Batch Scheduling Using Matrix Approach Under Supply Change

by

Chan Choon Hoong

Dissertation submitted in partial fulfillment of

the requirements for the

Bachelor of Engineering (Hons)

(Chemical Engineering)

JANUARY 2010

Universiti Teknologi PETRONAS

Bandar Seri Iskandar

31750 Tronoh

Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Batch Scheduling Using Matrix Approach Under Supply Change

by

CHAN CHOON HOONG

A project dissertation submitted to the
Chemical Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfillment of the requirement for the
BACHELOR OF ENGINEERING (Hons)
(CHEMICAL ENGINEERING)

Approved by,

(AP. Dr. Mohamed Ibrahim Abdul Mutalib)

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
January 2010

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

CHAN CHOON HOONG

ABSTRACT

Batch processing is the predominant mode of production operations for the low volume manufacturing of chemical, polymers and food products. Batch processing can be classified as single product batch process or multiple product batch process. Single product batch process in which single product is produce as compared to multiple product batch process where more than one product is produced using the same batch facility in successive campaigns. More recent works have considered the more complicated cases of processes in which each of the products has its own production sequence and make use of processing units in different combinations. In batch processing, the profitability in economics lies heavily on the scheduling of the production sequence. Scheduling optimization normally aimed at minimizing the makespan (i.e. completion time of the batch process.), leading to overall optimization of the production cost. The complication in scheduling is amplified when the feed change is taken into account. Disruption of feed typically requires a large amount of time to generate an optimal schedule. The proposed approach to address these issues in order to optimize batch production uses matrix to represent the batch recipes which is then solved optimal makespan based on a selected sequence. The arrangement of the matrix rows is according to the best sequence based on the availability or the disruption of supply. The user is then provided with production sequence options based on process requirement and supply.

ACKNOWLEDGEMENT

First and foremost, all the praise to the Almighty God for granting me the opportunity to complete this dissertation for my Final Year Project, which has been proven to be very enriching experience.

It is with pleasure that I express my heartfelt thanks to all who have assisted me either directly or indirectly during and throughout the research works of this project. My great gratitude goes to my capable supervisor, AP. Dr. Mohamed Ibrahim Abdul Mutalib, who helped and guided me to achieve the project objectives successfully. His contribution on this project was invaluable. I would like to acknowledge that without his guidance, all my effort would not have been fruitful.

Secondly, I would like to express my full appreciation towards the FYP coordinator, Dr. Khalik Mohamad Sabil, who has been excellent in coordinating this course.

I would also like to thanks all the internal examiners of Chemical Engineering Department, who have been very corporative in helping and sharing their experience throughout seminars.. Finally, I would also thank my parents for their continuous support, encouragement and understanding along my research.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLE

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND OF STUDY

In processing industry, the selection of technology is solely based on criterion to bring highest profitability to the organization taking consideration of criteria such as being safe and environmentally friendly. Continuous process was a leading choice in the processing industry in the middle of the twentieth century and still remains the processing method for organizations which produce product in bulk. Continuous process is defined as an industry process in which material is produced continuously without interruption.

Due to the competitive and fluctuating economy, it has resultant in instability to the product demand in the petrochemical, chemical and pharmaceutical industries. This factor has influenced and attracted these industries towards batch processes. Batch processing is the manufacturing technique of producing product requiring multiple operations in production. The booming growth of petrochemical, chemical and pharmaceutical industries using the batch processes is mainly due to their flexibility and suitability for the production of relatively small volume and high variety products which offer advantages in the present economic and business situations. Batch processing is the predominant mode of production operations for the low-volume manufacturing of paint, food products, pharmaceuticals and polymers industries.

Batch processes are generally categorized as single product and multiple products. Single product batch process refers to the production of only one type of product in repetition while multiple products batch process offers production of different products using the same batch plant facility. Multiple product batch plant offers the flexibility of producing a variety range of products with the same plant configuration.

Batch processing often require multiple stages such as mixing, reaction and separation. The batch processing scheduling problem deals with the optimal allocation of time and resources to ensure timely and cost effective.

Two main important issues in manufacturing are production planning and scheduling. The proper application of these techniques results in reducing manufacturing cost, satisfying customer demands in a timely manner and overall better planning and control of manufacturing operations (Sule, D.R. 2008). Generally, the aim of scheduling is to minimize the process completion time (i.e. makespan) in order to optimize production and increase the profitability of the organization. Good scheduling leads to the achievement of these goals and, therefore, are integral parts of every professionally run organization. Scheduling consists of planning and prioritizing activities that need to be performed in an orderly sequence of operation (Sule, D.R. 2008).

The general parameters for batch scheduling normally consists of product sequencing i.e. the order of producing different products using the same batch facility, intermediate transfer policies adopted, transfer and setup time between process stages and the overall structure of processing network for the production of specific products (Shafeeq, A. 2008a).

Several scheduling methods have been proposed such as Gantt Chart Method, Mixed Integer Linear Programming (MILP) and Mixed Integer Nonlinear Programming (MINLP) methods. The Gantt Chart Method is a widely used method due to its simplicity. However, this method becomes complex when there are a lot of variables involved. As for MILP and MINLP methods, these applications have been used in industrial level. The complexity of the mathematical approach has caused the approach being programmed and run using computer. The limitation of this method would be increasing computational time when possible number of sequences increases. A more simplified method which is based on matrix representation is developed by Shafeeq, A.

(2008) addresses all the drawbacks in all the conventional scheduling method available in industry.

In real plant, there are a lot of challenges in scheduling. Scheduling generally does not take consideration into uncertainties which normally arises throughout production. These uncertainties which include feed and supply uncertainties, unexpected machine breakdown, cancellation or modification of existing orders, etc. pose a problem to scheduler to resolve the arising problems. The reaction time to address these problems is crucial in as it will defer production which would result in losses if the client's dateline is not met.

Based on different treatment of uncertainty, methods for process scheduling can be classified into two groups: preventive scheduling and reactive scheduling (Li, Z., et al. 2008). Preventive scheduling generates policies before uncertainty occurs by taking account uncertainty in generating schedules that can tolerate parameter variability. As for reactive scheduling, it reschedules after the occurrence of the uncertainty and is implemented based on up-to-date information regarding the state of the system. Reactive scheduling actions are based on various underlying strategies. It can rely on simple techniques or heuristic rules to seek a quick schedule consistency restoration.

1.2 PROBLEM STATEMENT

The scheduling objectives can be in many ways such as minimizing the time required to complete all the tasks (i.e. maksespan), minimizing the number of orders completed after their committed due dates, maximizing customer satisfaction by completing orders in a timely fashion, maximizing plant throughput, maximizing profit or minimizing the production cost. Generally, the main objective of scheduling is to minimize makespan to optimize production.

The scheduling problem is the organization over time of the execution of a set of tasks, taking into account time constraints, supply and demand changes and capability constraints on the resources required for these tasks. Scheduling involves taking decisions regarding the allocation of available capacity of resources (equipment, labor, space, feedstock) (Lopez, P., et al. 2001).

Simple methods commonly used in scheduling may not provide good results, and an analyst who is not aware of other techniques, may not even realize that the solutions may be improved. Another disadvantage on simple scheduling method such as Gantt Chart Method would be the problematic when the case is of a large scale. This method would become extremely tedious when there are a lot of parameters involved.

On the other hand, complex and mathematically methods (e.g. MILP and MINLP) require substantial and extensive knowledge. We would not be able to expect every scheduler to possess such expertise in the industry. Due to the fact that such techniques often go unused in business because of their intricacies and mathematical complications, it is difficult to generate the most efficient result. In order to address such issues, MILP and MINLP methods have been generated on computational approach. Although it manages to solve the complexity of the mathematical approach, the computational time problem occurs when the number of possible sequences increases. The number of possible sequences can be known through permutation. The processing time is directly proportional with the number of possible sequences. This method generated on complete enumeration which prolonged the processing time. Thus, the efficiency of this method is questionable since time is essential whenever an unexpected event happens and it requires a quick rescheduling to address the issue.

It is hard to predict the future with complete certainty. Prices go up and down, and so can the demand. New competitors can come into market, or product can experience obsolescence (Sule, D.R. 2008).

Batch processing scheduling generates all the possible combination sequences based of the number of products and process stages. The makespan for each combination is determined and the best sequence would be employed. The common batch processing scheduling assumed all the parameters associated with it are known. In real plant, there is supply uncertainty which would affect the batch processing causing the inconsistency between the predicted makespan using various scheduling approaches (i.e. Gantt Chart Method, MILP and MINLP) and the actual makespan of the employed sequence. Most of the work in the area of scheduling deals with the deterministic optimization model where all the parameters are considered to be known. In reality, uncertainty is a very important concern that is coupled with the scheduling process since many of the parameters that are associated with scheduling are not known exactly (Li, Z., et al. 2006). One main uncertainty which imposed a challenge to the batch processing scheduling would be the supply disruption. When there is a change in the supply, a scheduler would have to reschedule in order to optimize production based on the changes that take place.

1.3 OBJECTIVES

This project proposes a simple method for batch process scheduling using matrix approach under supply uncertainty. The project objectives are detailed as follows:

1) To identify general supply uncertainty scenario in batch processes industry.
2) To develop a procedure of analyzing and scheduling in batch process under supply change.
3) To develop a computer-based model to perform makespan calculation and screen the optimal batch processing sequence under supply change.
4) To verify and validate results obtained from the developed approach with available case studies.

1.4 SCOPE OF STUDY

The scope of study focuses towards developing a simple method for batch processing scheduling under supply change using the matrix approach. This approach would be able to resolve all the drawbacks or disadvantages specifications on current conventional scheduling methods. The development of the scheduling tool would increase the efficiency of the scheduling methods. The study also covers the method to develop a simple and user-friendly based tool in the sense that would allow scheduler which does not require exceptional operation management technical background to operate the tool.

The next part of the study is to address the changes in the batch production plant. In this study, only feed change scenario would be addressed. The strategy adopted to address the supply change problem would be rescheduling based technique which is known as reactive scheduling. Next, approaches to address the supply change would be seek and would be used as a basis to modify the reactive scheduling.

The extent of the study includes developing a computer based model which is able to perform reactive scheduling when there is an unexpected supply change. This is an improvement to the scheduling tool using matrix approach developed by Shafeeq, A. (2008a). The computer-based model developed using Microsoft Visual C++™ has the functionality to calculate the makespan of all the possible sequences using the matrix approach. The tool utilizes the heuristic approach which enables partial enumeration. This would shorten the processing and computational time which is a favorable advantage when compared to scheduling methods using the computational based MILP and MINLP approaches.

CHAPTER 2
LITERATURE REVIEW

2.1 Introduction

As mentioned in the earlier segment, batch process is the manufacturing technique of producing product requiring multiple operations in production. Generally, batch processing usually involves multiple operations such as mixing, blending, reaction and separation. These types of processes are arranged in order of the stages. The batch processing scheduling problem deals with the optimal allocation of time and resources to ensure timely and cost effective. Shafeeq, A. (2008a) mentioned that productivity of a batch process plant can be increased by reducing the batch process time known as makespan by minimizing the idle time of each process stage through efficient scheduling. Parameters such as process sequencing, transfer policies, etc. are important factors in efficient scheduling.

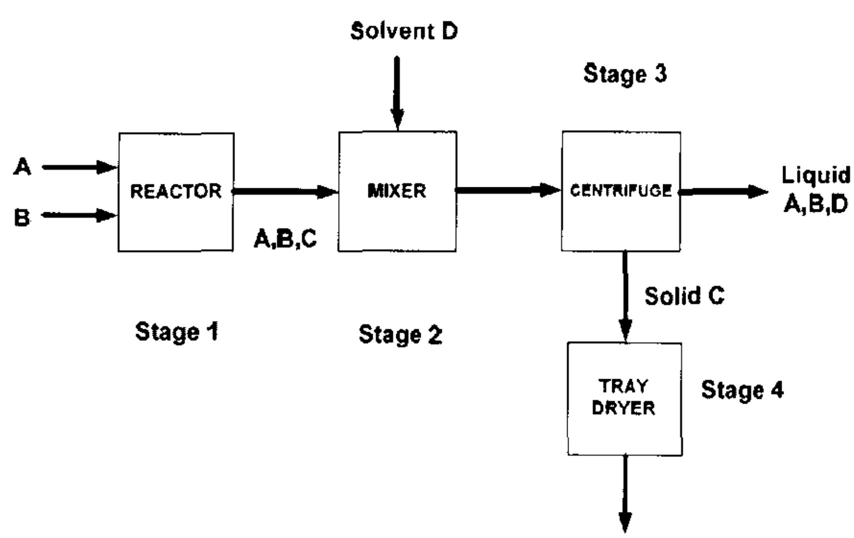

Figure 2.1: Example of batch process

(Source: Biegler, L.T. et al. 1997)

2.2 Batch Process Scheduling

Generally, batch process scheduling is an important aspect is optimizing the plant production. The objectives and benefits of batch process scheduling may vary from product to product. However, some of the objectives and benefits are common among the different products and these were explained by Morrison, S.M. (1996) and Shafeeq, A. (2008a) as follows:

a) The number of equipments involved in the production process can be optimized to minimize the cost and labor requirements. This can be achieved by adopting proper sequence of the products being produced.

b) The excess inventory could result in extra costs incurred in maintaining the quality of the stored products. Effective scheduling can help in managing the inventory level of products according to the raw material supply and to meet the sudden changes in the product demand.

c) The production time should be able to meet the due date set by the customers. The effective scheduling can decide the order of the products that can reduce the overall production time.

d) The most important benefit of scheduling is its flexibility to manage the unforeseen events such as equipment breakdown, rush orders, order changes and raw material availability.

Shafeeq, A. (2008a) stated that a typical batch process scheduling problem depends on the following specifications:

a) Transfer policies for product intermediates between processing stages
b) Processing order of various products
c) Transfer and setup time between different processing stages

Li, Z., et al. (2006) mentioned that scheduling is an important decision-making process where each task requires certain amounts of specified resources called processing time. The scheduling objective can take many forms such as minimizing the time required to complete all the tasks (the makespan), maximizing profit or minimizing production costs. Scheduling decisions to be determined include the optimal sequence of tasks taking place in each unit, the amount of material being processed at each time in each unit and the processing time of each task in each unit.

2.3 Makespan Criteria

Makespan is defined as the time duration of a sequence of jobs and tasks in processing. Minimizing makespan and idle time would maximize product throughput and maximizing profitability. Shafeeq, A. (2008a) stated that the objective of makespan minimization can be achieved by different methods. One of the possible methods is sequencing. Sequencing is defined as the order in which products are manufactured in a batch process.

2.4 Sequencing

Scheduling using sequencing approach provides a basis for assigning the order of products to be produced in a batch process. Heizer, J and Render, B. (2008) mentioned that sequencing specifies the order in which jobs should be executed. In this context, scheduling using sequencing approach is to sequence the products to be produced in order to produce the minimum makespan. Generally, minimizing the makespan would increase the plant throughput and indirectly increase the profitability of the plant.

2.5 Batch Scheduling Methods

Batch process scheduling generates all the possible combination sequences based of the number of products and process stages. The makespan for each combination is determined and the best sequence with the minimum makespan would be employed. There are a lot methods being utilized in the processing industry namely MILP, MINLP and Gantt Chart Methods.

2.5.1 MILP and MINLP Methods

Hong, J., et al. (2001) cited that Mixed Integer Linear Programming (MILP) and Mixed Integer Nonlinear Programming (MINLP) are popular methods applied to batch process industries. The advantages of the MILP and MINLP methods are that, in general, an optimal objective function can be created for a problem. However, due to the complexity of the mathematical approach, the generation of sequences and the makespan of each sequence requires a prolong time to be generated.

In response to these drawbacks, a computational and simplified mathematical approaches utilized by several authors such as Kondili et al. (1993), Pinto and Grossmann (1994), Graells et al. (1994), etc. appeared to have solved all the complexity of this approach. In general, time is an important factor in scheduling. A particular disadvantage appears to surface in this method would be the computational time to generate the sequence and the makespan time which is less efficient as compared to the matrix approach method that would be explained in the later section.

2.5.2 Gantt Chart Method

Gantt Chart is a very simple and widely used graphical representation for viewing scheduling. A horizontal segment of length that is proportional to the operation is associated with each task. In the chart, each horizontal line corresponds to a resource, which makes it possible to view its periods of operation or idleness as well as the sequence of operations using it and the scheduling duration.

The following example of batch scheduling using Gantt Chart method is adopted from Shafeeq, A., et al. (2008b).

Table 2.1: Processing time of 4 products in 3 stages

Products	Processing Time (h)		
	S1	S2	S3
A	5	8	6
B	9	3	2
C	4	5	3
D	4	5	2

(Source: Shafeeq, A., et al. (2008b))

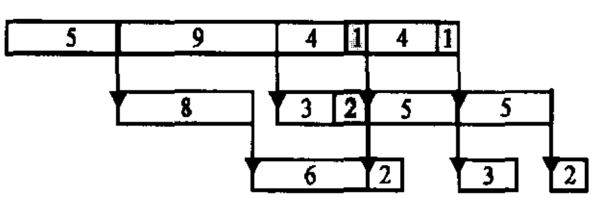

Figure 2.2: Gantt chart for four products in three stages in the sequence of A, B, C and D

(Source: Shafeeq, A., et al. (2008b))

In this example, the makespan calculation is performed for four products (i.e. A,B, C and D). Table 2.1 shows a processing time for three products in the sequence of A, followed by B and lastly C. There are many paths to calculate the makespan of this sequence and one of them is by taking the sum of AS_1, AS_2, AS_3, BS_3, idle time between BS_3 and CS_3, CS_3, idle time between C_3 and D_3, and D_3. The calculated makespan for the specified production sequence is 31 hours.

Although Gantt Chart method is well known for its simplicity approach in generating the makespan for all the possible sequence. However, this approach becomes extremely tedious when a large scale of possible sequences arises (Shafeeq, A. 2008a). The complexity of this approach becomes more apparent when more parameters are involved.

2.5.3 Matrix Approach Method

Shafeeq, A. (2008a) and Shafeeq, A., et al. (2008b) proposed a matrix approach which formulates and simplifies calculation to determine the makespan for specified batch production sequence. Shafeeq, A., et al. (2008b) cited that the ability to quickly calculate the makespan of specific sequence enables the matrix approach to be used to calculate the makespan for all possible production sequences derived from given batch process recipes. The inclusion of heuristic method enables the matrix approach proposed by Shafeeq, A., et al. (2008b) allows the scheduling done based on partial enumeration. This method improves computational and shorten the processing time as compared to MILP and MINLP computational.

The following are guidelines using matrix approach developed for the same example as in section 2.3.2:

Step 1: The product recipes arranged as shown in figure below where the sequencing is A, B, C and D.

	1	2	3
1	AS_1	AS_2	AS_3
2	BS_1	BS_2	BS_3
3	CS_1	CS_2	CS_3
4	DS_1	DS_2	DS_3

Figure 2.3: Matrix arrangement of four products in three stages in the sequence of A, B, C and D

(Source: Shafeeq, A., et al. (2008b))

Step 2: Common path to calculate makespan is through selecting of first element in the first row, the entire elements in the second column and the third element in the bottom most row of the matrix as shown in the figure below.

	1	2	3
1	AS_1	AS_2	AS_3
2	BS_1	BS_2	BS_3
3	CS_1	CS_2	CS_3
4	DS_1	DS_2	DS_3

Figure 2.4: Common path calculation of four products in three stages in the sequence of A, B, C and D

(Source: Shafeeq, A., et al. (2008b))

Step 3: Parameters such as the idle time exists between stages and the waiting time for intermediate products must be included in the makespan. The calculation of slack variable is made based on the value of the matrix elements located diagonally between the first two rows as shown in figure below. The calculation of slack variables is a series of formulas.

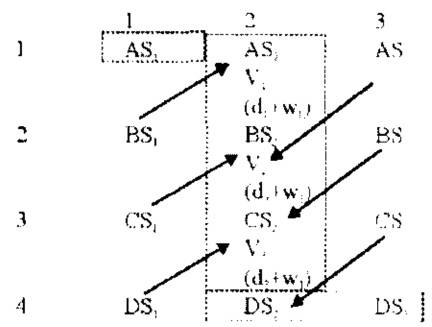

Figure 2.5: Calculation of slack variables

(Source: Shafeeq, A., et al. (2008b))

Step 4: The makespan for the batch process is calculated using the formula;

Makespan = $AS_1 + AS_2 + BS_2 + CS_2 + DS_2 + DS_3 + V_1 + V_2 + V_3$

The matrix approach proves to be a major step towards scheduling at higher efficiency. However, the tool developed by Shafeeq, A. (2008a) and Shafeeq, A., et al. (2008b) assumes that there is no uncertainty. (Li, Z., et al. 2006) mentioned that uncertainties are part of the relations that needed to be addressed in the real plant. This project will only focus on the feed changes which would be included to the existing functionality of the tool developed by Shafeeq, A. (2008a) and Shafeeq, A., et al. (2008b).

2.5.4 Heuristics and Metaheuristics

Generally, heuristics technique is a method which generates a solution that is hoped to be close to the best possible answer which is also known as optimal solutions in swift manner. Despite the fact that it does not always gurantee optimal solution, the present heuristics methods could produce reasonably optimal solutions for large size problems within shorter time period compared to the mathematical programming. In addition, the method is also known to be more stable (Shafeeq, A. 2008a).

The heuristics methods utilized simple iterative search technique to find the optimal solutions. The iterative search technique continues until there is no improved solution to the assigned values to the variable in the initial step.

Lately, there is a lot of modifications and improvements have been made to the conventional heuristics methods. The improved heuristics methods are better known as metaheuristic techniques. The following is a list of metaheuristics techniques:

a) Simulated Annealing

   Simulated Annealing is a class of metaheuristics algorithm for finding the global optimum solution in huge search space. In batch process scheduling problem which focuses in finding the minimum makespan, the Simulated Annealing starts with an initial solution i.e. a production sequence with some makespan value, followed by comparison with the makespan of the second possible production sequence until the search space having all the possible solutions is analyzed (Shafeeq, A. 2008a).

b) Tabu Search

   In most of the heuristic approach, the main limitation would be its failure to locate the global optima as it always trapped within the local optima (i.e. the iterative search stops as soon as the near optimum solution is found). Tabu

Search overcomes these limitations by maintaining a tabu list containing the solutions which have already been searched for optimal solution (Shafeeq, A. 2008a). The search continues until there is no more optimal solution within the solution search space (Tra. N.T.L. 2000, Edgar T.F. et. al. 2001)

2.6 Transfer Policies

Transfer policies for product intermediates between processing stages is considered to be an important aspect in batch scheduling. The option of transfer policy of batch process relies on the nature of the product namely the physical and chemical aspects. The following are general adopted transfer policies discussed by Shafeeq, A. (2008a).

2.6.1 Zero Wait (ZW)

In ZW transfer policy, the product intermediate is transferred immediately to the next stage upon completing its process due to its nature that requires immediate transfer (Biegler, L. et al. 1997; Ryu, J.H. et al. 2007). This transfer policy could result to a situation where the production of the next batch could be delayed even after the availability of the first process stage. The ZW policy results in the longest makespan as compared to other transfer policies. Referring to Figure 2.6, the processing of product B in stage 3 would only start when the processing of product A is done.

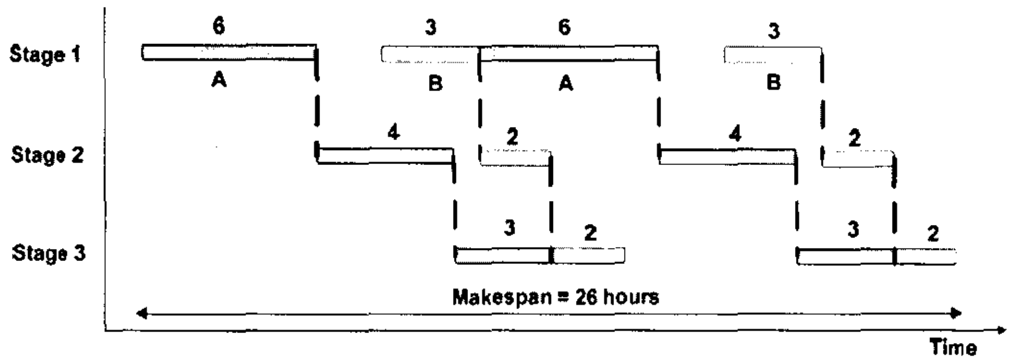

Figure 2.6: Gantt Chart for ZW transfer policy

(Source: Biegler, L.T. et al. 1997)

2.6.2 No Intermediate Storage (NIS) / Unlimited Wait (UW)

In NIS/UW transfer policy, the nature of the intermediates is such that they could stay in their current stage until the availability of the next processing stage (Ku, H.M. 1992; Biegler et al. 1997). Figure 2.7 shows that the processing of product B in stage 1 could start immediately after the processing of product A and does not have to depend on the timing availability of stage 2.

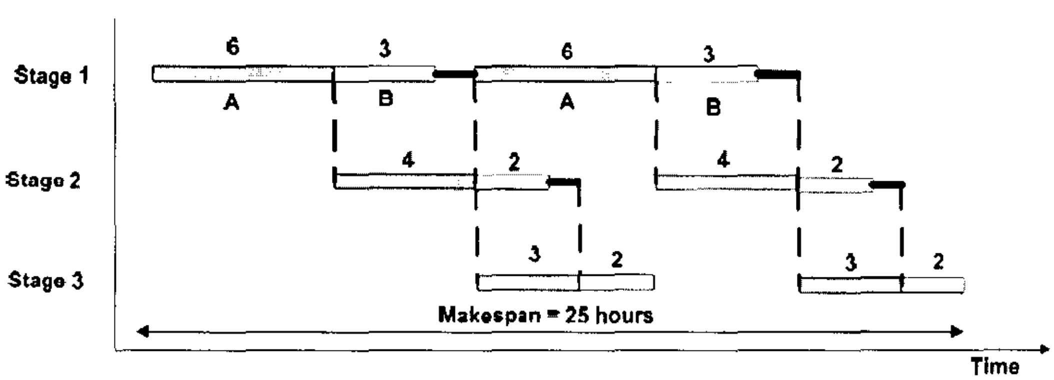

Figure 2.7: Gantt Chart for NIS/UW transfer policy

(Source: Biegler, L.T. et al. 1997)

2.6.3 Unlimited Intermediate Storage (UIS) / Unlimited Wait (UW)

For UIS / UW, intermediate storage tanks are used to store product intermediates until the availability of the next process stage. This situation is adopted when the product intermediate is not allowed to reside temporarily in the same process stage due to either process makespan restriction or product intermediate undergoing further reaction if remains within the process stage (Biegler, L.T. et al. 1997).

Due to the unlimited number of storages made available, there is no restriction at all on the temporary storage of product intermediates as shown in the figure below (Kim, M. et al. 1996). From Figure 2.8, the temporary storages are available for storing the product B intermediate after stage 1 and stage 2. Due to the physical and chemical nature of the product intermediates, the residence time in a temporary storage must be monitored carefully to meet the quality standards of the final product (Ha, J.K. et al. 2000).

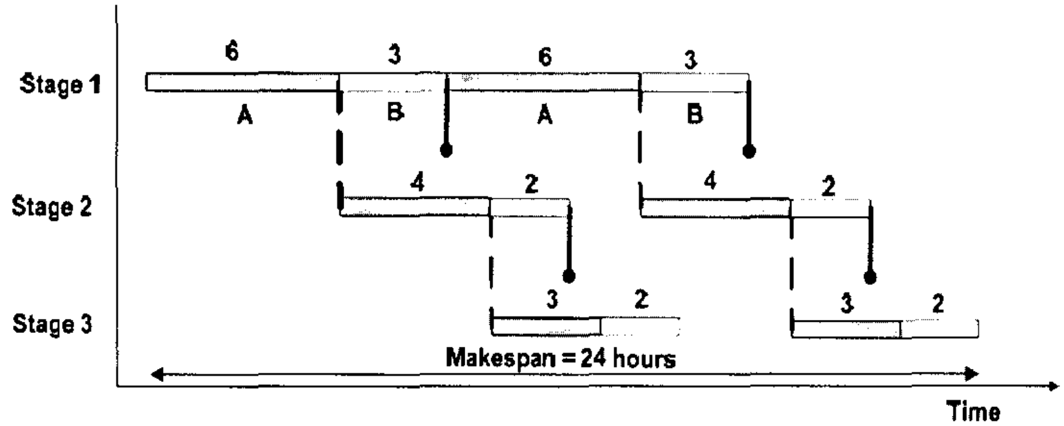

Figure 2.8: Gantt Chart for UIS/UW transfer policy

(Source: Biegler, L.T. et al. 1997)

2.6.4 Finite Intermediate Storage (FIS) / Unlimited Wait (UW)

For FIS / UW, the process has almost similar specifications as the UIS / UW except that the number of storages is limited. The storage system in FIS transfer policy results in better economics compared to UIS / UW as it tend to reduce the capital cost while optimizing the storage utilization (Kim, M. et al. 1996). Kim, M. et al. (1996) suggested the application of FIS / UW by combining the specifications of FIS / UW and NIS / UW. This combination would be deemed possible if storage is available where the product intermediate is transferred into it until availability of a temporary storage. From Figure 2.9, a temporary storage is used to store the product B intermediate after its processing in stage 1 has been completed as there is unavailability of stage 2 which is still processing intermediate product A.

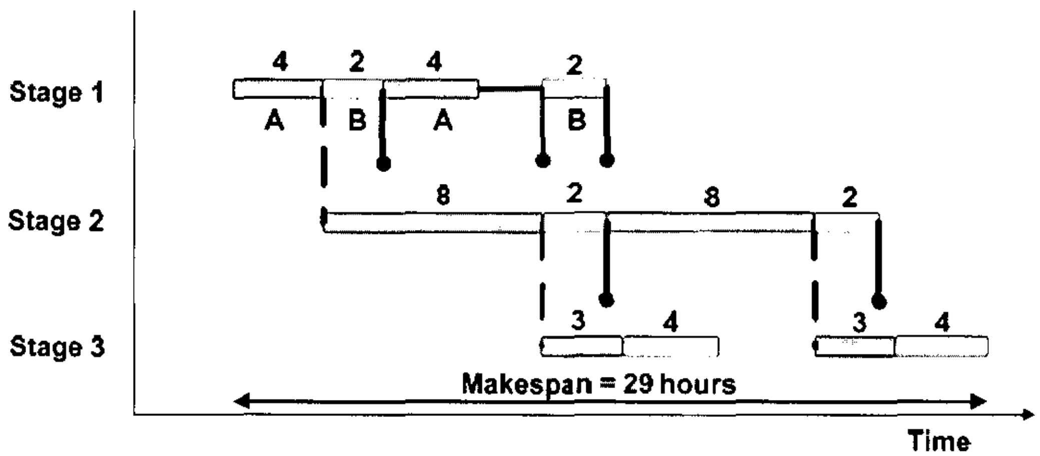

Figure 2.9: Gantt Chart for FIS/UW transfer policy

(Source: Biegler, L.T. et al. 1997)

2.7 Uncertainties in Batch Process Scheduling

It is hard to predict the future with complete certainty. Prices go up and down, and so can the demand. New competitors can come into market, or product can experience obsolescence (Sule, D.R. 2008). The need to account for uncertainty in the planning decisions can essentially be traced back to the core functionality of planning models, which is to allocate resources for the future based on current information and future projections. The foremost consideration in incorporating uncertainties into the planning decisions is the determination of the appropriate representation of the uncertain parameters (Gupta, A. et al. 2003). Uncertainty in process operations can originate from many aspects, such as demand or changes in product orders or order-priority, batch or equipment failures, processing time variability, resource changes, recipe variations, or both etc.

The common batch processing scheduling assumed all the parameters associated with it are known. In real plant, there is supply uncertainty which would affect the batch processing causing the inconsistency. A swiftly rescheduling would be required to deal with the occurring problem.

2.7.1 Supply Change

Supply disruptions can arise from many resources and often times without warning. These disruptions can be entirely external, such as a natural disaster, or they can be internal, rising from the failure to integrate all functions in a supply chain. Disruptions can result from attempts to create a more efficient, cost-conscious supply chain environment. Supply change can also be affiliated with disrupted raw materials and parts, supplier planning and communication issues, service failures caused by supply chain partners (delivery and quality), terrorist infiltration, port operations delays, unscheduled shutdown of the raw materials plant and logistic problems (Sule, D.R. 1997).

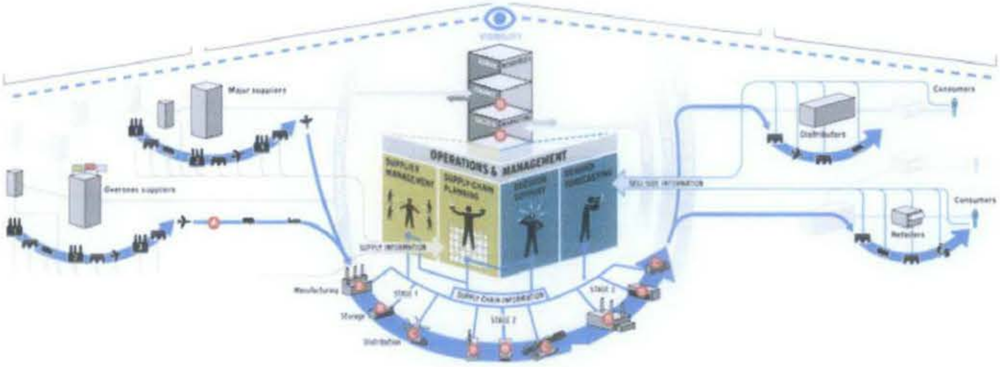

Figure 2.10: Graphical representation of supply chain

A supply chain must be fully integrated to operate at maximum efficiency. Failing to understand the potential vulnerabilities can compromise the supply chain's ability to handle unexpected and sudden shocks. By understanding risk within and external to the supply uncertainties, an organization can more clearly identify its options for optimizing the batch processing to ensure viability and strength.

2.8 Approaches to Supply Change

Li, Z., et al. (2006) pointed out there are a few methods been used to include description of uncertain parameters within the optimization model of the scheduling problems:

a) *Bounded form.* Interval mathematics is used for uncertainty estimation where this method does not require information about the type of uncertainty in the parameters. The bounds represent the ranges of all possible realizations of the uncertain parameters. The upper and lower bounds are determined from historical data from analysts.

b) *Probability description.* This is a common approach for the treatment of uncertainties when information about the behavior of uncertainty is available since it requires the use of probabilistic models to describe the uncertain parameters.

c) *Fuzzy description.* Fuzzy sets allow modeling of uncertainty in cases where historical data are not available. The resulting scheduling models are based on fuzzy sets have the advantage that they do not require the use of complicated integration schemes needed for continuous probabilistic models and they do not need large number of scenarios as the discrete probabilistic uncertainty representations.

These methods may seem as a preventive scheduling which generates policies for scheduling prior to the unexpected events happening. Sanmarti, E. et al. (1996), Rodrigues, M.T.M et al. (1996), Ferrer-Nadal, S. et al. (2007) and Li, Z. et al. (2008) pointed out that reactive scheduling is an approach where it is able to perform rescheduling when the unexpected event takes place. It requires the modification of the existing schedule during the manufacturing process to adapt to the changes which occured. The reactive scheduling actions are based on various underlying strategies where it relies on simple techniques or heuristic rules to seek a quick schedule consistency restoration (Li, Z. et al., 2008).

CHAPTER 3

THEORY AND METHODOLOGY

3.1 Introduction

In batch process, there are multiple stages (i.e. mixing, blending, reacting, separation, etc.) involved which are independent process for each stage of each product. Before the start of the process, a scheduler would perform scheduling which translate into capacity decisions, aggregate (intermediate) planning, and master schedules into job sequences and specific assignments.

In general, batch process scheduling utilizes sequencing approach which is to sequence the products to be produced to obtain minimum makespan. The rule of thumb states that minimum makespan increases plant productivity and indirectly increases plant profitability.

From the initial schedule produced by the scheduler, the objectives set are based on the target to produce the product to meet customers' demand. In this context, the supply to produce the targeted demand is assumed to be available. In real plant operation, there are many unforeseen events which can disrupt the supply of raw materials. Events such as tornadoes, snow storm, logistics and transportation delays, etc. which could result in the supply shortage.

In the event of a supply change, a scheduler would have to make a swift decision to make adjustment or modification to the initial schedule to continue production while waiting for the supply problems to be addressed.

3.2 Scheduling Approach For Supply Change

There are 2 main approaches which are widely used to address uncertainties (i.e. supply change) in scheduling. The following sections discuss the approaches mentioned.

3.2.1 Preventive Scheduling

Preventive scheduling is an approach which is used to generate the initial master schedule before supply changes take place. This approach requires history on the supply chain data. Data which relates on the probability of a supply disruption would be referred when generating the initial master schedule. In another word, this approach generates policies for scheduling prior to the unexpected supply disruption takes place.

This approach would be ineffective to make adjustment or modification to the initial master plan when a supply disruption has taken place.

3.2.2 Reactive Scheduling

Contradicting from preventive scheduling, reactive scheduling is an approach where it is able to perform rescheduling when there is an unexpected event that takes place. This approach makes adjustment and modification to the existing master schedule during the manufacturing process to adapt to the supply change.

In this project, reactive scheduling would be used as a basis of study due to its flexibility to perform rescheduling to the existing master plan to adapt to any supply change.

3.3 Development of Approach for Reactive Scheduling

Reactive scheduling on supply change is divided into two main parts. The first part of scheduling would be to address the supply change by setting an objective function for the products to be produced while the second part focuses on scheduling based on the data extracted from the first part.

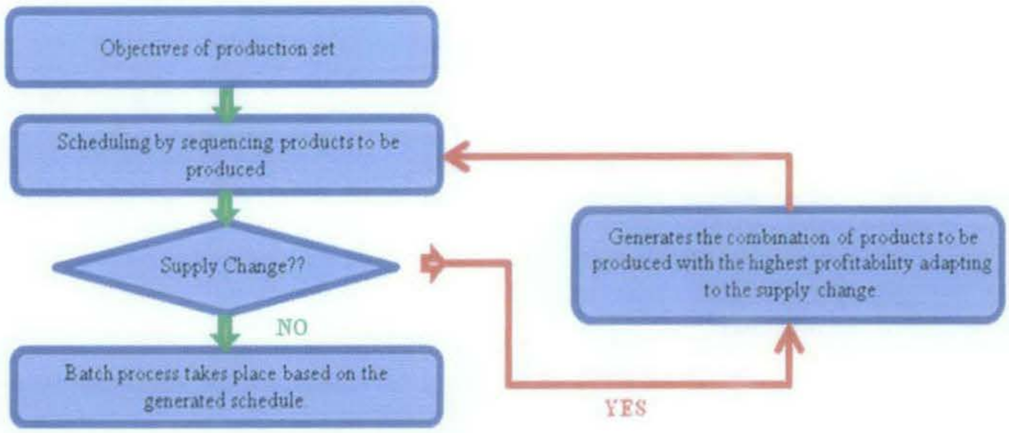

Figure 3.1: Flow chart on reactive scheduling under supply change

Figure 3.1 shows the flow chart on the reactive scheduling based on supply change. Initially, objectives of production would be set. Based from the objectives set, an initial master schedule would be produced to account to the products to be produced. At this point of time, the scheduler assumes that there is no supply change. If there is no supply change, the batch process would take place based on the initial master schedule.

However, if there is an expected supply change which take place, the scheduler would have to make a swift adjustment and modification to the initial master schedule. First, the scheduler would have to generate the combination of products to be produced based on the available supply which to give the highest profitability. Once the products to be produced have been identified, a rescheduling would take place.

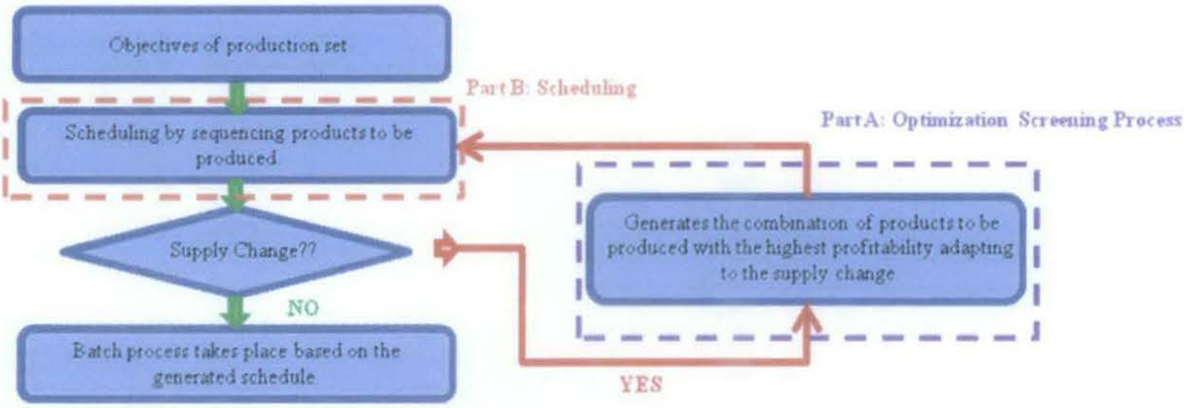

Figure 3.2: Breakdown of project.

Figure 3.2 summarizes the breakdown of the project. As mentioned, the first part of the reactive scheduling would be to perform optimization screening process in search of the combination of the products which can be produced with the available supply. The combination of products with the highest profitability would be chosen to be produced. The second part would be the scheduling tool which would perform scheduling based on the data extracted from the first part. A detailed clarification on both the part would be explained in the later stages.

3.3.1 Part A: Optimization Screening Process

The Optimization Screening Process tool is a computer based model developed using C language. The following detailed the functionality chronology of the Optimization Screening Process.

1) First, the tool will prompt the user to key input the required feedstock (in terms of stoichiometry) for every batch of product produced.

2) The tool will prompt the user to key input the profits generated for each batch of the product produced.

$$\textbf{\textit{Profit for each batch}}\ (\$) = \textbf{\textit{Sales price for each batch}}\ (\$) - \textbf{\textit{Production cost}}\ (\$)$$

Note: Production cost is inclusive of raw materials costs, utilities cost, etc.

3) Next, the tool will prompt the user to key input the number of feedstock (i.e. supply) available.

4) The tool will perform an optimization screening to generate a list of combination of product(s) which is producible from the available feedstock (i.e. supply).

5) From each combination of product(s) generated, the tool would calculate out the profits generated from each combination of product(s).

$$\textbf{\textit{Overall Profit}}\ (\$) = \sum (P_i \times N_i)$$

where P_i = Profit for each batch of product i

N_i = Number of batches of product i

6) The tool would screen out the most profitable combination of product(s).

7) The tool would display the result which would be used to be key input into (Part B: Scheduling) a computer based tool using matrix approach to perform sequencing of the products to be produced.

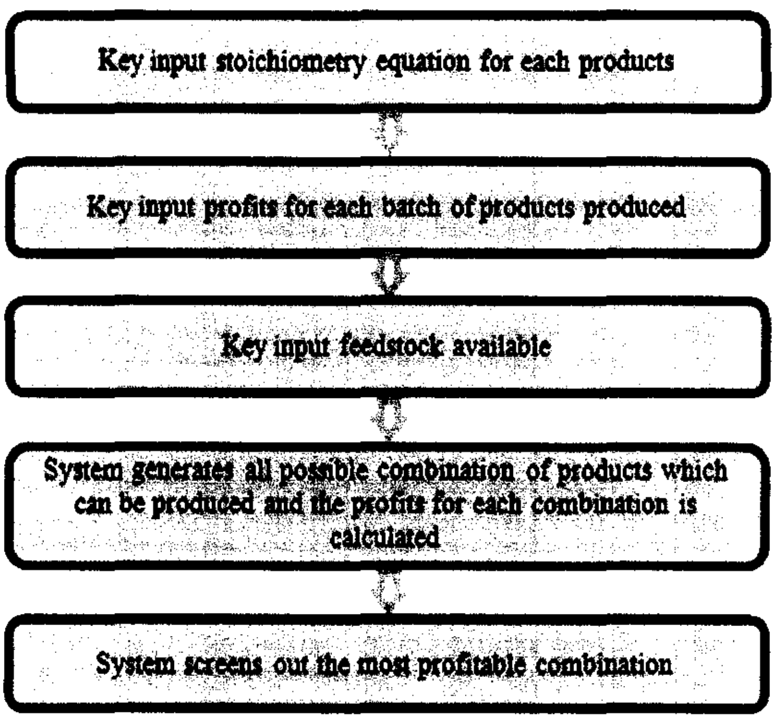

Figure 3.3: Flow Chart of Optimization Screening Process

Figure 3.3 illustrates chronology of the functionality of the computer based model (i.e. Optimization Screening Process).

3.3.2 Part B: Scheduling

The scheduling tool is a computer based model developed using C++ programming by Shafeeq, A. (2008a). This scheduling tool is developed using matrix approach which allow the user to run the iteration either on full enumeration or partial enumeration. The data obtained from Part A which is the Optimization Screening Process is key input into this scheduling tool to complete the reaction scheduling using matrix approach under supply change. The following detailed the functionality chronology of the scheduling tool developed by Shafeeq, A. (2008a).

1) First, the tool would prompt the user on the number of products in the batch process.

2) Then, the tool would prompt the user on the number of stages in the batch process.

3) Next, the tool would prompt the user on the processing time for every stage for each product.

4) The user would have to select the transfer policy.

5) The tool would generate a list of possible sequence for all the product(s) based on permutation rules.

6) If the user chooses to run the iteration using partial enumeration, step 6 is followed whereas if the user chooses to run the iteration using full enumeration, step 7 is followed.

7) Using heuristic rules, the tool would filter out the possible sequence which would not produce the minimum time. Makespan would be calculated using matrix

approach for the unfiltered sequence and the minimum makespan sequence and the makespan time would be displayed by the tool.

8) Makespan would be calculated using matrix approach for all the possible sequence and the minimum makespan sequence and the makespan time would be displayed by the tool.

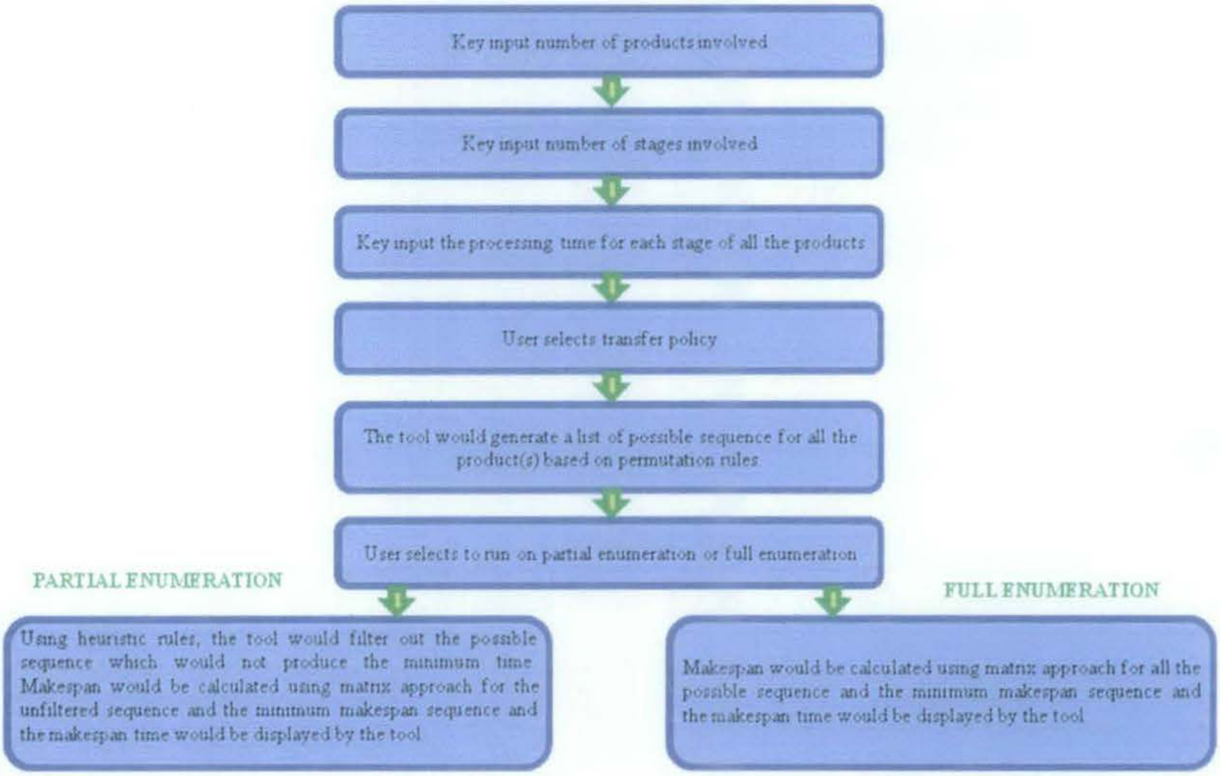

Figure 3.4: Flow Chart of Scheduling

Figure 3.4 illustrates chronology of the functionality of the computer based model developed by Shafeeq, A. (2008a) (i.e. scheduling process using matrix approach).

3.3.3 Development of Heuristic Rules

The heuristic rules has been developed by Shafeeq, A. (2008a) based on two critical observations made on the matrix representation of the batch process.

a) The optimal sequence can start with the product that has the least makespan in the first stage.

b) The optimal sequence can start with the product that has the sum of its processing recipe and processing time in the last stages of all other products with the least value compared to the value when calculated for other products using the same procedure.

3.4 Limitations of Tool

There are a few limitations and assumptions being considered in this tool. The following detailed the limitations of the tool:

1) The tool can only be run for 3 types of products.
2) The tool can only consider a maximum of 5 types of supplies.
3) Products demand is not being considered.
4) Feedstock replenishment policy is not being considered.

CHAPTER 4

RESULT AND DISCUSSION

4.1 Scheduling Based On Supply Change

As mentioned in the Section 3.3, the tool developed would utilize simple screening process in order to search for the possible products being produced from the available feedstock. Information such as required feedstock (in terms of stoichiometry) for every batch of product produced, price (i.e. profits) of the products, the feedstock available, number of stages, processing time for each stage for all the products and transfer policy used would be inputted by the scheduler.

From the information provided, the tool would screen and identify possible product(s) which can be produced from whatever feedstock is available. The example below detailed the functionality of the tool to performed scheduling using matrix approach under supply change.

Note: The example below is a hypothetical example which depicts the actual scenario of a given multiproduct batch process. This example is used to verify the functionality of the tool to perform reactive scheduling using matrix approach under supply change.

Example

The following would be the stoichiometry equations of the multiproduct batch process:

A +2B + C → P1 (Profit for each batch = $ 12.5k)

2B + D → P2 (Profit for each batch = $ 11.5k)

C + D → P3 (Profit for each batch = $ 11.0k)

Note: The profit given is the net profit generated based on the selling price of each batch of the product subtracted from the total cost to produce each batch.

The following is the available feedstock:

A = 8

B = 8

C =7

D = 8

E = 7

Table below shows the processing time of the 3 products in 3 stages.

Table 4.1: Processing time of 4 products in 3 stages

Products	Processing Time (h)		
	S1	S2	S3
P_1	5	8	6
P_2	9	3	2
P_3	4	5	3

The first step of the tool is to list out the combination of products which can are producible from the available feedstock available. Since there are 3 products, the possible sequences which are used to screen the possible combination of amount of batches to be produced for each product would be 6 (i.e. based on permutation rule).

No. of possible sequences = 3! = 6 sequences

Below shows the list of sequences used for the screening:

- P_1, P_2, P_3
- P_1, P_3, P_2
- P_2, P_1, P_3
- P_2, P_3, P_1
- P_3, P_1, P_2
- P_3, P_2, P_1

Using the loop function in the C programming, screening is done where according to each sequence as mentioned earlier. The following shows the screening process for the sequence of P_1, P_2, P_3.

$A + 2B + C \rightarrow P1$

$2B + D \rightarrow P2$

$C + D \rightarrow P3$

Table 4.2: Summary of screening process for sequence of P_1, P_2, P_3

Products			Feedstocks Available				
P_1	P_2	P_3	A	B	C	D	E
4			8	8	7	8	0
	0		4	0	3	8	0
		3	4	0	3	8	0

The Table 4.2 summarizes the screening process. The maximum batches of P_1 which is producible from the available feedstock (A=8, B=8, C=7, D=8, E=0) would be 4 batches. After screening process for P_1, the feedstock is left with (A=4, B=0, C=3, D=8, E=0). From that amount of feedstock, the maximum batches of P_2 producible would be 0 batches and the remaining feedstock would be (A=4, B=0, C=3, D=8, E=0). The screening is further done for P_3 in which the maximum number of batches producible would be 3 batches.

The Table 4.3 is a summary of all the combination of number of batches producible for P_1, P_2, P_3 based on the different screening sequence.

Table 4.3: Summary of batches producible based on different screening sequence

Sequence	P_1	P_2	P_3	Profit ($)
P_1, P_2, P_3	4	0	3	83.00
P_1, P_3, P_2	4	0	3	83.00
P_2, P_1, P_3	0	4	4	90.00
P_2, P_3, P_1	0	4	4	90.00
P_3, P_1, P_2	0	1	7	88.50
P_3, P_2, P_1	0	1	7	88.50

(Note: Profit price for A = 12.5; B = 11.5; C = 11.0)

The screening process would screen the most profitable combination of products to be produced which is 4 batches of P_2 and 4 batches of P_3.

The profit is calculated from using the following equation:

$$\text{Profit} = \Sigma(\text{Number of batches of product i})(\text{Profit price of one batch of product i})$$

The information from this section would be inputted into the second part of the program in which the scheduling takes places using matrix approach to sequence out the products to be produced to achieve the least makespan.

4.2 Scheduling Using Matrix Approach

The scheduling tool using matrix approach developed by Shafeeq, A. (2008a) and Shafeeq, A., et al. (2008b) started off by key inputting all the necessary data (i.e. no. of stages, time for every stage, policy transfer, etc.) as illustrated in the figure below.

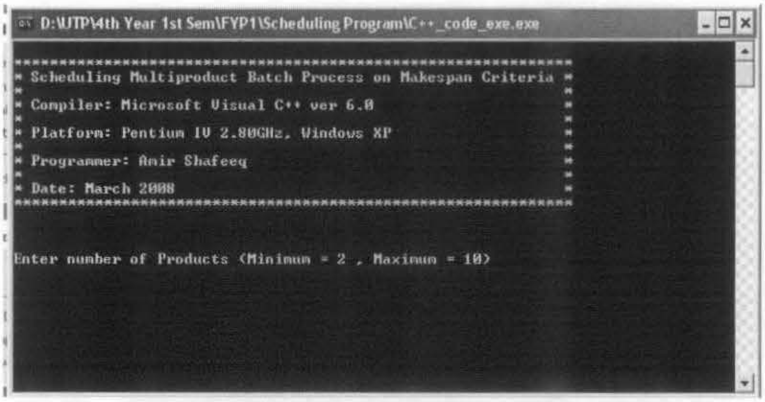

Figure 4.1: Snapshot of the scheduling tool

The following is the result obtained from the tool:

No. of Possible Production Sequences = 40320

No. of Partial Production Sequences = 20160

No. of Production Sequences with Minimum Makespan = 576

One of the possible product sequence with the minimum makespan = P_3, P_2, P_3, P_2, P_3, P_2, P_3, P_2

Concluding Remark

Scheduling generation acts as a predictive mechanism that determines planned start and completion times of production tasks based on given requirements and constraints prior to the production process. The optimal batch process schedule is often based on designer's choice for production sequence offering minimum makespan.

A repetition on whole procedures in the matrix approach to address different production sequence will enable it to calculate the makespan for the corresponding sequence. Repeating it for all possible production sequence will lead to the makespan for each of the possible sequence to be determined.

Generally, the concept used to address batch process under supply change is reactive scheduling (i.e. rescheduling). Whenever there is an unexpected supply change, a modification or an adjustment has to be made to the initial master schedule.

The tool developed in this project is separated into 2 parts in which the first part is to perform optimization screening towards all the available supply. From that, the tool would run and perform screening to screen out the combination of products producible from the available supply. The tool would then calculate out the profits of all the combination and screen the most profitable combination. The results from this tool would be referred to perform scheduling in the second part of the tool.

The second part of this tool which is programmed to perform scheduling using the matrix approach has been developed by Shafeeq, A. (2008a). Basically, the function of this tool is to sequence the products to obtain the sequence with the minimum makespan. As mentioned in the earlier section, minimum makespan would be considered to be an optimum solution in batch process scheduling. Minimum makespan increases plant throughput as well as increases profitability of the plant.

With the inclusion of heuristic approach, the computational time can be reduced significantly especially when performing scheduling for large tasks involved. Referring to the results, the heuristic approach has helped to reduce the iteration from 362,880 possible production sequences to 120,960 sequences which is a reduction of 66.7 percent. This has a significant reduction in computational time.

Generally, both parts of the tool are considered to be user-friendly. The tool does not require a scheduler with great understanding and knowledge in batch scheduling or operations management to perform reactive scheduling when there is a supply change.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

Conclusion

The batch process scheduling appears to be rather complex with various parameters along with supply change on top of that. MILP and MINLP were widely used in the past to overcome this complex optimization. The matrix approach provides simple formulation the computer programming allows makespan calculation to be executed swiftly. The improved functionality would be implemented to the matrix approach based developed by Shafeeq (2008a) to address the supply change.

In the nutshell, the objectives in this project have been successfully completed. The computer based model to perform scheduling using matrix approach under supply change has been successfully developed. The model has been tested and verified with a hypothetical example.

Recommendations

Reactive scheduling would be added to the functionality of the tool developed by Shafeeq (2008a). Another recommended mechanism to this study would be to include the preventive scheduling function which is based on probability of unexpected event occurring. This would generate policies to determine the optimum scheduling function.

REFERENCES

Argoneto, P., Perrone, G., Renna, P., LoNigro, G., Bruccoleri, M. and Noto La Diega, S. 2008. *Production Planning in Production Networks: Models for Medium and Short-term Planning*, London, Springer.

Biegler, L.T., Grossmann, I.E., and Westberg, A.W. 1997. *Systematic Methods of Chemical Process Design.* Prentice-Hall.

Edgar, T.F., Himmelblau, D.M. and Lasdon, L.S. 2001. *Optimization of Chemical Processes*, 2nd Edition, McGraw Hill.

Ferrer-Nadal, S., Mendez, C.A., Graells, M., Puigjaner, L. 2007. *Optimal Reactive Scheduling of Manufacturing Plants with Flexible Batch Recipes*, Ind. Eng. Chem. Res., 46, 6273-6283.

Gunther, H. and van Beek, P. 2003. *Advanced Planning and Scheduling Solutions in Process Industry*, Verlag Berlin Heidelberg, Springer.

Gupta, A. and Maranas, C.D. 2003. *Managing demand uncertainty in supply chain planning.* Science Direct Computers and Chemical Engineering 27 (2003) 1219-1227.

Ha, J.K., Chang, H.K., Lee, E.S., Lee, I.B., Lee, B.S. and Yi, G. 2000. *Intermediate Storage Tank Operation Strategies in the Production Scheduling of Multi-Product Batch Processes.* Computers & Chemical Engineering, 24: (2-7) 1633 – 1640.

Heizer, J. and Render, B. 2008. *Operations Management.* 9th Edition, Pearson International Edition.

Hong, J., Prabhu, V. and Wysk, R, 2001. *Real-Time Batch Sequencing Using Arrival Time Control Algorithm.* Int. J. Prod. Res., 2001, Vol. 39, No. 17, 3863-3880.

Ko, D., Na, S., Moon, I., Lee, I. 1999. *Development of a Batch Manager for Dynamic Sceduling and Process Management in Multiproduct Batch Processes*, Korean J. Chem. Eng., 17(1), 27-32 (2000).

Ku, H.M. and Karimi, I.A. 1992. *Multiproduct Batch Plant Scheduling.* In CACHE Process Design Case Studies. Grossmann, I and Morari, M. (Eds.), CACHE Corporation, Austin, TX.

Li, Z. and Ierapetritou, M. 2006. *Process Scheduling under Uncertainty: Review and Challenges.* Computers and Chemical Engineering 32 (2008) 715-727.

Li, Z. and Ierapetritou, G. 2008. *Reactive Scheduling Using Parametric Programming*, Wiley Interscience, DOI 10.1002/aic.11593.

Lopez, P. and Roubellat, F. 2001. *Production Scheduling*, London. Hermes Science Ltd.

Pochet, Y. and Wolsley, L. 2006. *Production Planning by Mixed Integer Programming*, USA, Springer.

Rodrigues, M.T.M., Gimeno, L., Passos, C.A.S, Campos, M.D. 1996. *Reactive Scheduling Approach For Multipurpose Chemical Batch Plants*, Computer Chem. Engng Vol. 20 , Suppl., pp. S1215-S1220.

Ryu, J.H. and Pistikopoulus, E.N. 2007. *A Novel Approach to Scheduling of Zero-Wait Batch Processes Under Processing Time Variations*, Computers & Chemical Engineering, 31: (3) 101-106.

Sanmarti, E., Huercio, A., Espuna, A., Puigjaner, L. 1996. *A Combined Scheduling/Reactive Scheduling Strategy To Minimize The Effect of Process Operations Uncertainty in Batch Plants*, Computer Chem. Engng Vol. 20, Suppl., pp. S1263-S1268.

Shafeeq, A. 2008. *Scheduling Multiproduct Chemical Batch Processes using Matrix Representation*, Ph.D. Thesis, Universiti Teknologi PETRONAS, Malaysia.

Shafeeq, A., M.I. Abdul Mutalib, K.A. Amminudin and Muhammad, Ayyaz 2008. *Scheduling Three Stage Flowshop Processes with No Intermediate Storage Using Novel Matrix Approach*. Journal of Applied Sciences 8 (11): 2136-2141.

Shafeeq, A., M.I. Abdul Mutalib, K.A. Amminudin and Muhammad Ayyaz, 2008. *A Heuristic Method To Search For Optimal Solution Using Partial Enumeration For A Multiproduct Chemical Batch Process*.

Sule, D.R. 1997. *Industrial Scheduling*, USA, PWS Publishing Company

Sule, D.R. 2008. *Production Planning And Industrial Scheduling*, USA, CRC Press.

Tra N.T.L 2000. *Comparison of Scheduling Algorithms For A Multi-Product Batch Chemical Plant With A Generalized Serial Network*, Master's thesis, Virginia Polytechnic Institute and State University.

Vob, S. and Woodruff, D. 2003. *Introduction to Computational Optimization Models for Production Planning in a Supply Chain*, Germany, Springer.

APPENDIX

The following is the pseudo code for the 1st part of the programming using C programming:

```
#include<stdio.h>
#include<conio.h>

int main()
{
int P1, P2, F11, F12, F13, F14, F15, F21, F22, F23, F24, F25, F31, F32, F33, F34, F35,
FS1, FS2, FS3, FS4, FS5, AA=0, AB=0, AC=0, m, AD=0, AE=0, AF=0, AG=0, AH=0,
AI=0, AJ=0, AK=0, AL=0, AM=0, AN=0, AO=0, AP=0, AQ=0, AR=0, AS=0, AT=0,
AU=0, AV=0, AW=0, AX=0, AY=0, AZ=0, AAA=0, AAB=0, AAC=0, AAD=0, n,
BA=0, BB=0, BC=0, p=0, o, q, r ,s, t, u, v, w, x, a, b, c, d, e, f, BD=0, BE=0, BF=0,
BG=0, BH=0, BI=0, BJ=0, BK=0, BL=0, BM=0, BN=0, BO=0, BP=0, BQ=0, BR=0,
BS=0, BT=0, BU=0, BV=0, BW=0, BX=0, BY=0, BZ=0, BAA=0, BAB=0, BAC=0,
BAD=0, CA=0, CB=0, CC=0, CD=0, CE=0, CF=0, CG=0, CH=0, CI=0, CJ=0, CK=0,
CL=0, CM=0, CN=0, CO=0, CP=0, CQ=0, CR=0, CS=0, CT=0, CU=0, CV=0, CW=0,
CX=0, CY=0, CZ=0, CAA=0, CAB=0, CAC=0, CAD=0;
double D1=0, D2=0, D3=0, pr1=0, pr2=0, pr3=0, pr4=0, pr5=0, pr6=0;


printf ("Enter stoichiometry of F1 for P1: \n");
scanf("%d", &F11);
printf ("Enter stoichiometry of F2 for P1: \n");
scanf("%d", &F12);
printf ("Enter stoichiometry of F3 for P1: \n");
scanf("%d", &F13);
printf ("Enter stoichiometry of F4 for P1: \n");
```

```
scanf("%d", &F14);
printf ("Enter stoichiometry of F5 for P1: \n");
scanf("%d", &F15);
printf("Enter price for P1: \n");
scanf("%lf", &D1);

printf ("\nEnter stoichiometry of F1 for P2: \n");
scanf("%d", &F21);
printf ("Enter stoichiometry of F2 for P2: \n");
scanf("%d", &F22);
printf ("Enter stoichiometry of F3 for P2: \n");
scanf("%d", &F23);
printf ("Enter stoichiometry of F4 for P2: \n");
scanf("%d", &F24);
printf ("Enter stoichiometry of F5 for P2: \n");
scanf("%d", &F25);
printf("Enter price for P2: \n");
scanf("%lf", &D2);

printf ("\nEnter stoichiometry of F1 for P3: \n");
scanf("%d", &F31);
printf ("Enter stoichiometry of F2 for P3: \n");
scanf("%d", &F32);
printf ("Enter stoichiometry of F3 for P3: \n");
scanf("%d", &F33);
printf ("Enter stoichiometry of F4 for P3: \n");
scanf("%d", &F34);
printf ("Enter stoichiometry of F5 for P3: \n");
scanf("%d", &F35);
printf("Enter price for P3: \n");
```

```
scanf("%lf", &D3);


printf ("\nEnter available feed of F1 : \n");
scanf("%d", &FS1);
printf ("Enter available feed of F2 : \n");
scanf("%d", &FS2);
printf ("Enter available feed of F3: \n");
scanf("%d", &FS3);
printf ("Enter available feed of F4: \n");
scanf("%d", &FS4);
printf ("Enter available feed of F5: \n");
scanf("%d", &FS5);

m=0;
while (AA >= 0 && AB >= 0 && AC >=0 && AD >=0 && AE >=0)
{
        AA = FS1 - (m*F11);
        AB = FS2 - (m*F12);
        AC = FS3 - (m*F13);
        AD = FS4 - (m*F14);
        AE = FS5 - (m*F15);
   m++;

        }
m=m-2;


n=0;
while (AF >= 0 && AG >= 0 && AH >=0 && AI >=0 && AJ >=0)
```

```
{
        AF = FS1 - (m*F11) - (n*F21);
        AG = FS2 - (m*F12) - (n*F22);
        AH = FS3 - (m*F13) - (n*F23);
        AI = FS4 - (m*F14) - (n*F24);
        AJ = FS5 - (m*F15) - (n*F25);
        n++;
}
n=n-2;


o=0;
while (AF >= 0 && AG >= 0 && AH >=0 && AI >=0 && AJ >=0)
{
   AK = FS1 - (m*F11) - (n*F21)- (o*F31);
        AL = FS2 - (m*F12) - (n*F22)- (o*F32);
        AM = FS3 - (m*F13) - (n*F23)- (o*F33);
        AN = FS4 - (m*F14) - (n*F24)- (o*F34);
        AO = FS5 - (m*F15) - (n*F25)- (o*F35);
        o++;
}
o=o-2;


p=0;
while (AP >= 0 && AQ >= 0 && AR >=0 && AS >=0 && AT >=0)
{
        AP = FS1 - (p*F11);
        AQ = FS2 - (p*F12);
        AR = FS3 - (p*F13);
```

```
        AS = FS4 - (p*F14);
        AT = FS5 - (p*F15);
  p++;

        }
p=p-2;



q=0;
while (AU >= 0 && AV >= 0 && AW >=0 && AX >=0 && AY >=0)
{
        AU = FS1 - (p*F11) - (q*F31);
        AV = FS2 - (p*F12) - (q*F32);
        AW = FS3 - (p*F13) - (q*F33);
        AX = FS4 - (p*F14) - (q*F34);
        AY = FS5 - (p*F15) - (q*F35);
        q++;
}
q=q-2;



r=0;
while (AZ >= 0 && AAA >= 0 && AAB >=0 && AAC >=0 && AAD >=0)
{
  AZ = FS1 - (p*F11) - (q*F31)- (r*F21);
        AAA = FS2 - (p*F12) - (q*F32)- (r*F22);
        AAB = FS3 - (p*F13) - (q*F33)- (r*F23);
        AAC = FS4 - (p*F14) - (q*F34)- (r*F24);
        AAD = FS5 - (p*F15) - (q*F35)- (r*F25);
        r++;
```

```
}
r=r-2;



s=0;
while (BA >= 0 && BB >= 0 && BC >=0 && BD >=0 && BE >=0)
{
        BA = FS1 - (s*F21);
        BB = FS2 - (s*F22);
        BC = FS3 - (s*F23);
        BD = FS4 - (s*F24);
        BE = FS5 - (s*F25);
   s++;

        }
s=s-2;


t=0;
while (BF >= 0 && BG >= 0 && BH >=0 && BI >=0 && BJ >=0)
{
        BF = FS1 - (s*F21) - (t*F11);
        BG = FS2 - (s*F22) - (t*F12);
        BH = FS3 - (s*F23) - (t*F13);
        BI = FS4 - (s*F24) - (t*F14);
        BJ = FS5 - (s*F25) - (t*F15);
        t++;
}
t=t-2;
```

```
u=0;
while (BK >= 0 && BL >= 0 && BM >=0 && BN >=0 && BO >=0)
{
   BK = FS1 - (s*F21) - (t*F11)- (u*F31);
         BL = FS2 - (s*F22) - (t*F12)- (u*F32);
         BM = FS3 - (s*F23) - (t*F13)- (u*F33);
         BN = FS4 - (s*F24) - (t*F14)- (u*F34);
         BO = FS5 - (s*F25) - (t*F15)- (u*F35);
         u++;
}
u=u-2;



v=0;
while (BP >= 0 && BQ >= 0 && BR >=0 && BS >=0 && BT >=0)
{
         BP = FS1 - (v*F21);
         BQ = FS2 - (v*F22);
         BR = FS3 - (v*F23);
         BS = FS4 - (v*F24);
         BT = FS5 - (v*F25);
   v++;

         }
v=v-2;
```

```
w=0;
while (BU >= 0 && BV >= 0 && BW >=0 && BX >=0 && BY >=0)
{
        BU = FS1 - (v*F21) - (w*F31);
        BV = FS2 - (v*F22) - (w*F32);
        BW = FS3 - (v*F23) - (w*F33);
        BX = FS4 - (v*F24) - (w*F34);
        BY = FS5 - (v*F25) - (w*F35);
        w++;
}
w=w-2;


x=0;
while (BZ >= 0 && BAA >= 0 && BAB >=0 && BAC >=0 && BAD >=0)
{
   BZ = FS1 - (v*F21) - (w*F31)- (x*F11);
        BAA = FS2 - (v*F22) - (w*F32)- (x*F12);
        BAB = FS3 - (v*F23) - (w*F33)- (x*F13);
        BAC = FS4 - (v*F24) - (w*F34)- (x*F14);
        BAD = FS5 - (v*F25) - (w*F35)- (x*F15);
        x++;
}
x=x-2;




a=0;
while (CA >= 0 && CB >= 0 && CC >=0 && CD >=0 && CE >=0)
```

```
{
        CA = FS1 - (a*F31);
        CB = FS2 - (a*F32);
        CC = FS3 - (a*F33);
        CD = FS4 - (a*F34);
        CE = FS5 - (a*F35);
  a++;

        }
a=a-2;


b=0;
while (CF >= 0 && CG >= 0 && CH >=0 && CI >=0 && CJ >=0)
{
        CF = FS1 - (a*F31) - (b*F11);
        CG = FS2 - (a*F32) - (b*F12);
        CH = FS3 - (a*F33) - (b*F13);
        CI = FS4 - (a*F34) - (b*F14);
        CJ = FS5 - (a*F35) - (b*F15);
        b++;
}
b=b-2;


c=0;
while (CK >= 0 && CL >= 0 && CM >=0 && CN >=0 && CO >=0)
{
  CK = FS1 - (a*F31) - (b*F11)- (c*F21);
        CL = FS2 - (a*F32) - (b*F12)- (c*F22);
```

```
        CM = FS3 - (a*F33) - (b*F13)- (c*F23);
        CN = FS4 - (a*F34) - (b*F14)- (c*F24);
        CO = FS5 - (a*F35) - (b*F15)- (c*F25);
        c++;
}
c=c-2;




d=0;
while (CP >= 0 && CQ >= 0 && CR >=0 && CS >=0 && CT >=0)
{
        CP = FS1 - (d*F31);
        CQ = FS2 - (d*F32);
        CR = FS3 - (d*F33);
        CS = FS4 - (d*F34);
        CT = FS5 - (d*F35);
   d++;

        }
d=d-2;


e=0;
while (CU >= 0 && CV >= 0 && CW >=0 && CX >=0 && CY >=0)
{
        CU = FS1 - (d*F31) - (e*F21);
        CV = FS2 - (d*F32) - (e*F22);
```

```
        CW = FS3 - (d*F33) - (e*F23);
        CX = FS4 - (d*F34) - (e*F24);
        CY = FS5 - (d*F35) - (e*F25);
        e++;
}
e=e-2;


f=0;
while (CZ >= 0 && CAA >= 0 && CAB >=0 && CAC >=0 && CAD >=0)
{
   CZ = FS1 - (d*F31) - (e*F21)- (f*F11);
        CAA = FS2 - (d*F32) - (e*F22)- (f*F12);
        CAB = FS3 - (d*F33) - (e*F23)- (f*F13);
        CAC = FS4 - (d*F34) - (e*F24)- (f*F14);
        CAD = FS5 - (d*F35) - (e*F25)- (f*F15);
        f++;
}
f=f-2;



pr1 = (m*D1) + (n*D2) + (o*D3);
pr2 = (p*D1) + (q*D3) + (r*D2);

pr3 = (s*D2) + (t*D1) + (u*D3);
pr4 = (v*D2) + (w*D3) + (x*D1);

pr5 = (a*D3) + (b*D1) + (c*D2);
```

```
pr6 = (d*D3) + (e*D2) + (f*D1);

if (pr1 >= pr2 && pr1 >= pr3 && pr1 >= pr4 && pr1 >= pr5 && pr1 >= pr6)
        {printf ("\nThe amount of Product 1 to be produced is: %d", m);
        printf ("\nThe amount of Product 2 to be produced is: %d", n);
        printf ("\nThe amount of Product 3 to be produced is: %d", o);
        printf ("\nThe calculated profit is: %.2f", pr1);
}

else if (pr2 >= pr1 && pr2 >= pr3 && pr2 >= pr4 && pr2 >= pr5 && pr2 >= pr6)
        {printf ("\nThe amount of Product 1 to be produced is: %d", p);
        printf ("\nThe amount of Product 2 to be produced is: %d", r);
        printf ("\nThe amount of Product 3 to be produced is: %d", q);
   printf ("\nThe calculated profit is: %.2f", pr2);
}

else if (pr3 >= pr1 && pr3 >= pr2 && pr3 >= pr4 && pr3 >= pr5 && pr3 >= pr6)
        {printf ("\nThe amount of Product 1 to be produced is: %d", t);
        printf ("\nThe amount of Product 2 to be produced is: %d", s);
        printf ("\nThe amount of Product 3 to be produced is: %d", u);
        printf ("\nThe calculated profit is: %.2f", pr3);
}

else if (pr4 >= pr1 && pr4 >= pr2 && pr4 >= pr3 && pr4 >= pr5 && pr4 >= pr6)
        {printf ("\nThe amount of Product 1 to be produced is: %d", x);
        printf ("\nThe amount of Product 2 to be produced is: %d", v);
        printf ("\nThe amount of Product 3 to be produced is: %d", w);
        printf ("\nThe calculated profit is: %.2f", pr4);
}
```

```
else if (pr5 >= pr1 && pr5 >= pr2 && pr5 >= pr3 && pr5 >= pr4 && pr5 >= pr6)
        {printf ("\nThe amount of Product 1 to be produced is: %d", b);
        printf ("\nThe amount of Product 2 to be produced is: %d", c);
        printf ("\nThe amount of Product 3 to be produced is: %d", a);
        printf ("\nThe calculated profit is: %.2f", pr5);
}

else if (pr6 >= pr1 && pr6 >= pr2 && pr6 >= pr3 && pr6 >= pr4 && pr6 >= pr5)
        {printf ("\nThe amount of Product 1 to be produced is: %d", f);
        printf ("\nThe amount of Product 2 to be produced is: %d", e);
        printf ("\nThe amount of Product 3 to be produced is: %d", d);
        printf ("\nThe calculated profit is: %.2f", pr6);
}

else
{       printf("\n Error");
}

getch();
return 0;
}
```