DL\ simulations} \times 100$$

(Equation 3.4)

$$\text{Percentage of NEH optimum result} = \frac{No.of\ occurences\ NEH\ makespan\ ratio=1}{Number\ of\ NEH\ simulations} \times 100$$

(Equation 3.5)

$$\text{Percentage of NH optimum result} = \frac{No.of\ occurences\ NH\ makespan\ ratio=1}{Number\ of\ NH\ simulations} \times 100$$

(Equation 3.6)

$$\text{Percentage of FP1DL<NEH result} = \frac{Number\ of\ occurences\ FP1DL<NEH}{Number\ of\ FP1DL\ simulations} \times 100$$

(Equation 3.7)

$$\text{Percentage of FP1DL>NEH result} = \frac{Number\ of\ occurences\ FP1DL>NEH}{Number\ of\ FP1DL\ simulations} \times 100$$

(Equation 3.8)

$$\text{Percentage of FP1DL<NH result} = \frac{Number\ of\ occurences\ FP1DL<NH}{Number\ of\ FP1DL\ simulations} \times 100$$

(Equation 3.9)

$$\text{Percentage of FP1DL>NH result} = \frac{Number\ of\ occurences\ FP1DL>NH}{Number\ of\ FP1DL\ simulations} \times 100$$

(Equation 3.10)

REFERENCES

Adams, J., Balas, E., & Zawack, D. (1988). The Shifting Bottleneck Procedure for Job Shop Scheduling. Management Science, 34(3), 391-401.

Bareduan, S. A., & Hasan, S. H. (2008, 19-21 March 2008). Makespan Computation for Cyber Manufacturing Center Using Bottleneck Analysis: A Re-entrant Flow Shop Problem. Paper presented at the International MultiConference of Engineers and Computer Scientists (IMECS), Hong Kong.

Bareduan, S. A. (2009). Bottleneck Adjacent Matching Heuristics for Scheduling a Re-entrant Flow Shop with Dominant Machine Problem. Unpublished PhD Thesis, Universiti Tun Hussein Onn Malaysia.

Campbell, H. G., Dudek, R. A., & Smith, M. L. (1970). A Heuristic Algorithm for the n Job, m Machine Sequencing Problem. Management Science, 16(10).

Che, A., Chabrol, M., Gourgand, M., & Wang, Y. (2012). Scheduling multiple robots in a no-wait re-entrant robotic flowshop. International Journal of Production Economics, 135(1), 199-208.

Choi, S.-W., & Kim, Y.-D. (2008). Minimizing makespan on an -machine re-entrant flowshop. Computers & Operations Research, 35(5), 1684-1696.

Choi, H.-S., Kim, J.-S., & Lee, D.-H. (2011). Real-time scheduling for reentrant hybrid flow shops: A decision tree based mechanism and its application to a TFT-LCD line. Expert Systems with Applications, 38(4), 3514-3521.

Danneberg, D., Tautenhahn, T., & Werner, F. (1999). A comparison of heuristic algorithms for flow shop scheduling problems with setup times and limited batch size. Mathematical and Computer Modelling, 29(9), 101-126.