1. INTRODUCTION

The highest form of modern manufacturing technologies is called Computer-Integrated-Manufacturing (CIM) which is a computerized integration of the material-processing functions and the information-processing functions of the manufacturing systems. “The concept of CIM as an example of Information Technology (IT), in the manufacturing world, was introduced favoring the enhancement of performance, efficiency ... responsive behavior to market differentiations, and time to market” [64].
CIM is a methodology and a goal rather than a group of machines and computers. It can be considered as a manufacturing philosophy in which the computer plays a central role for the manufacturing functions [2].
Nowadays, CIM is directing the technology of manufacturing towards Computer Integrated Factory (CIF) [1].

CIF can be considered as a fully automated factory (factory of the future). In order to develop CIF, we have to improve the performance of CIM. CIM performance highly depends on the performance of the Flexible Manufacturing System (FMS) which is a highly automated manufacturing system for producing different products according to predetermined schedule.
Nowadays, FMS become the more advanced machining resources for part manufacturing in the industry all over the world.
The macro-parameters of the production triangle are Productivity rate (utilization rate), delivery reliability, and inventory level.
This research deals with first two parameters, particularly, FMS meets two critical challenges: Highest Makespan Productivity (HMP) and Effective Delivery Reliability (EDR).

HMP is an extremely essential challenge to modern FMS. Usually traditional FMSs have significant idle times with low machine productivity (and utilization).
In modern CIM, HMP is a desirable objective to be achieved for three basic reasons: first, FMS involves high capital investments, thus, higher machine utilization is required; machines must not stand idle. Second, based on the strategy "Time-Based Competition (TBC)", and the economic concept “Survival of the Fastest”, FMS must be produced faster in an accelerated style to react quickly to the current and future competitive market conditions. Third, HMP leads to lower Work In Process (WIP) inventory and lower setup time (cost). Accordingly, HMP results lower cost and more sales, i.e. higher profitability. Profitability highly depends on the productivity (Appendix B.2). FMS with HMP could be describing as a kind of Lean Manufacturing System (LMS). For these reasons, HMP is a proper objective to be achieved.

One of the purposes of this research is to attain this objective.
The second major challenge of the current and future factory is an EDR. In some of the traditional FMSs, the products are manufactured later (after delivery date). The customers have to wait for the products until the processes are completed. This leads to low customer satisfaction or, usually, losing customers. The results are low sales and low profitability.
EDR leads to achieve four important features: first, higher customer satisfaction. Second, attract new customers. Third, have close to zero inventories. Forth, products are delivered without waiting time. Of course, we assume that the products to be produced are high-quality, low-cost products. This results in lower costs and better sales, consequently, higher profitability.

In modern FMS, the products must be produced and delivered directly to the well-known customer as Just In Time (JIT) with EDR. This approach is known as a pull system or Make To Order (MTO). So, EDR is a suitable objective to be achieved.

For this reason, the second purpose of this research is to obtain this objective.

The two objectives (HMP and EDR) highly depend on the quality of FMS scheduling. So, appropriate scheduling is extremely important, and it must be analyzed accurately. Scheduling is defined as the process of assigning operation to resources in the course of time with the objective of optimizing a criterion. Scheduling is playing a critical role in all types of production systems such as manufacturing systems, transportation systems, computer systems, airline company systems and education systems [11].

In manufacturing, scheduling is concerned with the allocation of machines to operations over time in order to meet a single or multiple objectives.

In a general manufacturing environment, scheduling is one of the basic functions of the manufacturing system (Appendix A.1). In classic manufacturing systems, scheduling is the central function of the Production Planning and Control (PPC) system. Scheduling engineer plans the right schedule of the various products that are to be made in the factory [17] (Appendix A.2).

In a typical shop floor control system, order scheduling is an essential phase of the manufacturing control system [1] (Appendix A.3).

Scheduling process starts with a strategic plan and it interacts with other planning functions at the medium or higher level. Decisions made at a higher planning level may impact the scheduling process. When the materials and machines are available at specified times, schedule must be ready to dispatch [12] (Appendix A.4).

The modern manufacturing functions (design, planning, scheduling, order release, control, verification) are arranged in a sequence where outputs of a function are inputs to the next (Figure 1) [2]. These functions are integrated in CIM system.

The basic information-processing functions of classic CIM are design, planning, scheduling and control. Figure 2 shows the functions of the simple model factory. The material flow starts from parts storage and from there materials are brought by Automated Guided Vehicle (AGV) or by other devices from one station to another for processing. The completed products are deposited in the finished goods storage for shipping to the customer. The planning and control of this factory are done using computer network called Local area network (LAN) [2].

CIM consists of the following computerized sub-systems: (Figure 3) [18] Computer Aided Design (CAD), Computer Aided Planning (CAP), Computer Aided Scheduling (CAS), Computer Aided Manufacturing (CAM), and Computer Aided Quality (CAQ).

In modern FMS, scheduling primarily deals with the scheduling of the essential resources. i.e. the machine tools (or groups). Machine scheduling is often the master key to open all the main doors of scheduling of other resources. Machine Scheduling is one of the topics most widely discussed in the literature. Classic work of French [3] and many others give extensive coverage of the scheduling problems [4, 5]. From the newer literature one could use, for example [6–12].

Machine scheduling is used as a tool of control system to follow up the progress of the manufacturing processes.

As the main purpose of FMS is to produce right products by the right process at right time, machine groups of FMS must be scheduled with the right schedule. Efficient and effective scheduling will improve the efficiency and effectiveness of FMS.

FMS Scheduling is a time plan in which different machines are scheduled to different jobs which may have different quantities, different processes, different setups, and/or different process sequences, etc. planned according to a certain priority rule and subjected to certain constraints in order to meet one or multi- objective.

There are many objectives of scheduling system (Appendix A.5). They are varying from system to system based on the strategic goals. Furthermore, the objectives may be single or multi-objectives. It is not easy to state the objectives in FMS scheduling.

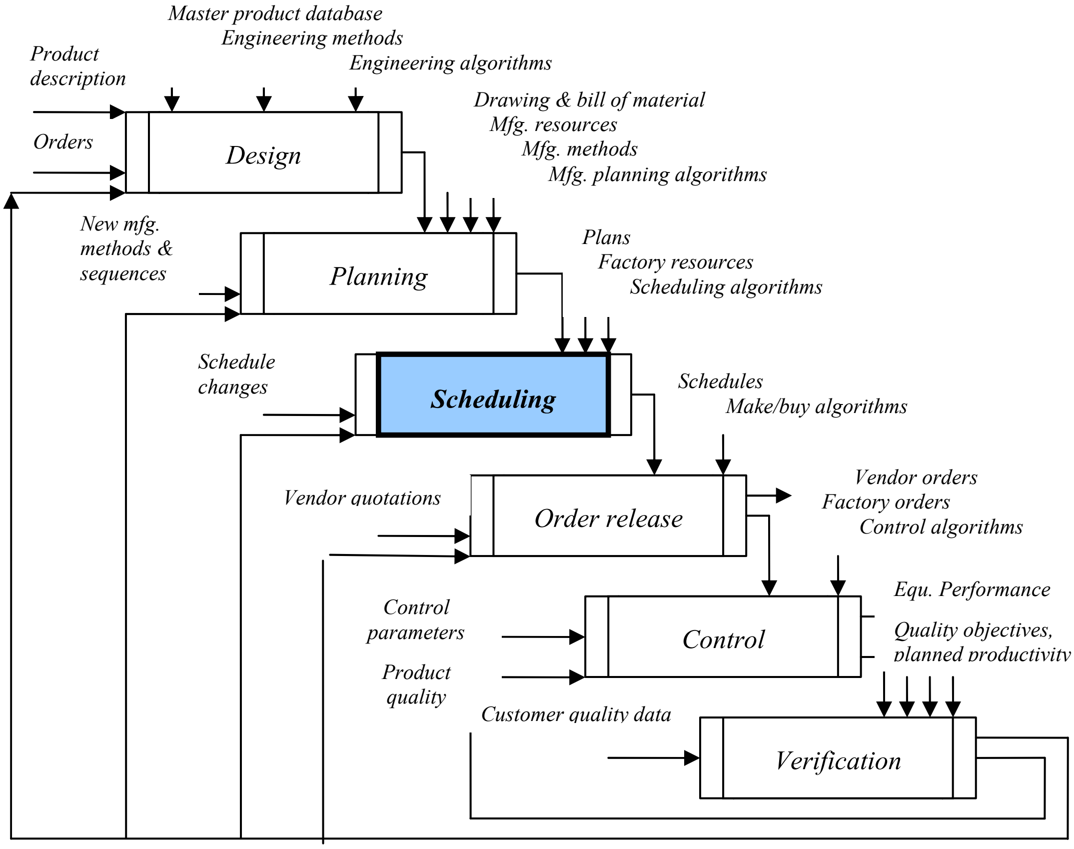


Figure 1: Functions of a manufacturing system

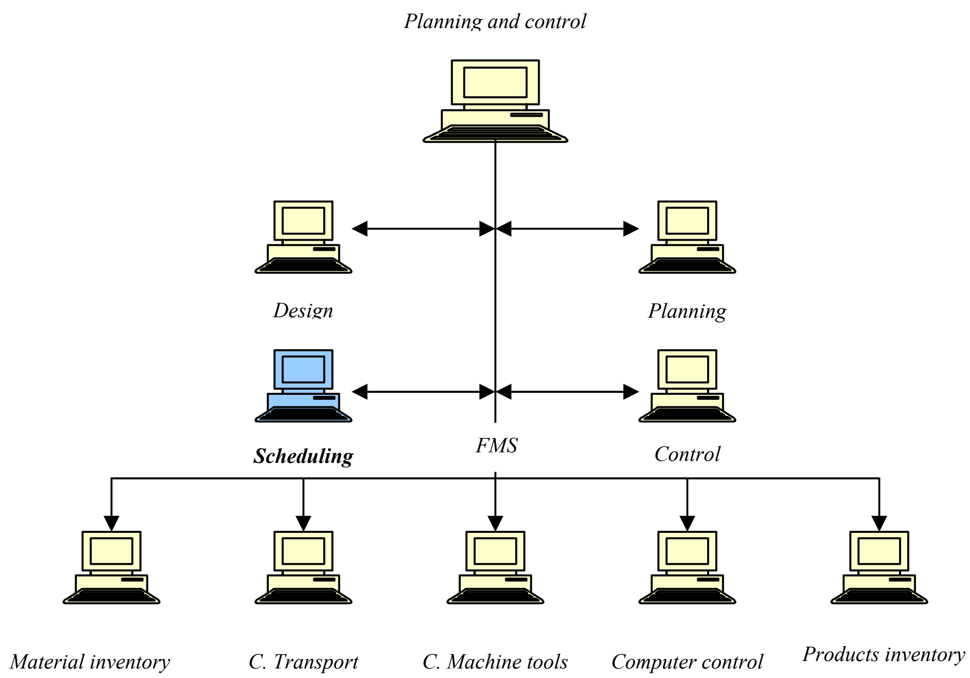


Figure 2: Schematic illustration of a CIM system

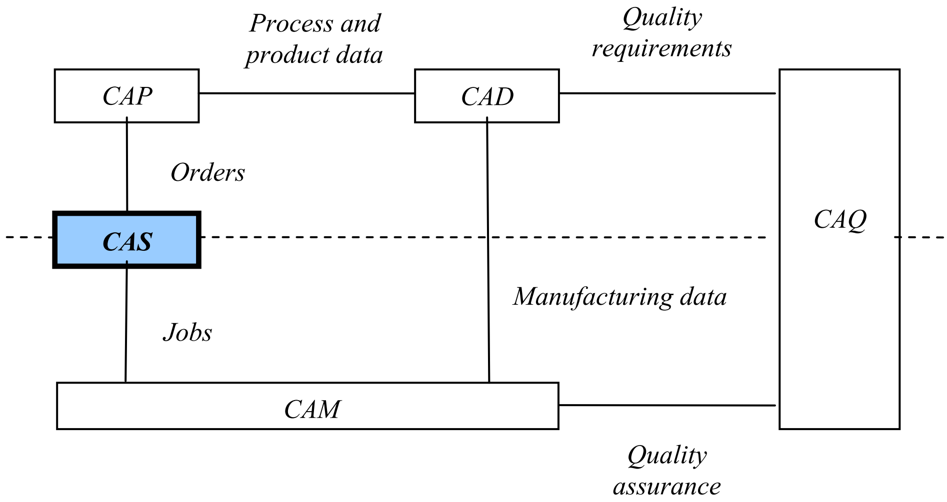


Figure 3: Sub-systems of CIM systems

The objectives usually are trade-off objectives with other manufacturing functions or activities. For example, long setup times or a large number of setup actions may result in low productivity. Also, the objectives HMP and EDR may conflict to each other. The system may have higher delivery reliability with low machine productivity. Thus, a trade-off optimization for FMS scheduling problems is extremely required.

This research deals with the multi-objective FMS scheduling problem.

FMS scheduling problems are, in general, subjected to a number of constraints (Appendix A.7). A solution of a scheduling problem must satisfy these constraints. Usually, there are four kinds of constraints in scheduling problems: processing constraints, capacity constraints, precedence constraints and job constraints.

In this research the scheduling problem to be solved is subjected to certain constraints.

Generally, FMS scheduling problems are complicated NP-hard combinatorial problems. It is difficult to find a feasible solution of FMS scheduling problems even with a small number of machines. For this reason, the development of FMS scheduling solutions remains an important and active research area.

Minimization of minimum of maximum production time close to the maximum of minimum production time is a suitable goal in FMS scheduling problems and may result in HMP and/or EDR. But answering the question how close this goal can be approached is a very complicated problem.

This research tries to answer this question.

In many FMS scheduling problems, the setup times play a significant role. The setup times have high effect on the planned schedules. As setup times increase, makespan increases. Accordingly, the resultant feasible schedule has a long makespan usually, with low machine productivity and/or ineffective delivery reliability.

To reduce setup times (minimize makespan), similar products of the jobs are grouped into batches. The batches are indivisible (unbreakable). Once an upstream operation starts in processing certain batch, the whole batch must be completed before it is transferred to its successor downstream operation.The batch spends a significant time waiting for a machine due to processing the previous batch, also the next machine has usually, an idle time waiting to process that batch. These idle times are increasing when size of batches is large. As idle times increase, makespan increases. Accordingly, again, the resultant feasible schedule has long makespan usually, with low machine productivity and/or ineffective delivery reliability.

Scheduling methods (Appendix A.9) such as mathematical programming methods, heuristic methods (such as scheduling priority rules (SPR)), etc. for solving typical scheduling problems can be used to solve FMS scheduling problems (see for example: [3-12]). Scheduling problems may be solved by nonlinear programming methods (like: ILOG OPL [13] or by simulation methods [14, 15] and with others. The literature on this topic is enormous. The publication of J. M. Proth [63] is one of the last survey papers on this topic.

These methods very frequently provide solutions with significant idle times, low machine productivity and/or ineffective delivery reliability.

There are several modern approaches to solve FMS scheduling problems, for example, methods based on Artificial Intelligence (AI) such as Neural Network (NN), Fuzzy Logic (FL), Genetic Algorithms (GA), Tabu Search (TS), Simulated Annealing (SA) and others. These methods could give effective solutions but these methods are outside of the scope of this dissertation.

An effective way to improve the machine productivity and/or delivery reliability is called Lot streaming. Lot streaming attracted extensive attention in the scheduling community. Most of works are focused on single-job, two-or three-machine problems. Works on multi-job/multi-machine problems are limited.

This research is focused on multi-job /multi-machine lot streaming problems.

Lot streaming becomes a common strategy in batch FMS environments, but most literature of lot streaming is concerned with one system of FMS called Flow-Shop Systems (FSS). Job-Shop Systems (JSS) have received much less attention.

This research is concerned with lot streaming problems for both systems of FMS: FSS and JSS. (Appendix A.6).

As it is known, FMS scheduling problems are complicated NP hard combinatorial problems, Lot streaming makes the problems even much more complicated.

In this research, two new optimization methods were developed to find the optimal lot streaming in order to achieve HMP and EDR. These methods are named as *Break and Test Method (BTM) and Joinable Schedule Approach (JSA).*

1.1 Content of the Dissertation

This dissertation work consists of nine chapters. It begins with an introduction, dealing with FMS scheduling definitions, objectives, constraints, modeling methods, etc. Also definitions of makespan productivity, utilization, workflow acceleration, and delivery reliability are given. Definitions of efficient, effective and perfect schedules, characteristics, and an application of the mathematical models to a scheduling problem on example 1 are analyzed in chapter 1.

In chapter 2, lot streaming definitions are given. Application of the zero setup time lot streaming problem using two new rules (Largest Sub-batch Size rule and Smallest Sub-batch Size rule) is demonstrated. Setup time consideration is given, the research problem is defined, and then the history and state-of-the-art developments are introduced.

In chapter 3, Global minimum of production time of job shop systems. Non-zero and zero setup time job shop models are given. Excess time coefficient, real and ideal schedules are defined. Investigations of lot streaming, rang, coefficient, and efficiency are introduced. The effect of the setup time values, bottleneck utilization and idle time coefficient and sub-batch size determination are investigated.

In chapter 4, Joinable Schedule Approach (JSA) definition, its procedure, characterization of lot streaming, joinability test and application of JSA for one and two jobs flow shop lot streaming problems are given.

In chapter 5, Global minimum of production time of flow shop systems. Flow shop systems with non-zero and zero setup time models are given. Excess time coefficient optimization is investigated. Characteristics of JSA for flow shop lot streaming problems are introduced. Effective delivery reliability by JSA is introduced. Conclusions of using JSA for FMS lot streaming problems are given.

In chapter 6, Break and Test Method (BTM) definition and its procedure are given; makespan productivity and delivery reliability tests are performed. Applications of BTM for flow shop and job shop lot streaming problems are given.

In chapter 7, Case studies of job shop and flow shop lot streaming problems using BTM and JSA are described. The effect of setup times is estimated. Comparing the BTM and JSA results are presented.

In chapter 8, conclusions and recommendations are discussed.

1.2 Mathematical model for FMS scheduling

In this section, the mathematical model for FMS scheduling is formulated. This model is valid for the determination of suitable schedules for FMS. The model consists of parts as follows:

- ***Jobs, machine groups***

The problem considered in this research is an FMS scheduling problem which can be formulated as the following question:

How to schedule FMS consisting of a set of **different** machine groups *M* (*1, 2…, m-1, m … M*) to process a set of **different** jobs (bathes) *J* (*1, 2… j-1, j …J*) of **different** volumes (number of products) n_1, $n_{2....}$ n_j through a set of **different** operations (*1, 2…k…K*) by **different** processing time $p_{j,m}$?

- ***Processing time data***

The processing time of one part is determined as $\tau_{k,k',m}$ where *k* and *k′* express the given operation and the next operation to be processed on machine group *m*. The processing time can be expressed as $\tau_{j,m,k}$ where *j* indicates the job index. The integer *k* expresses the order number of the given sequence. In this research, as in the most of the cases, only $\tau_{j,m}$ will be used (*k* – is omitted) because the order number of operations does not have any role. This time values are expressed in the necessary machine group time units. (That is, if the machine group consists of M_m number of machines, the time necessary for one machine is divided by M_m). The processing time values are determined by the process planning sub-system of CAPP, including manufacturing data determination (optimization).

- ***Production time period***

The time period of production will be considered here is the same for all of the parts (common time period). The time interval of the production is given by the time value "T_{sch}". So the production interval is $[0, T_{sch}]$.The T_{sch} value is, for example, the length of a shift, of a day, of the week, or so. The other case when the jobs have individual time periods is the topic of future research.

- ***Production system types***

This research deals with FMS scheduling problem for both types: Flow shop system (FSS) and Job Shop System (JSS) (Appendix A.6).

These systems are based on the route (processing order) of the jobs to be processed. (Considering CIM systems it is supposed that the order of processing is determined by the manufacturing sequences planning sub-system of Computer Aided Process Planning (CAPP)). In FSS, all jobs visit the machine groups in the same route (identical) whereas in JSS, any route of the jobs among the machine groups is possible. The patterns of job-flow in FSS and JSS are illustrated in Figures 4, 5 respectively. JSS is more complicated than FSS. In JSS, jobs may not require all *M* machines; and they may visit the same machines more than once. Any given machine may process new jobs arriving from outside the factory and process jobs arriving from WIP. The same machine may be the last step for a particular job, or it may be an intermediate step. Every job has a particular route. For example, in order to processing job 2, the route starts from third machine, last machine, then, certain machine, *3 →M→ m*.

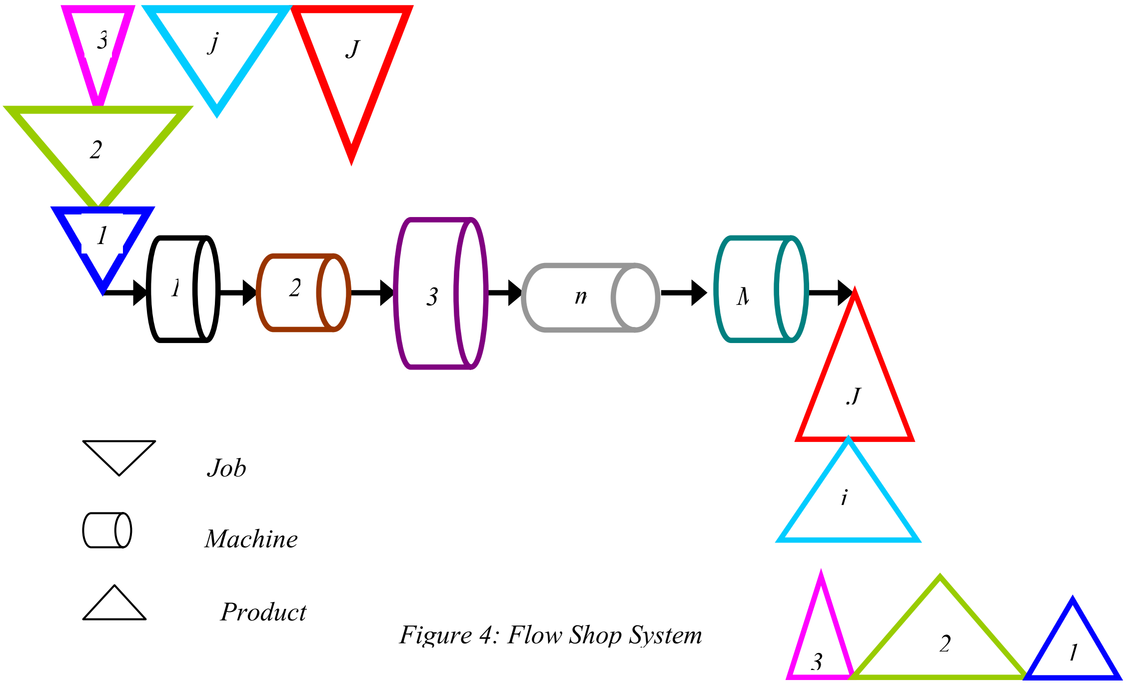


Figure 4: Flow Shop System

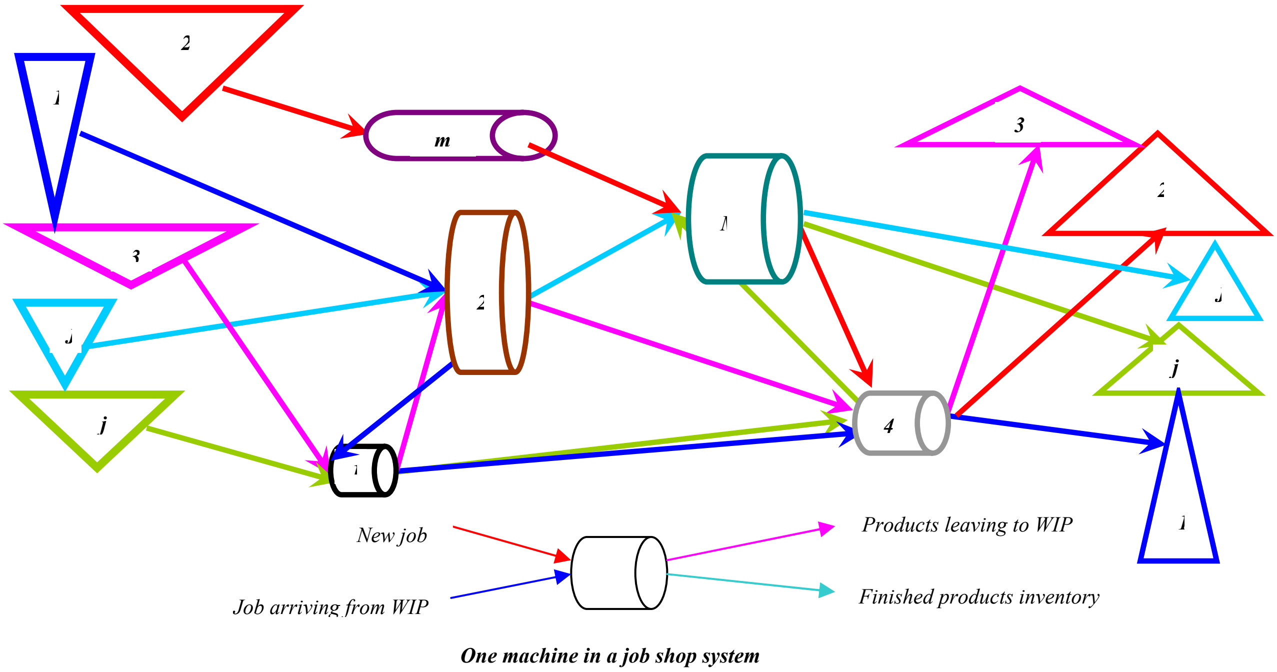


Figure 5: Job Shop System

• ***Setup times considerations***

When a machine group switches from processing one batch to another, setup time is necessary for preparing the new process. This is concerned with the necessary changes in machine conditions, tools, fixtures conditions; parts transport delays and so on. The setup time could be composed of different elements: changing time, handling time, break time, delay time and so on.

It is remarkable that the transportation times and setup times are very different quantities. But, sometimes, these can be treated in the same manner.

In FMS, the setup times are usually much shorter than the process times; otherwise the systems could be hardly called flexible.

In the present research, setup times are considered. For the sake of simplicity, for any of the machine-groups, setup times are indicated as $\delta_{j,m}$ be the setup time of machine group *m* to process job *j*.

• ***Scheduling criterion***

In the present research, we will use the value of makespan in estimation the goodness of schedules. Makespan is the maximum production (completion) time required to process all jobs. This dimension is highly used in practice, simple and effective to apply. The makespan is usually indicated in the literature as $C_{\max}$ (see: [3, 12]). Here we use for that t_{pr}. Of course, $t_{pr} = C_{\max}$. It is remarkable that there is a difference between lead time and production time. Production time is machine-related time, Lead time is job-related time.

The value of makespan is the most important dimension using to measure the quality of the schedule, but it is remarkable, in point of view of this research that the makespan value is not enough to measure the goodness of the schedule. Let us assume that the makespan values of certain schedules are, for example: 28, 7 or 1961. The question is: Are these schedules good or not? Which one is the best? In comparing between two or more schedules, makespan value may not give a clear picture for the evaluation. Its value usually can not provide an accurate meaning unless if it is estimated relative to another value as a reference value in order to create a new quantity such as a ratio or coefficient.

In this research, a new proposed quantity is used to measure the quality of FMS scheduling system called Excess time coefficient. Later on this quantity will be discussed.

1.3 The objectives

As it was mentioned, this research deals with the multi-objective FMS scheduling problem. Specifically, the objectives to be achieved are: to improve the productivity of the system and to obtain effective delivery reliability.

1.3.1 Makespan Productivity

Generally, Productivity is defined as the ratio of output to input of the production system. Productivity is expressed as: [68] (Appendix B.1).

$$Productivity = \frac{Output}{Input} \tag{1}$$

Where: output refers to the total production quantity which is the total number of parts to be produced from the system. Input refers to the amount of resources used.

In FMS, productivity is used as an indicator of how efficiently the machine tools or groups of FMS are utilized. One of the primary responsibilities of the manufacturing engineer is to improve the productivity of the available machine tools.

In this research, makespan productivity as a new criterion (unknown in the literature) is proposed to measure the scheduling efficiency of machine groups of FMS.
Makespan productivity can be defined as the ratio of the production quantity per unit time of the makespan.
The available time length of the schedule (makespan, t_{pr}) can be considered as the input of the system.
Makespan productivity can be compute as follows:
Let n_j be the number of products of job j to be produced.
We assume that all part type of the jobs have same weigh as a fair consideration.

The production quantity is computed by $n = \sum_{j=1}^{J} n_j$ (2)

According to Equation (1), the makespan productivity,

$$\eta = \frac{n}{t_{pr}} \text{ [part/ unit time]} \quad (3)$$

According to Equation (3), as makespan decreases, makespan productivity increases.
At minimum makespan, t^*_{pr}, makespan productivity is maximum which is proposed to name as Highest Makespan Productivity (HMP). It is computed by:

$$HMP = \eta^* = \frac{n}{t^*_{pr}} \quad (4)$$

Based on Equation (4), the generated feasible schedule which has HMP is so-called efficient (productive) schedule.

The productivity rate is computed by $\eta_r = \frac{\Delta\eta}{\eta_0} * 100$ (5)

Where, $\Delta\eta$ is the makespan productivity change, $\Delta\eta = \eta - \eta_0$
η_0, η are the initial and the current makespan productivity
The system can be evaluated by Equation (5) in order to specify whether the system had improved or not; one of the following three results can be obtained:

a. If $\eta_r = +$ → There is an improvement for the FMS performance. (Good change)
b. If $\eta_r = 0$ → There is no improvement (No change),
c. If $\eta_r = -$ → There is a failure (Bad change)

One of the objectives of this research is to improve the performance of the system and an attempt to generate an efficient schedule.

1.3.2 Makespan Utilization

Utilization is defined as the degree of available capacity of the machine(s) under certain condition (Appendix B.3).
In this research, makespan utilization as a new measure is proposed to measure the utilization of machine groups of FMS during makespan.

The processing time required to process the job j on machine m is computed by

$$p_{j,m} = \tau_{j,m} n_j \quad (6)$$

The load time required to process J jobs on machine group m is

$$L_m = \sum_{j=1}^{J} p_{j,m} \quad (7)$$

The sum of setup times S_m required to process J jobs on machine m is computed by

$$S_m = \sum_{j=1}^{J} \delta_{j,m} \tag{8}$$

The production time of the machine group m is computed by

$$t_m = L_m + S_m = \sum_{j=1}^{J} (\tau_{j,m} n_j + \delta_{j,m}) \tag{9}$$

The machine makespan utilization of one machine ρ_m is computed by:

$$\rho_m = \frac{t_m}{t_{pr}} \tag{10}$$

The sum of load times required to process all jobs in the system is determined by

$$L = \sum_{m=1}^{M} L_m = \sum_{j=1}^{J}\sum_{m=1}^{M} p_{j,m} = \sum_{m=1}^{M}\sum_{j=1}^{J} \tau_{m,j} n_j \tag{11}$$

The sum of setup times of the system is

$$\mathrm{S} = \sum_{m=1}^{M} S_m = \sum_{m=1}^{M}\sum_{j=1}^{J} \delta_{m,j} \tag{12}$$

The total production time of the system is computed by:

$$t = L + S = \sum_{m=1}^{M} t_m = \sum_{m=1}^{M}\sum_{j=1}^{J} (\tau_{m,j} n_j + \delta_{m,j}) \tag{13}$$

The system makespan utilization is determined by

$$\rho = \frac{t}{Mt_{pr}} \tag{14}$$

$$\rho = \frac{\bar{t}}{t_{pr}} \tag{15}$$

Where, $\bar{t}$ is the average production time in the system, $\bar{t} = \frac{t}{M}$

According to Equation (15), if the makespan is minimum t^*_{pr}, the system makespan utilization is maximum ρ^*.

$$\rho^* = \frac{\bar{t}}{t^*_{pr}} = \frac{t}{Mt^*_{pt}} \tag{16}$$

Let ρ_0 be the initial makespan utilization, the utilization rate ρ_r is computed by:

$$\rho_r = \frac{\Delta\rho}{\rho_0} *100 \tag{17}$$

Where, the makespan utilization change is $\Delta\rho = \rho - \rho_0$

1.3.3 Workflow Acceleration

In this research, we propose a new quantity to measure the acceleration of the production flow through FMS during makespan called workflow acceleration which is defined as the ratio of production rate to the makespan.

Since, the production rate is determined by $R = \frac{n}{p}$, $\quad p = \tau * n$

$$R = \frac{1}{\tau} \tag{18}$$

Let γ_1 be the workflow acceleration of one machine to process one job of *n* products during a given makespan. It can be computed as follow:

$$\gamma_1 = \frac{R}{t_{pr}} \quad [part/\ (unit\ time)^2] \tag{19}$$

From Equations (18, 19)

$$\gamma_1 = \frac{1}{\tau t_{pr}} = \frac{n}{\tau n\, t_{pr}} = \frac{\eta}{\tau n} \tag{20}$$

According to Equation (20), improving makespan productivity leads to acceleration of the workflow of the products.

The workflow acceleration of one machine with different production rates to process multiple jobs of multiple products in a multi-machine system is called machine workflow acceleration. It can be computed as follow:

Let $R_{j,m}$ be the production rate of machine *m* to process a set of jobs *j*.

The average production rate of the machine group *m is*

$$\overline{R}_m = \frac{\sum_{j=1}^{J} R_{j,m}}{J} \tag{21}$$

Machine workflow acceleration is determined by

$$\gamma_{j,m} = \frac{\overline{R_m}}{t_{pr}} \tag{22}$$

The average production rate of the system is computed by

$$\overline{R} = \frac{\sum_{m=1}^{M} \overline{R_m}}{M} \tag{23}$$

System workflow acceleration is determined by

$$\gamma = \frac{\overline{R}}{t_{pr}} = \frac{\sum_{m=1}^{M}\sum_{j=1}^{J} R_{j,m}}{JMt_{pr}} \tag{24}$$

According to Equation (24), maximum workflow acceleration γ^* is achieved when the makespan is minimum, t^*_{pr}

$$\gamma^* = \frac{\overline{R}}{t^*_{pr}} \tag{25}$$

Let γ_0 be the initial workflow acceleration. The acceleration rate γ_r is computed by:

$$\gamma_r = \frac{\Delta\gamma}{\gamma_0} * 100 \tag{26}$$

Where, workflow acceleration change, $\Delta\gamma = \gamma - \gamma_0$

It is remarkable to identify that for a given production system, the rates of makespan productivity, makespan utilization, and workflow acceleration are equal.

$$\eta_r = \rho_r = \gamma_r = \frac{\Delta\rho}{\rho_0} = \frac{\Delta\gamma}{\gamma_0} = \frac{\Delta\eta}{\eta_0} [\%] \tag{27}$$

It can be concluded that by decreasing the makespan of the system, the productivity, utilization and workflow acceleration implicitly improve with same rate.

1.3.4 Delivery Reliability

Many literatures such as [5] introduce the lateness as a criteria to measures the performance of the scheduling systems.

For job *j,* the lateness is computed by

$$L_j = C_j - d_j \tag{28}$$

Where, C_j , d_j is the completion time and the delivery due date of job *j*

For the whole system, the system lateness can be computed by

$$L_S = t_{pr} - d \tag{29}$$

Where, *d* is the common delivery (due) date.

According to Equation (29), there are three cases of lateness:

a. Tardiness: if the production is finished after its due date (Delay), $t_{pr} > d$, $L_S = +$

b. Earliness: if the production is finished before its due date (Earlier), $t_{pr} < d$, $L_S = -$

c. Timeliness: if the production is finished at its due date (On time), $t_{pr} = d$, $L_S = 0$

Usually, in a comparative evaluation between two or more due date schedules, lateness criterion may not provide a clear picture for the effectiveness of the system

For example, let us consider two FMS scheduling systems 1, 2 with two makespans $(t_{pr})_1 = 12\ h$ *and* $(t_{pr})_2 = 1200\ h$, respectively. The due dates of system 1 is $d_1 = 10\ h$ and the due date of system 2 is $d_2 = 1198\ h$.

The lateness of system *1,* $L_{S1} = 12 - 10 = 2\ h$

The lateness of system *2,* $L_{S2} = 1200 - 1198 = 2\ h$

The results demonstrate that the lateness of the two systems is equal, but, in fact, the effectiveness of system 2 is better than that of system 1 because the delay of system 1 *(2h)* is very long relative to *(10 h)* but comparatively the delay of system 2*(2h)* is a short delay relative to *1189 h*. Therefore, we need another criterion to give more accurate evaluation for the systems.

In this research, a new criterion called delivery reliability is proposed to measure the scheduling effectiveness of machine groups of FMS.

Delivery reliability is proposed to define as the ratio of delivery date to the makespan. It can be used as an indicator of how effectively the production time is utilized in matching the delivery date (Appendix B.4).

$$\text{Delivery reliability, } \mu = \frac{d}{t_{pr}} \tag{30}$$

For the given example, $\mu_1 = 83.3\ \%$, $\mu_2 = 99.8\ \%$. Based on these results, it is clear that the effectiveness of system 2 is better than of system 1 which corresponds to the reality. Therefore, delivery reliability can be used as a good criterion to measure the system performance. Delivery reliability could be effective or ineffective.

Effective Delivery Reliability (EDR) can be achieved:

$$\text{If} \quad d = t_{pr} \quad then \quad EDR = \mu^* = 1 \tag{31}$$

The feasible schedule which has EDR is proposed to call *effective (reliable) schedule.* If not, the schedule is ineffective. The ineffective schedule is proposed to classify into two types:

a. Ineffective lateness schedule. It is achieved if:

$$d < t_{pr}, \quad \mu < 1 \tag{32}$$

b. Ineffective earliness schedule. It is achieved if:

$$d > t_{pr}, \quad \mu > 1 \tag{33}$$

In this research we are dealing with the first type (a) ineffective lateness schedule.

1.4 Perfect Schedule

As mentioned earlier, in competitive environment, the manufacturing organizations try to attain two main objectives: improving the machine productivity and delivering the production on the promised date as possible as they can. The two objectives are named in this research as highest makespan productivity (HMP) and effective delivery reliability (EDR). These objectives may conflict to each other; the system may have effective delivery reliability with low makespan productivity.

Based on that, we can propose a new expression so-called Perfect schedule.

Perfect schedule is a schedule in which the two goals HMP and EDR are achieved. Perfect scheduling is a scheduling in which the efficient schedule and effective schedule are generated in one schedule called perfect schedule. This type of scheduling problems may solved by mathematical optimization techniques such as goal-programming or other methods.

The present research is not dealt with the perfect scheduling problem. It concerned with the efficient scheduling and effective scheduling individually.

As we mentioned that this research is a multi-machine/multi-job FMS scheduling problem and the goal is to achieve the two objectives; HMP and/or EDR.

In order to clarify the objectives to be achieved we investigate a simple flow shop scheduling problem in the following example.

Example 1

The FMS scheduling problem considered in this example can be characterized according to four basic parameters as proposed in [3]. ($\boldsymbol{\alpha_1/\alpha_2/\alpha_3/\alpha_4}$) Where, α_1: Type of the FMS system, α_2: Number of machine. α_3: Number of jobs. α_4: measure criteria. This example is characterized by ***FSS/2/2/***t_{pr} describes that the FMS is a flow shop system consisting of 2 machines (M1, M2 to process 2 jobs (A, B) using makespan criteria. The objective is to generate a schedule with highest makespan productivity and/or effective delivery reliability if $d = 2700$ *min.* Engineering data are summarized in Table 1. Figures 6, 7 illustrate the job chart and load chart, respectively.

Table 1: Engineering database of Example 1

Job	n_j	M_1 $\tau_{j,1}$	M_1 $p_{j,1}$	M_2 $\tau_{j,2}$	M_2 $p_{j,2}$	P_j
A	500	1.8	900	1.4	700	1600
B	200	5	1000	4	800	**1800**
L_m			**1900**		1500	3400

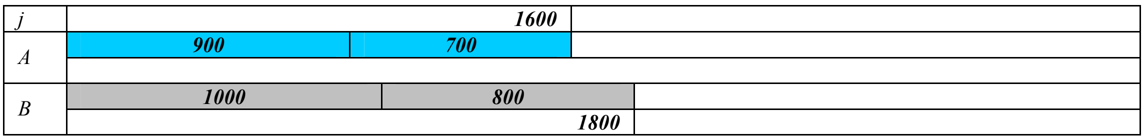


Figure 6: Job chart

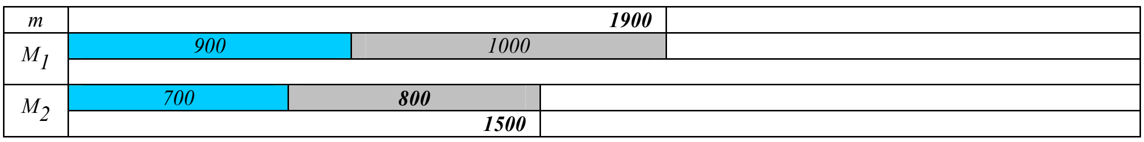


Figure 7: Load chart

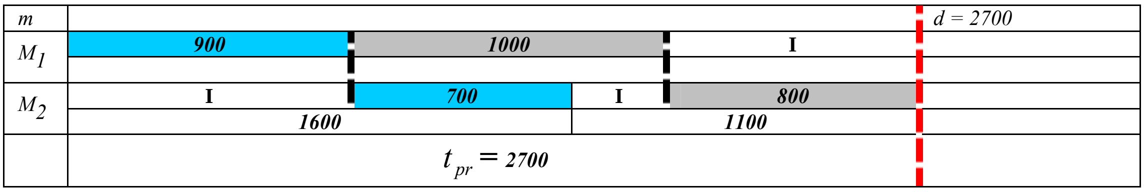


Figure 8: Gantt chart of feasible effective schedule, job A is scheduled first, I: Idle time

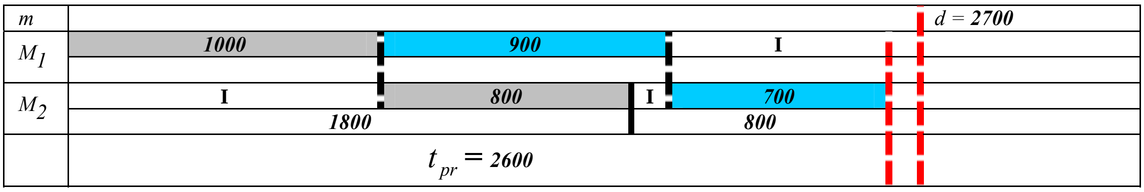


Figure 9: Gantt chart of feasible inefficient ineffective schedule, job B is scheduled first, I: Idle time

Table 2: Results of Example 1

Case	t_{pr}	ρ %	γ	η_r %	μ	**Schedule result**
1	*2700*	*62.96*	$1.59*10^{-4}$	-	***1***	*Inefficient, Effective*
2	*2600*	*65.38*	$1.65*10^{-4}$	*3.84*	*1.04*	*Inefficient, Ineffective*

For the given example using Equations (14, 24, 27, 30) the values of t_{pr}, ρ, γ, η_r, μ can be found. The results are presented in Table 2 and Figures (8, 9).

In Figure 8, job A is scheduled first. The schedule is a feasible effective schedule**.** The makespan meets the delivery date (*μ=EDR =1)*. The schedule is inefficient, the utilization is low (*62.96%).*

In Figure 9, there is a change of the job priority, job B is scheduled first. The productivity is improved by rate *(3.84%)* but the schedule is still an inefficient schedule because the utilization is low (*65.38 %)*, and the schedule is an ineffective schedule because the delivery reliability is higher than one (*μ=1.04).*

It can be concluded that in many cases the effective schedule could be inefficient. The schedule still has idle time (low utilization) and low productivity. In other cases even its productivity is improved but the schedule is still an inefficient and ineffective.

In order to achieve a schedule with higher efficiency and/or higher effectiveness, there is an effective technique in literature much more effect in improving the schedules quality than the changes of the scheduling priority rules. This technique called Lot streaming.

The definitions of the lot streaming and the definition of the problem are presented in the next chapter.

2. LOT STREAMING

2.1 Lot streaming definition

There are many different definitions of lot streaming in the literature (Appendix C) the most important definitions are given below:

"Lot streaming is a technique for accelerating the flow of batch by splitting a job lot into sub-lots, thereby creating transfer batches, and by overlapping consecutive operations. The optimal scheduling of sub-lots is a relatively difficult problem and has been solved efficiently only for simple cases" [19].

"Lot streaming or lot splitting is a concept in which a large production lot is split into smaller sublots so that its operations at successive stations can be overlapped and its progress accelerated." [23].

"Lot splitting is a shop-floor control technique that has recently received much attention by both researchers and practitioners. Lot splitting is commonly used in practice to break large orders into smaller transfer lots in an attempt to move parts more quickly through the production process." [20].

"Lot streaming is a procedure in which a large production lot is split into smaller sub-lots and the operations at the successive stages are overlapped in time." [21].

"Lot streaming is the process of splitting a production batch (or lot), comprised of several identical items, into smaller portions, often called transfer batches (or sublots), and then scheduling the sublots in an overlapping fashion on the machines, in order to accelerate the progress of orders in production and to improve the overall performance of the production system." [22].

"Lot streaming (lot sizing) is the process of creating sub-lots to move the completed portion of production sub-lots to downstream machines, however, the planning decisions become more complex when lot streaming is allowed." [24].

Lot streaming, lot sizing, and scheduling are strongly interconnected by each other; Lot streaming is termed sometimes in the literature [30] as lot sizing. But, they are definitely absolute different (Appendix C.5)

"The basic idea of lot streaming (LS) is a breaking the batches into some sub-batches then the sub-batches can be processed in an overlapping fashion." [25]

"Most scheduling models allow a job to be transferred to the next machine only when it is completed on the current machine. In a model which allows lot sizing, however, a sublot can be transferred to the next machine and processed, while other items from the same job, but of a different sublot, are processed on the current machine. We refer to this process of allowing overlaps through the creation of sublots as lot streaming. Thus, the decomposition of jobs is allowed, a solution procedure requires the creation of sublots through lot streaming, as well as the scheduling of sublots. The scheduling literature nearly always assumes that batching and lot sizing decisions are already taken. Similarly, research on batching and lot sizing seldom considers sequencing issues. There are surprisingly few publications that contain elements of both fields. From the discussion above, however, it should be clear that batching, lot sizing and scheduling are strongly interrelated. Moreover, in the advent CIM batching, lot sizing, and scheduling decision will have to be taken concurrently, i.e. they will be integrated and computer-controlled." [25]

2.2 Lot streaming point of view of the present Dissertation

It seems to us that lot streaming (LS) can be defined as a manufacturing philosophy in which two stages are performed: batch breaking and overlapping processing, subjected to certain constraints to satisfy a single or multiple objectives.

Batch breaking means "The batches are divided into a number of sub-batches"

Overlapping processing means "The sub-batches are processed in parallel through different machines".
Using lot streaming, many objectives are satisfied (many birds stoned at once)

1. It is possible to maximize productivity.
2. It is possible to maximize machine utilization.
3. It is possible to accelerate the production.
4. It is possible to attain effective delivery reliability.

In order to clarify the effectiveness of lot streaming to achieve the previous objectives we investigate a simple flow shop lot streaming scheduling problem with zero setup time in the following example.

Example 2

To illustrate the concept of lot streaming we consider the following problem which is characterized by $FSS/2/3/t_{pr}$. Each job consists of 400 products, $d= 2600$ *min*. The data are given in Table 3. We assume that the jobs have similar products. We also assume that the setup time is included in the processing time (zero setup time, $\delta =0$)

Table 3: Engineering database of Example 2

		M_1		M_2		
Job	n_j	$\tau_{j,1}$	$p_{j,1}$	$\tau_{j,2}$	$p_{j,2}$	P_j
A	400	1	400	2	800	1200
B	400	1	400	2	800	1200
C	400	1	400	2	800	1200
L_m			1200		**2400**	**3600**

In batch scheduling, the three jobs (similar products) are grouped into one batch consisting of *1200* products. The batch schedule is formulated. The batch will be processed first on M_1 with time *1200 min,* then, the sub-batches will be processed on M_2 with 2400 *min* (Figure 10).

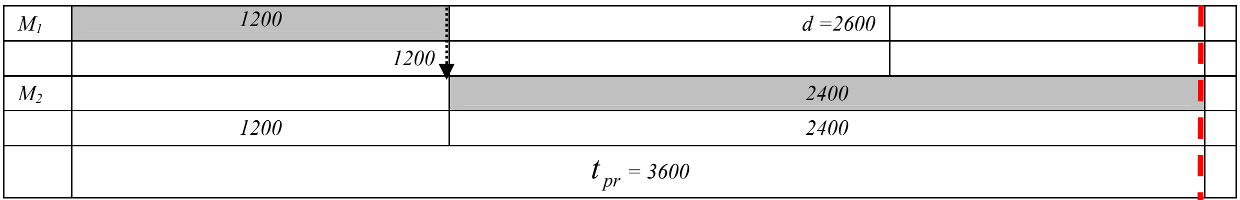


Figure 10: Gantt chart of initial batch schedule, t_{pr} = 3600 min

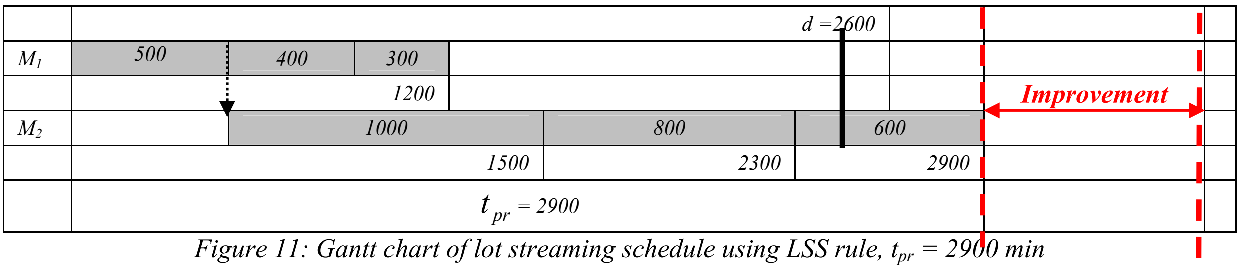


Figure 11: Gantt chart of lot streaming schedule using LSS rule, t_{pr} = 2900 min

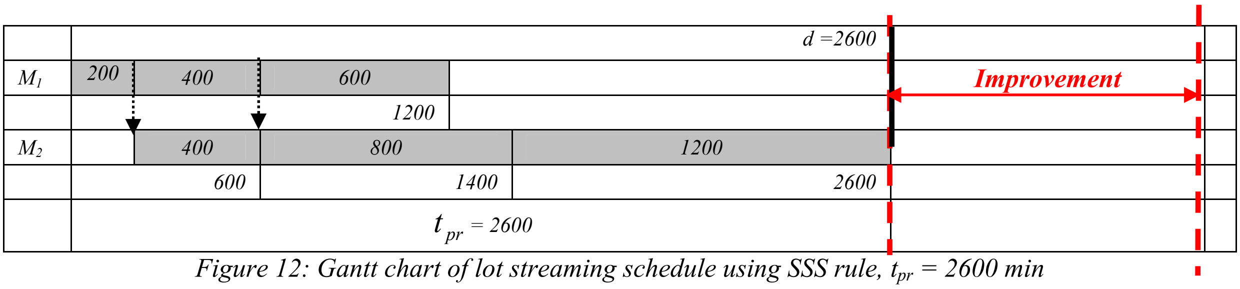


Figure 12: Gantt chart of lot streaming schedule using SSS rule, t_{pr} = 2600 min

Table 4: Results of Example 2

case	t_{pr}	ρ %	γ	η_r %	μ	*Schedule Type*
1	3600	50	$2.1*10^{-4}$	0	0.72	Feasible batch
2	2900	62.1	$2.6*10^{-4}$	24	0.90	High Efficiency & Ineffective
3	2600	69.2	$2.9*10^{-4}$	**39**	**1**	**Higher efficiency effective**

In lot streaming scheduling: the batch will be broken into several sub-batches, say; 3-sub-batches using the following proposed two rules (*unknown in literature*).

a. Largest Sub-batch Size Rule (LSS)

In this rule, the batch is divided into sub-batches in descending size mode; the largest sub-batch size is the first. For the given example, if the batch is divided into 3 sub-batches of sizes 500, 400, and 300 product/sub-batch and these sub-batches will be processed in an overlapping manner, the 3 sub-batches will be processed on machine group M_1 with times *500 min, 400 min, and 300 min,* respectively. Then, they will be processed on M_2 with times *1000 min, 800 min, and 600 min* (Figure 11**).**

b. Smallest Sub-batch Size Rule (SSS).

In this rule, the batch is divided into sub-batches of ascending size mode; the smallest sub-batch size is the first. For the given example, if the batch is divided into 3 sub-batches of sizes 200, 400, and 600 product/sub-batch and these 3 sub-batches will be processed on M_1 with times *200 min, 400 min, and 600 min,* respectively. Then, they will be processed on M_2 with times *400 min, 800 min, 1200 min* (Figure 12).

Using Gantt charts and Equations (14, 24, 27, 30), the values of t_{pr}, η, ρ, γ, η_r and μ can be found. The results are presented in Table 4.

From Table 4 and Figures (10, 11, 12) the following conclusions can be drawn:

For *case 1,* in Figure 10, the jobs are grouped into one batch. The Gantt chart of feasible batch schedule without lot streaming is illustrated. The utilization value is 50%. The schedule is inefficient schedule. The delivery reliability value is 72%, the schedule is ineffective.

For *case 2,* in Figure 11, the Gantt chart of lot streaming schedule using LSS rule is illustrated. The productivity is improved by rate 24%. The delivery reliability value is 0.9 (90%). The schedule is still an ineffective.

For *case 3,* in Figure 12, the Gantt chart of lot streaming schedule using SSS rule is illustrated. The productivity is improved by rate 39%. The delivery reliability value is 1 (100%). The schedule is an effective schedule.

It can be concluded that lot streaming approach using the SSS rule is an effective approach to improve the productivity and the delivery reliability of the systems.

For the given example, the efficient schedule with highest makespan productivity (HMP) can be achieved by dividing the batch into a number of sub-batches with one product per sub-batch. HMP value is restricted by the maximum load (for zero setup time models). The second machine group has the maximum load. (This group is named as bottleneck machine group and it will be explained later).

The minimum makespan (or HMP) can be determined as follows:

Let $Max\ t_m$ be the maximum load of machine group m.

Let $Min\ t_{pr}$ be the minimum makespan to be achieved.

$$Min\ t_{pr} = Max\ t_m + \tau_{m-1} \tag{34}$$

For the given example, $Min\ t_{pr} = 2400+1*1=2401\ min.$

HMP = 0.5 part/min. The productivity is improved by rate *51.51%.*

In the given example, the setup time is not taken into consideration.
In case of lot streaming with setup time taken into consideration, the problem is totally different. There is a trade-off optimization problem between the setup times and overlapping processes.
Because the setups have a significant effect on the solution of lot streaming problems, it should be taken into consideration very seriously.

2.3 Setup time considerations in lot streaming

In lot streaming, due to the breaking process, new setup actions appear for the sub batches. These setups can be classified into a number of types. It could be attached or detached setup time, dependent or independent-sequence setup time (Appendix C.7).

The setup time value could be estimated by maximum value estimation or sum value estimation (Appendix C.8).

In this research, the setup time is assumed as equal attached independent-sequence setup time. The setup time value is estimated by maximum value estimation.

It is remarkable that the setup time model is a deterministic model and risk analysis of the setup risk is performed (assumption). Human Reliability Analysis (HRA) and Common-Cause-Failure (CCF) are performed.

In order to investigate the effect of setup time on the lot streaming problem we establish the following example of flow shop lot streaming problem with setup time.

Example 3

The problem in this example is characterized as *FSS/2/3/t_p*. Data are summarized in Table 5, $d = 3200$ *min*, $\delta=100$ *min.* We assume that the jobs have similar products. We assume also that the first setup time of first operation of first machine group is included into the processing time. But, it can be omitted this assumption by setting the setup time individually (it has not any effect on the general solution).

Table 5: Engineering database of Example 3

Job	n_j	M_1		M_2		P_j
		$\tau_{j,1}$	$p_{j,1}$	$\tau_{j,2}$	$p_{j,2}$	
A	600	1	600	2	1200	1800
B	400	1	400	2	800	1200
C	200	1	200	2	400	600
L_m			1200		2400	**3600**

The problem can be solved as follows:

In *case 1: N=1,* batch scheduling, the jobs are grouped into a single batch of *1200* products. The initial schedule is illustrated in Gantt chart Figure 13. $t_{pr} = 3700$ *min*.

In lot streaming, we break the batch into a number of equal sub-batches randomly, for example, $N = 2, 3, 4, 6, 12$ sub-batches. The cases are given in Gantt charts in Figures (14, 15, 16, 17, 18).

Using Gantt charts and Equations (14, 24, 27, 30), the values of t_{pr} , η, ρ, η_r and μ can be found. The results are presented in Table 6.

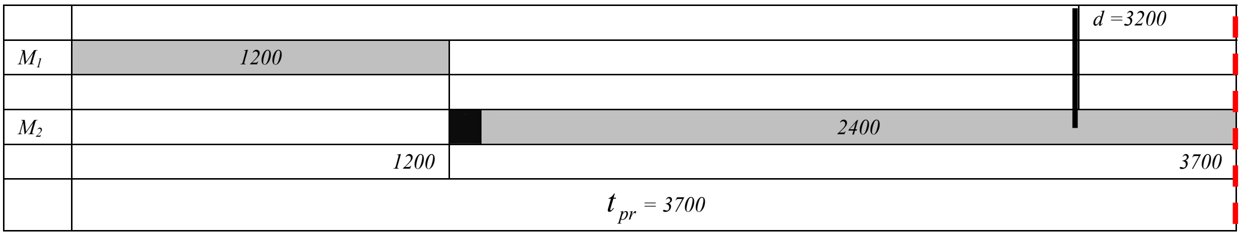


Figure 13: Gantt chart of initial batch schedule, t_{pr} = 3700 min

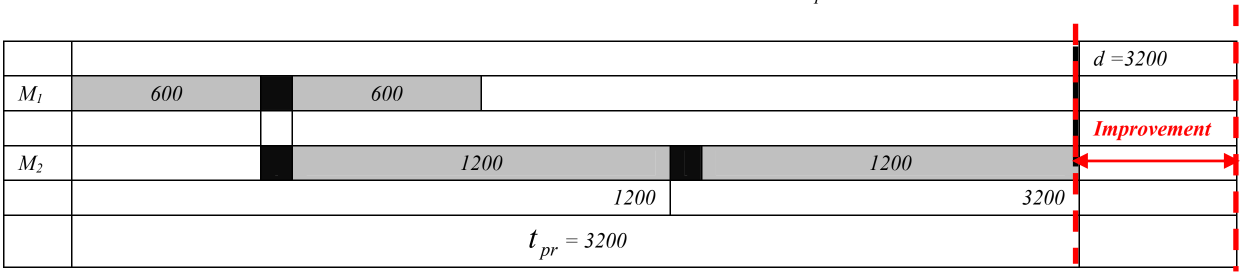


Figure 14: Gantt chart of lot streaming schedule, N=2, t_{pr} = 3200 min

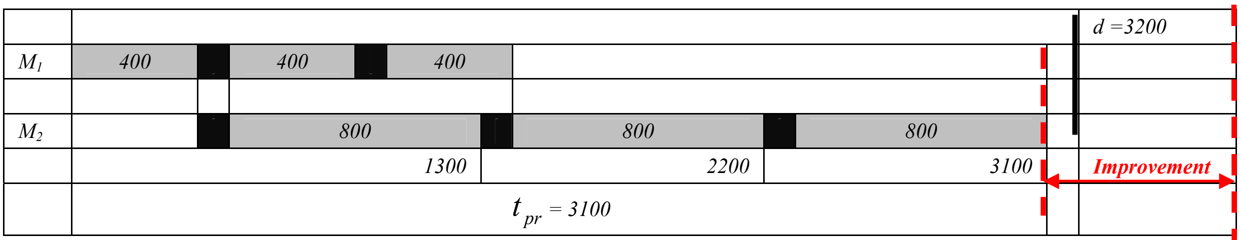


Figure 15: Gantt chart of lot streaming schedule, N=3, t_{pr} = 3100 min

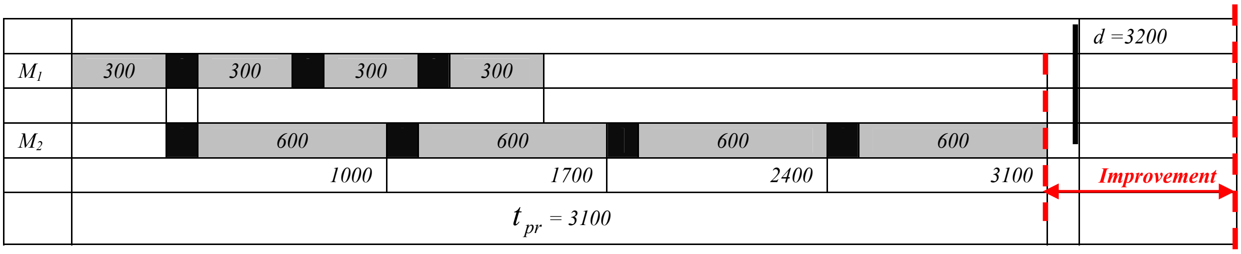


Figure 16: Gantt chart of lot streaming schedule, N=4, t_{pr} = 3100 min

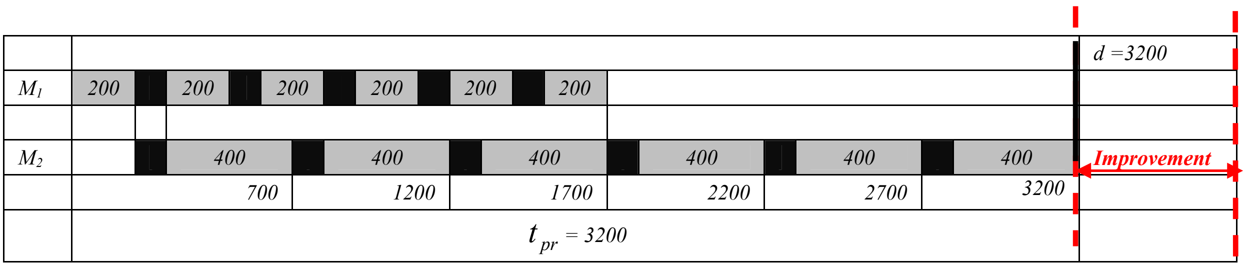


Figure 17: Gantt chart of lot streaming schedule, N=6, t_{pr} = 3200 min

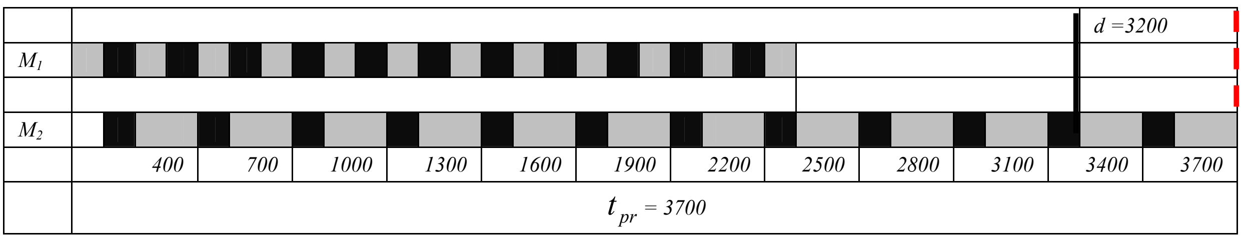


Figure 18: Gantt chart of lot streaming schedule, N=12, t_{pr} = 3700 min

Table 6: Results of Example 3

Case	N	t_{pr}	ρ %	η_r %	μ	Schedule Type
1	1	3700	48.64	**0**	0.86	Ineffective lateness
2	2	3200	56.25	+15.6	**1**	**Effective**
3	**3**	3100	58.06	+19.35	1.03	Efficient, Ineffective earliness
4	**4**	3100	58.06	+19.35	1.03	Efficient, Ineffective earliness
5	6	3200	56.25	+15.6	**1**	**Effective**
6	12	3700	48.64	**0**	0.86	Inefficient, Ineffective

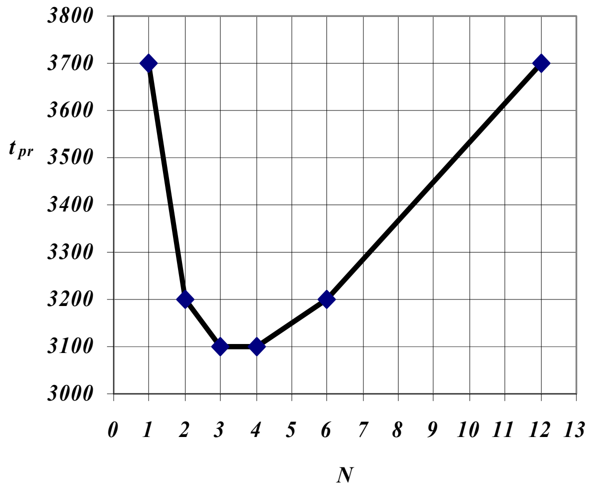


Figure 19: Makespan curve of FSS/2/1/lot streaming

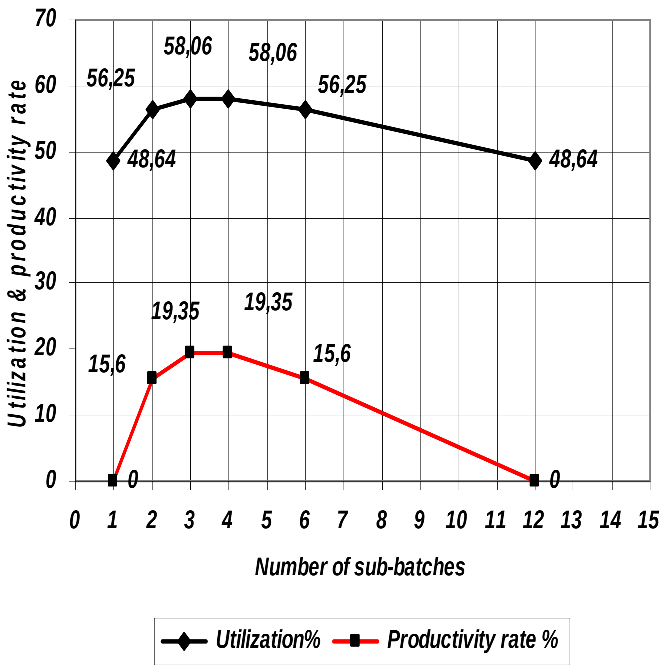


Figure 20: Curves of utilization and makespan productivity rate

From Gantt charts in Figures (13, 14, 15, 16, 17, 18), Table 6 and Figure 19, 20 the following conclusions can be made:

In case 1, *N=1,* (without lot streaming) the schedule is inefficient ineffective schedule.

In case 2, *N=2,* and case 5, *N=6,* the productivity is improved by rate *(15.6%).*The schedule is effective schedule.

In case 3, *N=3*, and case 4*, N=4* the productivity is improved by maximum rate *(19.35%).* The schedule is ineffective earliness schedule.

Based on the above, in cases 3 and 4, when the batch is divided into *N=3 or N=4,* the generated schedules are efficient but ineffective. In cases 2 and 5, when the batch is divided into *N=2 or N=6,* the generated schedules are effective but inefficient. Therefore, on selection *N*, the trade-off problem may exist between the productivity and the delivery reliability. The selection of the proper *N* is based on the strategic goal of the factory to be achieved and the constraints to be satisfied such as capacity of inventories and the demand quantity.
It is notable that for case 6, *N=12.* The productivity rate has declined down to zero ($\eta_r = 0$); the lot streaming is not effective. The schedule is inefficient ineffective schedule (as in case 1).

It can be concluded that the lot streaming is effective approach until reaching a certain number of sub-batches where it is not any more. Of course, the lot streaming approach has the highest effect on the performance of the system where the optimal number of sub-batches is determined. The question to be answered is: how to determine this optimal number for general complicated lot streaming problems of FMS consists of multi-machine to process multi-job? This research is tried to answer this question.

In order to solve the problem successfully, we have to define the problem accurately. This is; what will be introduced in the next section.

2.4 Problem definition

In Figure 20, the curves of productivity rate and utilization values are concave downward curves. The productivity rate initially is increased due to overlapping processing of sub-batches. As number of sub-batches increases, the productivity rate increases until a certain point where the optimal objective is reached and then, the productivity rate decreases due to the additional setup time of sub-batches until reaching a certain number of sub-batches. Generally, as the number of sub-batches increases the idle times decrease and in the same time, the setup times increase.
As mentioned earlier, Lot streaming consists of two processes; batch breaking and overlapping processing. In breaking process, new setup actions appear for the sub batches. As the number of sub-batches increases the setup times increase, consequently, the makespan increases and in the same time, in overlapping processing as the number of sub-batches increases the idle time decreases, consequently, the makespan decreases. So, there is a trade-off optimization problem between the time saved by overlapping processing and the extra-time caused by additional setup times of sub-batches. It is difficult to find the optimal solution even in the case of small number of machines.
The question to be answered is: what is the optimal number of sub-batches to be divided to attain the optimal objective and how to find it?
Particularly, the problem is concerned with the following questions:

• How to find the optimal number of sub-batches?
• What is the optimal number of sub-batches to achieve the HMP?
• What is the optimal number of sub-batches to achieve the EDR?

In this research, for this problem we try to give a solution as it will be outlined later.

FMS scheduling problems, in general are complicated NP hard combinatorial problems. Lot streaming makes the problems even much more complicated because of the formation of a large number of sub-batches.

There are many classifications of lot streaming according to sub-batch size, type or idle time form [Appendix C.6].

In this research, we consider the equal lot streaming type with sub-batch attached setups.

2.5 History and State-of-the-art developments

Hybrid Dynamical Approach (HDA) [55-60] is a highly effective approach for automatic lot streaming. One of the developers of the HDA was my supervisor. The proposed approach is used to solve an FMS scheduling problems. HDA realizes lot streaming as it was shown by Somlo [55]. HDA was proposed by Perkins and Kumar in 1989 [56]. General theory was developed by Matveev, Savkin [59]. Pragmatic aspects of using HDA were published in [58] and [60]. Many other works discuss the features of this approach, too. The problems with the general application of HDA are that it is valid only for high number of parts and it requires filling up auxiliary buffers. I tried to improve the applications of HDA approach and, at the same time, to solve the same problems in another way. My supervisor proposed to me to try to use the concept of a brute force approach as a new direction for low-scale problems (like 2-machine groups, 2 jobs). After a lot of work in this direction I developed a new technique. By using this technique, the performance of the system is improved. The results are a favorite. A scientific paper on this technique was accepted (and by me presented) in an international conference [62]. The technique is confirmed and named as ***Break and Build Method (BBM).***

After a lot of efforts and much time with hard research about applications of this method for complicated problems, I was surprised to discover that there were a number of researchers using the same concept of BBM, most of them, in different fields. They had published few papers. They were named the approach by different names such as lot streaming, batches transfer and lot splitting, etc. Lot streaming has been used many times under different names. The most famous name used in the literature is the lot streaming. I used the same name (Lot streaming) in this research in substitute of BBM.

The literature on lot streaming is extremely limited. For good survey of lot streaming technique, there is just one reference book called “Flow Shop Lot Streaming” published in 2007 by Professor S. C. Sarin and P. Jaiprakash [32]. This book is concerned with lot streaming for the flow shop systems only. In its chapter one (1.6, pp 20), there is a brief historical perspective, evolution of lot streaming and research in lot streaming.

Even this book has the basic knowledge of lot streaming but it is interesting to introduce a summary of the literature review as follows:

The term lot streaming was first introduced by Reiter in 1966 [33] but the work at that time was limited. In the late 1980’s to early 1990’s, lot streaming is rediscovered as a result of searches to improve the efficiency and effectiveness of production systems. The importance of lot streaming was increased when it became consistent with the JIT manufacturing philosophy of making small sub-batches which is used for Make To Order (MTO) environment as a pull system.

Another philosophy known as the OPT is strongly based on the theory of lot streaming. It is referred to as transfer batches. Until the late 1980s, lot streaming was researched independently of OPT. OPT has received little attention in the literature. Currently, it is popularly referred to as the Theory Of Constraints (TOC). Comprehensive review of lot streaming is discussed in [34].

Lot streaming has been extensively studied in academic circles as well as in industrial fields and has been shown to be an effective technique for compressing Manufacturing Lead Time (MLT).

Nowadays, the concept of lot streaming is used in FMS where large batches of similar items are to be streamed on the machines [16].

Lot streaming scheduling problem is not popular as the batch scheduling problem. Most literature of lot streaming scheduling problems is focused on the FSS problems; lot streaming in JSS problems has received much less attention. The majority of the analytical works has focused on the impact of a single lot, split into sublots, on the makespan, in two and three-machine flow-shops [e.g.28, 35, 36]. One exception is the work by Baker and Pyke [29] who proposed a computationally-efficient procedure of consistent two-sublots for the *m*-machine flow-shop.

The limited literature on more than three-machine flow-shops is due to the fact that it becomes a complicated task to determine sublot sizes that are not of equal size. However, the assumption of equal sublots seems to be a suitable solution as it has been demonstrated in the literature. Based on experiments, Kropp and Smunt [37] concluded that, for relatively small setup times, sublots of equal size (except for the first sublot) tend to be optimal. They give two other justifications for the use of equal sublots. First, a standardized container size is more convenient. Secondly, using equal sublots would most likely reduce the number of implementation errors. There are several other indications in the literature that equal sublots can, under certain operating conditions, be near-optimal. Baker and Jia [22] present results concerning the impact of equal and consistent sublots on system performance. Performance comparisons of equal and consistent sublots in a three-machine flow-shop are provided. Their results indicate that as the number of sublots increases, the ratio of the makespan of equal and consistent sublots approaches 1.00, implying that it is less attractive to use consistent sublots instead of equal sublots.

There are many papers for FSS lot streaming scheduling problems such as: Two-machine/one-job (2/1) lot streaming with setup time is given in [38]. 2-machine/ multi-job (2/J) with setup time was presented in [39, 40, 41], (3/1) was presented in [28], (3/J) in [30], multi-machine, multi-job M/J without setup time in [42, 43], J/M with setup time using dynamic programming algorithm in [44], J/M with setup time using Mixed Integer Linear Programming (MILP) in [45], M/ J with setup time using Genetic Algorithm (GA) was proposed in [46]. The analysis of batch splitting in an assembly scheduling environment was presented in [27]. Tabu Search (TS) and Simulated Annealing (SA) were proposed in [24].

Most papers on lot streaming problems consider the objective of minimizing the maximum production time (makespan) in an FSS [47, 48], and Baker [22] independently developed a conceptual framework for the problem. Trietsch and Baker [35] present a classification outline and review the most important results.

Lot streaming scheduling problem in JSS was proposed in [49]. Benefits of lot streaming in JSS Problem are presented by [23]. Lot streaming to JSS Problem using Genetic Algorithm (GA) was proposed by [50]. An effective optimization-based algorithm for JSS Problem with fixed-size transfer lots is presented in [51].
Lot streaming is not so popular as the general problem of scheduling with fixed batches but it is widely used.
An attempt to formulate the lot streaming problem from the point of view of inventory control was made by Szendrovits [52]. Dauzere-Peres and Lasserre [53, 54] proposed computation aspects for lot streaming in job-shop scheduling problems. Genetic Algorithms for the above problem were proposed by F.T.S. Chan, Wong, and P.L.Y. Chan [50]. Many other works discuss the problem, too.

In a new book on flow shop lot streaming the authors proposed to use the following solutions.

In 2-machine, single-lot, no setups lot streaming problem of continuous and consistent sub-lots, the sizes of the sublots can be shown to be geometric. The resulting solution is formulated for the no idle time between sublots cases.
Critical Path Method (CPM) has been used effectively to address the lot streaming problems. The processing of the sublots on the machines is viewed as a directed network. The critical path is the largest path from the start node to the end node.
Generally, the consideration of setup makes the problem more complicated;
For lot-attached or lot-detached setups, geometric sublot sizes are optimal for the case of continuous sublot sizes. The optimal discrete sublot sizes can be obtained by iteratively modifying the optimal continuous sublot sizes.
Linear Programming (LP) can be used for sublot-attached transfer time lot streaming problem to obtain optimal continuous sublot sizes. For lot-detached setup, the optimal lot sizes are geometric except for the first sublot. Algorithms are also available to obtain integer sublot sizes.
The geometric sublot sizes remain optimal when the lot-detached setup is involved, and the sublots are processed in a no-wait manner, i.e., each sublot is processed continuously on the machines. When the size of each sublot is required to take an integer value, the problem is difficult to solve, and a heuristic solution procedure is available that does not guarantee the optimal solution.
Johnson's algorithm can also be used to solve multiple lot/2-machine problems.

For 3-machine, single-lot, lot streaming problem, the optimal sublot sizes (continuous or discrete) can be determined by using mathematical programming.
For the case of consistent and continuous sublot sizes and no setups, the problem can be solved by analyzing the network representation.
When sublot sizes are variable and no idling among the sublots is permitted, optimal sublot sizes can be obtained by decomposing the problem into two sub-problems.
The situation of consistent sublot sizes with lot-detached/attached setups can also be analyzed by using the network representations.

The *m*-machine lot streaming problem is more difficult than its two and three machine versions. One of the first efforts in this regard is the work of Baker and Pyke [29] who developed a two-sublot-based heuristic procedure for single lot to determine continuous and consistent sublot sizes in order to minimize makespan. The discrete version of this problem is addressed by Chen and Steiner [36]. Kalir and Sarin address the same problem with sublot-attached setups and equal sublot sizes.
For the single batch problem, a linear programming formulation can be used to obtain continuous optimal sublot sizes. Integer sublot sizes can be obtained from the continuous optimal sublot sizes by using heuristic procedures.

As the number of lots increases to more than one, also the issue of sequencing the lots is addressed. Kalir and Sarin [43] presented this sequencing issue of equal-size sublots for each lot.
For the multiple lot case, the lot sequencing and sublot sizing problems are dependent on each other. A procedure is to schedule the lots by solving a Traveling Salesman Problem (TSP).
The lot streaming problem in no-wait flow shops was addressed by Sriskandarajah et al. [44]. The *m*-machine lot streaming problem can also be analyzed using network representation.
The single-lot *m*-machine lot streaming problem is solved by determining the optimal number of sublots for each lot, and then, a heuristic procedure, termed bottleneck minimal idleness heuristic, is developed to minimize the idleness caused at the bottleneck.

In this research, we try to show when the heuristic methods for general FMS scheduling may not give high-quality schedules, the goodness of schedules can significantly be improved by lot streaming.
In this research, two new methods were developed to find the optimal lot streaming in order to achieve HMP and EDR. These methods can be used in cases when the HDA may not be applied. The proposed methods are named as *Joinable Schedule Approach (JSA) and Break and Test Method (BTM) [61-67]*.
The solutions of these two methods are based on the global minimum of production time which we will be discussed in the following chapter.

3. GLOBAL MINIMUM OF PRODUCTION TIME FOR JOB SHOP SYSTEMS

3.1 Bottleneck Scheduling Approach (BSA)

In FMS, usually, there is a machine group which has maximum load time among all machine groups. This group is called bottleneck machine group.

The basic idea of Bottleneck Scheduling Approach (BSA) is that "The higher the utilization of the bottleneck, the higher utilization the whole system can be provided". Therefore, improving the performance of bottleneck will improve the performance of the whole system. Bottleneck machine group can be considered as a constraint within the system that restricts the production time of the whole system. In job shop systems, it is impossible to construct a schedule with makespan less than the load of bottleneck machine group. For this reason, bottleneck must be scheduled first.

By BSA the feasible schedule is generated relative to the bottleneck machine group in order to get certain objective(s).

Base on the above, bottleneck machine group must be productive (busy) all the time. To keep the bottleneck productive, batch size should be large. But, large batch sizes create longer production times. To solve this problem while still maintaining bottleneck to be productive, lot streaming can be used. Furthermore, the idle time of bottleneck machine group is the lowest idle times among all machine groups. Minimization of the lowest idle times is easier than of the idle time of the others (non-bottleneck) which they implicitly will be decreased.

As mentioned earlier, bottleneck machine group has the maximum of minimum production time. Therefore, minimization of minimum of maximum production time close to the maximum of minimum production time is a suitable goal in FMS scheduling problems and may result in *HMP* and/or *EDR*. But answering the question how close this goal can be approached is a very complicated problem.

The present research tries to answer this question using scheduling bottleneck approach to solve lot streaming problems of FMS scheduling as we will see later.

For large-scale systems, the minimum of maximum production time refers to the minimum makespan. The production time of bottleneck machine group is the maximum of minimum production time of the system which is called global minimum of production time.

To minimize the minimum makespan of multi-machine/multi-job FMS lot streaming problems close to the global minimum of production time as a reference value we have to follow the global determination procedure. The global determination procedure for JSS and FSS problems is different.

For job shop systems, the global minimum of production time is determined by the maximum summation of the production time and the sum of setup time (if it is considered) of certain machine group among all machine groups of the system. Clearly, it is impossible to construct a minimum makespan schedule less than this minimum.

Based on the consideration of setup time, the global minimum of production time can be classified into two models as follows:

3.2 Non-zero setup time job shop systems model

The setup times can be taken into consideration in different ways. The global minimum of production time can be determined as follows:

According to Equations (6) and (7)

The load time required to process J jobs on machine group m

$$L_m = \sum_{j=1}^{J} p_{j,m} = \sum_{j=1}^{J} \tau_{j,m} n_j \tag{35}$$

The production time of machine group m is

$$t_m = L_m + s_m\ \delta_m \qquad m = 1, 2, 3, m...M \tag{36}$$

Where, s_m , δ_m are the number of the setups and the setup time of machine group m.

According to the setup time matrix, let us suppose δ_m satisfies the following condition:

$$\delta_{\min} \le (\delta_{\min})_m \le \delta_m \le (\delta_{\max})_m \le \delta_{\max} \qquad m = 1, 2, 3, m...M \tag{37}$$

Where, $\delta_{\min}$, $\delta_{\max}$ are the minimum and maximum setup times of the system.

$(\delta_{\min})_m$, $(\delta_{\max})_m$ are the minimum and maximum setup times of machine group m.

According to the load time matrix, let us suppose L_m satisfies the following condition

$$L_{\min} \le L_m \le L_{\max} \qquad m = 1, 2, 3, m...M \tag{38}$$

Where, $L_{\min}$, $L_{\max}$ are the minimum and maximum load times of the system.

Let us select two arbitrary machine groups from the system, for example x and y.

The production time of machine group x is

$$t_x = L_x + s_x\ \delta_x \tag{39}$$

We suppose that the load time of machine group x is the maximum load.

$$L_x = L_{\max} \tag{40}$$

The production time of machine group y is

$$t_y = L_y + s_y\ \delta_y \tag{41}$$

We suppose that the setup time of machine group y is the minimum setup time.

$$\delta_y = (\delta_{\min})_y \tag{42}$$

Where, $(\delta_{\min})_y$ is the minimum setup of machine group y

According to the values of setup time (at constant number of setups ($s_x = s_y$)) one of two cases is resulted as follow:

a. Non-bottleneck setup time case

$$\text{If } t_x < t_y \text{ then } \delta_x < \delta_y \tag{43}$$

$$\text{In general, If } t_x < t_m \text{ then } \delta_x < \delta_m \qquad m = 1, 2, 3...M, m \neq x \tag{44}$$

In this case, the setup time of machine group x is called a non-bottleneck setup time and the production time of x is the minimum production time of the system.

b. Bottleneck setup time case

$$\text{If } t_x > t_y \text{ then } \delta_x > \delta_y \tag{45}$$

$$\text{In general, If } t_x > t_m \text{ then } \delta_x > \delta_m \qquad m = 1, 2, 3...M, m \neq x \tag{46}$$

In this case, the setup time of machine group x is called *bottleneck setup time* and the production time of x is called the global minimum of production time and the machine group x is called bottleneck machine group.

We assume that the bottleneck machine group has the maximum setup time and maximum load time

$$\delta_x = \delta_{\max} = \delta_b \tag{47}$$

$$L_x = L_{\max} = L_b \tag{48}$$

Let S_b be the bottleneck sum of the setup times.

$$S_b = Max\, S_m = Max \sum_{j=1}^{J} \delta_{j,m} \tag{49}$$

The global minimum of production time of the system

$$t_g = Max\ t_m = Max\ [L_m + S_m] \tag{50}$$

$$t_g = L_b + S_b \tag{51}$$

Let s_b be the maximum number of setups on the bottleneck machine group,

Since, $S_b = s_b\ \delta_b$. This usually leads to

$$t_g = L_b + s_b\ \delta_b \tag{52}$$

We highly propose to use the value determined using Relation (52). Sometimes, in stead of δ_b an estimated $\delta \le \delta_b$ can be used

The global minimum production time is

$$t_g = L_b + s_b\ \delta \tag{53}$$

3.3 Zero-setup time job shop systems model

If the setup time is not taken into consideration *($S_m = \delta_b = 0$).*

The global minimum production time is

$$t_g = L_b \tag{54}$$

3.4 Excess time coefficient of job shop systems

In this research, a new proposed quantity to measure the quality of FMS scheduling system so-called excess time coefficient which is the ratio of makespan value to the global minimum of production time. To find the value of the Excess time coefficient we have to follow the following:

The initial makespan based on the bottleneck machine group is [see Figure (21)]

$$t_{pr} = t_g + I_b \tag{55}$$

$$t_{pr} = L_b + S_b + I_b$$

Where, I_b is the last idle time of bottleneck machine group, $I_b = S_{nb} + I_0$ (56)

Where, I_0 is the net idle time (without setup times, $\delta = 0$) and S_{nb} is the non-bottleneck stream setup time during the last idle time of bottleneck.

$$0 \le S_{nb} \le [\sum_{m=1}^{M} \delta_m] - \delta_b \tag{57}$$

If we use equal setup times, and s_{nb} is the number of setups of the last operations of non-bottleneck machine group. Then, $s_{nb} \le M-1$ (58)

So, the non-bottleneck stream setup time is $S_{nb} \le s_{nb}\delta \le (M-1)\ \delta$ (59)

The last idle time of bottleneck machine group $I_b = s_{nb}\delta + I_0$ (60)

The value of s_{nb} usually is a very small relative to bottleneck sum of setup time and in some cases it is not exist as we will see later, $s_{nb} = S_{nb} = 0,\ I_b = I_0$

For zero setup time models, the makespan is $t_{pr} = t_g + I_0$ (61)

For non-zero setup models, the makespan becomes

$$t_{pr} = L_b + S_b + S_{nb} + I_0 \tag{62}$$

In general, the initial feasible makespan of job shop systems is

$$t_{pr} = Max \sum_{j=1}^{J} \tau_{j,m,k} n_j + [\sum_{m=1}^{M} \delta_m] - \delta_b + [Max \sum_{j=1}^{J} \delta_{j,m}] + I_0 \tag{63}$$

The Excess time coefficient can be found by:

For zero-setup time job shop systems model, $C_{r0} = \frac{t_{pr}}{L_b}$ (64)

For non zero-setup time job shop systems model $C_r = \frac{t_{pr}}{t_g}$ (65)

We can formulate the planning goal as to construct a schedule which satisfy the following quality requirement condition:

$$1 \leq C_r \leq C_{\max} \leq \Omega \quad (66)$$

Where, $C_{\max}$ is the maximum excess time coefficient. Its value is chosen from the planning range permission of general excess time coefficient, Ω.

According to the above conditions (66) and based on practical experience, for example: we propose to classify the real schedule into three types according to the goodness parameter, for example: ($C_{\max}$ *=1.1, 1.15, 1.2*) as follow:

a) *High quality schedule,* if $C_r = 1.1$
b) *Medium quality schedule,* if $C_r = 1.15$
c) *Low quality schedule,* if $C_r = 1.2$

One can recognize that these values seem very much satisfy practical goals. If the above condition is satisfied the completion time is close to the global minimum and this can be a good criterion for the systems performance. For suitable planning, the quality requirement conditions (66) should be checked. If it is not valid, the planning is not successful.

We propose to classify the schedules of JSS according to the value of C_r into two types as follows:

a. Real Schedule

In job shop systems, if the value of the excess time coefficient of the schedule is

$$C_r > 1 \quad (t_{pr} > t_g) \quad (67)$$

The schedule is called real schedule.

b. Ideal Schedule

In job shop system, if the schedule is generated with the best value of C_r which is

$$C_r = 1 \quad (t_{pr} = t_g) \quad (68)$$

The schedule is called ideal schedule.

In job shop systems; there are a number of characteristics concerning the excess time coefficient as follow:

- C_r never has a lower value than one.
- For high quality schedule C_r must be close to or equal to one. (i.e. makespan must be close to or equal to the global minimum of production time).
- For real schedule, lot streaming can be used.
- The initial feasible schedule, sometimes, is generated an ideal schedule ($C_r = 1$). In this case, the lot streaming is not needed.

Example 4

In this example, JSS consists of 5 machine-groups to produce 4 jobs. The Gantt chart and the engineering database are illustrated on the Figure 21.

For zero-setup time model**,** the global minimum value is: L_b *=280 [h].*

We emphasize here that it is not analyzed how feasible schedule is good. The maximum production time for this case is: t_{pr} *=410 [h]*.

So, $C_r = \frac{410}{280} = 1.46$

This value for excess time coefficient is not very good.

Now, let us use lot streaming by dividing the "lengths" of the operations into 2-equal sublots. The effect is given in Figure 22. It can be recognized that the maximum production time has significantly decreased. Because, t_{pr} *=345 [h]*, the excess time coefficient decreases to C_r *=1.23*. This is a much more favorable value.

Going on with this procedure to *N=3, 4, 5, 6,* etc. values, if the idle time of bottleneck machine group, I_0 *=410-280=130 [h]*. The production time becomes

$$t_{pr} = 280 + \frac{130}{N} \quad [h]$$

That is: t_{pr} *=323,3; 312,5; 306,0; 301,7;* and C_r *=1,15; 1,12; 1,09; 1,08;* etc.

In this way the production time goes closer and closer to the global minimum.

Based on the above, it is necessary that the new concept and procedure which are given above must be formulated in a general analytical approach.

In this research, a new approach is developed to generalize an analytical approach for a class of lot streaming problems. This approach is so-called "***Joinable Schedule Approach (JSA)***" which is given in next chapter.

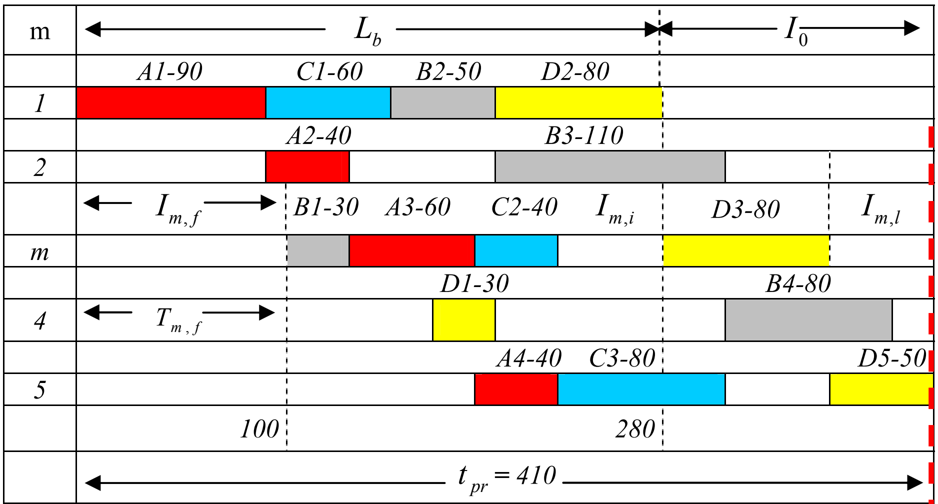


Figure 21: JSS Gantt chart of initial schedule

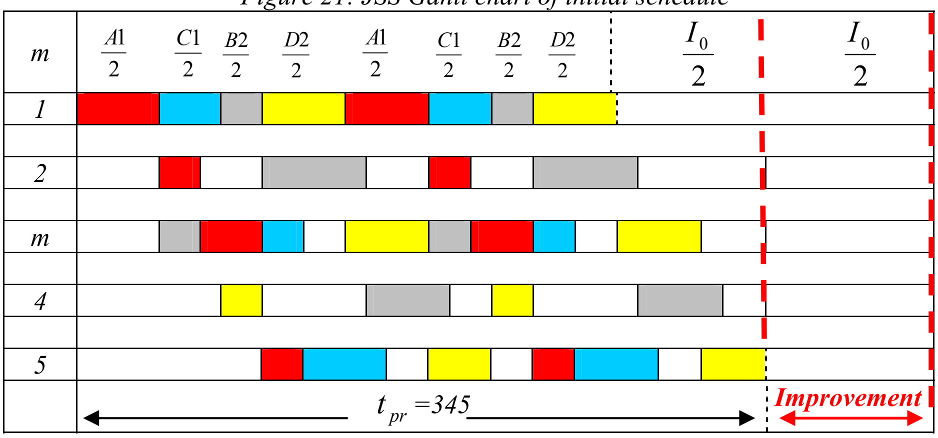


Figure 22: JSS Gantt chart of lot streaming schedule

4. THE JOINABLE SCHEDULE APPROACH

4.1 Definition

Joinable Schedule Approach (JSA) is a new analytical approach which can be used for a class formulated problems to find the optimal number of sub-batches, named optimal lot streaming. The basic idea of JSA is: "The feasible batch schedules are generated by a proper scheduling method. Then, joinability test is performed to check the possibility of joining the feasible schedule with itself until they touch each other without overlapping. If it is possible, the feasible schedule is a joinable schedule. Then, by lot streaming, the batches of feasible schedules are divided into integer number of sub-batches. The optimal (minimum) number of sub-batches can be found using the differentiation for the mathematical model of joinable schedule".

4.2 Joinable Schedule Approach Procedure

There are three phases of the procedure of JSA to determine the suitable lot streaming for FMS scheduling problem:

4.2.1 Building Phase

In this phase, a number of feasible batch schedules are generated using certain scheduling priority rules (SPR) (Appendix A.9.2). Then the best feasible schedule is selected according to the predetermined rule and objective. Gantt chart is represented, and values of some criteria such as bottleneck load time, makespan, idle time, utilization are determined. This phase is proposed to name as bottleneck scheduling phase. Later, this phase will be discussed.

In this research, two of the most important SPR are used; First In First Out rule (FIFO), Minimum Slack (MS) rule using a simulation method such as Taylor ED or LEKIN.

4.2.2 Testing Phase

In this phase, a test is proposed to specify whether the JSA method can be used or not. This test is name as *Joinability Test*. Generally, a feasible schedule is a joinable schedule if it can be joined with itself. Later on this test will be discussed.

The work of E. K. Nagy [61] on the joinable schedule approach which is based on this work [64, 65] is very important for extended research. He proposed other types of joinable schedules than it is proposed in this dissertation. The joinable schedules of the present dissertation may be interpreted us “Trivial Joinable Schedules”.

4.2.2 Solving Phase

In this phase, the optimal solution of lot streaming problem is achieved using a mathematical optimization formula. In this research, we consider the equal sub-batches and sub-batch attached setups within an analytical framework. This phase is proposed to name as mathematical solution phase. Later on the mathematical model will be formulated.

4.3. Joinability Test

Joinability test is concerned with the configuration structure of the Gantt charts of the schedule. The Gantt charts of JSS are illustrated in Figures (21, 22). We assume that the first machine group is the bottleneck.

A schedule is joinable if

a) No idle times exists in-between the operations of the bottleneck machine-group $I_{b,i} = 0$ and the first idle time $I_{b,f} = 0$.

b) No overlapping occurs of the active sections of the non-bottleneck machine-group during moving together.
Now, let us try to analyze the features of joinable schedules. The following notations are introduced:

$T_{m,f}$ - is the first time of the active sections on the machine-groups m

$T_{m,l}$ - is the last time of the active sections on the machine-groups m.

$I_{m,f}$ - is the first (front) idle time of machine group m.

$I_{m,l}$ - is the last idle time of machine group m.

According to Figure 21

$$I_{m,f} = T_{m,f} \tag{69}$$

$$I_{m,l} = t_{pr} - T_{m,l} \tag{70}$$

For zero setup time model, the bottleneck machine-group ($m = b$), the last idle time is computed by

$$I_0 = t_{pr} - L_b \tag{71}$$

4.3.1 The Condition of Joinability

If the initial feasible schedule is satisfying the following basic initial joinability condition, then, the schedule is joinable.

$$I_{m,f} + I_{m,l} \geq I_0 \qquad m = 1, 2, 3 \ldots M \tag{72}$$

Returning to the Example 4, for zero setup time models, the maximum production time when lot streaming is not applied is

$$t_{pr} = L_b + I_0 \tag{73}$$

When lot streaming is applied the maximum production time becomes

$$t_{pr}(N) = L_b + \frac{I_0}{N} \tag{74}$$

4.4 The Effect of the setup times

Now, let us consider the non-zero setup time models when the setup time between any two manufacturing operation is present (may not be neglected). We suppose that the schedules are joinable.

Let us apply JSA for JSS lot streaming problem by break the batches into, for example, 2 equal sub-batches (Figure 22).

The makespan for 2 sub-batches is

$$t_{pr}(2) = \frac{A1}{2} + \frac{C1}{2} + \frac{B2}{2} + \frac{D3}{2} + \frac{A1}{2} + \frac{C1}{2} + \frac{B2}{2} + \frac{D3}{2} + 2*4*\delta + \frac{I_0}{2} + s_{nb}\delta$$
$$= A1 + B1 + C1 + D1 + 2*s_b*\delta + \frac{I_0}{2} + s_{nb}\delta$$

$A1 + B1 + C1 + D1 = L_b$, $s_b\delta = S_b$, $s_{nb}\delta = S_{nb}$

The lot streaming makespan for 2 sub-batches is

$$t_{pr}(2) = L_b + 2S_b + \frac{I_0}{2} + S_{nb}$$

Similarly, breaking the batches into N sub-batches; the job shop lot streaming makespan function becomes

$$t_{pr}(N) = L_b + S_{nb} + N S_b + \frac{I_0}{N} \tag{75}$$

Clearly,
$$S_{nb} = I_b - I_0 \tag{76}$$

$$S_b = s_b\,\delta_b \tag{77}$$

Let us divide Equation (75) by t_g , we get

$$C_r(N) = \frac{L_b}{t_g} + \frac{S_{nb}}{t_g} + N\,\frac{S_b}{t_g} + \frac{I_0}{t_g}\,\frac{1}{N}$$

Let us introduce the following notations,

$$\beta_r = \frac{L_b}{t_g}, \qquad \varepsilon_r = \frac{S_{nb}}{t_g}, \qquad \theta_r = \frac{S_b}{t_g}, \qquad \phi_r = \frac{I_0}{t_g} \tag{78}$$

Where, β_r **,** ε_r , θ_r , ϕ_r are so-called bottleneck load coefficient, non-bottleneck stream setup time coefficient, setup relation coefficient, bottleneck idle time coefficient, respectively. The excess time coefficient becomes

$$C_r(N) = \beta_r + \varepsilon_r + N\,\theta_r + \frac{\phi_r}{N} \tag{79}$$

4.5 Optimization for Excess Time Coefficient Function

To minimize C_r we differentiate $C_r\ (N)$ *w. r. t.* N of Equation (79) and equalizing to zero.

$$\frac{\partial C_r(N)}{\partial N} = \theta_r - \frac{\phi_r}{N^2} = 0$$

The optimal number of sub-batches is

$$N^* = \sqrt{\frac{\phi_r}{\theta_r}} \tag{80}$$

The optimum excess time coefficient is determined by substitute N^* in Equation (79).

$$C_r^* = \beta_r + \varepsilon_r + 2\sqrt{\theta_r \phi_r} \tag{81}$$

It is remarkable from Equations (80, 81), the bottleneck load coefficient and non-bottleneck stream setup time coefficient have not any effect on the determination of optimal number of sub-batches but they have a significant effect on the value of the optimal excess time coefficient.

The optimum makespan can be determined as

$$t_{pr}^* = t_g\ C_r^* = L_b + S_{nb} + 2\sqrt{S_b I_0} \tag{82}$$

It is remarkable that the optimal number of sub-batches can be determined also by differentiate $t_{pr}(N)$ *w. r. t.* N of Equation (75) and equalizing to zero.

The Equations (79, 80, 81, 82) give an opportunity to investigate a number of values such as the range of the lot streaming effective, the effect of the setup time on the quality of scheduling, the optimal value of setup time, the optimal sub-batch size, the optimal bottleneck utilization, and the lot streaming efficiency.

4.6 Lot Streaming Effective Range

It is very easy to estimate the improvement value achieved by JSA method. We can determine the range of the productivity improvement caused by the effective of lot streaming. This improvement value can be determined by the proposed term so-called "*Excess time coefficient decrease,* ΔC_r *"* as follow:

If the initial excess time coefficient is C_r

$$C_r = \beta_r + \varepsilon_r + \theta_r + \phi_r \tag{83}$$

From Equations (79, 83), the Excess time coefficient decrease is computed by:

$$\Delta C_r = C_r - C_r(N) \tag{84}$$

$$= -\,\theta_r\,(N\text{-}1) + \frac{\phi_r}{N}(N\text{-}1)$$

$$\Delta C_r = (N\text{-}1)\left(\frac{\phi_r}{N} - \theta_r\right)$$

The value of the productivity improvement can be achieved if $\Delta C_r > 0$

$$(N\text{-}1) > 0 \rightarrow N > 1 \tag{85}$$

and

$$(\frac{\phi_r}{N} - \theta_r) > 0 \rightarrow N < N^{*2} \tag{86}$$

From inequalities statements (85, 86), the effect of lot streaming can be realized in the so-called "*Lot streaming effective range*" where,

$$1 < N < N^{*2} \tag{87}$$

There is no improvement ($\Delta C_r = 0$) by lot streaming if $N \geq N^{*2}$ (88)

Relation (87, 88) clearly demonstrates that the effect of lot streaming can be achieved when N is lower than the square of the optimal value of number of sub-batches N^{*2}. There is disadvantage or useless if we use lot streaming for $N \geq N^{*2}$.

It is remarkable that due to the lot streaming effect, the decrease in the makespan can be computed by so-called *makespan improvement value*, Δt_{pr}, where

$$\Delta t_{pr} = t_{pr} - t_{pr}(N) \tag{89}$$

4.7 Lot Streaming Coefficient

To define the lot streaming coefficient, Let us introduce the following notations:

$S_{b\,LS}$ is so-called *Lot streaming setup time,* $S_{b\,LS} = N\ S_b$ (90)

$I_{b\,LS}$ is so-called the *Overlapping time,* $I_{b\,LS} = \frac{I_0}{N}$ (91)

T_{LS} is so-called the *Lot streaming time,* where

$$T_{LS} = t_{pr}(N) - L_b - S_{nb} \tag{92}$$

$$= S_{b\,LS} + I_{b\,LS} \tag{93}$$

By substituting Equations (90, 91) into Equation (93), we get:

$$T_{LS} = N\ S_b + \frac{I_0}{N} \tag{94}$$

Let us divide Equations (92, 94) by t_g

$$\zeta(N) = C_r(N) - \beta_r - \varepsilon_r \tag{95}$$

$$= N\ \theta_r + \frac{\phi_r}{N} \tag{96}$$

Where, $\zeta(N)$ is so-called *lot streaming coefficient,* $\zeta(N) = \frac{T_{LS}(N)}{t_g}$ (97)

By differentiating ζ in Equation (96) *w. r. t. N*, and equalizing to zero

$$N^* = \sqrt{\frac{\phi_r}{\theta_r}} \tag{98}$$

Formula (98) is similar to the Formula (80)

By substituting N^* into Equation (96) we get

The *optimal lot streaming coefficient* is determined by $\zeta^* = 2\sqrt{\theta_r \phi_r}$ (99)

Let us introduce the following notations:

- $\theta_{r,LS}$ is the *lot streaming setup coefficient,* $\theta_{r,LS} = N\ \theta_r$ (100)

By substituting N^* into Equation (100), the optimal lot streaming setup coefficient is

$$\theta^*_{r,LS} = \sqrt{\theta_r \phi_r} \tag{101}$$

- $\phi_{r,LS}$ is the *overlapping coefficient,* $\phi_{r,LS} = \frac{\phi_r}{N}$ (102)

By substituting N^* into Equation (102), the optimal overlapping coefficient is

$$\phi^*_{r,LS} = \sqrt{\theta_r \phi_r} \quad (103)$$

From Equations (101, 102) we can get $\theta^*_{r,LS} = \phi^*_{r,LS} = \sqrt{\theta_r \phi_r}$ (104)

The optimal lot streaming coefficient is $\zeta^* = \theta^*_{r,LS} + \phi^*_{r,LS} = 2\sqrt{\theta_r \phi_r}$ (105)

The optimal lot streaming time $T^*_{LS} = 2\sqrt{I_b S_b}$ (106)

It can be concluded that at the optimal number of sub-batches for joinable schedule,

- The lot streaming time (or coefficient) is minimum
- The lot streaming setup time (or coefficient) is optimal
- The overlapping time (or coefficient) is optimal.
- Optimal lot streaming setup coefficient is equal to optimal overlapping coefficient.

Equation (105) gives an opportunity to investigate the effect of setup time on the quality of scheduling

Indeed, let us assume $I_0 = 0.5\, t_g$, then $\phi_r = 0.5$

From Equation (105), we get $\theta_r = \frac{1}{2} \zeta^{*2}$

Let us investigate the effect of lot streaming coefficient (ζ <1), for setup relation coefficient, θ_r for example, ζ = 0.15, 0.1 or 0.2, then

$\theta_r(\zeta = 0.15) = 0.01125$ or $\theta_r(\zeta = 0.1) = 0.005$ and $\theta_r(\zeta = 0.2) = 0.02$

So, if the relation of sum of setup times to the global minimum of production time is less than one hundreds, the processes obtained by using the JSA seem favorable.

The optimal setup relation coefficient is determined by

$$\theta^*_r \le \frac{1}{4}\left(\frac{\zeta^{*2}}{\phi_r}\right) \quad (107)$$

For suitable production planning and production system design, the conditions (107) should be provided. In practical cases the fulfillment of this condition should be checked. If it is not valid, the planning is not successful.

4.8 The Effect of the Setup Relation Coefficient

The *maximum excess time coefficient decrease* ΔC^*_r is computed by

$$\Delta C^*_r = C_r - C^*_r \quad (108)$$

$$= \theta_r + \phi_r - 2\sqrt{\theta_r \phi_r}$$

$$\Delta C^*_r = (\sqrt{\phi_r} - \sqrt{\theta_r})^2 \quad (109)$$

By Relation (109) it can be estimated how the setup relation coefficient values affect the opportunity of improving the system performance. For bigger ϕ_r values significant improvement occurs. This improvement is restricted by the relation $\phi_r > \theta_r$. Lot streaming is always worth to apply if the following relation is satisfied

$$\phi_r >> \theta_r \quad (110)$$

The lot streaming is not needed if the ϕ_r values for some feasible schedules are small.

$$\text{If} \quad \Delta C^*_r = 0 \;\rightarrow\; \phi_r = \theta_r \quad (111)$$

The lot streaming is impossible to apply if there is no idle time on the bottleneck machine group and the objective is to minimize C_r.

$$\phi_r = 0 \quad (I_b = I_0 = 0) \quad (112)$$

This case may be occurring when the maximum production time of the initial feasible is equal to the global minimum of the production time.
$t_{pr} = t_g$ for non-zero setup time models, $t_{pr} = L_b$ for zero setup time models
It is concluded that although lot streaming is an effective approach to improve the quality of the schedule, in some cases, lot streaming is not effective or it is not necessary to apply. This may be happen when the makespan is equal to the global minimum of production time as we will see later.

4.9 Optimal Bottleneck Utilization

Bottleneck utilization is the largest ratio of global minimum of production time to the makespan. The bottleneck utilization ρ_b is computed by

$$\rho_b = \frac{t_g}{t_{pr}} \tag{113}$$

From Equations (65) we can get an interesting relation between the excess time coefficient and the bottleneck utilization. It can be expressed as follows:

$$C_r = \frac{1}{\rho_b} \tag{114}$$

The optimal bottleneck utilization

$$\rho_b^* = \frac{1}{C_r^*} \tag{115}$$

C_r^* is the multiplicative inverse (reciprocal) of ρ_b^*. From Formulas (14, 16)

$$\rho_b^* \geq \rho_b \geq \rho^* \geq \rho \tag{116}$$

Condition (116) specifies the range of minimum and maximum values of the utilization of the machine group and the whole system during the scheduling length.

4.10 N-Integer

In the discrete manufacturing, when dividing the batches into N sub-batches it might happen that the items in sub-batches are continuous with real number. The real value of N^* have not any physical meaning. Because only the integer values of N have real meaning; real values must be changed to integer value. Usually, the real values obtained for continuous sub-batches sizes are very close to these for discrete. Frequently, the discrete lot streaming can be solved by simple rounding. Applying the closest integer number does not change significantly the obtained results. The results should numerically be estimated and the better value applied which does not affect the goodness of the results significantly. It is a very motivating development that the real number of N^* could lead to interesting results concerning the setup times.
The integer values should be computed then substituted into Equation (75, 79).

$$Real\ N^* = \begin{cases} N_1^* = Int(N^*) \\ or \\ N_2^* = Int(N^*) + 1 \end{cases} \tag{117}$$

The one of two which gives the minimum is the optimal solution. So, the optimal $(C_r^*)_i$ value of integer N^* is

$$(C_r^*)_i = \underset{i=1,2}{Min}\left\{\beta_r + \varepsilon_r + N_i^*\theta_r + \frac{\phi_r}{N_i^*}\right\} \tag{118}$$

Accordingly, the optimal makespan is $(t_{pr}^*)_i = t_g (C_r^*)_i$ (119)

4.11 Lot Streaming Efficiency (LSE)

The lot streaming efficiency is determined by $LSE = \frac{\Delta C_r}{C_r(N)} *100\ [\%]$ (120)

The maximum lot streaming efficiency is $LSE_{\max} = \frac{\Delta C_r^*}{C_r^*} *100\ [\%]$ (121)

4.12 Sub-batch Size Determination

The sub-batch size of job j (Z_j) can be computed by $Z_j = \frac{n_j}{N_j}$ (122)

The optimal sub-batch size of job j is determined by $Z_j^* = n_j \sqrt{\frac{S_b}{I_b}}$ (123)

Example 5

Let us return to Example 4. As it can be recognized, the numbers of setups are $s_b = 4$, $s_{nb} = 1$. Let consider $\delta_b = \delta_{pr} = \delta_{\max} = 1\ [h]$.

$N^* = \sqrt{\frac{130}{4}} = 5.7 \rightarrow N_1^* = 5;\ N_2^* = 6$. Then, $t^*_{pr1} = 280+1+5*4+\frac{130}{5} = 327\ [h]$,

Similarly, $t^*_{pr2} = 326.7\ [h]$

One can recognize that these values are very close to each other.

At the given setup times the optimal excess time coefficient is $C_r^* = 1.17$

If the setup times have lower values the situation is better. For example at $\delta_b = \delta_{pr} = \delta_{\max} = 0.5\ [h]$ one gets: $N^* = 8.06 \rightarrow N_1^* = 8 \rightarrow t^*_{pr1} = 312.75\ [h] \rightarrow C_r^* = 1.12$.

At the decrease of setup times the parameters of planning become better and better.

Example 6

In Table 7, engineering database of $JSS/3/3/t_{pr}$ lot streaming problem is given.
$\delta = 10\ min$. Three jobs A, B, C ($j = 1, 2, 3$) to be processed on three machine groups.

Table 7: Engineering database of Example 6

Job j	n_j	Machine group m: 1 $\tau_{j,1}$	$p_{j,1}$	k	2 $\tau_{j,2}$	$p_{j,2}$	k	3 $\tau_{j,3}$	$p_{j,3}$	k	P_j
A	60	1	60	2	1	60	1	1	60	3	180
B	60	3	180	1	2	120	3	2	120	2	420
C	60	1	60	3	1	60	2	1	60	1	180
L_m			300			240			240		780

The solution of the given FMS/JSS batch scheduling problem is as follow:

We assume that the best feasible schedule is achieved by FIFO rule (Figure 23). The bottleneck machine index is 1. Using the joinability test for the given example, the schedule is not joinable schedule. For the demonstration of the application of JSA for non-joinable schedule, we need to transform the initial schedule to a joinable schedule. We get "joinable schedule" from the schedule in Figure 23 if we shift the processing times A1, C2 on machine group 2 and, C1 on machine group 3 to the right. A "joinable schedule" is given in Figure 24

Let us break the batches into 2 sub-batches. The Gantt chart is shown in Figure 25. Using JSA mathematical model, we get the following results:

$$L_b = 300\ min,\ \delta = 10\ min,\ S_b = 30\ min,\ t_g = 330\ min,\ t_{pr} = 450\ min,$$

$$S_{nb} = 0 \rightarrow I_b = I_0 = t_{pr} - t_g = 450-330 = 120\ min$$

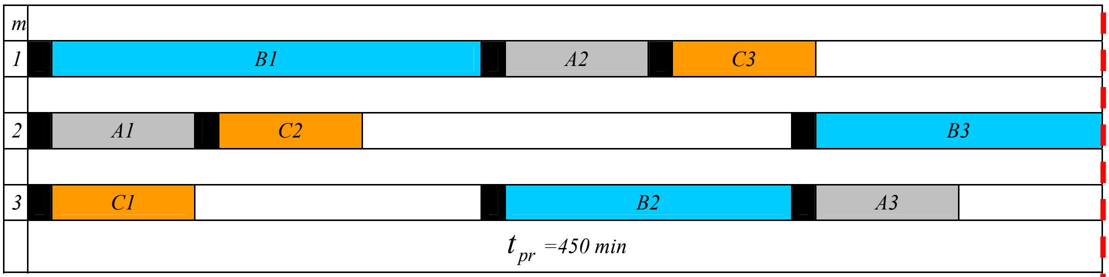


Figure 23: JSS/ FIFO/3/3 Gantt chart of initial schedule

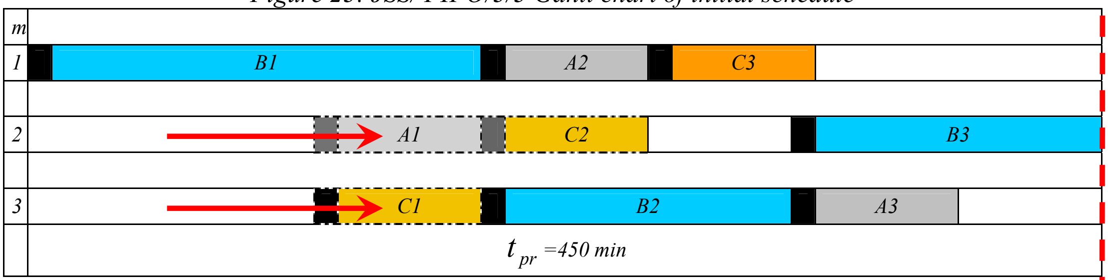


Figure 24: JSS/ FIFO/3/3 Gantt chart of modified schedule

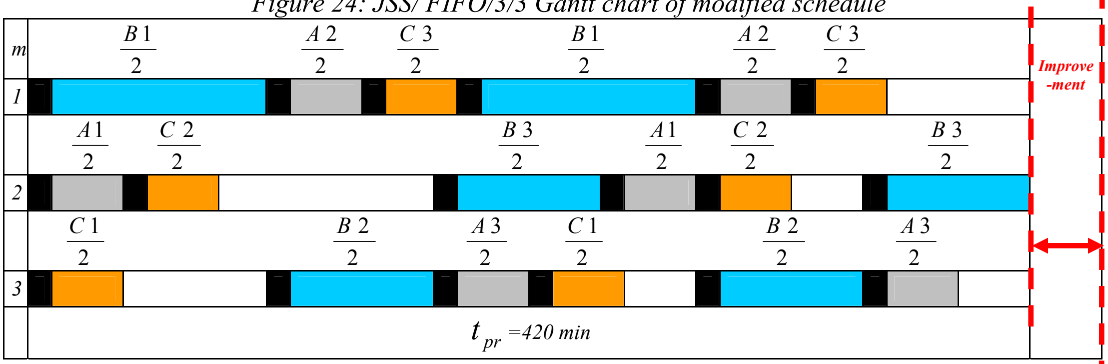


Figure 25: JSS/FIFO/3/3 Gantt chart of lot streaming schedule, N=2

Using Equations (78, 79), $C_r = 1.363636,\ \beta_r = 0.9090,\ \varepsilon_r = 0,\ \theta_r = 0.0909,\ \ \phi_r = 0.3636$

Using Equations (80, 81, 82), $N^* = 2$ *sub-batches,* $C_r^* = 1.272727,\ t_{pr}^* = 420$ *min*

Using Equations (108,121,123) $\Delta C_r^* = 0.0909,\ LSE_{\max} = +7.143\%,\ Z_j^* = 30$ *pc/sub-batch*

The maximum productivity is improved by rate *7.143%* with the optimal size 30 product per sub-batch for each job.

It can be concluded that even the initial schedule is not joinable, the productivity of JSS lot streaming problem can be improved by JSA. The schedule can be transformed into joinable by some modification and then we can apply the JSA procedure.

4.13 Conclusions of JSA for FMS Lot Streaming of Job Shop Systems

When looking for the solution of scheduling problems for an FMS problem we can use a number of different approaches and devices (standard software). In some cases (depending on the database) we can find at least one feasible schedule which gives very close to the global minimum of production time. If the schedule is suitable from other point of views, too, the planning may be finished. If the production time do not satisfy the quality requirements, and one can find joinable schedules, JSA can be used to find suitable schedules. The success of the JSA methodology depends on the values of the setup times. Of course, when the setup time values are big the JSA may not be used. What is big and what is small is always an open question. We have the conjuncture that when the sum of setup times on the bottleneck machine is less than a hundreds of the production time on this machine-group the given approach gives good results. The reasoning which is given above highly supports this conjuncture.

Now, let as consider how the idea could be extended for flow shop problems.

5. GLOBAL MINIMUM OF PRODUCTION TIME OF FLOW SHOP SYSTEMS

The difference between JSS and FSS problems, from the point of view of the global minimum of production time, is that for JSS problems the production time on any of the machine groups may give the global minimum of the production time for the overall system. The situation for FSS problems is totally different. To determine the global minimum the production time for the system, not only the production time of the bottleneck machine group should be considered but the production time of the upstream and downstream processes of the bottleneck should be included.

We remark that when the first or last machine-groups are bottleneck the problems may be solved similarly to job shop ones.

The global minimum of production time can be determined for two models (just like in JSS problem) based on the consideration of setup time.

5.1 Non-Zero Setup Time Flow Shop Systems Model

In the case of JSS models, it was possible, at the analysis of global minimums of production time, to separate the problems considering individual machine groups. In FSS case it is impossible.

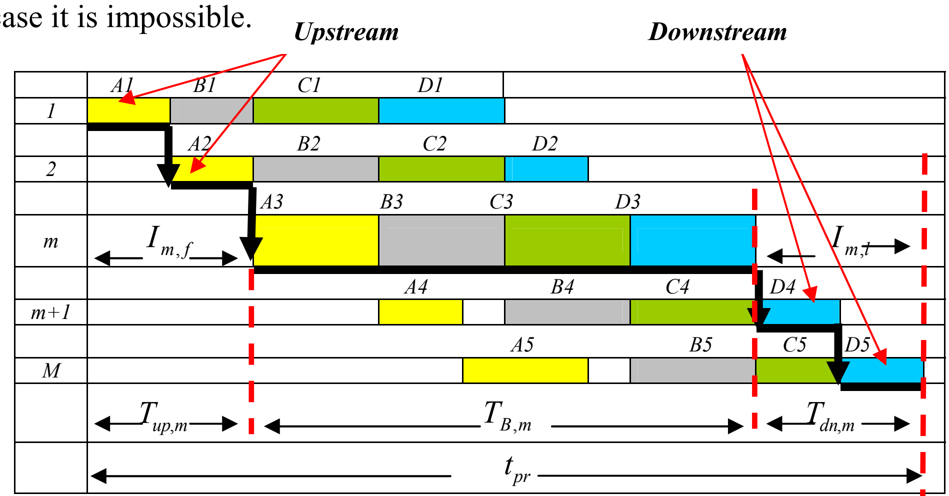


Figure 26: FSS Gantt chart with upstream, body and downstream production times

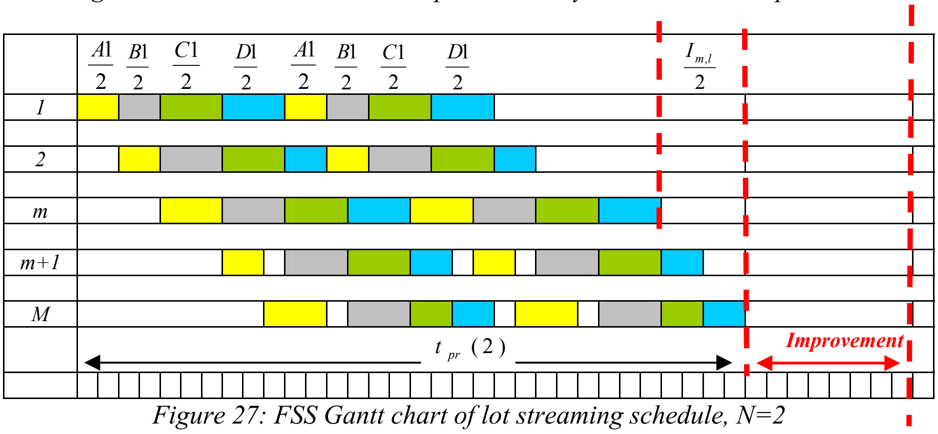


Figure 27: FSS Gantt chart of lot streaming schedule, N=2

To determine the global minimum of production time of the system we have to investigate the path production time for each machine group. The path where the global minimum of production time existing is called critical production path. Critical path because any operation in this path is delayed; there will be a corresponding delay

in the makespan of the whole system. This path restricts the minimum of production time of the system. We can find the global minimum of production time as follow:
Concerning the setup time, we will not analyze the different variants like we have done in job shop case.
It is remarkable that the setup time of a given machine group is a properly chosen value between the minimum and maximum values of setup times.

$$\delta_{\min} \le \delta \le \delta_{\max} \tag{124}$$

For simplify, in flow shop systems, usually the number and the values of setup times required to process all jobs on all machine groups are equal and constant.

$$S_{j,m} = S_m \qquad m=1,\ 2,\ 3,\ldots\ m\ldots M \tag{125}$$

Let us introduce the following definitions:
Path production time ($t_{m,F}$) is the production time of the processes along the production path of certain machine group. It consists of: upstream production time, body production time and downstream production time (Black bold line in Figure 26).

- *Upstream production time* ($T_{up,m}$) is the production times of processing the first job before a given machine group *m*.
- *Body production time* ($T_{B,m}$) is the production time of a given machine group *m*.
- *Downstream production time* ($T_{dn,m}$) is the production time of processing the last job after a given machine group *m*. So,

The path production time of machine group *m* (see Figure 26) is computed by.

$$t_{m,F} = T_{up,m} + T_{B,m} + T_{dn,m} \tag{126}$$

- The upstream production time is computed by

$$T_{up,m} = \sum_{m=1}^{m-1} (\tau_{1,m}\, n_1) + S_{up,m} = \sum_{m=1}^{m-1} p_{1,m} + \delta(m-1) = L_{up,m} + \delta(m-1) \tag{127}$$

Where, $L_{up,m} = \sum_{m=1}^{m-1} p_{1,m}$, $S_{up,m} = \delta(m-1)$ are the upstream load time and upstream setup time of machine group *m*.

- The body production time is computed by

$$T_{B,m} = \sum_{j=1}^{J} (\tau_{j,m}\, n_j) + S_{B,m} = \sum_{j=1}^{J} p_{j,m} + s_m\, \delta = L_{B,m} + s_m\, \delta \tag{128}$$

Where, $L_{B,m} = \sum_{j=1}^{J} p_{j,m}$, $S_{B,m} = s_m\, \delta$ are the body load time and body setup time of machine group *m*.

- The downstream production time is computed by

$$T_{dn,m} = \sum_{m=m+1}^{M} (\tau_{J,m}\, n_J) + S_{dn,m} = \sum_{m=m+1}^{M} p_{J,m} + \delta(M-m) = L_{dn,m} + \delta(M-m) \tag{129}$$

Where, $L_{dn,m} = \sum_{m=m+1}^{M} p_{J,m}$, $S_{dn,m} = \delta(M-m)$ are the downstream load time and downstream setup time of machine group *m*.

From Equations (127, 128, 129), the path production time is

$$t_{m,F} = L_{up,m} + \delta(m-1) + L_{B,m} + s_m\, \delta + L_{dn,m} + \delta(M-m) \tag{130}$$

Let us introduce $t_{g,F}$ be the global minimum of production time of flow shop systems.

It is remarkable that the body production time does not depend on the order of the jobs, but the upstream and downstream production times depend on it. In order to get the global minimum of production time we have to investigate the maximum body production time and the minimum values of the sum of the upstream and downstream production times. So, the global minimum of production time is determined as follows:

$$t_{g,F} = Min\,T_{up,m} + Max\ T_{B,m} + Min\,T_{dn,m} \tag{131}$$

Let us introduce the following notations:

- $T_{up,g} = Min\,T_{up,m} = Min\ [L_{up,m} + \delta(m-1)] = L_{up,g} + \delta(m_b - 1)$ (132)

  Where, $T_{up,g}$ is so-called bottleneck upstream production time, $m = m_b$

  $L_{up,g} = Min\,L_{up,m}$ is so-called bottleneck upstream load time

- $T_{B,g} = Max\ T_{B,m} = Max\ (L_{B,m} + s_m\ \delta) = L_b + s_b\ \delta$ (133)

  Where, $T_{B,g}$ is the bottleneck production time.

  $L_b = Max\,L_{B,m}$ (bottleneck load time).

- $T_{dn,g} = Min\,T_{dn,m} = Min\ [L_{dn,m} + \delta(M - m_b)] = L_{dn,g} + \delta(M - m_b)$ (134)

  Where, $T_{dn,g}$ is so-called the bottleneck downstream production time.

  $L_{dn,g} = Min\,L_{dn,m}$ is so-called bottleneck downstream load time.

From Equations (131, 132, 133, 134), the global minimum of production time which is the critical production time can be determined as follows:

$$t_{g,F} = T_{up,g} + T_{B,g} + T_{dn,g} \tag{135}$$

$$= L_b + L_{up,g} + L_{dn,g} + \delta\,(s_b + M\text{-}1) \tag{136}$$

The global load time is

$$L_{g,F} = L_b + L_{up,g} + L_{dn,g} \tag{137}$$

The global setup time is $S_{g,F} = \delta\ (s_b + M\text{-}1)$ (138)

The bottleneck sum of setup times is $S_b = s_b\ \delta$ (139)

The non-bottleneck stream setup time $S_{nb} = (M-1)\delta$ (140)

The global minimum of production time for FSS non-zero-setup time model is

$$t_{g,F} = L_b + L_{up,g} + L_{dn,g} + S_b + S_{nb} \tag{141}$$

5.2 Zero Setup time flow shop systems model

If the setup time is not taken into consideration $S_b = S_{nb} = 0$.

The global minimum of production time is

$$t_{g,F} = L_{g,F} \tag{142}$$

5.3 Excess Time Coefficient of flow shop systems

The makespan of non-zero setup time model for FSS is

$$t_{pr} = t_{g,F} + I_{g,F} \tag{143}$$

Where, $I_{g,F}$ is the *production idle time*. It is the non-productive time of the overall production time.

The maximum production time is

$$t_{pr} = L_b + L_{up,g} + L_{dn,g} + S_b + S_{nb} + I_{g,F} \tag{144}$$

The excess time coefficient for flow shop systems is

$$C_{r,F} = \frac{t_{pr}}{t_{g,F}} \qquad 0 \langle C_{r,F} \le 1 \tag{145}$$

The value of $C_{r,F}$ describes the goodness of the schedule.

The value of $C_{r,F}$ may be less than one, in contradiction with job shop case.

According to the values of $C_{r,F}$ we can classify the real schedule into 4-types:

a. Feasible schedule if $t_{pr} \rangle t_{g,F} \rightarrow C_{r,F} \rangle 1$ (146)

b. Low quality schedule if $t_{pr} = t_{g,F} \rightarrow C_{r,F} = 1$

c. Medium quality schedule if $t_{pr} \langle t_{g.F} \rightarrow C_{r,F} \langle 1$

d. High quality schedule if $t_{pr} \langle\langle t_{g.F} \rightarrow C_{r,F} \approx 0$

5.4 Optimization of Excess Time Coefficient for flow shop systems

We suppose that the FSS initial feasible schedule in Figure 26 is joinable schedule approach the relations on are valid.

From Figure 26, we get

$$L_{up,g} = I_{g,f} \ , \quad L_{dn,g} = I_{g,l} \qquad For\ m=g \tag{147}$$

Then,

$$t_{pr} = L_b + I_{g,f} + I_{g,l} + S_b + S_{nb} + I_{g,F} \tag{148}$$

We assume that the net idle time for flow shop system, I_0 is

$$I_0 = I_{g,f} + I_{g,l} + I_{g,F} \tag{149}$$

The maximum production time of flow shop system becomes

$$t_{pr} = L_b + S_b + S_{nb} + I_0 \tag{150}$$

Equation (150) is similar to the Equation (62) of job shop system.

In Figure 27, the batches are divided into 2 equal sub-batches.

The lot streaming makespan for 2 sub-batches is

$$t_{pr}(2) = L_b + 2\,S_b + \frac{I_0}{2} + S_{nb}$$

So, the production time of lot streaming is

$$t_{pr}\,(N) = L_b + S_{nb} + N\ S_b + \frac{I_0}{N} \tag{151}$$

Equation (151) is similar to the Equation (75) of job shop system. So, the lot streaming flow shop problem can be treated similarly as the job shop problem; the equations for determining the optimal number of sublots stay the same; the estimation process of the effects of setup times is the same, too.

Generally, the equations of JSA using to solve JSS lot streaming problems can be used to solve FSS problems. The different is the determination procedure to find the values of net idle time (I_0) and the non-bottleneck stream setup time (S_{nb}). This is because of the operation characteristics on upstream and downstream production times of the bottleneck machine group have important role.

Example 7

To solve the problem of the Example 3 (*FSS/2/1/t_{pr}/ELS/DLS/NI/100 lot streaming problem)* by JSA, JSA procedure must be followed. The feasible schedule is built. Then, joinability condition is tested; the original schedule is a joinable. So,

The makespan for *N*-equal sub-batches is

$$t_{pr}\ (N) = L_b + N\,\delta + \frac{I_0}{N} \tag{152}$$

The optimal number of sub-batches

$$N^* = \sqrt{\frac{I_0}{\delta}} \tag{153}$$

From Equations (152, 153), the optimal makespan can be determined by:

$$t^*_{pr} = L_b + 2\sqrt{I_0 \delta} \tag{154}$$

To specify whether the optimal value of t_{pr} is minimum or maximum, second derivative of $\frac{\partial t_{pr}}{\partial N}$ is determined. $\frac{\partial^2 t_{pr}}{\partial N^2} = \frac{2I_0}{N^3} = +$, this indicates that N^* is the minimum value, the shape of the makespan curve is concave up (convex)

By using Equations (90, 91, 94, 75, 27), the values of lot streaming setup times, overlapping time, lot streaming time, makespan and the productivity rate can be determined for different *N= 2, 3, 4, 5, 6.* The results are given in Table 8.

The behavior of lot streaming setup times, overlapping time and lot streaming time is shown in Figure 28. The lot streaming setup time increases in straight line (linearly) as number of sub-batches increases. The overlapping time decreases nonlinearly as number of sub-batches is increased. The shape of the lot streaming time curve is concave up (convex), and lowest point in the lot streaming time curve is T^*_{LS}, and the number of the sub-batch in (*N-axis*) is N^*.

In Table 9, there are two solutions for the given problem in examples 3 and 5
The results of Example 3 are given in Tables 8 and *OLST* diagram in Figure 28.
The results of Example 5 are determined by JSA Formulas (98,106, 82).
The summary of the results of the two Examples (3, 5) are given in Table 9.
If we round the real value of $N^* = 3.4641$ to closest integer value 3 or 4 and then we use Equation (151)

For N=3, $\quad t_{pr}(3) = 2400 + 3(100) + \frac{1200}{3} = 3100 min.$

For N=4, $\quad t_{pr}(4) = 2400 + 4(100) + \frac{1200}{4} = 3100 min.$

$$t_{pr}(3) = t_{pr}(4) < t_{pr}(2) = t_{pr}(5)$$

From Table 9 to compare the results of the Example 3 and 5, they are same.

It is clear from Figure 28 that the minimum of T_{LS} occurs at the intersection of the curves of the function $S_{b\,LS}$ with $I_{b\,LS}$. Therefore, at this intersection,

$$S_{b\,LS} = I_{b\,LS} \tag{155}$$

$$N S_b = \frac{I_b}{N} \tag{156}$$

From Equation (156) we can find out $N^* = \sqrt{\frac{I_b}{S_b}}$ (157)

Table 8: Lot streaming data of Example 5

N	S_{bLS}	I_{bLS}	T_{LS}	t_{pr}	η_r %
1	100	1200	1300	3700	0
2	200	600	800	3200	+15.6
3	**300**	**400**	**700**	**3100**	**+19.35**
4	**400**	**300**	**700**	**3100**	**+19.35**
5	500	240	740	3140	+15.6
6	600	200	800	3200	0

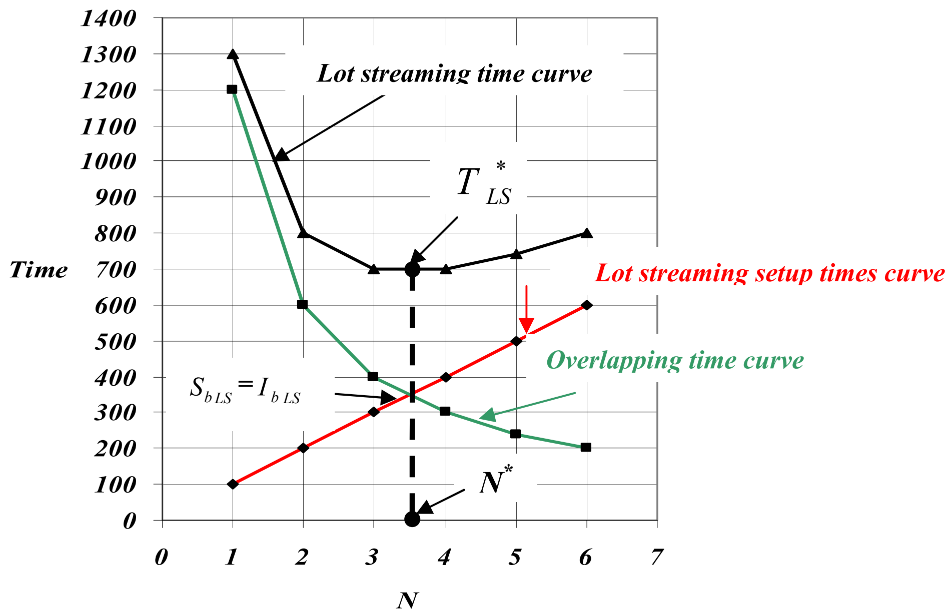


Figure 28: Optimal Lot Streaming Time (OLST) diagram

Table 9: Summary of the results of Examples 3, 5

Example	N^* (sub-batch)	T^*_{LS} (min)	t^*_{pr} (min)	η^*_r %
3	3 , 4	700	3100	19.35
5	3.4641	692.82	3092.82	19.63
	3	700	3100	19.35
	4	700	3100	19.35

The optimal values of lot streaming setup times and overlapping time are equal.

$$S^*_{bLS}=I^*_{bLS}=\sqrt{I_b S_b}=346.41\ min,$$

It can be concluded that the maximum productivity of *FSS/2/2/t_{pr}/ELS/DLS/NI/δ lot streaming problem* can be achieved when the lot streaming setup time is equal to the overlapping time.

Example 8

The Engineering database of *FSS/3/3/ t_{pr} lot streaming problem* is given in Table 10.

δ =10 min.

Table 10: Engineering database of Example 8

Job j	n_j	Machine group m									P_j
		1			2			3			
		$\tau_{j,1}$	$p_{j,1}$	k	$\tau_{j,2}$	$p_{j,2}$	k	$\tau_{j,3}$	$p_{j,3}$	k	
A	300	0.20	60	1	0.13	40	2	0.06	20	3	120
B	350	0.17	60	1	0.29	100	2	0.11	40	3	200
C	400	0.15	60	1	0.15	60	2	0.15	60	3	180
L_m			180			**200**			120		**500**

- In building phase, the initial feasible schedules are generated by simulation method using two important rules FIFO and MS. (See Figures 28, 29)

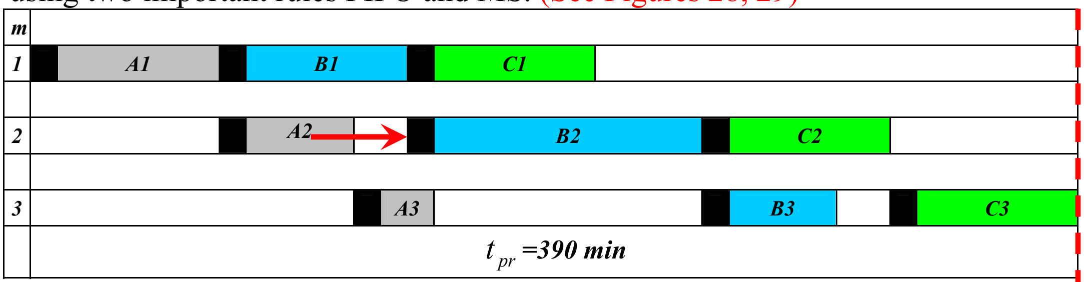


Figure 29: FIFO Gantt chart of initial schedule

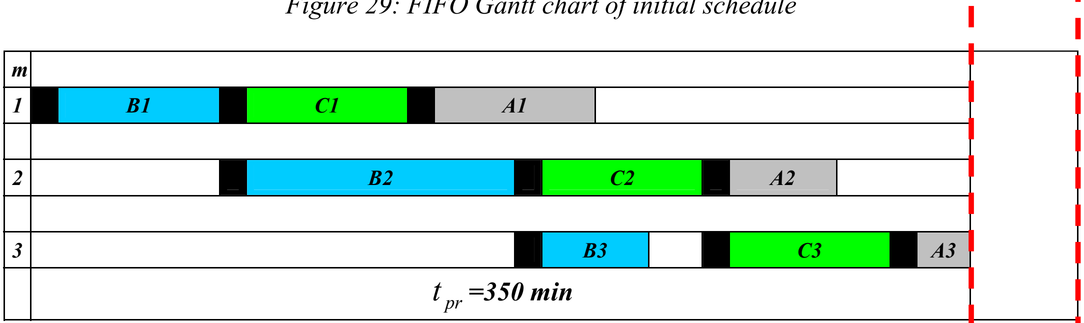


Figure 30: MS Gantt chart of initial schedule

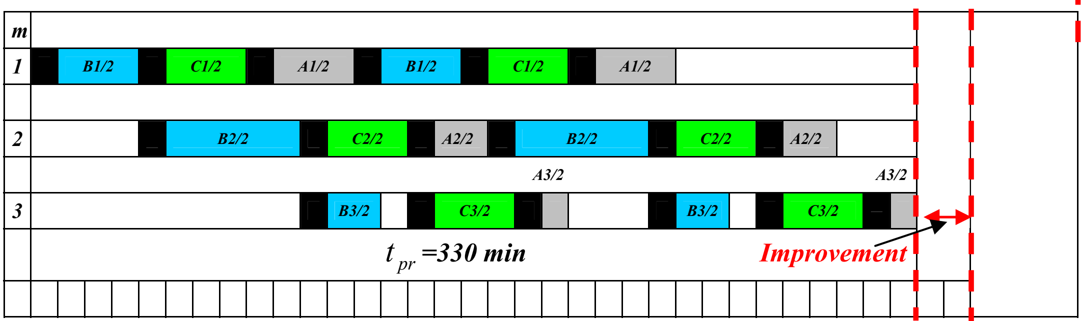


Figure 31: MS Gantt chart of lot streaming schedule, N=2

- In testing phase, the joinability condition is compatible with the MS schedule condition; so, MS schedule is joinable schedule. FIFO schedule needs small shifting switch to the right (Figure 29).

To select the best feasible schedule among FIFO and MS rules we have to determine the minimum of maximum production time (minimum makespan). Minimum makespan of MS schedule (*350 min)* is lower than that of FIFO schedule (*390 min)*.

- In solving phase, the upstream, body and downstream production times for the three machine groups are determined. The results are given in Table 11.

From Table 10, the bottleneck index, $m=2,\ L_b=200min,$

From Table 11, $L_{up,g}=60\ min,\ L_{dn,g}=20\ min,\quad s_b=3setup,\quad M=3\ machine\ group$

Using Equation (141), $t_{g,F}=330\ min,\ I_{g,F}=20\ min$

From the Gantt chart in Figure 30, the makespan of MS schedule t_{pr}=350 min,

By Equation (149), I_0 *=60+20+20=100 min,*

Using Equations (139, 140), S_b *=30 min.* S_{nb} *=20 min*

Using Equation (78), the coefficients values of FSS are almost the same of JSS coefficient, the difference is the determination of the variables of the coefficients. So, $C_{r,FL}$ *=1.0606,* $\beta_{r,F}$ *=0.6060,* $\varepsilon_{r,F}$ *=0.0606,* $\theta_{r,F}$ *=0.09090,* $\phi_{r,F}$ *=0.3030*

Table 11: Upstream and downstream load times

job	$L_{up,m}$	$L_{dn,m}$	Σ
A	***60***	***20***	***80***
B	*60*	*40*	*100*
C	*60*	*60*	*120*

Using Equation (80, 81,121, 123) (By substituting the values of FSS)
The optimal real number of sub-batches, N^* *=1.825 sub-batches*
The closest integer number of N^* *=1.825 is* N^* *=2 sub-batches*
The optimal excess time coefficient, $C^*_{r,FL}$ *= 1*

The optimal makespan, $t^*_{pr} = C^*_{r,FL}\ t_{F,g}$ *=330 min* (Figure 31).

The productivity is improved by the rate, $LSE_{\max}$ *=+ 6.06%*

The optimal sizes of the sub-batches: Z^*_A *=150,* Z^*_B *=175,* Z^*_C *=200* product/sub-batch,

Based on the above, it is concluded that by JSA the productivity is improved even the initial feasible schedule is not joinable. By small modification, the non-joinable schedule can be transformed into joinable, then, we can apply JSA method.

5.5 Effective Delivery Reliability by Joinable Schedule Approach

As mentioned earlier, the delivery reliability is an indicator of how effectively the production of FMS is delivered on the promised date. The effective delivery reliability (EDR) can be achieved when the makespan matches the delivery date. The number of sub-batches to be delivered at the EDR is so-called optimal delivery sub-batches number N^*_d.

Using JSA basic Equation (75), the effective delivery reliability and the optimal delivery sub-batches number can be achieved for both systems job shop systems and flow shop systems as follows:
To formulate the delivery reliability model of job shop lot streaming problems, we assume that the feasible schedule is a joinable schedule and the delivery date is a common delivery date, *d.* from Equations (30, 31), EDR = μ^* *=1,* $t_{pr} = d$.
Equation (75) becomes

$$d = L_b + S_{nb} + N\ S_b + \frac{I_0}{N} \tag{158}$$

$$S_b N^2 + (L_b + S_{nb} - d)\ N + I_0 = 0 \tag{159}$$

Equation (159) is a quadratic equation; so,

$$N^*_d = \frac{-(L_b + S_{nb} - d) \pm \sqrt{(L_b + S_{nb} - d)^2 - 4\,S_b I_0}}{2S_b} \tag{160}$$

Let us propose the following term so-called Bottleneck Discriminant, D_b**,** where,

$$D_b = (L_b + S_{nb} - d)^2 - 4S_b I_0 \tag{161}$$

The optimal delivery sub-batch number becomes as

$$N^*_{d1,2} = \frac{(d - L_b - S_{nb}) \pm \sqrt{D_b}}{2S_b} \tag{162}$$

There are three values of D_b as follows:

a. *Positive bottleneck discriminant*

When the bottleneck discriminant is positive, the delivery dates of the customer greater than the minimum makespan.

$$If\, D_b > 0 \rightarrow d > t^*_{pr} \tag{163}$$

According to Equation (162), there are two optimal delivery sub-batch numbers (N^*_{d1}, N^*_{d2}). The curve will cross the *N-axis* twice.

b. *Zero bottleneck discriminant*

When the bottleneck discriminant is zero, the delivery dates of the customer matches the minimum makespan.

$$If\ D_b = 0 \rightarrow d = t^*_{pr} \tag{164}$$

There is only one value of optimal delivery sub-batch number N^*_d. The curve will cross the *N-axis* once.

$$N^*_d = \frac{d - L_b - S_{nb}}{2S_b} \tag{165}$$

c. *Negative bottleneck discriminant*

When the bottleneck discriminant is negative, the delivery date of the customer is lower than the minimum makespan.

$$If\ D_b < 0 \rightarrow d < t^*_{pr} \tag{166}$$

There is no optimal delivery sub-batches. The curve will not cross the *N-axis* at all.

The optimal delivery sub-batch size can be computed by $Z^*_d = \dfrac{n}{N^*_d}$

It can be concluded that the JSA can be used effectively to obtain the optimal delivery sub-batches number which will be delivered on the promised date. The optimal delivery sub-batches number depends on the bottleneck discriminate. For $D_b > 0$, there are two options for the customer of the optimal delivery sub-batch numbers.

For $D_b = 0$, there is only one optimal value. At these values the delivery reliability is effective (EDR) = $\mu^* = 1$, and the schedule is effective schedule.

For $D_b < 0$, it is impossible to delivery the sub-batches on the promised date.

Example 9

For the given Example 3, let us assume that the delivery date, *d=3200 min*.

The data of the example are L_b*=2400 min,* S_b *=100 min,* S_{nb} *=0,* I_0 *=1200 min,*

Using Equation (161), the bottleneck discriminate $D_{bF} = 160000 > 0,\ d > t^*_{pr}$.

There are two optimal delivery sub-batches numbers: (see Figure 32)

Using Equation (162) we get N^*_{d1} *=2 sub-batches* and N^*_{d2} *= 6 sub-batches.*

The optimal delivery sub-batches sizes are Z^*_{d1} *= 600* and Z^*_{d2} *= 200 pc/sub-batch.* The values of both sizes *(600, 200)* are given in Figures (14, 17), respectively.

It can be concluded that the joinable schedule approach is an effective method to get the effective delivery reliability for both systems (flow shop systems and job shop systems).

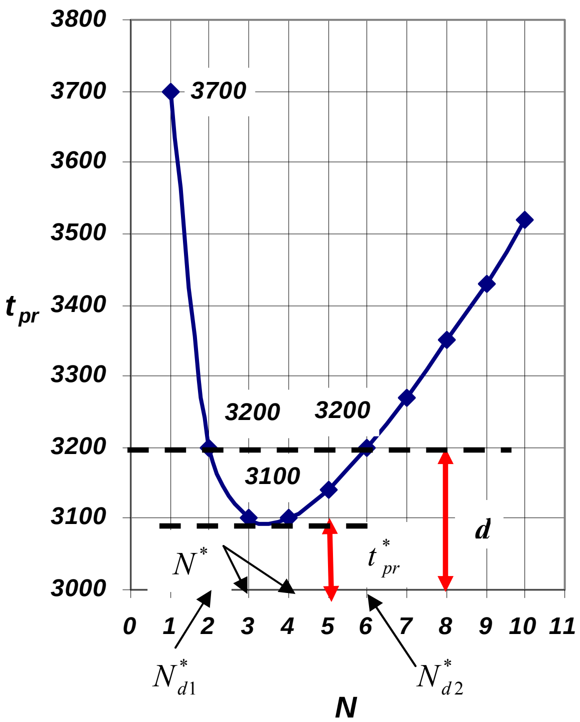


Figure 32: Makespan curve and common delivery date.

5.6 Conclusions of JSA for FMS Lot Streaming Problems

The use of the "Joinable Schedules Approach" may lead to significant improvement of the quality of FMS schedules. The investigation of the problems and the realization of the results are extremely simple.

The results outlined in the previous sections and through the determination of the optimal number of sub-batches, give an opportunity to analyze the effect of the parameters of feasible schedules, setup times values, the effect of lot streaming, etc. In general, by applying JSA we can achieve the following:

1. Optimal number of sub-batches N^*
2. Minimum makespan, t^*_{pr} ,
3. Minimum excess time coefficient, C^*_r
4. Maximum makespan utilization, ρ^* ,
5. Maximum workflow acceleration, γ^*
6. Minimum idle time, I^*
7. Optimal sub-batch size, Z^*
8. Maximum Lot Streaming Efficiency (*Max LSE).*
9. Optimal lot streaming time, = T^*_{LS}
10. Optimal delivery sub-batches number N^*_d

Finally, not the last, by JSA method, the objectives to be achieved are achieved, namely, Highest Makespan Productivity, *(HMP* $=\eta^*$) and Effective Delivery Reliability (EDR= μ^* =1).

6. THE BREAK AND TEST METHOD

6.1 Definition

When using JSA we suppose that the original feasible schedules are joinable schedules and they can be improved by the proposed JSA procedure.
One of the features of the JSA was that we kept the original sequences of the operations for sub-batches, too. JSA is used for a class of FMS scheduling problems which fulfill the joinability condition.
The basic difficulty in practical use of JSA in solving lot streaming problems for both systems job shop systems and flow shop systems is that it is not easy to find the initial joinable schedule. To eliminate this difficulty, we propose in this research a new method called *Break and Test Method (*BTM*)*.
The essence of BTM is an enumeration search optimization method to find the optimal lot streaming in order to achieve HMP and/or EDR.
The basic idea of BTM is "Breaking the batches of proper feasible schedule into many possible different sub-batches, and comparison test is made at each break until finding the proper number of sub-batches according to the specified single objective or multiple objectives ".
BTM is a sort of brute force approach which is based on the concept "Solve the problem by any solution is better than no solution".
"Brute Force Method is an algorithm that inefficiently solves a problem often by trying every one of a wide range of possible solutions" (*National Institute of Standards and Technology (NIST))*.
In BTM, we solve the problem by a large number of solutions and compare them until a suitable solution is found; Try many times until getting the best.
Comparing BTM with JSA, both methods are based on the theory of bottleneck scheduling approach and simulation technique. BTM has a different basic idea, different procedure and different test.

In some cases of lot streaming problems, the sizes for different sub-batches may lead to combinatorial explosion problem. Eliminating this problem we use equal lot streaming in which all the batches are divided into the same number of sub-batches.
It seems to us that Expert Systems (ES), Genetic Algorithms (GA) and other modern approaches could give very effective solutions but these directions are outside of the scope of this research.
In this research, we use BTM to solve FMS lot streaming problems for both basic systems: FSS and JSS.

6.2 Break and Test Method Procedure

Generally, BTM procedure consists of three phases in a sequence:

6.2.1 Building Phase: This phase is the same as the first phase of JSA procedure.

6.2.2 Breaking Phase:

In this phase, the batches are divided into number of sub-batches at certain setup time using BTM computer program as described in [58]. This phase is proposed to name as lot streaming phase.

6.2.3 Testing Phase:

In this phase, the number of sub-batches is compared with the previous number until finding the proper number of sub-batches at which the predetermined objective is met. This phase is proposed to name as objective test phase.

According to the scheduling objective and the initial condition of the problem we can perform one of two tests or both as follows:

a) ***Makespan Productivity Test***

This test is required if the objective of the scheduling system is to improve the productivity which can be performed by comparing the current excess time coefficient with the previous one until we achieve the minimum coefficient.

b) ***Delivery Reliability Test***

This test is required if the objective of the scheduling system is to attain the effective delivery reliability which can be achieved by using lot streaming and comparing the given due date with the generated maximum production time (Makespan) until the makespan matches the due date.

In the following section BTM is applied for both systems job shop systems and flow shop system. At certain setup time we divide all batches into equal sub-batches *N* and then testing is made to compare the new number of sub-batches with the previous number until finding the optimum number of sub-batches at which the objective is met.

6.3 Break and Test Method for Job Shop Systems

As it was written earlier, most of the literature on lot streaming focuses on the flow shop problems. Job shop problems have received much less attention.
In this section, lot streaming of FMS scheduling problem for job shop systems is formulated. BTM is used to solve the problem in order to improve the productivity of the system. The effect of setup times is considered. The problem can be investigated as a numerical example:

Example 10

As an example for FMS scheduling we consider the system analyzed in (55, 57, 58, 60). Engineering database is given in Table 12, *δ =0.2 min. d= 310 min*. Gantt chart For FIFO and MS are given in Figure (33, 38). L_b*= 280 min*

Table 12: Engineering Database of Example 10

		Machine group m														
		1			2			3			4			5		
Job	n_j	τ	t	k	τ	t	k	τ	t	k	τ	t	k	τ	t	k	p_j
A	300	0.3	90	1	0.133	40	2	0.2	60	3	0	0	0	0.133	40	4	230
B	300	0.167	50	2	0.367	110	3	0.1	30	1	0.267	80	4	0	0	0	270
C	350	0.171	60	1	0.114	40	5	0.114	40	2	0.171	60	4	0.229	80	3	280
D	400	0.2	80	3	0.175	70	1	0.2	80	4	0.075	30	2	0.125	50	5	310
L_m			**280**			260			210			170			170		1090

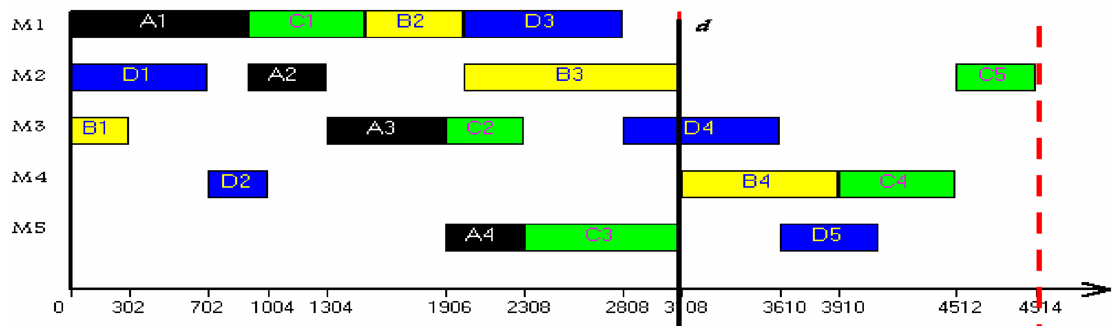


Figure 33: JSS/FIFO Gantt chart of initial schedule, N = 1, t_{pr} = 491.4[h]

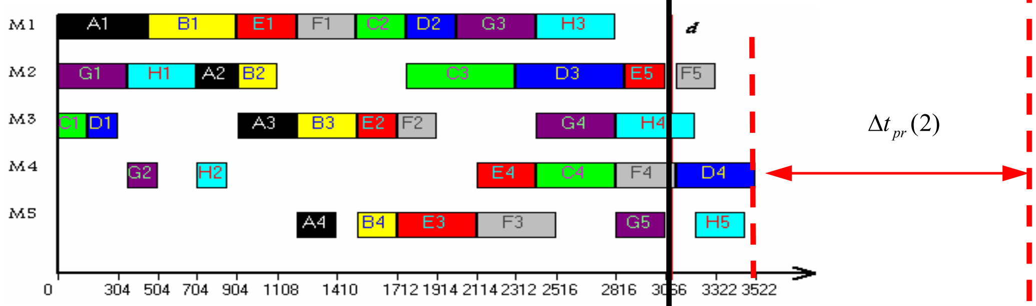


Figure 34: JSS/FIFO Gantt chart of Lot streaming schedule, N = 2, δ = 0.2, t_{pr} = 352.2 min

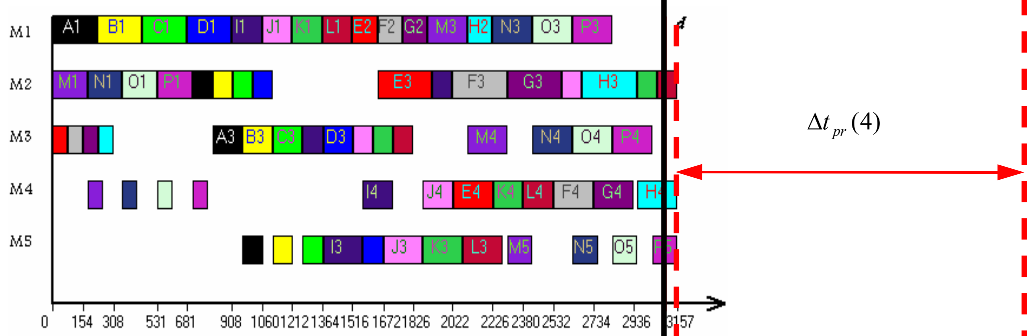


Figure 35: JSS/FIFO Gantt chart of Lot streaming schedule, N = 4, δ = 0.2, t_{pr} = 315.7 min

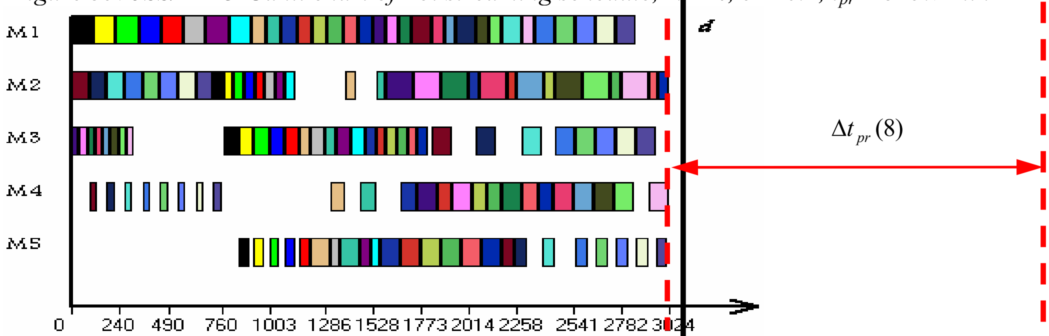


Figure 36: FIFO Gantt chart of Lot streaming schedule N = 8, δ = 0.2, t_{pr} = 302.4 min

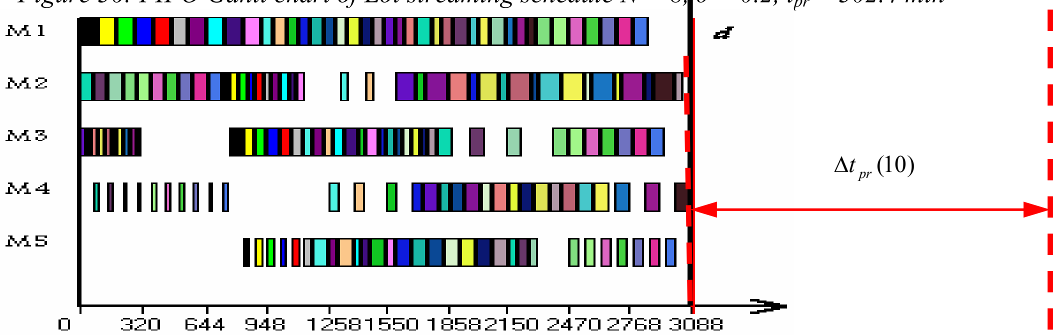


Figure 37: JSS/FIFO Gantt chart Lot streaming schedule, N = 10, δ = 0.2, t_{pr} = 308.8 min

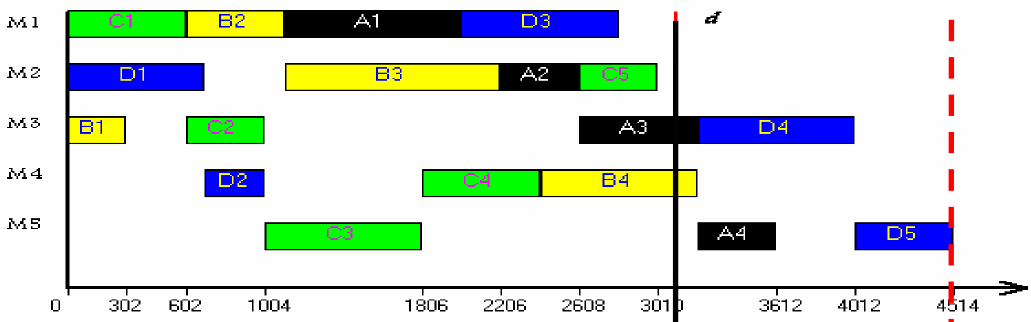


Figure 38:JSS/MS Gantt chart of initial schedule, N = 1, δ = 0.2, t_{pr} = 451.4 min

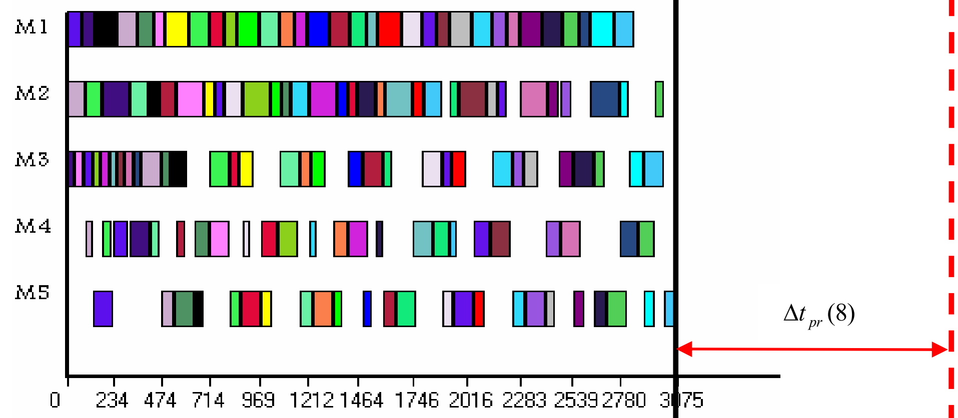


Figure 39: JSS/MS Gantt chart of Lot streaming schedule, N = 8, δ = 0.2, t_{pr} = 307.5 min

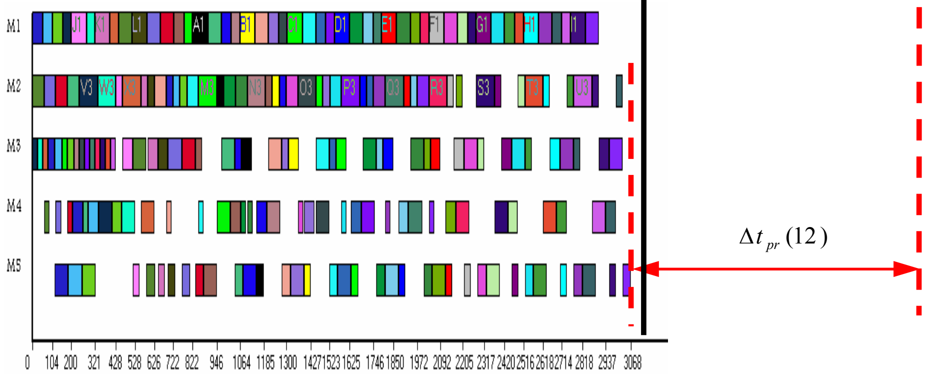


Figure 40: JSS/MS Gantt chart of Lot streaming schedule, N =12, δ = 0.2, t_{pr} = 306.8 min

Let us divide the original batches by *N = 2, 4, 6, 8, 10, 12, 14* and realize scheduling with sub-batches obtained in this way. The Gantt charts of FIFO for *N = 2, 4, 8, 10* are shown in Figures (54-57). The Gantt charts of MS for *N = 8, 12* are shown in Figures (39, 40).The results are given in Table 13 and Figure 61.

Table 13: Makespan & Excess time coefficient values of FIFO and MS

	FIFO		**MS**	
N	t_{pr}	C_r	t_{pr}	C_r
1	491.4	1.76	451.4	1.61
2	352.2	1.26	377.2	1.33
4	316.1	1.13	326.3	1.17
6	306.4	1.09	313.3	1.12
8	302.6	**1.08**	307.5	1.10
10	308.8	1.10	306.6	1.10
12	306.9	1.10	306.8	1.10
14	309.5	1.11	304.7	**1.09**

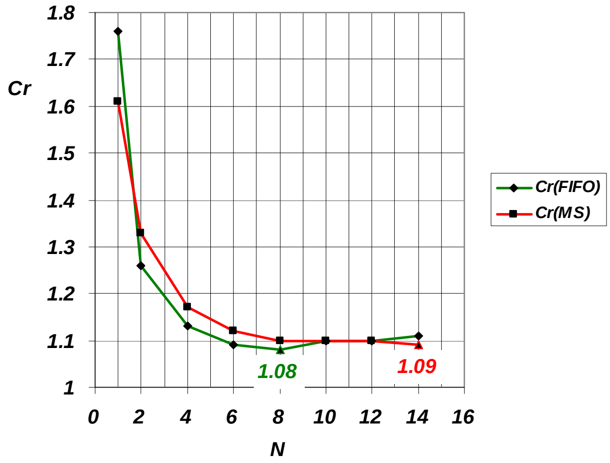


Figure 41: Excess time coefficient curves of FIFO and MS for job shop systems

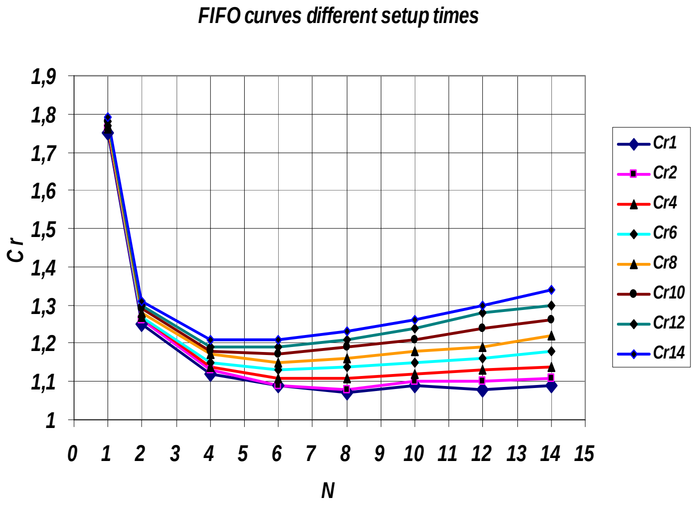
The effect of setup time on the system performances is presented in Tables 14, 20 and Figure 42.

Table 14: Effect of setup time values using FIFO rule

	C_r **(FIFO)**							
δ	0.1	0.2	0.4	0.6	0.8	0.10	0.12	0.14
θ_r	0.00143	0.00286	0.0057	0.0086	0.011	0.0014	0.0017	0.002
N	C_{r1}	C_{r2}	C_{r4}	C_{r6}	C_{r8}	C_{r10}	C_{r12}	C_{r14}
1	1.75	1.76	1.76	1.77	1.77	1.78	1.78	1.79
2	1.25	1.26	1.27	1.27	1.28	1.29	1.30	1.31
4	1.12	1.13	1.14	1.15	1.17	1.18	**1.19**	**1.21**
6	1.09	1.09	**1.11**	**1.13**	**1.15**	**1.17**	1.19	1.21
8	**1.07**	**1.08**	**1.11**	1.14	1.16	1.19	1.21	1.23
10	1.09	1.10	1.12	1.15	1.18	1.21	1.24	1.26
12	1.08	1.10	1.13	1.16	1.19	1.24	1.28	1.30
14	1.09	1.11	1.14	1.18	1.22	1.26	1.30	1.34

Table 15: Effect of setup time values using MS rule

	C_r **(MS)**							
δ	0.1	0.2	0.4	0.6	0.8	0.10	0.12	0.14
θ_r	0.00143	0.00286	0.0057	0.0086	0.011	0.0014	0.0017	0.002
N	C_{r1}	C_{r2}	C_{r4}	C_{r6}	C_{r8}	C_{r10}	C_{r12}	C_{r14}
1	1.61	1.61	1.62	1.62	1.63	1.63	1.64	1.64
2	1.33	1.33	1.34	1.35	1.35	1.36	1.37	1.38
4	1.16	1.17	1.18	1.19	1.21	1.22	1.23	1.25
6	1.11	1.12	1.14	1.16	1.18	**1.20**	**1.22**	**1.23**
8	1.09	1.10	**1.12**	**1.15**	**1.17**	1.20	1.22	1.23
10	1.08	1.10	1.13	1.16	1.19	1.20	1.23	1.26
12	**1.07**	1.10	1.13	1.17	1.21	1.22	1.26	1.29
14	**1.07**	**1.09**	1.13	1.17	1.20	1.24	1.28	1.32



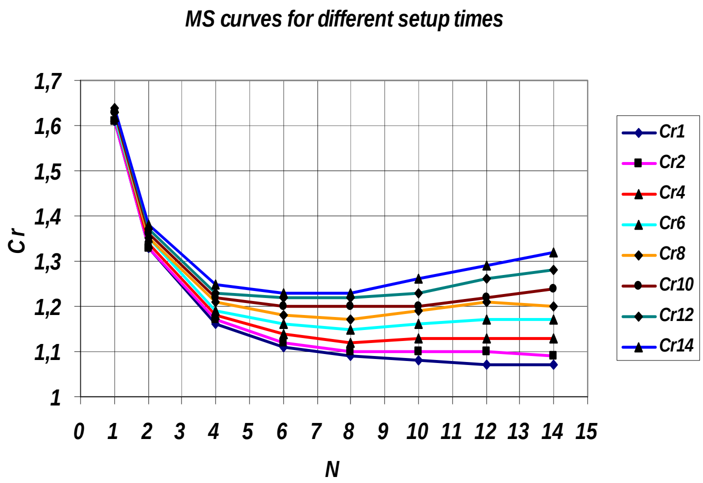


Figure 42: Excess time coefficient curves of FIFO and MS for different setup times (0.1- 1.4 min)

6.4 Analysis of the Results of BTM for Job Shop Systems

As we see from Figures 41, 42 and Tables 13, 14, 15, BTM can be effectively used for job shop lot streaming problems. As it is straightforward, the application range of this method depends very much on the value of setup relation coefficient θ_r. It seems to us (based on the results of the analyzed example, see: Tables 14 and 15, that the values $\theta_r \leq 0.006$ provide very favorable conditions for the use of this method. For these values the makespan values are less than 1.15 times more than the global minimum. Concerning the number of sub-batches N it is interesting to note that $N=8$ seems to be the most favorable number for $0.00143 \leq \theta_r \leq 0.0057$ ($0.1 \leq \delta \leq 0.4$), for bigger values of θ_r, 6 sub-batches seem to be the most suitable choice.

According to the results it is hard to apply BTM when $\delta > 0.8$ [h] ($\theta_r > 0.011$).

When the setup times are small, the lot streaming can be effectively used.

Interestingly, MS rule, which is much better for the original case, is worse at lot streaming. Rather detailed knowledge about the use of BTM can be obtained by case studies, given later, which can lead to deeper understanding of the effectiveness of the method.

6.5 Break and Test Method for Flow Shop Systems

Investigation of the application of BTM for lot streaming problems of flow shop systems is based on the objective to be achieved as follows:

6.5.1 Improving Productivity Objective

To investigate the application of BTM in order to improve the productivity of the system, we present the following example:

Example 11

In Table 16, Engineering database of *FSS/3/3/* t_{pr} *lot streaming problem* is given.
δ =2 h.

Table 16: Engineering database of Example 11

Job	n_j	Machine group m									p_j
		1			2			3			
		τ	t	k	τ	t	k	τ	t	k	
A	300	0.20	60	1	0.13	40	2	0.06	20	3	120
B	350	0.17	60	1	0.29	100	2	0.11	40	3	200
C	400	0.15	60	1	0.15	60	2	0.15	60	3	180
L_m			180			200			120		500

In building phase, by simulation method using two rules: FIFO and MS, the initial feasible schedules are generated. The Gantt charts are given in Figures 43 and 45.
In breaking phase, using BTM computer program [58] with dividing the batches into different numbers *N=2, 3, 4, 5, 6, 7, 8, 9, 10.*
The Gantt chart of FIFO rule for *N=4* is illustrated in Figure 44.
The Gantt chart of MS rule for *N=9* is illustrated in Figure 46.
The values of t_{pr}, C_r, ρ and *LSE* for both *FIFO* and *MS* rules are presented in Tables 17 and 18 and Figures 47, 48, and 49.
In Testing phase, using makespan productivity test to compare the values of t_{pr}, C_r, ρ and *LSE*.
The change of values of t_{pr}, C_r, ρ and *LSE* is presented in Table 19, where Δ indicates the difference between two values (change values).
In Figure 47, the excess time coefficient starts to decrease as number of sub-batches increases until *N =4* and *5* for *FIFO*, *N=3* for *MS* and then starts to increase.
In Figure 48, the system utilization starts to increase as number of sub-batches increases until *N =4* and *5* for *FIFO*, *N=3* for *MS* and then starts to decrease.
In Figure 49, lot streaming efficiency as indication of the improvement starts increasing as number of sub-batches increases until *N =4* and *5* for *FIFO* (*25.90%*), *N=3* for *MS* (*25.51%*), and then starts to decrease.

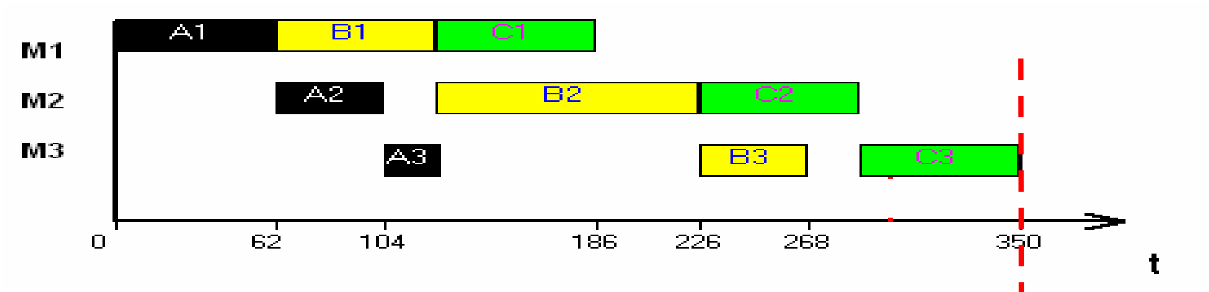


Figure 43: FSS/FIFO Gantt chart of initial schedule, N=1, δ =2

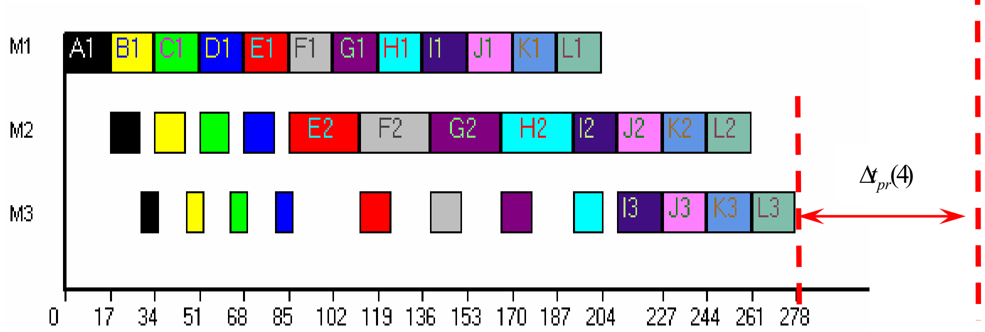


Figure 44: FSS/FIFO Gantt chart of lot streaming schedule, N=4, δ =2

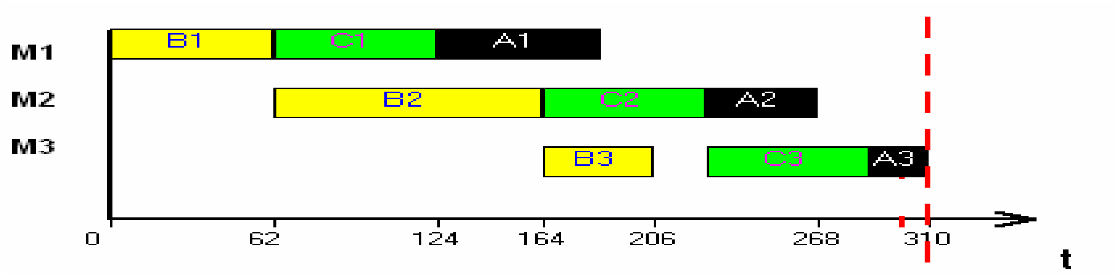


Figure 45: FSS/MS Gantt chart of initial schedule, N=1, δ =2

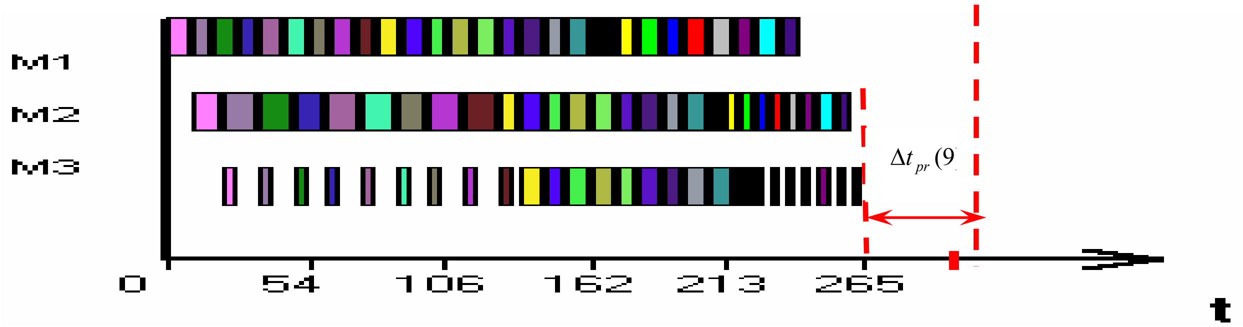


Figure 46: FSS/MS Gantt chart of lot streaming schedule, N = 9, δ=2

Table 17: Results of FIFO rule using BTM

N	t_{pr}	C_r	ρ %	LSE %
1	**350**	**1.75**	**47.62**	**0.00**
2	296	1.48	56.31	18.24
3	281	1.405	59.31	24.56
4	**278**	**1.39**	**59.95**	**25.90**
5	**278**	**1.39**	**59.95**	**25.90**
6	282	1.41	59.10	24.11
7	288	1.44	57.87	21.53
8	292	1.46	57.08	19.86
9	297	1.485	56.12	17.85
10	296	1.48	56.31	18.24

Table 18: Results of MS rule using BTM

N	t_{pr}	C_r	ρ %	LSE %
1	**310**	**1.55**	**53.76**	**0.00**
2	256	1.28	65.10	21.09
3	**247**	**1.235**	**67.48**	**25.51**
4	248	1.24	67.20	25.00
5	250	1.25	66.67	24.00
6	257	1.285	64.85	20.62
7	261	1.305	63.86	18.77
8	262	1.31	63.61	18.32
9	265	1.325	62.89	16.98
10	272	1.36	61.27	13.97

Table 19: Results of BTM for FIFO and MS rules

	FIFO			**MS**		
	Initial values	**Optimal LS**	**Δ**	**Initial values**	**Optimal LS**	**Δ**
N^*	0	4-5	-		3	-
t_{pr}	350	278	72	310	247	63
C_r	1.75	1.39	0.36	1.55	1.235	0.315
ρ %	47.62	59.95	12.33	53.76	67.48	13.72
LSE_{max} %	0	25.90	25.90	0	25.51	25.51

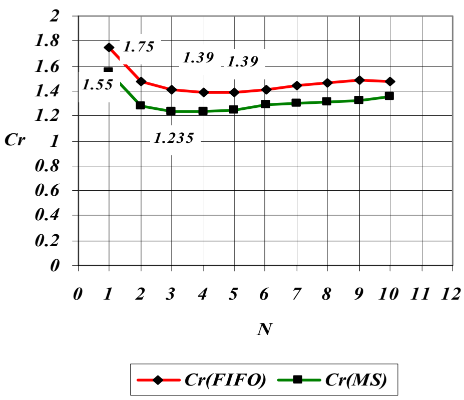


Figure 47: Excess time coefficient curves by BTM of FIFO and MS rules for flow shop systems

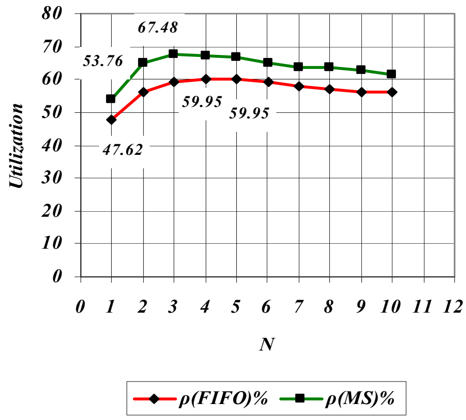


Figure 48: System utilization curves by BTM of FIFO and MS rules

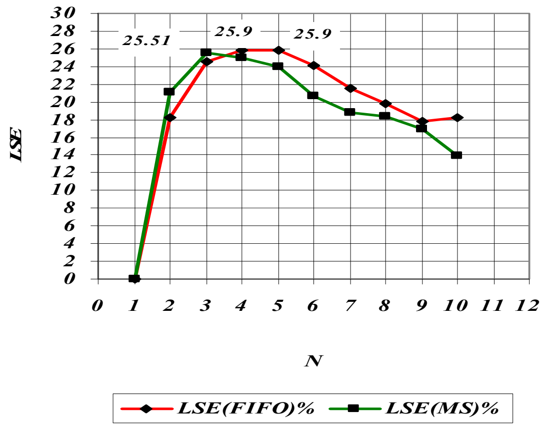


Figure 49: Lot streaming efficiency curves by BTM of FIFO and MS rules

Based on the above, the optimal numbers of sub-batches for FIFO are *4* and *5 sub-batches,* and for MS*,* the optimal number is *3 sub-batches*.

It is interesting to clarify that the highest utilization by MS (*67.48%*) is higher than by FIFO (*59.95%*), whereas, the highest improvement of productivity by FIFO (*25.90%*) is higher than of MS (*25.51%*).

It is concluded that by BTM method we can improve the productivity with different rates based on the used rule. We can obtain two values of maximum productivity by two different optimal numbers of sub-batches.

6.5.2 Effective Delivery Reliability Objective

From Table 17, for *d=281*, by delivery reliability test, EDR can be achieved by FIFO rule with N_d^* *=3 sub-batches.*

From Table 18, for *d=250*, EDR can be achieved by MS rule and N_d^* *= 5 sub-batches.*

6.6 Hybrid Dynamical Approach (HDA) to FMS Scheduling

The following outline of HDA is based on [55-60]. In Figure 50 a diagram is shown representing the schedule for the bottleneck (numbered by 1) machine group for the given example (11). In this Figure the part demands are plotted as straight lines the slopes of which are the demand rates. The piecewise sections below these lines represent the production. During a horizontal section the given part is not produced (there is no change in the number of produced parts). After a horizontal section a production section follows represented by a sloped section. The slope of the line is

equal to the production rate (number of produced parts during the time unit). When this line intersects the “demand line”, the virtual buffer of the given part type becomes empty. (We call “demand line” the one representing the part demand.) At this moment, a decision is made according to the switching law about which part will be processed next. The production of the next part begins after the proper setup time. The diagram in Figure 50 can easily be transformed to corresponding Gantt chart, which is shown in Figure 51.
The above scheduling example was examined using the simulation program developed by T. Koncz [58].
We remark that the outlined above BTM can be used in cases when the HDA is not suitable. These cases are:

a) when the number of parts is not high enough (less than 200).
b) when it is difficult to provide auxiliary buffer contents.

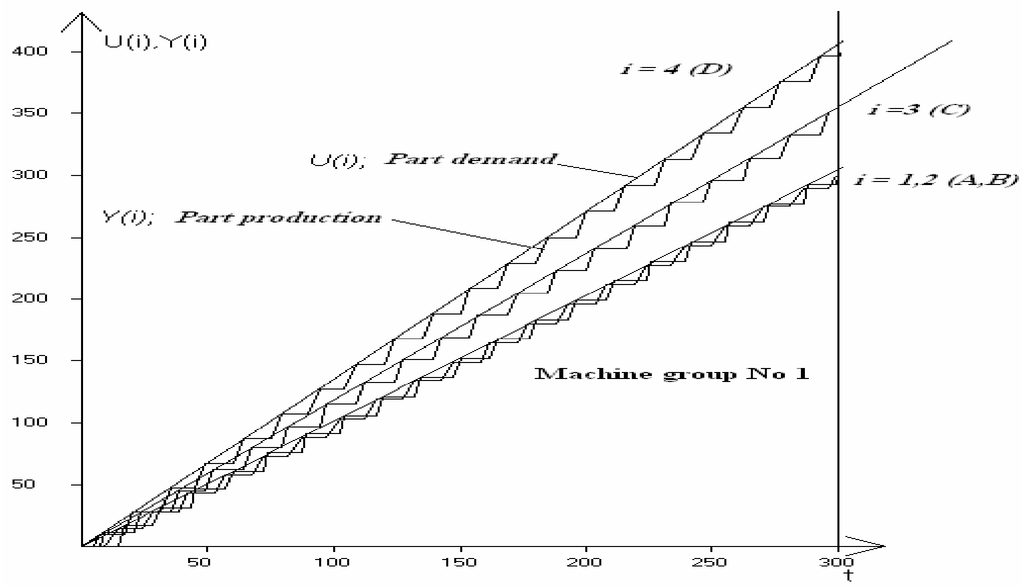


Figure 50: Part demands and part production using HDA

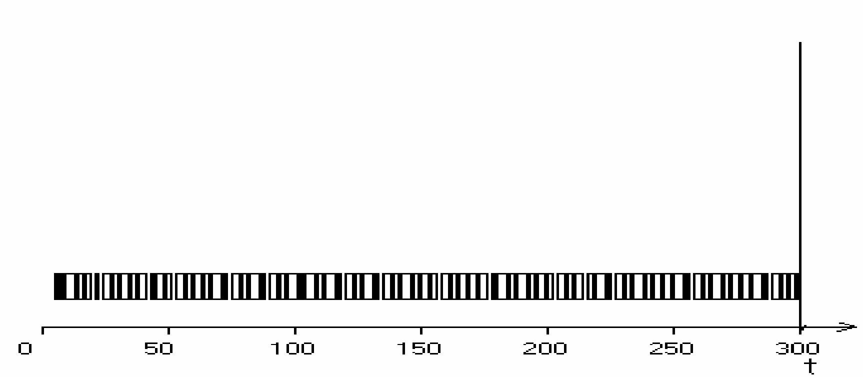


Figure 51: Gantt chart for machine group No. 1 when using HDA

7. CASE STUDIES FOR THE APPLICATION OF BREAK AND TEST METHOD & JOINABLE SCHEDULE APPROACH

In the following case studies, we applied the two methods BTM and JSA for different cases of both systems JSS and FSS and then, we made a comparison between the results of these methods.

7.1 Lot streaming characterization

It is convenient to have a simple notation to represent types of lot streaming scheduling problems. There are many types of characteristics. The general lot streaming scheduling problem is characterized by 9-parameters.

$$\alpha_1/\ \alpha_2/\ \alpha_3/\ \alpha_4/\ \alpha_5/\ \alpha_6/\ \alpha_7/\alpha_8/\ \alpha_9$$

α_1: manufacturing system type, α_2: Number of machines to be used, α_3: Number of jobs to be produced, α_4: Bottleneck index. α_5: Objective to be achieved, α_6: Setup time value, α_7: Sub-batch type, α_8: Sub-batch size, α_9: Idling type.

For example: *FSS/2/2/1/ρ/2/ELS/DLS/II*: The FMS system is a Flow Shop System (α_1) consisting of two machine groups ($\alpha_2=2$) to process two jobs ($\alpha_3=2$), the bottleneck machine group index is 1 ($\alpha_4=1$) and the objective (α_5) is to obtain a higher utilization ρ, the setup time value α_6 ($\delta=2h$) sub-batch type is an equal lot streaming ($\alpha_7=ELS$), Sub-batch size is a discrete lot streaming ($\alpha_8=DLS$) and intermediate idle time (α_9 =II) is allowed.

7.2. Case Studies for job shops systems using BTM and JSA

In this section, applications and analysis of *BTM* and *JSA* for 5 different case studies of job-shop lot streaming problems are performed. A comparative study is made between these methods through these case study examples.

7.2.1 Characterization of job shops systems case studies

The case studies data are introduced in Table 20: To clarify the content of the table we give an example which is the third case: ***Case No3: JSS/3/4/2/ρ/3.*** It means that the manufacturing system is a Job Shop System (α_1=JSS) which consists of three machine groups ($\alpha_2=3$) to process four jobs ($\alpha_3=4$), and the bottleneck index is ($\alpha_4=2$). The objective is to obtain higher utilization ($\alpha_5=\rho$), and the value of setup time ($\alpha_6=\delta=3h$).

From Table 20 we can see that there are five different cases of different load times, number of machine groups. the cases 1 and 3 have the same index of bottleneck group, the cases 1, 2 have the same setup times (*δ=2)* and also the cases 3 and 4 have the same setup times of (*δ=3)*.

Table 20: Five case studies of JSS with different machine group index

Case	α_2	α_3	α_4	α_5	α_6	L_b	L
1	2	2	2	ρ	2	160	280
2	3	3	1	ρ	2	200	500
3	3	4	2	ρ	3	320	840
4	4	4	3	ρ	3	380	1180
5	5	4	5	ρ	4	360	1360

7.2.2 Engineering Database of case studies of job shop systems

The Engineering databases of the case studies are given in Tables (21, 22, 23, 24, 25).

Case No 1: JSS/2/2/2/ρ/2

Table 21: Engineering Database of case N0 1

Job	n_j	Machine group m: 1			2			P_j
		τ	l	k	τ	l	k	
A	200	0.40	80	2	0.50	100	1	180
B	150	0.27	40	1	0.40	60	2	100
L_m			120			160		280

Case No 2: JSS/ 3/3/1/ρ/2

Table 22: Engineering Database of case No 2

Job	n_j	Machine group m: 1			2			3			P_j
		τ	l	k	τ	l	k	τ	l	k	
A	100	0.40	40	1	0.40	40	2	0.20	20	3	100
B	150	0.40	60	2	0.40	60	1	0.27	40	3	160
C	150	0.67	100	2	0.53	80	3	0.40	60	1	240
L_m			200			180			120		500

Case No 3: JSS/ 3/4/2/ρ/3

Table 23: Engineering Database of case No 3

Job j	n_j	Machine group m: 1			2			3			P_j
		τ	l	k	τ	l	k	τ	l	k	
A	100	0.40	40	3	0.80	80	2	0.60	60	1	180
B	150	0.27	40	1	0.40	60	3	0.40	60	2	160
C	150	0.40	60	2	0.53	80	1	0.53	80	3	220
D	200	0.40	80	3	0.5	100	2	0.5	100	1	280
L_m			220			320			300		840

Case No 4: JSS/ 4/4/3/ρ/3

Table 24: Engineering Database of case No 4

Job j	n_j	Machine group m: 1			2			3			4			P_j
		τ	l	k	τ	l	k	τ	l	k	τ	l	k	
1(A)	200	0.50	100	4	0.40	80	3	0.50	100	2	0.40	80	1	360
2(B)	250	0.24	60	1	0.32	80	4	0.40	100	3	0.24	60	2	300

3(C)	300	0.27	80	2	0.20	60	1	0.27	80	4	0.13	40	3	260
4(D)	250	0.24	60	3	0.24	60	2	0.40	100	1	0.16	40	4	260
L_m			300			280			**380**			220		**1180**

Case No 5: JSS/ 5/4/5/ρ/4

Table 25: Engineering Database of case No 5

Job j	n_j	Machine group m															P_j
		1			2			3			4			5			
		τ	l	k	τ	l	k	τ	l	k	τ	l	k	τ	l	k	
1(A)	100	0.60	60	2	1	100	3	0.80	80	1	0.80	80	4	1	100	5	420
2(B)	150	0.40	60	3	0.53	80	2	0.40	60	4	0.40	60	5	0.53	80	1	340
3(C)	150	0.27	40	4	0.40	60	3	0.53	80	2	0.40	60	1	0.67	100	5	340
4(D)	200	0.20	40	5	0.30	60	1	0.20	40	3	0.20	40	2	0.40	80	4	260
L_m			200			300			260			240			**360**		**1360**

7.2.3 Application of BTM for job shop case studies

In order to apply BTM to the given case studies we have to follow the BTM procedure as we did in flow shop problems. In building phase, the initial feasible schedules of the case studies are generated using the computer program described in [58]. Then, breaking phase, we divide the batches into a number of sub-batches, *N= 2, 3, 4, 5, 6.* In testing phase, we test the number according to the predetermined objective. Let us suppose the objective is to improve the productivity of the system.

Case No 1: JSS/2/2/2/ρ/2

Table 26: Excess time coefficient & utilization values of case No 1

N	FIFO,MS		
	t_{pr}	C_r	ρ %
1	**184**	**1,122**	**76,09**
2	**168**	**1,024**	**83,33**
3	171	1,043	81,87
4	176	1,073	79,55
5	180	1,098	77,78
6	186	1,134	75,27

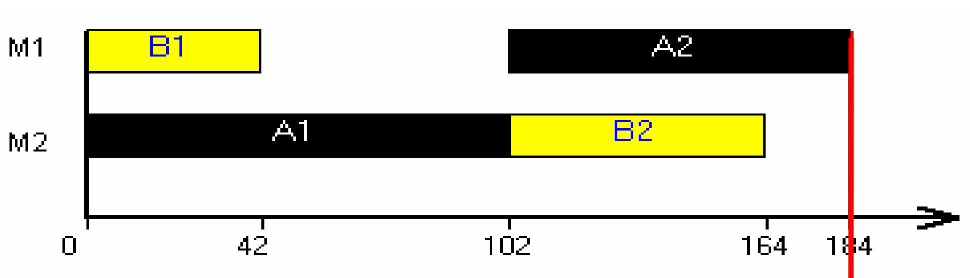
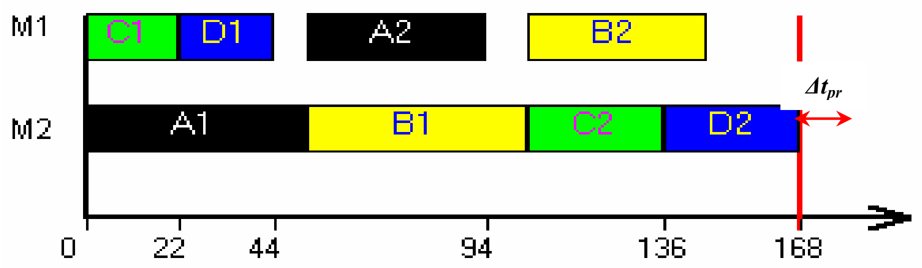


Figure 52: a) JSS/FIFO Gantt chart of initial schedule for case No 1



b) JSS/FIFO Gantt chart of lot streaming schedule for case No 1, N = 2

Case No 2: JSS/3/3/1/p/2

Table 27: Excess time coefficient & utilization values of case No 2

	FIFO			MS		
N	t_{pr}	C_r	ρ %	t_{pr}	C_r	ρ %
1	***308***	***1,495***	***54,11***	***268***	***1,301***	***62,19***
2	*254*	*1,233*	*65,62*	***234***	***1,136***	***71,23***
3	***245***	***1,189***	***68,03***	*235*	*1,141*	*70,92*
4	*246*	*1,194*	*67,75*	*244*	*1,184*	*68,31*
5	*248*	*1,204*	*67,20*	*250*	*1,214*	*66,67*
6	*255*	*1,238*	*65,36*	*259*	*1,257*	*64,35*

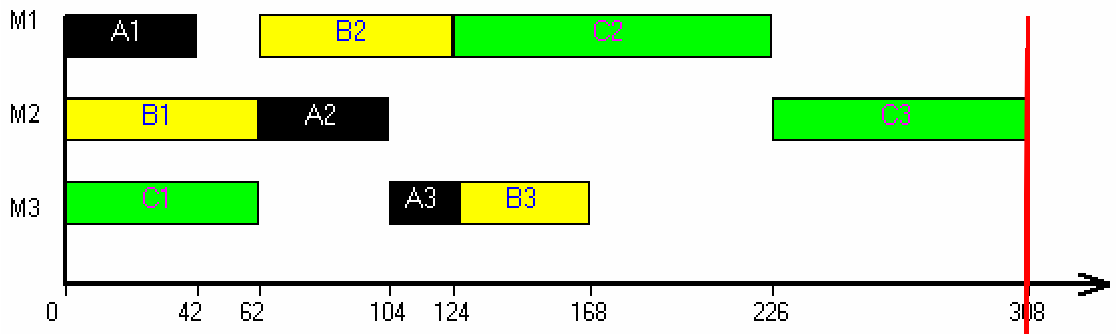
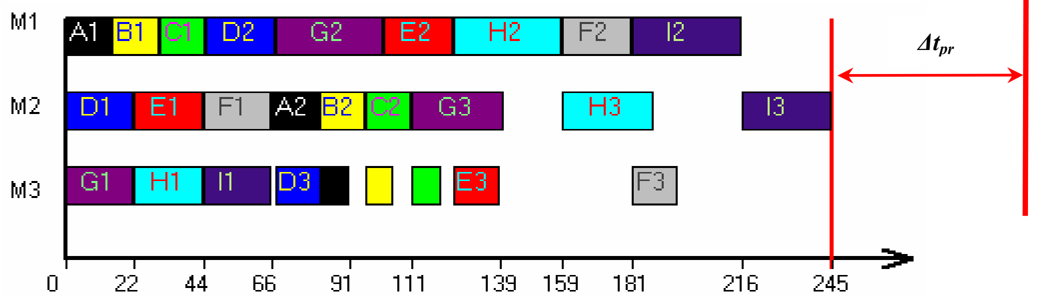


Figure 53: a) JSS/FIFO Gantt chart of initial schedule for case No 2



b) JSS/FIFO Gantt chart of lot streaming schedule for case No 2, N = 3

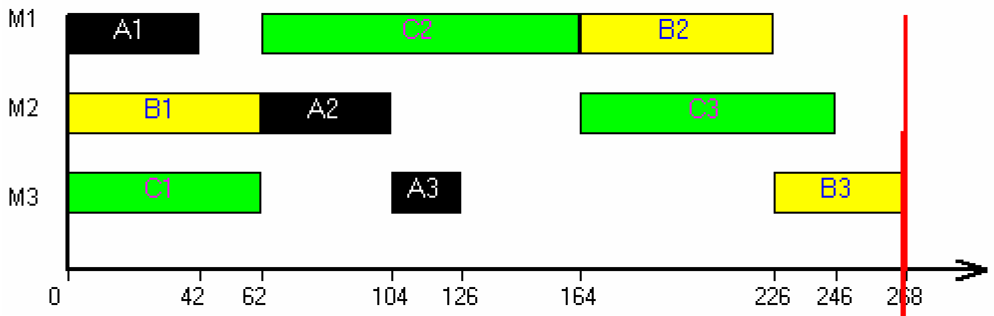
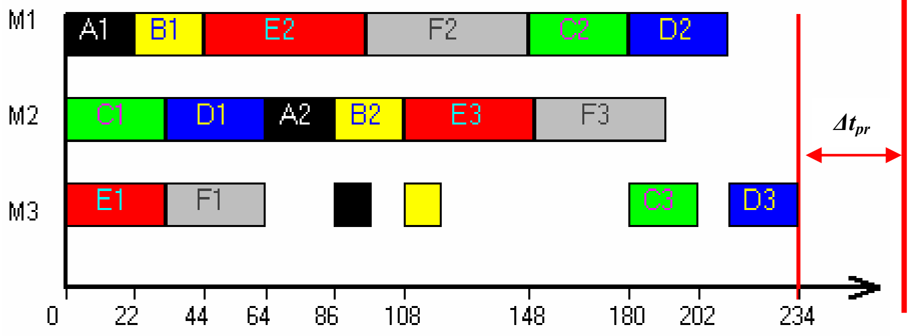


Figure 54: a) JSS/MS Gantt chart of initial schedule for case No 2



b) JSS/MS Gantt chart of lot streaming schedule for case No 2, N=2

Case No 3: JSS/3/4/2/ρ/3

Table 28: Excess time coefficient & utilization values of case No 3

	FIFO,MS		
N	t_{pr}	C_r	ρ %
1	***352***	***1,06***	***79,55***
2	***344***	***1,036***	***81,40***
3	*357*	*1,075*	*78,43*
4	*368*	*1,108*	*76,09*
5	*380*	*1,145*	*73,68*
6	*390*	*1,175*	*71,79*

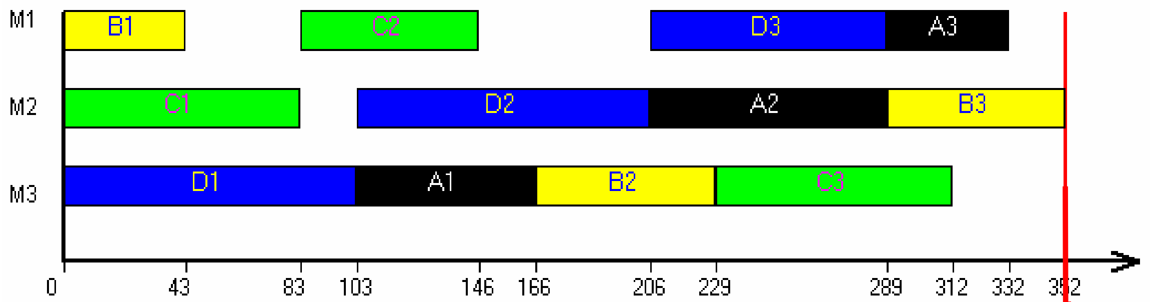
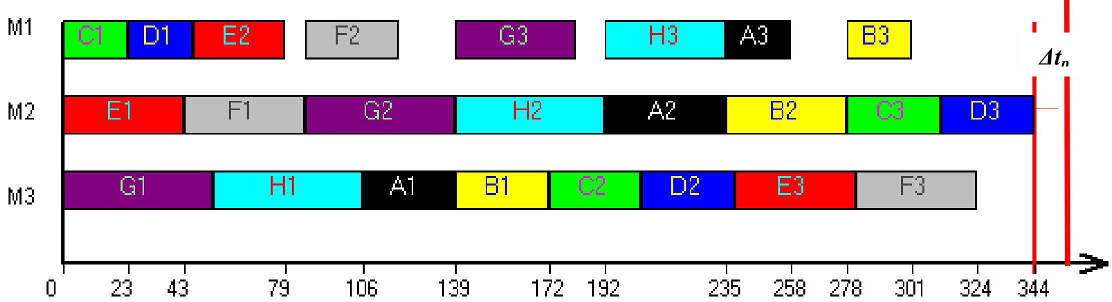


Figure 55: a) JSS/MS Gantt chart of initial schedule for case No 3



b) JSS/MS Gantt chart of lot streaming schedule for case No 3, N = 2

Case No 4: JSS/4/4/3/p/3

Table 29: Excess time coefficient & utilization values of case No 4

N	FIFO, MS		
	t_{pr}	C_r	ρ %
1	***392***	***1***	***75,26***

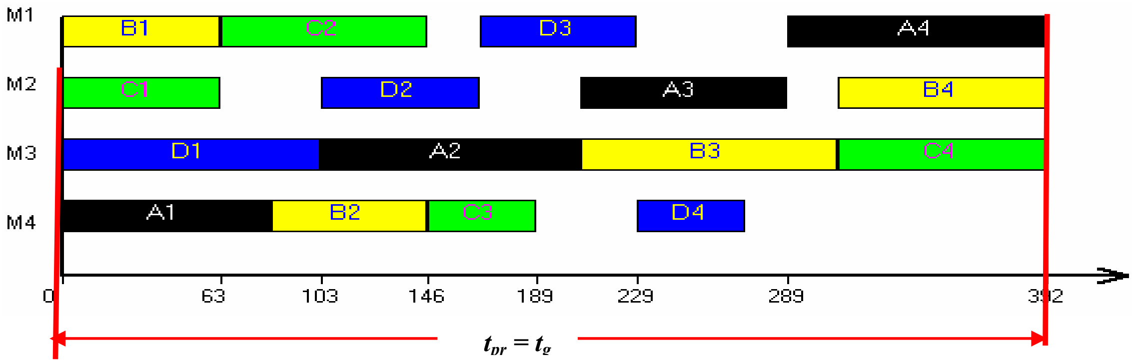


Figure 56: JSS/MS Gantt chart of initial schedule for case No 4

Case No 5: JSS/5/4/5/p/4

Table 30: Excess time coefficient & utilization values of case No 5

	FIFO			MS		
N	t_{pr}	C_r	ρ	t_{pr}	C_r	ρ %
1	***564***	***1,5***	***48,23***	***564***	***1,5***	***48,23***
2	504	1,34	53,97	504	1,34	53,97
3	***486***	***1,293***	***55,97***	***469***	***1,247***	***58***
4	505	1,343	53,86	484	1,287	56,2
5	504	1,34	53,97	500	1,33	54,4
6	527	1,402	51,61	521	1,386	52,21

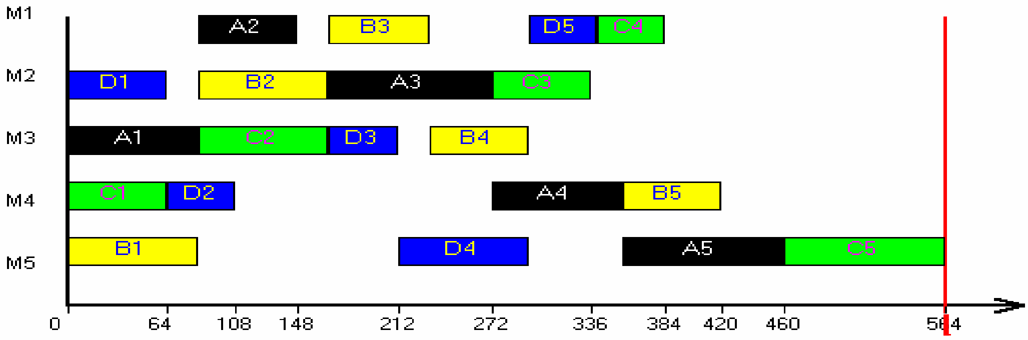
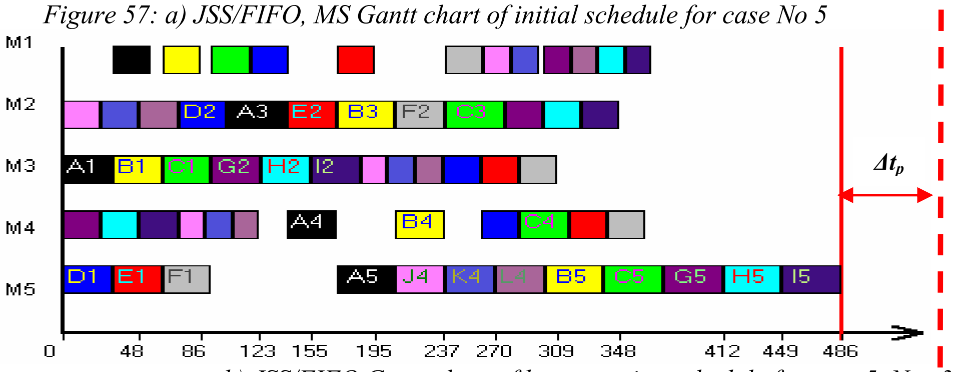
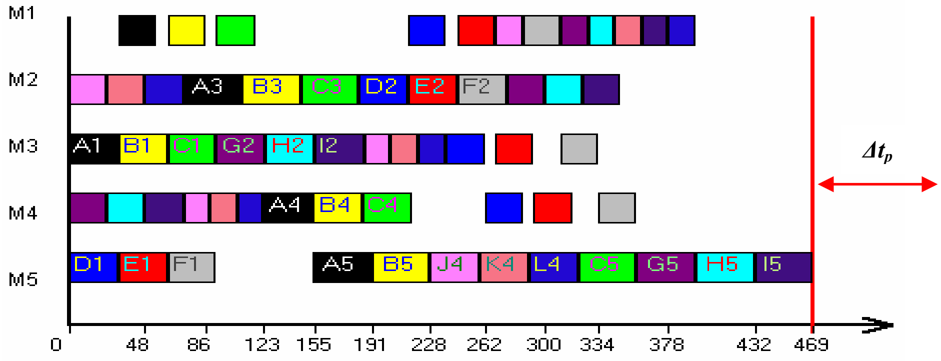


Figure 57: a) JSS/FIFO, MS Gantt chart of initial schedule for case No 5



b) JSS/FIFO Gantt chart of lot streaming schedule for case 5, N = 3



c) JSS/MS Gantt chart of lot streaming schedule of case 5, N = 3

We represent the Gantt charts of FIFO and MS rules of all cases in Figures (52-57). Let us remind that the setup times are given in Table 20.

7.2.4 Results of BTM Applications for job shop systems

The values of t_{pr}, and C_r, and Gantt charts of all cases are represented in Tables (26, 30) and summary of the values of C_r in Table 31.

The optimal values: N^*, t^*_{pr}, C^*_r, ρ^* and *max LSE* can be computed. The values are represented in Table 32.

Table 31: JSS Excess time coefficient values of five cases using BTM

N	C_{r1}	C_{r2} FIFO	C_{r2} MS	C_{r3}	C_{r4}	C_{r5} FIFO	C_{r5} MS
1	1,122	1,495	1,301	1,060	1	1,500	1,500
2	**1,024**	1,233	**1,136**	**1,036**	-	1,340	1,340
3	1,043	**1,189**	1,141	1,075	-	**1,293**	**1,247**
4	1,073	1,194	1,184	1,108	-	1,343	1,287
5	1,098	1,204	1,214	1,145	-	1,340	1,330
6	1,134	1,238	1,257	1,175	-	1,402	1,386

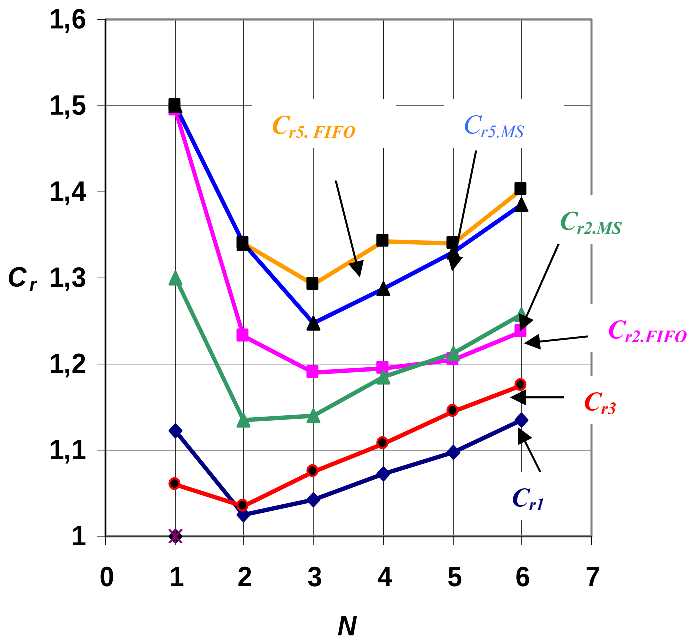


Figure 58: JSS Excess time coefficient curves of 5- cases using BTM

Table 32: Maximum lot streaming efficiency values of 5-cases using BTM

Case	***Rule***	N^*	C_r	C^*_r	$LSE_{\max}$ %
1	*FIFO*	*2*	*1,122*	*1,024*	*9,57*
	MS	*2*	*1,122*		
2	*FIFO*	*3*	*1,495*	*1,189*	***25,74***
	MS	*2*	*1,301*	*1,136*	*14,52*
3	*FIFO*	*2*	*1,06*	*1,036*	*2,32*
	MS	*2*	*1,06*		
4	***FIFO***	*-*	*1*	***1***	***0***
	MS	*-*	*1*		
5	*FIFO*	*3*	*1,5*	*1,293*	*16,01*
	MS	*3*	*1,5*	*1,247*	*20,29*

7.2.5 Analysis of the Results of BTM for job shop systems

It can be concluded from Table 32 and Figure 58 that the optimum number of sub-batch N^* for all the cases is in the range *(2, 3)* and the optimum excess time coefficient C^*_r in the range *(1, 1.293)*

<u>Case 1</u>

The optimal number of sub-batches for both rules FIFO and MS are the same *(*N^* = 2). The excess time coefficient is much closed to 1 (C^*_r*=1.024)* because the initial schedule is a high quality schedule. The maximum rate of the productivity improvement is relatively low *(*$LSE_{\max}$ *= 9.57%).*

<u>Case 2</u>

There are two different N^* = (3, 2) of FIFO and MS, respectively. The maximum rate of the productivity improvement of FIFO *(*$LSE_{\max}$ *=25.74%)* has the highest value among the case studies.

<u>Case 3,</u>

(Like the values of case 1 with different number of machine groups and jobs to be processed) the optimal number of sub-batches for both rules FIFO and MS are the same *(*N^* *= 2).* The maximum rate of the productivity improvement is very low relatively *(*$LSE_{\max}$ *=2.32 %)* because the initial schedule is a very high quality schedule (close to ideal schedule).

<u>Case 4</u>

The initial schedule is ideal schedule for both FIFO and MS *(*C_r *=1),* in this case there is no productivity improvement rate *LSE= 0.* Lot streaming is not efficient.

<u>Case 5,</u>

The optimal number of sub-batches for both rules FIFO and MS are the same *(*N^* *= 3)* but the maximum rate of the productivity improvement is different $LSE_{\max}$ of FIFO *(16.01%)* is lower than that of MS *(20.29%).*

C^*_r of the two cases 2, 5 are: for FIFO *(1.189), (1.293)* is higher than that of MS *(1.136), (1.247),* respectively.

7.3 Comparing the Results of Applications of BTM and JSA for job shop systems

By substituting the given values into Equations (60, 61, 65, 80, 117, 82, 81, 14, 121) we can get the results given in Table 33.

Table 33: Results of applications of JSA for job shop systems

Case	**Rule**	L_b	S_b	t_g	t_{pr}	I_0	C_r	N^*	N_i^*	t_{pr}^*	C_r^*	ρ %	$LSE_{\max}$ %
1	*FIFO,MS*	*160*	*4*	*164*	*184*	*20*	*1,122*	*2,23*	*2*	*177,89*	*1,085*	*78,70*	*3,33*
2	*FIFO*	*200*	*6*	*206*	*308*	*102*	*1,495*	*4,12*	*4*	*249,48*	*1,211*	*66,81*	***18,99***
	MS	*200*	*6*	*206*	*268*	*62*	*1,301*	*3,21*	*3*	*238,57*	*1,158*	*69,86*	*10,98*
3	*FIFO,MS*	*320*	*12*	*332*	*352*	*20*	*1,06*	*1,29*	*2*	*350,98*	*1,057*	*79,78*	*0,27*
4	*FIFO,MS*	*380*	*12*	*392*	*392*	*0*	*1*	*1*	*1*	*392,00*	*1*	*75,26*	***0***
5	*FIFO,MS*	*360*	*16*	*376*	*564*	*188*	*1,5*	*3.42*	*3*	*469,69*	*1,249*	*57,91*	*16,72*

Table 34: Optimum excess time coefficient values of JSS 5-cases using BTM and JSA

		N^*		C^*_r	
Case	***Rule***	***JSA***	***BTM***	***JSA***	***BTM***
1	*FIFO,MS*	*2*	*2*	*1,085*	*1,024*
2	*FIFO*	*4*	*3*	*1,211*	*1,189*
	MS	*3*	*2*	*1,158*	*1,136*
3	*FIFO,MS*	*2*	*2*	*1,057*	*1,036*
4	*FIFO,MS*	*1*	*1*	*1*	*1*
5	*FIFO*	*3*	*3*	*1,249*	*1,293*
	MS	*3*	*3*	*1,249*	*1,247*

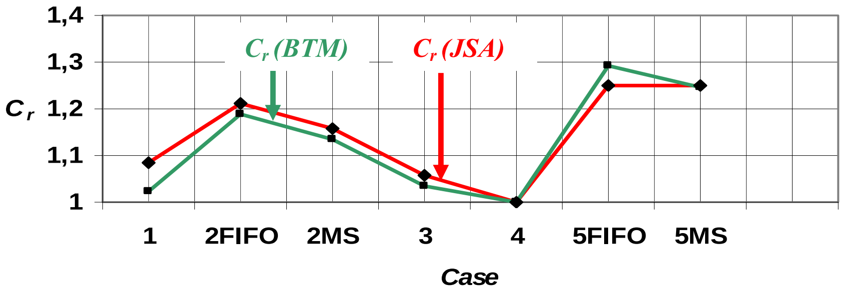


Figure 59: Optimum excess time coefficient curves of JSS 5-cases using BTM and JSA

From Table 33, the range of the optimal numbers of sub-batch N^* for all the cases is *(2, 4)* and the range of the optimal excess time coefficient C_r^* is *(1.057, 1.249)*. In case 4, the initial schedule is ideal schedule and the initial excess time coefficient (C_r *=1)*. Lot streaming is not efficient.

Comparing the results given in Table 34 and Figure 54, the quantities obtained by the application of both methods BTM and JSA using FIFO and MS rules are very close for the five cases but there is a small difference between the two methods due to the following reasons:

For case 1, The bottleneck index is number 2 and there is no back and front idle times on the non-bottleneck machine group. The joinability condition is not fulfilled.

For case 2, 3, 5, The bottleneck indexes are number 1, 2, 5, respectively and there is in-between idle time on the bottleneck machine group. The joinability condition is not fulfilled.

For case 4, Its initial excess time coefficient value is one which is the same of BTM.

In general, from Table 34 and Figure 59 it can be recognized that the two methods BTM and JSA can be used effectively to solve job shop lot streaming problems.

7.4 Estimation of the effect of setup times

In case study 5, for FIFO and MS, we can recognize the effect of setup times on lot streaming. We analyze the effect of different setup times (1, 2, 3, 4, 6, 8, 10, 20, 60, 80, 90 100). We use the JSA. By substitution in Relation (77, 89, 84) the results are given in Table 35**.** Characterization of the effect of setup time is given in Figure 60.

Table 35: Setup times values and its effects

δ	S_b	t_g	t_{pr}	Δt_{pr}	ΔC_r	*LSE %*
1	*4*	*364*	*546*	*132,04*	*0,36*	*24,18*
2	*8*	*368*	*552*	*115,27*	*0,31*	*20,88*
3	*12*	*372*	*558*	*103,51*	*0,28*	*18,55*
4	***16***	***376***	***564***	***94,31***	***0,25***	***16,72***
6	*24*	*384*	*576*	*80,24*	*0,21*	*13,93*
8	*32*	*392*	*588*	*69,61*	*0,18*	*11,84*
10	*40*	*400*	*600*	*61,11*	*0,15*	*10,19*
20	*80*	*440*	*660*	*34,67*	*0,08*	*5,25*
60	*240*	*600*	*900*	*3,34*	*0,01*	*0,37*
80	*320*	*680*	*1020*	*0,30*	*0,00*	*0,03*
90	*360*	*720*	*1090*	*0,07*	*0,00*	*0,01*
100	***400***	*760*	***1160***	*0*	*0*	*0*

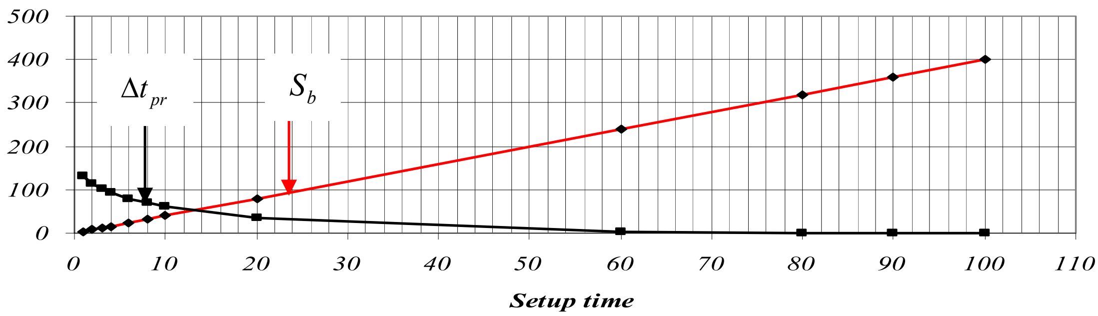


Figure 60: Effect of setup time on the makespan

We can conclude that as the setup times increases, makespan improvement**,** Δt_{pr} decreases until a certain point at which no more improvement can be achieved *(*Δt_{pr} *=0)* where the setup time δ = *100* min. The improvement by lot streaming approach highly depends on the setup time values.

7.5 Case Studies for flow shop systems using BTM and JSA

In this research, BTM is applied for 7-random case studies of flow-shop lot streaming problems. By a certain simulation computer program, as described in [58], we break the batches at certain setup time into equal sub-batches and then comparative testing is made between the new number and the previous one until finding the proper number at which the objective is attained.

7.5.1 Characterization of case studies

It is suitable to simplify the data of the case studies by notation to represent the types of lot streaming scheduling problems. The case studies can be characterized according to basic six parameters. Each case is characterized as a category $\alpha_1/\ \alpha_2/\ \alpha_3/\ \alpha_4/\ \alpha_5/\ \alpha_6$
Where**,** α_1 is indicating to the manufacturing system type, α_2*:* number of machine to be used, α_3*:* number of jobs to be produced, α_4**:** Bottleneck index.α_5*:* Objective to be achieved, α_6**:** Setup time value.
The case studies data are introduced in Table 36: To demonstrate the content of the table we give an example which is the first case study: *FSS/2/2/1/ρ/2*. The flexible manufacturing system is a Flow Shop System (α_1=FSS) consisting of two machine

groups (**α_2**=*2*) to be processed two jobs (**α_3**=*2*), and the bottleneck index *(**α_4**=1).* The objective is to obtain higher utilization (**α_5**=*ρ),* and the setup tim*e (**α_6**=δ=2h).*
As it is clear from Table 32, the case studies to be presented in this section are different cases and they are not selected statistically but so-called semi-randomly. The cases have different number of machine groups (2 to 5), different number of jobs (2 to 4), different bottleneck indexes (1 to 5) and different setup times values (2 to 4 min). It is remarkable that cases 1, 2 have the same $\alpha_2, \alpha_3, \alpha_5, \alpha_6$, and L, different bottleneck index α_4 and different bottleneck load L_b. Cases 3, 4 have the same $\alpha_2, \alpha_3, \alpha_5, \alpha_6$, and L_b, different bottleneck index α_4.

Table 36: Seven case studies of FSS with different machine group indexes

Case	α_2	α_3	α_4	α_5	α_6	L_b	L
1	2	2	1	ρ	2	180	280
2	2	2	2	ρ	2	160	280
3	3	3	1	ρ	2	200	500
4	3	3	2	ρ	2	200	500
5	3	4	2	ρ	3	320	840
6	4	4	3	ρ	3	380	1180
7	5	4	5	ρ	4	360	1360

7.5.2 Engineering Database of Case Studies

The Engineering Databases of the case studies are given in Tables (37-43).

Table 37: Engineering Database of Case No 1: FSS/2/2/1/ρ/2

Job j	n_j	Machine group m 1 τ	1 t	1 k	2 τ	2 t	2 k	P_j
1(A)	150	0.67	100	1	0.40	60	2	160
2(B)	200	0.40	80	1	0.20	40	2	120
L_m			**180**			100		**280**

Table 38: Engineering Database of Case No 2:FSS/2/2/2/ρ/2

Job j	n_j	Machine group m 1 τ	1 t	1 k	2 τ	2 t	2 k	P_j
1(A)	200	0.40	80	1	0.50	100	2	180
2(B)	150	0.27	40	1	0.40	60	2	100
L_m			**120**			160		**280**

Table 39: Engineering Database of Case No 3: FSS/3/3/1/ρ/2

Job j	n_j	Machine group m 1 τ	1 t	1 k	2 τ	2 t	2 k	3 τ	3 t	3 k	P_j
1(A)	100	0.40	40	1	0.40	40	2	0.20	20	3	100
2(B)	150	0.40	60	1	0.40	60	2	0.27	40	3	160
3(C)	150	0.67	100	1	0.53	80	2	0.40	60	3	240
L_m			**200**			180			120		**500**

Table 40: Engineering Database of Case No 4: FSS/3/3/2/ρ/2

Job j	n_j	Machine group m 1 τ	1 t	1 k	2 τ	2 t	2 k	3 τ	3 t	3 k	P_j
1(A)	100	0.40	40	1	0.40	40	2	0.20	20	3	100
2(B)	150	0.40	60	1	0.40	60	2	0.27	40	3	160
3(C)	150	0.53	80	1	0.67	100	2	0.40	60	3	240
L_m			180			**200**			120		**500**

Table 41: Engineering Database of Case No 5: FSS/3/4/2/ρ/3

Job j	n_j	Machine group m 1			2			3			P_j
		τ	t	k	τ	t	k	τ	t	k	
1(A)	100	0.40	40	1	0.80	80	2	0.60	60	3	180
2(B)	150	0.27	40	1	0.40	60	2	0.40	60	3	160
3(C)	150	0.40	60	1	0.53	80	2	0.53	80	3	220
4(D)	200	0.40	80	1	0.5	100	2	0.5	100	3	280
L_m			220			**320**			300		**840**

Table 42: Engineering Database of Case No 6: FSS/4/4/3/p/3

Job j	n_j	Machine group m 1			2			3			4			P_j
		τ	t	k	τ	t	k	τ	t	k	τ	t	k	
1(A)	200	0.50	100	1	0.40	80	2	0.50	100	3	0.40	80	4	360
2(B)	250	0.24	60	1	0.32	80	2	0.40	100	3	0.24	60	4	300
3(C)	300	0.27	80	1	0.20	60	2	0.27	80	3	0.13	40	4	260
4(D)	250	0.24	60	1	0.24	60	2	0.40	100	3	0.16	40	4	260
L_m			300			280			**380**			220		**1180**

Table 43: Engineering Database of case No 7: FSS/5/4/5/U/4

Job j	n_j	Machine group m 1			2			3			4			5			P_j
		τ	t	k	τ	t	k	τ	t	k	τ	t	k	τ	t	k	
1(A)	100	0.60	60	1	1	100	2	0.80	80	3	0.80	80	4	1	100	5	420
2(B)	150	0.40	60	1	0.53	80	2	0.40	60	3	0.40	60	4	0.53	80	5	340
3(C)	150	0.27	40	1	0.40	60	2	0.53	80	3	0.40	60	4	0.67	100	5	340
4(D)	200	0.20	40	1	0.30	60	2	0.20	40	3	0.20	40	4	0.40	80	5	260
L_m			200			300			260			240			**360**		**1360**

7.5.3 Application of BTM for Flow shop case studies

Using LEKIN computer program [12] and applying BTM for the 7 case studies using another computer program given in [58]. The Gantt charts of the 7 cases are illustrated in Figures (61-66) and Tables (44-49).

By applying the BTM to the case studies using FIFO and MS rules and computing the values of $N^*, t^*_{pr}, C^*_r, \rho^*$.

Table 44: Makespan and excess time coefficient values of Case No 1

N	FIFO,MS t_{pr}	$C_{r,F}$
1	226	1
2	210	0,929204
3	**207**	**0,915929**
4	208	0,920354
5	210	0,929204
6	213	0,942478
7	211	0,933628
8	215	0,951327

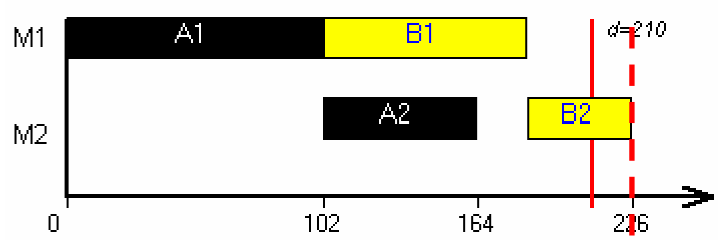
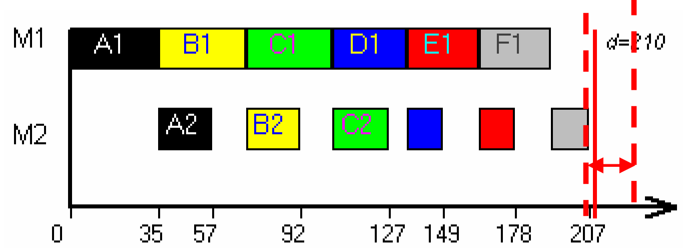


Figure 61: a) FSS/FIFO Gantt chart of initial schedule for case No1



b) FSS/FIFO Gantt chart of lot streaming schedule for case No 1, N=3

Table 45: Makespan and excess time coefficient values of Case No 2

N	FIFO,MS	
	t_{pr}	$C_{r,F}$
1	246	1,194175
2	210	1,019417
3	200	0,970874
4	**198**	**0,961165**
5	**198**	**0,961165**
6	201	0,975728
7	202	0,980583
8	204	0,990291

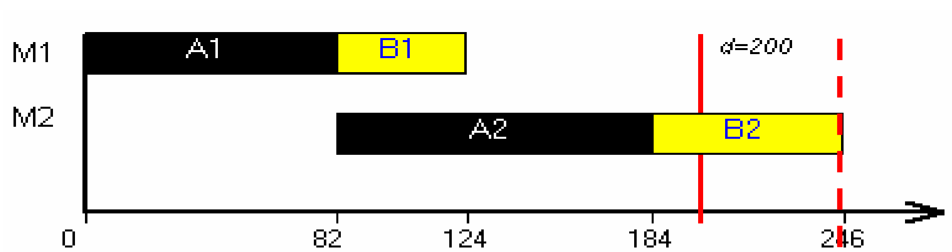
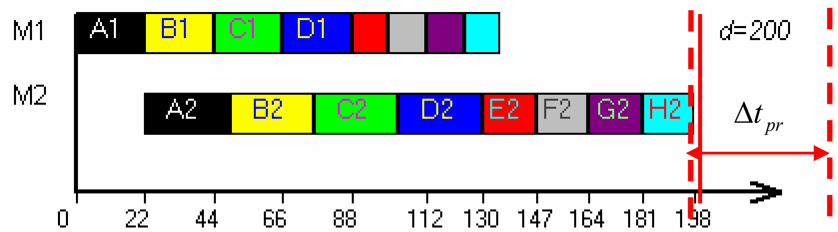
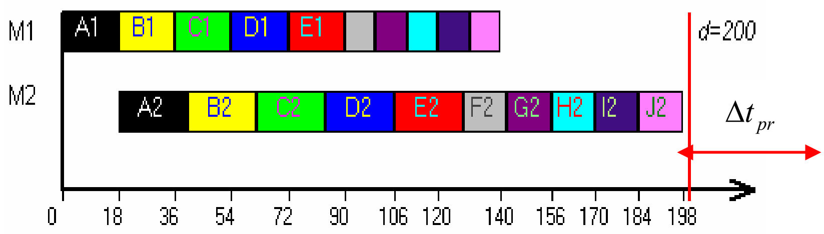


Figure 62: a) FSS/FIFO Gantt chart of initial schedule for case No 2



b) FSS/FIFO Gantt chart of lot streaming schedule for case No 2, N=4



c) FSS/FIFO Gantt chart of lot streaming schedule for case No 2, N=5

Table 46: Makespan and excess time coefficient values of Cases No 3, 4

N	*FIFO*		*MS*	
	t_{pr}	$C_{r,F}$	t_{pr}	$C_{r,F}$
1	*350*	*1,296296*	*310*	*1,148148*
2	*286*	*1,059259*	*266*	*0,985185*
3	*267*	*0,988889*	*254*	*0,940741*
4	*263*	*0,974074*	***253***	***0,937037***
5	***262***	***0,97037***	*254*	*0,940741*
6	*267*	*0,988889*	*260*	*0,962963*
7	*269*	*0,996296*	*263*	*0,974074*
8	*270*	*1*	*264*	*0,977778*

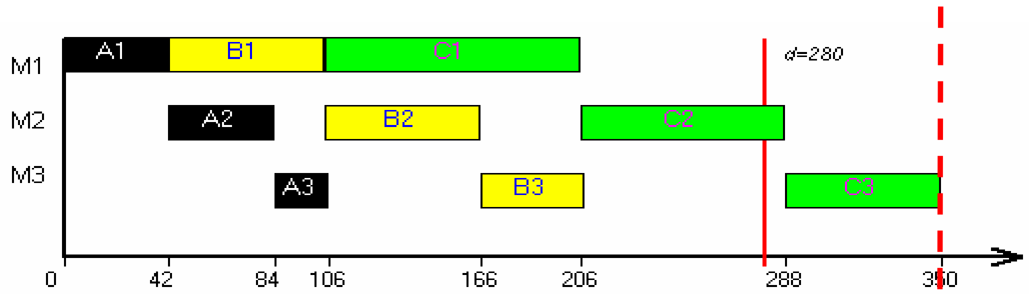
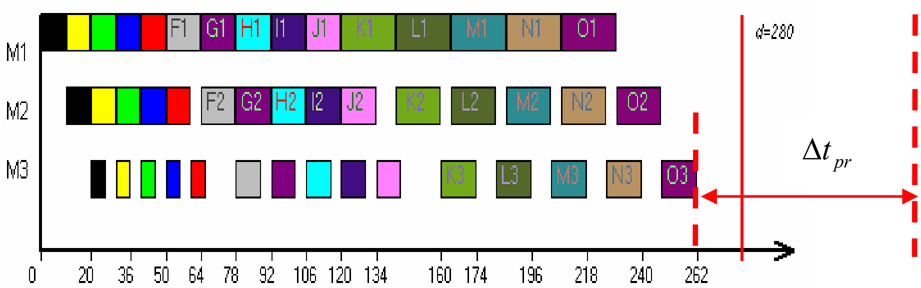


Figure 63: a) FSS/FIFO Gantt chart of initial schedule for case No 3



b) FSS/FIFO Gantt chart of lot streaming schedule for case No 3, N=5

Table 47: Makespan and excess time coefficient values of Case No 5

N	*FIFO*		*MS*	
8	*439*	*1,002283*	*440*	*1,004566*
	t_{pr}	$C_{r,F}$	t_{pr}	$C_{r,F}$
1	*478*	*1,091324*	*498*	*1,136986*
2	*420*	*0,958904*	*420*	*0,958904*
3	*409*	*0,93379*	*410*	*0,936073*
4	*409*	*0,93379*	*409*	*0,93379*
5	*414*	*0,945205*	*414*	*0,945205*
6	*420*	*0,958904*	*419*	*0,956621*
7	*425*	*0,97032*	*425*	*0,97032*

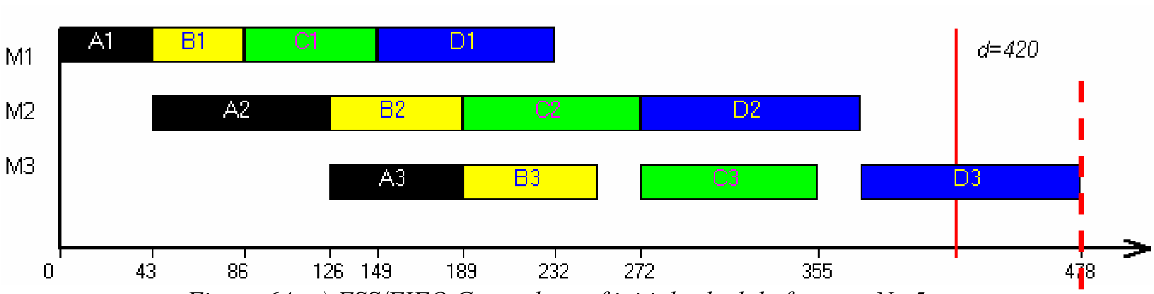
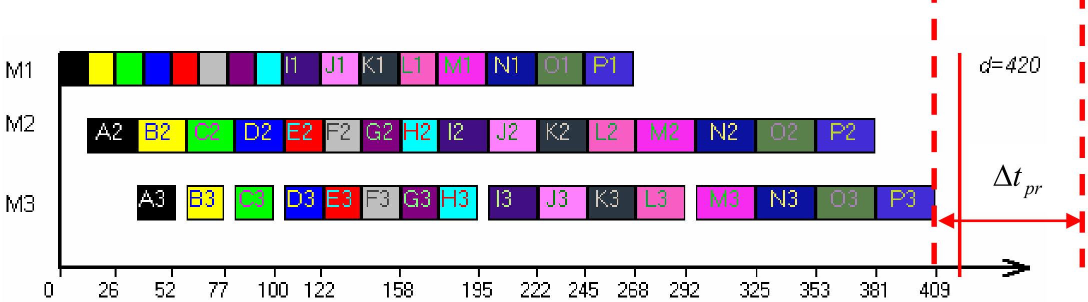


Figure 64: a) FSS/FIFO Gantt chart of initial schedule for case No 5



b) FSS/FIFO Gantt chart of lot streaming schedule for case No 5, N=4

Table 48: Makespan and excess time coefficient values of Case No 6

N	*FIFO,MS*	
	t_{pr}	$C_{r,F}$
1	*621*	*1,106952*
2	*523*	*0,932264*
3	*496*	*0,884135*
4	*492*	*0,877005*
5	*493*	*0,878788*
6	*502*	*0,894831*
7	*495*	*0,882353*
8	*500*	*0,891266*

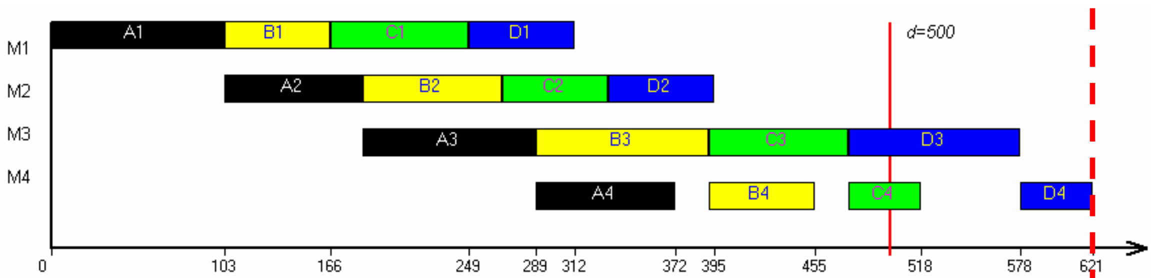
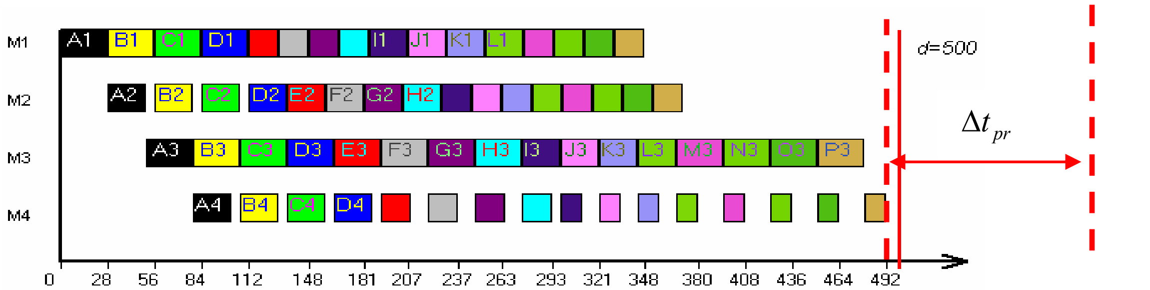


Figure 65: a) FSS/FIFO Gantt chart of initial schedule for case No 6



b) FSS/FIFO Gantt chart of lot streaming schedule for case No 6, N=4

Table 49: Makespan and excess time coefficient values of Case No 7

N	*FIFO,MS* t_{pr}	*FIFO,MS* $C_{r,F}$
1	*712*	*1,244755*
2	*568*	*0,993007*
3	*531*	*0,928322*
4	*520*	*0,909091*
5	*520*	*0,909091*
6	*525*	*0,917832*
7	*523*	*0,914336*
8	*536*	*0,937063*

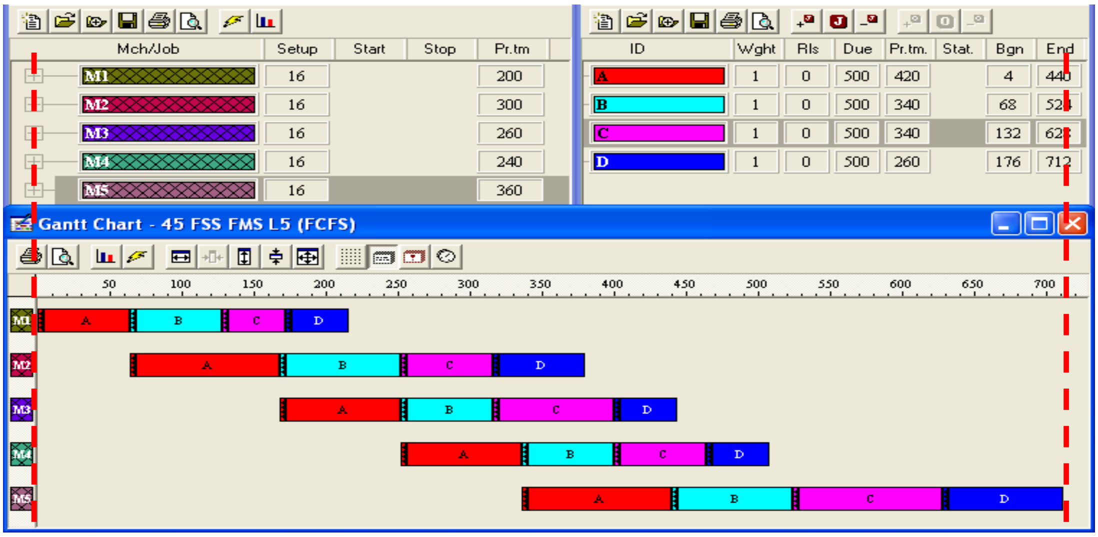
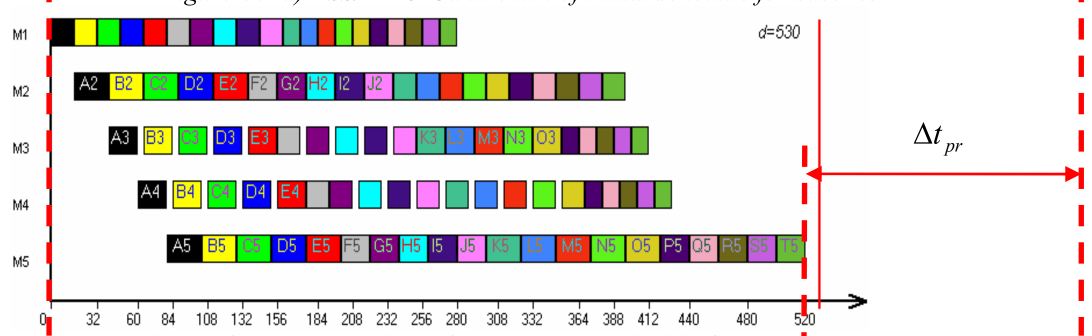


Figure 66: a) FSS/FIFO Gantt chart of initial schedule for case No 7



b) FSS/FIFO Gantt chart of lot streaming schedule for case No 7, N=5

7.5.4 Results of BTM Applications for flow shop systems

The results of application of *BTM* for the seven cases are given in Tables (50, 51) and illustrated in Figure 57.

Table 50: Makespan values of 7-case studies using BTM

	Case1	***Case 2***	***Case 3,4***		***Case 5***		***Case 6***	***Case 7***
N	*FIFO,MS*	*FIFO,MS*	*FIFO*	*MS*	*FIFO*	*MS*	*FIFO,MS*	*FIFO,MS*
1	*226*	*246*	*350*	*310*	*478*	*498*	*621*	*712*
2	*210*	*210*	*286*	*266*	*420*	*420*	*523*	*568*
3	***207***	*200*	*267*	*254*	***409***	*410*	*496*	*531*
4	*208*	***198***	*263*	***253***	***409***	***409***	***492***	***520***
5	*210*	***198***	***262***	*254*	*414*	*414*	*493*	***520***
6	*213*	*201*	*267*	*260*	*420*	*419*	*502*	*525*
7	*211*	*202*	*269*	*263*	*425*	*425*	*495*	*523*
8	*215*	*204*	*270*	*264*	*439*	*440*	*500*	*536*

Table 51: Excess time coefficient values of 7-case studies using BTM

	Case1	**Case 2**	**Case 3,4**		**Case 5**		**Case 6**	**Case 7**
N	*FIFO,MS*	*FIFO,MS*	*FIFO*	*MS*	*FIFO*	*MS*	*FIFO,MS*	*FIFO,MS*
1	*1*	*1,194175*	*1,296296*	*1,148148*	*1,091324*	*1,136986*	*1,106952*	*1,244755*
2	*0,929204*	*1,019417*	*1,059259*	*0,985185*	*0,958904*	*0,958904*	*0,932264*	*0,993007*
3	***0,915929***	*0,970874*	*0,988889*	*0,940741*	***0,93379***	*0,936073*	*0,884135*	*0,928322*
4	*0,920354*	***0,961165***	*0,974074*	***0,937037***	***0,93379***	***0,93379***	***0,877005***	***0,909091***
5	*0,929204*	***0,961165***	***0,97037***	*0,940741*	*0,945205*	*0,945205*	*0,878788*	***0,909091***
6	*0,942478*	*0,975728*	*0,988889*	*0,962963*	*0,958904*	*0,956621*	*0,894831*	*0,917832*
7	*0,933628*	*0,980583*	*0,996296*	*0,974074*	*0,97032*	*0,97032*	*0,882353*	*0,914336*
8	*0,951327*	*0,990291*	*1*	*1,148148*	*1,002283*	*1,004566*	*0,891266*	*0,937063*

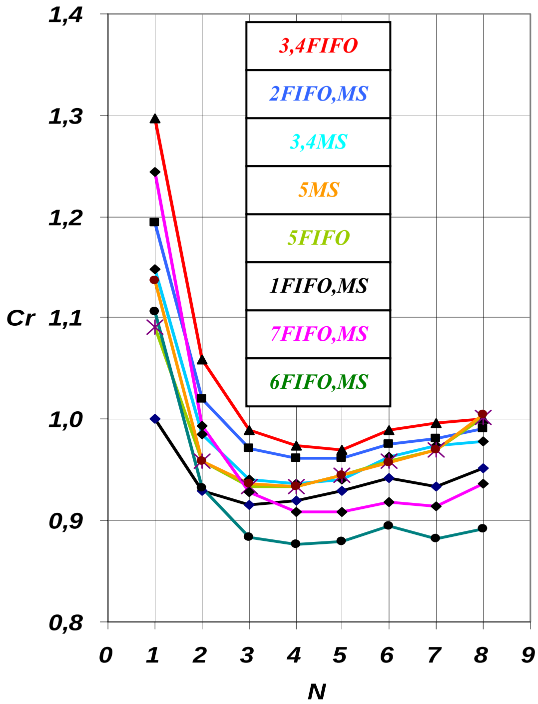


Figure 67: Excess time coefficient curves of FSS 7-case studies using BTM

Table 52: Productivity improvement rate by Lot streaming efficiency using BTM

Case	*1*	*2*	*3,4*		*5*		*6*	*7*
			FIFO	*MS*	*FIFO*	*MS*		
LSE_{max} %	*9.178*	*24.24*	*33.587*	*22.529*	*16.87*	*21.76*	*26.219*	*36.923*

From Table 52 and Figure 67, the BTM is used for the given flow shop lot streaming case studies, the productivity of the systems is improved by different rates from *9.178 %* to *36.923 %.* The improvement value depends on the structure of the system and on the engineering database of the problem.

7.6 Comparing the Results of Applications of BTM and JSA for flow shop systems

By the substitution of the given values into the JSA Formulas (132, 134, 139, 140, 141) we can get the results as given in Tables (53, 54, 55).

The values of the global minimum of production time are given in Table 53.
The initial values of makespan, excess time coefficient and utilization using JSA are given in Table 54.

The optimal values of the case studies are attained by JSA Formulas (80, 117, 81, 82, 16, 121) are presented in Table 55 and can be explained as follows:

Cases 1, 2

Cases 1 and 2 have the same number of machine group, number of jobs to be processed, setup times and total production time but their results are different such as maximum production times (makespan), optimal number of sub-batches and the optimal excess time coefficients.

Case 3, 4

Cases 3 and 4 have different bottleneck index and different operation order time but their results are the same.

Case 5

The results of the two rules FIFO and MS are different; the values of makespan and excess time coefficient of FIFO are lower than that of MS.

For FIFO rule we obtain two optimal values of the number of sub-batches whereas we obtain one optimal value by MS.

The utilization value of FIFO is higher than that of MS. In contrast, the productivity improvement rate of MS is higher than that of FIFO.

Case 6

Comparing to the previous cases, this case has higher number of machine groups and lower utilization but the lowest (best) excess time coefficient.

Case 7

This case is much more complicated (5 machine groups/ 4 jobs). The utilization value is the lowest but the rate of productivity improvement has the highest value.

Table 53: Global minimum of production time Values for 7-case studies using JSA

Case	Job	$L_{up,m}$	$L_{dn,m}$	Σ	MinΣ	L_b	S_b	S_{nb}	$t_{g,F}$
1	A	0	40	40	**40**	180	4	2	**226**
	B	0	60	60					
2	A	80	0	80	**40**	160	4	2	**206**
	B	40	0	40					
3	A	0	40+20	60	**60**	200	6	4	**270**
	B	0	60+40	100					
	C	0	80+60	140					
4	A	40	20	60	**60**	200	6	4	**270**
	B	60	40	100					
	C	80	60	140					
5	A	40	60	100	**100**	320	12	6	**438**
	B	40	60	100					
	C	60	80	140					
	D	80	100	180					
6	A	100+80	80	260	**160**	380	12	9	**561**
	B	60+80	60	200					
	C	80+60	40	180					
	D	60+60	40	160					
7	A	60+100 80+80	0	320	**180**	360	16	16	**572**
	B	60+80 60+60	0	280					
	C	40+60 80+60	0	240					
	D	40+60 40+40	0	180					

Table 54: Initial values of makespan, excess time coefficient and utilization using JSA

Case	t_{pr}		$C_{r,F}$		ρ	
	FIFO	MS	FIFO	MS	FIFO	MS
1	226	226	1	1	63.716	63.716
2	246	246	1.194	1.194	58.53	58.53
3	350	310	1.296	1.148	49.333	55.698
4	350	310	1.296	1.148	49.333	55.698
5	478	498	1.091	1.136	61.087	58.63
6	621	621	1.106	1.106	49.436	49.436
7	712	712	1.244	1.244	34.157	34.157

From the results given in Table 55 it is concluded that JSA method is an effective method to solve different cases of flow shop lot streaming problems. The productivity improvement rate is up to *36.897%.*

Table 55: Optimal values of FSS 7-case studies using JSA

Case	I_0		N^*		Int. N^*		t^*_{pr}		$C^*_{r,F}$		ρ^* %		$LSE_{\max}$ %	
	FIFO	MS	FIFO	MS	FIFO	MS	FIFO	MS	FIFO	MS	FIFO	MS	FIFO	MS
1	0	0	3.162	3.162	3	3	207	207	0.9159	0.9159	69.565	69.565	9.179	**9.179**
2	40	40	4.472	4.472	4-5	4-5	198	198	0.9611	0.9611	72.727	72.727	24.255	24.255
3	80	40	4.830	4.082	5	4	262	253	0.9703	0.9370	65.90	68.24	33.59	22.517
4	80	40	4.830	4.082	5	4	262	253	0.9703	0.9370	65.90	68.24	33.59	22.517
5	40	60	3.415	3.651	3-4	4	409	414	0.9337	0.9452	71.39	70.53	16.866	20.296
6	60	60	4.2817	4.2817	4	4	492	492	0.8770	0.8770	62.39	62.39	26.203	26.203
7	140	140	4.472	4.472	4-5	4-5	520	520	0.9090	0.9090	46.76	46.76	36.897	**36.897**

Table 56: Optimal values of makespan and excess time coefficient using BTM and JSA

Case	Rule	N^*(JSA)	N^*(BTM)	C^*_r(JSA)	C^*_r(BTM)
1	FIFO,MS	3	3	0.9159	0,9159
2	FIFO,MS	4 -5	4 -5	0.9611	0,9611
3	FIFO	5	5	0,9703	0,9703
	MS	4	4	0.9370	0,9370
4	FIFO	5	5	0,9703	0,9703
	MS	4	4	0.9370	0,9370
5	FIFO	3 - 4	3 - 4	0.9337	0,9337
	MS	4	4	0,9452	0,9337
6	FIFO,MS	4	4	0.8770	0,8770
7	FIFO,MS	4 - 5	4 -5	0.9090	0,9090

Generally, from Table 56 and Figure 68 we can conclude that the optimal numbers of sub-batches for the different cases *N = 3, 4 or 5* sub-batch.
The optimal values of excess time coefficient are lower than 1.

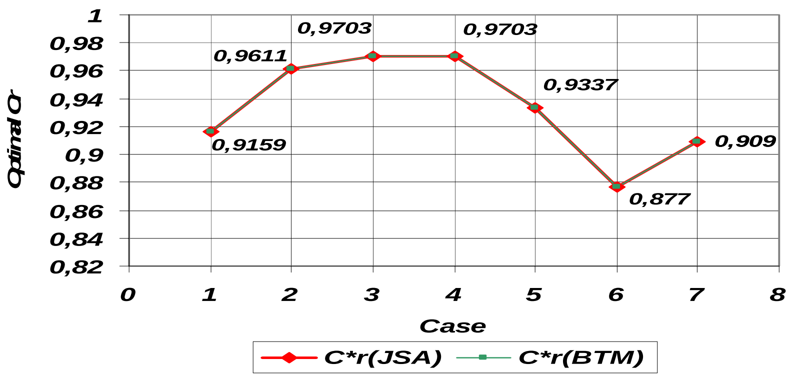


Figure 68: Optimal excess time coefficient of FSS/FIFO using BTM and JSA

When we compare the values of application of both methods BTM and JSA for the given seven cases of FSS lot streaming problems the same results are achieved.

8. Conclusions and Recommendations

8.1 Conclusions

It can be concluded that lot streaming can be used for the solution of FMS scheduling problems with great effectiveness.
Like the results of scheduling in general, the effectiveness of lot streaming depends on the structure and quantities of the database.
It seems to us that the effectiveness of lot streaming for flow shop systems is even better than for job shop systems. Furthermore, lot streaming is effective for all flow shop systems whereas for some cases of job shop systems, lot streaming is not effective where excess time coefficient $C_r = 1$.
In flow shop systems, the value of excess time coefficient could be equal one for two cases: at the initial feasible schedule and lot streaming schedule, whereas in job shop systems this value could be achieved only at the initial feasible schedule. In flow shop systems, when we use lot streaming the value of excess time coefficient could be lower than one, whereas in job shop systems that value never has a lower value than one even when we use lot streaming.

There is a difference between JSS and FSS problems in determination of the global minimum of production time; for JSS problems the production time on any of the machine groups may give the global minimum the production time of the system. The situation for FSS problems is totally different. To determine the global minimum of the production time, not only the load times on a given bottleneck machine group should be considered but also the minimum of production time belonging to upstream and downstream machine groups of bottleneck machine group should be included.
The data applied in the case studies examples are quite general. So, it may be supposed that the results are widely applicable.
It is shown in this research that the two new proposed methods Break and Test Method and Joinable Schedule Approach can effectively be used to optimize lot streaming of FMS scheduling problems for both systems: Flow Shop Systems and Job Shop Systems.
Generally, the advantages of BTM and JSA can be characterized as follow:
1.It is possible to optimize lot streaming for the two systems Flow shop and job shop systems.
2.It is possible to improve the productivity, utilization and workflow acceleration (efficient schedule).
3.It is possible to improve the delivery reliability (effective schedule).

The applications of both methods highly depend on the selected scheduling priority rule such as First In First Out (FIFO) rule and Minimum Slack (MS) rule.
Break and Test Method is a general search method for solving lot streaming scheduling problems.
In many cases Break and Test Method leads to suitable results in a very simple and realizable way.
A disadvantage of Break and Test Method is that it consumes too much time, especially for large-scale problems even when using a high quality computer. It is not analytical method.
Joinable Schedule Approach (JSA) is a mathematical optimization method, valid for the specially stated conditions of lot streaming scheduling problems.
JSA is easy to use and it is possible to obtain quickly the optimal values of the systems parameters even for large-scale problems. The possibilities of use depend on the selected rule and on the joinability condition.
It seems that for many Flow shop system problems, the condition of joinability is satisfied and the feasible schedules are joinable schedules.
The analytical results obtained by JSA can easily be obtained for extended applications.
The optimization mathematical model of JSA can be used as a general optimization model of lot streaming for FMS scheduling problems for both flow shop systems with attached independent sequence setup time.
A difficulty of the JSA is that the optimal number sometimes is a real number. But, it seems to us that by rounding to integer, this difficulty can easily be solved (see: the case studies).

8.2 Recommendations

We concluded that the lot streaming methods proposed in this dissertation might improve the performance of any discrete production system in competitive environment.
There are a number of future research problems which could lead to more applications of the methods. A few questions to be cleared are given below:

1. How can one specify the optimum number of sub-batches when the setup time is detached?
2. How can one produce an optimum number of sub-batches in integer number without round off when we use Joinable Schedule Approach?
3. What is the relation and place of HDA and the proposed Break and Test Method and Joinable schedule Approach in the overall problem range?

References

1. Groover P., 1987, “Automation, Production Systems, and Computer Integrated Manufacturing”, PP.808.Prentice-Hall.
2. Rembold, B.O.Nnaji,A. Storr, 1993, “Computer Integrated Manufacturing and Engineering”, PP.640.Addison-Wesley.
3. French S., B.A., M.A., D.Phil., 1982.”Sequencing and Scheduling: An Introduction to the Mathematics of the Job-shop” ELLIS HORWOOD Ltd. England.
4. Bellman R., A. O. Esogbue and I. Nabeshima, 1982. “Mathematical Aspects of Scheduling and Applications” Pergamon Press.
5. Conway R.W., W.L. Maxwell, L.W. Miller, 1967. “Theory of Scheduling” Dover Publications, Inc. New York.
6. Morton T. E. and D. W. Pentico, 1993. “Heuristic Scheduling Systems” John Wiley & Sons, Inc. USA.
7. Parker R., 1995. “Deterministic Scheduling Theory” 1st Edition, Chapman& Hall.

8. Plenert G. and B. Kirchmier, 2000. “Finite Capacity Scheduling” John Wiley & Sons, Inc. USA.
9. T’kindt V., J. Billaut, 2002. “Multicriteria Scheduling, Theory, Models and Algorithms”, Springer, NewYork.
10. Brucker P., 2004. “Scheduling Algorithms” 4th Edition, Springer-Verlag, Berlin, Heidelberg, New York.
11. Pinedo M., 2001.“Scheduling Theory, Algorithms, and Systems”, 2nd Edition, Prentice Hall, New Jersey.
12. Pinedo M., 2005. “Planning and Scheduling in Manufacturing and Services” Springer series in Operations Research.
13. Gentilly, Trance, 2003.” ILOG OPL Studio 3.7, Language Manual ILOG”.
14. Chan F. and H. Chan, 2004. “A Comprehensive Survey and Future Trend of Simulation Study on FMS Scheduling” Journal of Intelligent Manufacturing, 15, 87-102.
15. F&H Simulations B. V., 2000.” Taylor Enterprise Dynamics Software for Modeling, Simulation, Visualization and Control”, The Netherlands.
16. Chen, J. and Steiner G., 1999,“Discrete lot streaming in two-machine flow shops” INFOR, 37, 160-173.
17. Harold t., .John a., Oliver s., 1975.“Manufacturing Organization and Management” , PP. 588.Prentic-Hall
18. Blazewicz J., K. H. Ecker, E. pesch, G. Schmidt, J. Weglarz, 2001, "Scheduling Computer and Manufacturing Processes". 2nd Edition, Springer-Verlag Berlin.
19. Baker K. R. 1995 “Lot Streaming in Two-Machine Flow Shop with Set up time” Annual of Operations Research 57.1-11
20. Ruben, R.A. and Mahmoodi, F., 1998 "Lot Splitting in Unbalanced production systems" Decision Science, 29, 921-949.
21. Ramasesh R. V., H. Fu, Duncan K. H. Fong, J. C. Hayya, 2000, " Lot streaming in multistage production systems" International Journal of Production Economics, 66, 199-211.
22. Baker K.R., D. Jia, 1993, "A comparative Study of Lot Streaming Procedures" OMEGA International Journal of Management Science 21(5), 561-566.
23. Low C., C. M. Hsu, Kai I Huang, 2004, "Benefits of lot splitting in job shop scheduling" International Journal Advanced manufacturing Technology, 24, 773-780.
24. Ganapathy C., S. Marimuthu, S. G. Ponnambalam, 2004,"Tabu Search and Simulated Annealing Algorithms for Lot Streaming in two-machine Flow shop"
25. Potts C.N. and L. N. Van Wassenhove, 1992, “Integrating Scheduling with Batching and Lot-sizing: a Review of Algorithms and Complexity” Operations Research Society V. 43, No. 5, pp. 395-406.
26. Liu S. C., 2003."A heuristic method for discrete lot streaming with variable sub-lots in a Flow Shop" International Journal Advanced manufacturing Technology,22, 662-668.
27. Dastidar S. G., R. Nagi, 2004. “Batch Splitting in an Assembly Scheduling Environment” International Journal of Production Economics 105, 372-384.
28. Glass C. A., Gupta J. N. D. and Potts C. N.1994 “Lot Streaming in Three-Stage production processes” European Journal of Operations Research, 75:378-394.
29. Baker K. R. and D. F. Pyke, 1990," Solution Procedure for the lot streaming problem" Decision Sciences, 21, 3; pp 475.
30. Vickson R. and B. Alfredsson. 1992. “Two and Three Machine Flow Shop Scheduling Problems with Equal Sized Transfer Batches” International Journal Production Research 30, 1551-1574.
31. Cornelius Leondes, 2000, "Computer-aided Design, Engineering, and Manufacturing ", 1st Edition, CRC press. USA.
32. Sarin S. C., P. Jaiprakash, 2007, "Flow Shop Lot Streaming" Springer, Science Business Media, LLC, Virginia, USA.
33. Reiter, S., 1966, " A System for Managing Job Shop production" Journal of Business, 39: 371-393.
34. Chang J.H., H.N. Chiu,. 2005 “A Comprehensive Review of Lot Streaming” International Journal of Production Research 43, 1515-1536,.
35. Baker K. R. and D. Trietsch, 1993, "Basic Techniques for lot streaming" operations research, v.41, 6, 1065-1076.

36. Chen J., G. Steiner, 1997," Approximation methods for discrete lot streaming in flow shops" operations research letters 21, 139-145..
37. Kropp, D. H., Smunt, T.L., 1990," Optimal and heuristic models for lot splitting in a flow shop" Decision Sciences, 21, 4, 691-709.
38. Baker K. R. 1995 “Lot Streaming in Two-Machine Flow Shop with Set up time” Annual of Operations Research 57.1-11.
39. Vickson R.G.1993 “Optimal Lot Streaming for multiple products in a two-machine flow shop” European Journal of Operational Research 85, 556-575.
40. Cetinkaya, F. C. 1994. “Lot Streaming in a Two-Stage Flow-Shop with Set up Processing and Removal times separated”. Journal of the Operational Research Society 45(12):1445-1455.
41. Kalir A. A. and Sarin S. C. 2003 ”Constructing near Optimal Schedules for the Flow Shop Lot Streaming problem with Sublot-Attached Setups". Journal of Combinatorial Optimization 7(1) 23-44.
42. Kalir A. A. and Sarin S. C. 2000. “Evaluation of the potential benefits of Lot Streaming in Flow Shop systems” International Journal of production economics 66.131-142.
43. Kalir A. A., Sarin S. C. 2001 “A near-optimal heuristic for the sequencing problem in multiple-batch flow-shops with small equal sublots”. Omega 29, 577-584.
44. Hall N. G., G. Laporte, E. Selvarajah, and C. Sriskandarajah, 2003 “Scheduling and Lot Streaming in Flowshops with no-wait process” Journal of Scheduling 6:339-354.
45. Zhang W., C. Yin, J. Liu, R. J. Linn, 2005 “Multi-job Lot Streaming to minimize the mean completion time in m-hybrid flow shops” International Journal of Production Economics 96, 189-200.
46. Subodha Kumar,T. P. Bagchi and C. Sriskandarajah, 2000 “Lot Streaming and Scheduling Heuristics for m-machine No-wait Flow shops” Computers and Industrial Engineering 38, 149-172. GA
47. Trietsch, D. 1987, " An efficient transfer lot sizing procedure for batch processing on several machines (working paper). Noval Postgraduate School.
48. Trietsch, D. 1987, "Optimal transfer lots for batch manufacturing: A basic case and extensions (working paper). Noval Postgraduate School.
49. Dauzere-Peres S. and J. Bernard Lasserre, 1993. “An Iterative Procedure for Lot Streaming in Job-shop Scheduling” Computer and Industrial Engineering, Vol. 25, no 1-4, pp. 231-234.
50. Chan F.T.S., T. C. Wong and P.L. Chan, (2004). “Equal Size Lot Streaming to Job shop Scheduling Problem using Genetic Algorithms”, proceedings of the IEEE International Symposium on Intelligent Control, Taiwan.
51. Bin J. and Peter B.L, (1999). “An Effective Optimization-Based Algorithm for Job Shop Scheduling with Fixed-size Transfer Lots” Jour. of Manufacturing Systems Vol. 18, no. 4.
52. Szendrovits A.Z., 1975, "Manufacturing cycle time determination for multi-stage economic production quantity model" Management Science, 22, 3, 298-308.
53. Dauzere-Peres S., J. Bernard Lasserre, 1993 “An Iterative Procedure for Lot Streaming in Job-shop Scheduling” Computer and Industrial Engineering, V. 25, Nos 1-4, pp. 231-234.
54. Dauzere-Peres S., J.-Bernard Lasserre 1997 ”Lot Streaming in Job-Shop Scheduling” Operations Research, 4, V. 45, pp. 584-595.
55. J. Somlo, 2001. “Hybrid Dynamical Approach Makes FMS Scheduling More Effective” Periodica Polytechnica Series Mechanical Engineering. V.45, No. 2, pp. 175-200
56. J. R. Perkins and P. R. Kumar, 1989. “Stable, Distributed, Real Time Scheduling of Flexible Manufacturing/ Assembly/ Disassembly Systems” IEEE transaction automatic control, V. 2.
57. J. Somlo, A. V. Savkin, 2006. “Periodic and Transient Switched Server Schedules for FMS” Robotics and Computer-Integrated-Manufacturing, 22, 93-112.
58. J. Somlo, A.V. Savkin, A. Anufriev, T. Koncz, 2004. “Pragmatic Aspects of the Solution of FMS Scheduling Problems Using Hybrid Dynamical Approach” Robotics and Computer-Integrated-Manufacturing 20, 35- 47.
59. Matveev, A. S., Savkin, A. V., 2000. “Qualitative Theory of Hybrid Dynamical Systems” Birkhauser, pp. 348.
60. Koncz T. 2002 “Scheduling of Production of Flexible Manufacturing Systems Using Hybrid Dynamical Method” Master of Science Thesis, Budapest University of Technology and Economics, Hungary.
61. Elemer K. Nagy, 2009 “Novel method to neutralize loops and equal-effect action sequences in resource-action-constraint models used in scheduling” Ph.D. Dissertation, Department of

Control Engineering and Information Technology, Budapest University of Technology and Economics” Budapest, Hungary.

62. William J. Stevenson, 2005, "Operation Management", 8^{th} Edition, international edition, McGraw-hill, USA. pp. 47-53.

63. J-M. Proth, 2007. „Scheduling: New trends in industrial enviroment” Annual Reviews in Control. Vol.31. Issue1, pp. 157-166.

64. G Chryssolouris, D Mavrikios, N Papakostas, D Mourtzis, G Michalos, K Georgoulias, 2009 “Digital manufacturing: history, perspectives, and outlook” Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture. Volume 223, Number 5, pp 451-462.

Related publications of the author

65. **Ezedeen Kodeekha**, 2004 "A New Method of FMS Scheduling using Optimization and Simulation" Proceedings 16th European Simulation Symposium, SCS Press.

66. **Kodeekha Ezedeen** and Janos Somlo**,** 2008 **“**Improvement of FMS Scheduling Efficiency by Optimal Lot Streaming for Flow Shop Problems”, Proceedings of 6th National Conference ME2008, Faculty of Mechanical Engineering , BME, 29-30 May. Budapest-Hungary.

67. **Kodeekha Ezedeen** and Janos Somlo 2007 “Improving FMS Scheduling by Lot Streaming” Journal of Periodica Polytechnica Series (BUTE), Mechanical Engineering, 51/1, 3-14 Budapest -Hungary.

68. **Kodeekha Ezedeen** and Janos Somlo, 2007, “Optimal Lot Streaming for a Class of FMS Scheduling Problems-Joinable Schedule Approach-” Proceedings of IFAC International Federation of Automatic Control Symposium CEFIS’07 on October 9-11.Istanbul- Turkey.

69. **Kodeekha Ezedeen,** 2007 “Brute Force Method for Lot Streaming in FMS Scheduling Problems” Proceedings of 11^{th} International Conference INES2007 at June 29 -July 1, pp.:179 – 184 on Intelligent Engineering Systems, , Budapest Hungary**.**

70. **Ezedeen Kodeekha,** 2008, "Case Studies for Improving FMS Scheduling by Lot Streaming in Flow-Shop Systems". International Journal of Applied Sciences**,** Vol. 5, No. 4, pp. 125-143

71. **Kodeekha Ezedeen,** and Janos Somlo J., 2008 “Optimal Lot Streaming for FMS Scheduling of Flow Shop Systems” proceedings of 12^{th} International Conference INES2008 Intelligent Engineering Systems, on February 25-29 2008. Miami, USA.

Appendix A

A.1 Scheduling in manufacturing

In industrialized nations, the manufacturing industry is the most important contributor to the economic health. The higher the level of manufacturing technologies in a country, the higher the standard of living of its people [1].

In order to improve the manufacturing technologies the performance of the basic functions of manufacturing system must be improved which are given in the following hierarchical structure (Figure 1).

a) ***Design***. This is a long-term strategic level function concerned with the following activities: Product design, Process design, Cost Engineering, Engineering analysis, and Market analysis.

b) ***Planning***. This is a medium-term tactic level function concerned with the following planning: Master Production Scheduling (MPS), Material Requirement Planning (MRP), Capacity Requirements Planning (CRP), Process planning (PP).

c) ***Scheduling***. This is a short-term operational level function concerned with the right allocation of machines to operations over time.

d) ***Control.*** This is a physical level function concerned with the monitoring, follow-up the actual operations according to the schedule, and corrective action if needed.

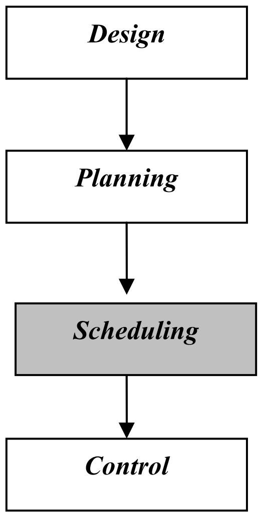


Figure 1: Basic functions of manufacturing systems

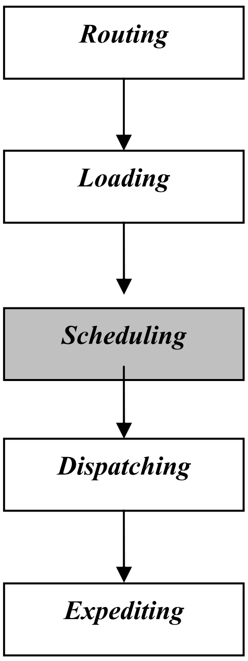


Figure 2: PPC functions

A.2 Scheduling in Production Planning and Control (PPC)

Production Planning and Control System (PPC) as sub-system of manufacturing system is an operational system concerned with the planning and control of the Work-In-Process (WIP).

Scheduling is an essential function not only on the planning level but also on the control level. It is a common function of both planning and control levels. In some literature, scheduling is used as production planning and control. In the typical PPC system, the role of PPC functions is defined as follows (Figure 2): Once the orders have been released and grouped into jobs, the route of processes of these jobs on the machines have to be specified by *routing function*. Once the route has been established, the load time required to perform the processes of the job has to be computed by *loading function*. Once the loads have been determined, the staring and finishing times for all jobs must be specified and schedule can be developed by *scheduling function* [2]. Once the schedule is planned, the schedule is put into effect to execution production processes by *dispatching function*. Actual performance is observed, recorded, evaluated and compared to the schedule. Decision is made (accept, corrective action or rescheduling) by *expediting function*.

In summary, the plan for the processing of materials through the plant is established by the functions of routing (where?), loading (how much time?) and scheduling (when?).

Scheduling engineer designs the right schedule to the various products that are to be made in the factory. Based on the schedule, dispatcher issues the individual work instructions to the machine operators to accomplish the processing of the parts. Expediter collects the information and writes certain reports (such as order status report, progress report and exception report). Even with the best schedules, things sometimes go wrong (e.g., machine breakdowns, material delayed). Expediter compares the actual progress against the schedule. For orders that fall behind, the expediter takes the necessary corrective action to complete the order on time [3].

A.3 Scheduling as tool of Control

A typical shop control system consists of the following three phases (Figure 3) [3].

a) ***Order Release*** phase provides the documentation needed to process a production order through the factory. The documents consist of rout sheet – material requisition – job cards – move tickets – parts list.

b) ***Order Scheduling*** phase assigns the production orders to the various work-centers in the plant. It executes the dispatching function. It prepares a dispatch list that indicates which production orders should be accomplished at the various work-centers. It also provides information about relative priorities of the different jobs.

The order scheduling is planned to solve two problems in production control.

i. The ***machine loading*** problem. To schedule a given set of orders in the factory, the orders must first be assigned to work-centers. Allocating the orders to the work-centers is referred to as machine loading.

ii. The ***job sequencing*** problem. Since the total number of production orders will probably exceed the number of work-centers, each work-center will have a queue of orders waiting to be processed. The question is: in what sequence should the orders be processed? Which is first? Answering this question is the problem in job sequencing. Job sequencing involves determining the order in which the jobs will be processed through a given work-center. To determine this sequence, priorities are established for the queue of jobs, and the jobs are processed in the order of their relative priorities.

c) ***Order progress*** phase monitors the status of the various orders in the plant, work-in-process, and other characteristics that indicate the progress and performance of production. The function of order progress module is to provide information that is useful in controlling the factory based on data collected from factory. The information is often summarized in the form of reports.

A.4 Information flow of scheduling

Scheduling process starts with a strategic plan and it interacts with other planning functions in the medium or higher level. Decisions made at a higher planning level may impact the scheduling process. Figure 4 illustrates a diagram of the information flow in a classic manufacturing system. When the materials and machines are available at specified times, schedule must be ready to dispatch. Scheduling is the last task of planning functions to determine an actual implementation plan for every job included in the process plan [4].

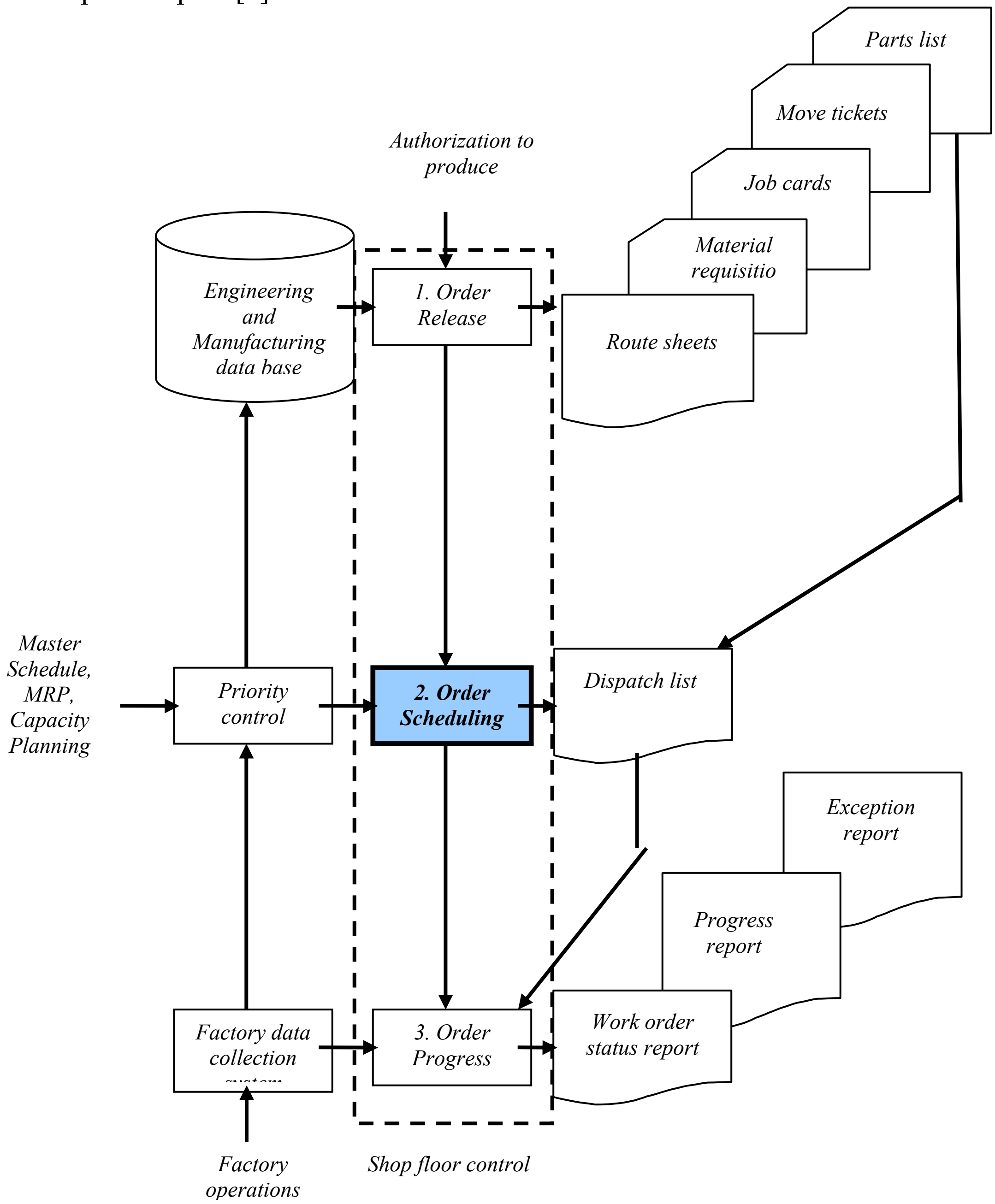


Figure 3: Three phases in a shop floor control system

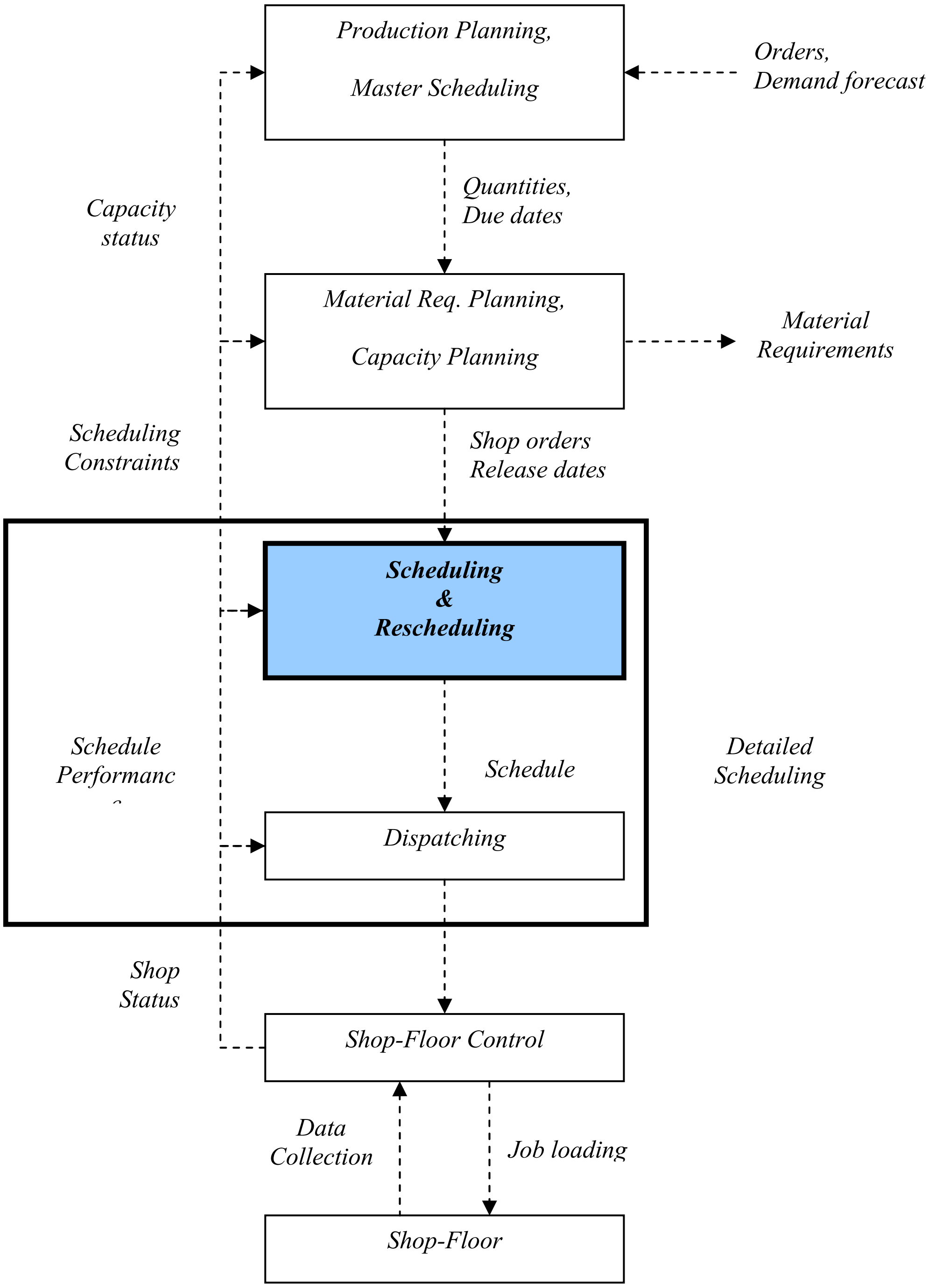


Figure 4: Information flow diagram in a manufacturing system

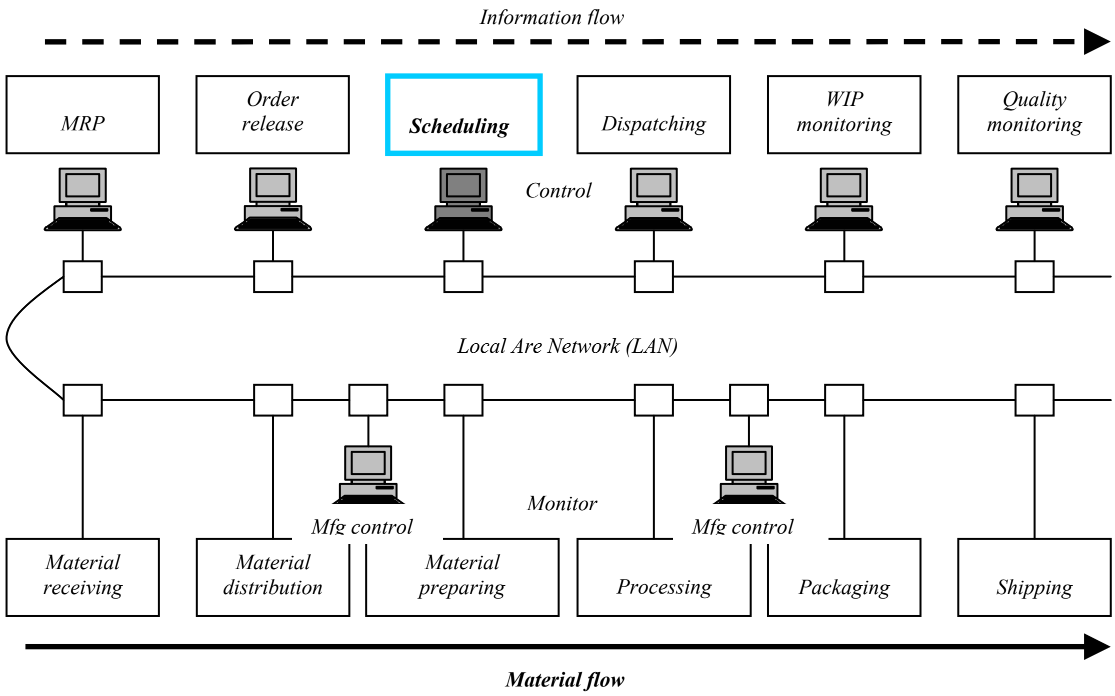


Figure 5: Information flow and material flow of CIM

In CIM, when the order is released it competes with several other orders of different delivery dates; for this reason, a proper schedule process must be designed. To assure that the material flow is at the right machine at the right time, a distributed computer system is used (Figure 5). The control function ensures that the information flow triggers the material flow in the correct sequence according to the predetermined schedule. Local area network (LAN) interconnects the various computing systems which coordinate and control the factory operations [5].

A.5. Scheduling system

We proposed the following diagram (Figure 6) for the elements of basic scheduling system. It consists of four elements as follows:

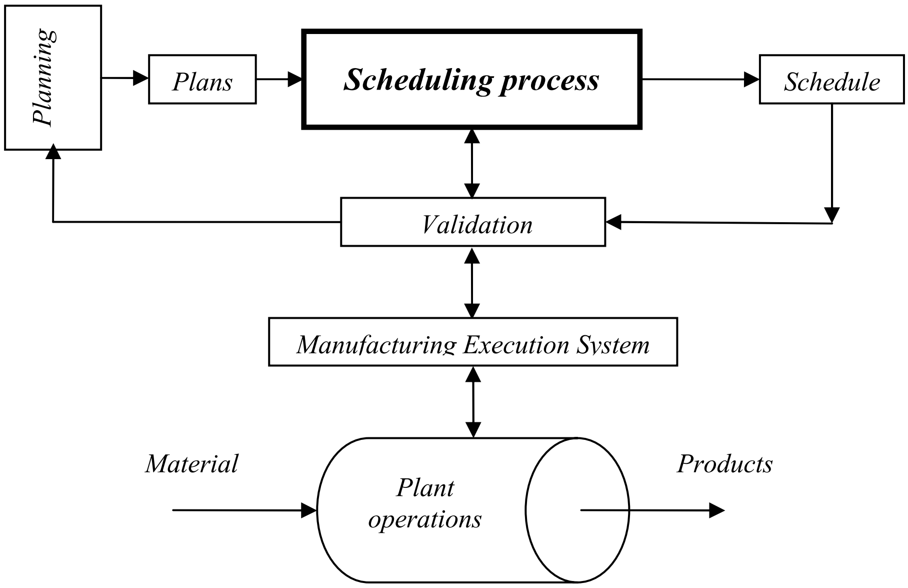


Figure 6: Elements of basic scheduling system

A.5.1 ***System input***

The input to the scheduling system is the collection of production plans. Specifically, the basic input data needed to process a schedule is: Job data, Machines data, Scheduling type and Objectives (Criteria).

A.5.2. ***Scheduling process***

The process of scheduling system consists of the following steps:

a) ***Modeling,*** In this step, the scheduling problem is formulated and simplified. Defined assumptions are presented, constraints and the objectives are specified.

b) ***Method***, a mathematical, heuristic, or/and simulation method is selected in this step according to the nature of the scheduling problem and specified objective.

c) ***Analyzing,*** In this step, the solution procedure is followed in order to achieve predetermined objectives. If the model fits into the specified method, an optimal solution may be obtained by using this method. If the mathematical relationships of the model are complex to permit analytic solution, then a heuristic method may be more appropriate. If the scheduling problem is too complicated, i.e. NP-hard problem, then a simulation method is used and the search for optimal solution must be replaced by the search for good solution.

A.5.3. System output

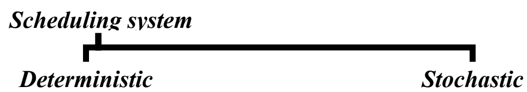
Output of the scheduling system are schedules.

A.5.4.Validation. Validation involves testing the schedule models using simulation to establish their validity. A common approach for testing is to compare its current performance with others. Validation is used as scheduling control step to evaluate and modify the generated schedules and make a corrective action if needed. The action that may be taken is a rescheduling action or replanning action.

Manufacturing Execution System (MES) is an information system that schedules, dispatches, tracks, monitors, and controls production on the factory floor. Such systems also provide real time linkages to MRP systems, product and process planning [6].

A.6. Scheduling classification (Figure 7)

There are many types of scheduling system; I propose the following classification:



A.6.1. Scheduling system classification according to the certainty degree

Scheduling system can be classified according to the certainty degree, if one or some of the input data of the scheduling system are of probabilistic uncertainty **(**random), the scheduling is called *Stochastic Scheduling,* if all input data are known in advance and have certainty; the system is called *Deterministic Scheduling.* Most scheduling problems are either deterministic or can be closely approximated by the deterministic models. The most common way to convert a stochastic problem into a deterministic problem is to work with the average values. Deterministic model is applicable to the most real-life scheduling problems. Deterministic and stochastic scheduling can be classified into two types**:**

a) ***Project Scheduling****,*

Project is defined as a combination of interrelated activities that must be executed in a particular order to complete an entire task. In some manufacturing environments, the size is large or/and weight of the product is heavy, this product remains in one location (fixed position) and the machines are brought to it, for example, aircraft assembly. The scheduling function of a project is called *Project Scheduling.* The most important techniques used to solve project scheduling problems are called *Program Evaluation and Review Technique (PERT)* and *Critical Path Method (CPM).*

Figure 7: Scheduling system classification

b) ***Machine Scheduling***

Machine scheduling is a scheduling of jobs to be processed through a single or multi-machine. It can be classified into the two following types:

b.1. Single-Machine Scheduling

It is concerned with the scheduling of single machine systems. This system consists of one machine *(m =1)* to process a number of different jobs *J* **(***1, 2, 3… j…J).* (Figure 8)

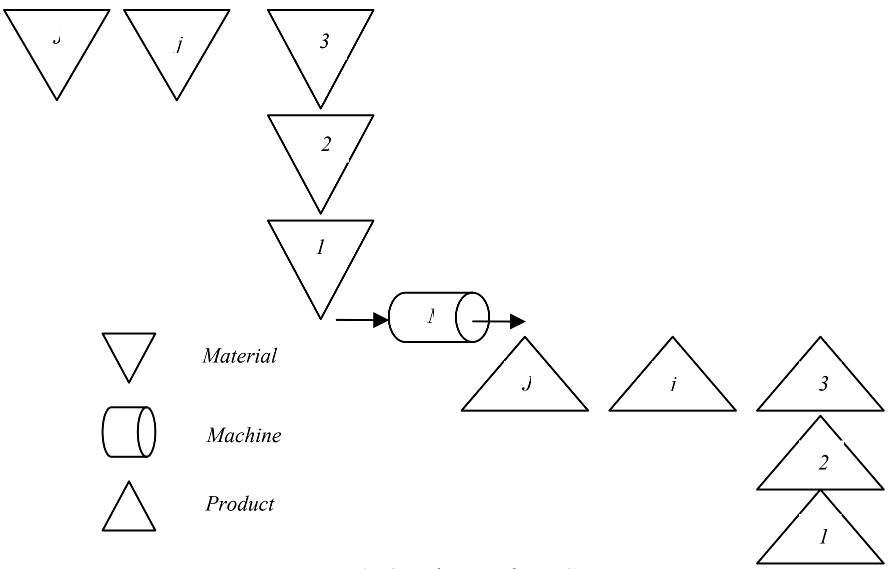


Figure 8: Single-Machine System

b.2. Multi-Machine Scheduling

It is concerned with the scheduling of multi-machine systems. This system consists of different machines group *(m =1, 2, 3…m…M)* to process a number of different jobs

(j=1, 2, 3… j…J). According to the type of route, there are two basic types of multi-machine scheduling are the following.
In this research, the FMS scheduling problem is considered for two types of FMS: Flow Shop Systems (FSS) and Job Shop Systems, (JSS)
FSS is a manufacturing system in which the routes of all jobs to be processed on certain machine groups are identical, i.e., all jobs visit the same machine in the same sequence.
JSS is a manufacturing system in which the routes of all jobs to be processed are different, i.e., each job has a particular route.

b.2.1. Flow Shop Scheduling

Flow shop scheduling is concerned with the scheduling of flow shop system **(FSS).** In FSS, the routes of all jobs to be processing are identical.

b.2.2. Job Shop Scheduling

Job shop scheduling is concerned with the scheduling of job shop systems **(JSS).** In JSS, each job requires *m* operations, one on each machine, in a specific order, but the order can be different for each job. JSS are more complicated. Jobs may not require all *M* machines; and they may visit the same machines more than once. The same machine may be the last machine for a particular job, or it may be an intermediate processing step. Any given machine may process new jobs arriving from outside the factory, and process jobs arriving from WIP inventory inside the factory.
The difference between Job shop and Flow shop systems are shown in Table 1.

Table 1: Types of classic manufacturing systems

	Job shop		***Flow shop***
1	*Vary*	*Route*	*Single*
2	*Process*	*Layout*	*Product*
3	*General*	*Machines*	*Special*
4	*Make to order*	*Type of production*	*Make to stock*
5	→	*Production rate*	→
6	→	*Flexibility*	→
7	→	*Demand rate*	→
8	←	*Product variety*	←
9	←	*Capital investment*	←
10	←	*System reliability*	←
11	←	*WIP*	←

A.6.2. Scheduling system classification according to the job arrival pattern

According to the job arrival pattern, job shop scheduling can be classified into two types: when the jobs arrive simultaneously and no extra jobs will arrive before these jobs are completed, the scheduling is called *Static Scheduling.* When the jobs arrive intermittently (continually) and possibility of new job arrivals over time is allowed, the scheduling is called *Dynamic Scheduling*.

A.6.3. Scheduling system classification according to the starting date

If the scheduling starts from first operation or on the earliest date and each operation must be completed forward in time, the scheduling is called *Forward Scheduling.*
If the scheduling starts from last operation or some date in the future (usually, due or delivery date), the scheduling is called *Backward Scheduling.*

A.6.4. Scheduling system classification according to the active time

If the scheduling is designed as planning function and made ahead of time, the scheduling is called *Predictive (Off-line) Scheduling.*
Reactive (On-line or Real-time) Scheduling is a control function to adjust predictive schedules when unexpected events force changes (Rescheduling).
Predictive scheduling and reactive scheduling are working together. Predictive scheduling generates the schedules for reactive scheduling, and reactive scheduling generates feedback to predictive scheduling. The complement of Predictive scheduling is the reactive scheduling which can be regarded as deterministic scheduling with a shorter planning horizon.

A.6.5 Scheduling system classification according to the production quantity

Production quantity is crucial in determining the type of scheduling required.
Production quantity is defined as the total number of products to be produced. It can be produced in individual batches of various lot sizes. This quantity is collected from one job or multi- job. Job may consist of a single-product or multi-product (batch) to be produced. The multi-product jobs can be classified into three types Low-number of parts (LNP) job, Medium-number of parts (MNP) job and High-number of parts (HNP) job. The scheduling of multi-product jobs is called *batch scheduling.* The batch scheduling can be proposed to classify into three types: *LNP scheduling, MNP scheduling and HNP scheduling.*

A.6.6 Scheduling system classification according to Group Technology (GT)

GT is a manufacturing philosophy in which similar parts are identified and grouped together to take advantages of their similarities in manufacturing and design such as shape, dimension and/or process route. This is a technique to increase a production batch size [3]. A most significant advantage of applying GT to production activity is that setup time for each job is reduced, because several jobs are grouped and processed in a sequence, and the same jigs and tools may be used. With GT, the scheduling is simplified. This scheduling is called *group scheduling* [7].

A. 7 Scheduling Constraints

Any real scheduling problem is subjected to some constraints. A solution of a scheduling problem must always satisfy a certain number of constraints. Four kinds of feasibility constraints are commonly found in scheduling problems:

a) Processing constraint. Each machine must process only one process at time.
b) Capacity constraint. Each machine has certain capacity.

c) Precedence constraints. Each job must be processed according to a given route.
d) Job constraint. Each job must be processed on one machine at time.

A. 8 Graphical scheduling model (Gantt chart)

Gantt chart is a powerful method to solve simple scheduling problems. It has the answers for the following questions:
What will happen in the shop? What is ahead of each machine? How many machines are needed and how many jobs will be processed? When should start the jobs and operations and when will they be finished? Which job is the first and which is the last? How much time does each job need for processing? How many operations are needed for each job? What are the due dates of the jobs? How long is the schedule?

Gantt chart is a powerful tool for the control system. It has the answers for the following questions:
What have been achieved at a point in time? Which job is ahead and which is behind the schedule? It helps answer the question what type of corrective actions can be made if needed.

For example, Figure 9 is a JSS Gantt chart. Each job has a particular color. Each processing time has a particular index. For example, $P_{2,1,3}$ is the processing time of job 2 on machine 1 by operation 3. DK vertical dash color is used for showing idle times.

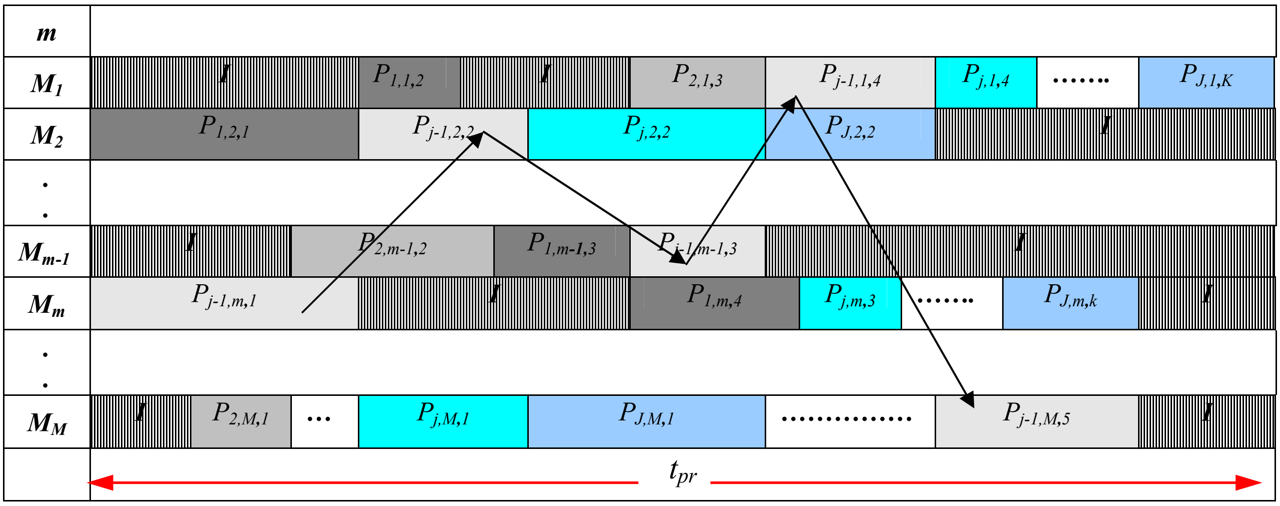


Figure 9: Gantt chart of (j, m) FMS scheduling of Job-Shop System

A. 9 Scheduling methods

The theory of scheduling has many methods for solving scheduling problems. Indeed, the scheduling field has become a central topic for research; creation, development, application, and evaluation of methods. No specific method to solve general scheduling problems. The selection of a proper method depends mainly on the type of the scheduling model, number of machines, number of jobs, the constraints, and the objective function. For this reason, it is important to study some scheduling methods.
Traditionally, many scheduling problems have been viewed as discrete optimization problems subject to certain constraints.
Generally, scheduling problems are solved mainly using the following methods:

A.9.1 Mathematical programming methods

Some scheduling problems can be defined as allocation problems. Mathematical programming methods **(**Operations Research (OR)) are usually appropriate to determine the optimal solutions for such problems. There are a set of mathematical programming methods in the literature. Here, we present some of these programming:

a) ***Linear Programming (LP).***

LP is an assignment method to solve a scheduling problem of J jobs that can be performed on *any* of M machines. LP has complete independence limitation and it is not applicable for some scheduling constraints such as precedence constraint.

b) ***Dynamic programming (DP).***

DP is applicable to solve some scheduling problems. It is based on Bellman's principle of optimality "*An optimal policy has the property that whatever the initial state and the initial decision are, the remaining decision must constitute an optimal policy with regard to the state of resulting from the first decision"*. It is a recursive or multistage optimization approach (means that optimizes on a step by step basis using information from the preceding steps). It proceeds by breaking the scheduling problem into smaller sub-problems called stages, and developing optimal solution for each stage. The stages are tied together through recursive relationships, so that when the last stage is calculated, the solution of the overall problem is determined.
The disadvantages of DP are: First, there is no standard form of solutions. Second, it has usually excessive computational requirements. Third, it is not particularly efficient when compared to other mathematical programming algorithms such as linear programming based solution methods.

c) ***Full Enumeration Method (FEM)***

Scheduling problems, usually, are combinatorial optimization problems.
FEM is an approach to generate all feasible schedules of the scheduling problem and then select the proper schedule according to the predetermined objective.
In job shop scheduling problems, if all jobs J are processed on all machines M, then there are $(J!)$ different schedules on each machine, so that, there are a total of $(J!)^M$ possible schedules. At least one of them must be the optimal schedule according to a certain measure of performance. This definitely exists and can, theoretically, be found in a finite number of computational iterations. However, it requires a lot of computations, particularly as the size of the scheduling problem becomes large, for example, how to schedule 5 jobs on 8 machines? To establish a schedule, we must decide which job to process first, second, and so on. As it was mentioned above, there are 5 choices for the job to process first. Once this job has been chosen, there are four jobs left to choose from for the second job to process. Then, there are three jobs to choose from for the third job in order; then, there are two jobs to choose from for the fourth job in order; and finally, the last remaining job is scheduled to run fifth. In other words, there are $(5!)^8 = 4.3*10^{16}$ schedules. This is a huge number.
In a flow shop scheduling problem, all J jobs must visit machines in the same sequence. It is convenient in this case to number machines in accordance with the processing order such that jobs visiting machine 1 first and M last. This is referred to as a *permutation schedule.* At time 0 the first job is started on machine 1. As soon as this operation is completed, the first job begins on machine 2 and the second job begins on machine 1. This process is continued until the last job finishes on machine M. The good news is that we have only to consider $J!$ schedules instead of $(J!)^M$. The bad news is that $J!$ still grows quickly by J.
FEM is a useful practical method to solve low-scale scheduling problems (J is small) but, of course, infeasible or impossible even with the fastest computer where there

may be several hundred or more jobs even if the number of machines is low. For this reason, we have to use partial enumeration methods called **Branch And Bound (BAB).**

d) ***Branch And Bound (BAB)***

This method is a partial tree search enumeration method. It is an iterative procedure for obtaining the optimal solution of any combinatorial optimization problem that has a finite number of feasible solutions by systematically examining the subsets of integer values of feasible solutions. It is a way to reduce the computational steps.
BAB is often described as a (Divide and Conquer) method. The basic idea of BAB is "dividing the set of all solutions into smaller subsets and search among promising subsets. Initially, the method breaks the set of all solutions into two mutually exclusive subsets. These two smaller are divided into smaller subsets and so on".
BAB procedure consists of two phases: first**, Branching** (process to determine how to divide the subsets). Second, **Bounding** (process to determine upper /or lower bounds of the best solution contained in any subset of solutions).
BAB has some disadvantages: First, it must be modified for the specific problem to be solved. Second, the computational requirements can be severe for large problems. Third, there is no guarantee that the solution can be obtained quickly. Because of these disadvantages, we have to use other scheduling methods such as heuristic rules.

A.9.2 Heuristic methods

Some scheduling problems can be formulated as LP, DP or BAB; they can be solved easily through the use of existing efficient algorithms. Other scheduling problems with hundreds or even thousands of jobs can be solved through polynomial time algorithms that are also efficient in short time on a computer.
Some Scheduling problems are NP-hard problems (which means that the amount of computation required to solve a scheduling problem increases as the exponential of the number of entities in the system) and computationally expensive. There are no simple algorithms that yield optimal solutions in a limited amount of computer time. It may be possible to formulate these problems as integer or disjunctive programs, but optimal solutions may require a huge amount of computer time. Since in practice that amount of computer time is not always available, an acceptable feasible solution that is sub-optimal is usually satisfying.
For this reason, heuristic methods are used as logic methods to find a "good-enough" or satisfactory solution in short time. The most important heuristic methods using in scheduling problems are presented as follows:

a) ***Scheduling Priority Rules (SPR)***

SPR are one of heuristics methods to specify which job should be selected for processing next from among a queue of jobs. SPR is a job-based method.
SPR are approximation method to solve large-scale scheduling problem (J is large). SPR are not optimization methods (because they simply do not evaluate all the available alternatives) and not guarantee the optimal solution (because the amount of error and degree of optimality is not known). SPR are self-explanatory,
SPR usually produce schedules with long production time and low system utilization.
In practice, SPR are necessity and probably the most frequently applied heuristics for solving FMS scheduling problems. The advantages of SPR are: simplicity, ease of implementation, low time complexity. The interesting question is: which rule to select to obtain good performance? In general, some of SPR are considered to be better than

the others. This depends on some factors, such as the measure of performance, degree of certainty, and job arrival pattern. A large number of SPR have appeared in research and in practice. In Table 2, some well-known rules:

Table 2: Some of common SPR

No.	*Rule*	*Description*
1	*FIFO*	*First In First Out*
2	*MS*	*Minimum Slack*
3	*LIFO*	*Last In First Out*
4	*SPT*	*Shortest Processing Time*
5	*LPT*	*Longest Processing Time*
6	*EDD*	*Earliest Due Date*
7	*LCR*	*Least Critical Ratio*
8	*LSU*	*Least Set-Up*
9	*HW*	*Highest Weight*
10	*FO*	*Fewest Operation*
11	*RANDOM*	*Random*

Research in SPR has been active for several decades, and many different rules have been studied in the literature. These rules can be classified in various ways. For example, static and dynamic SPR rules; Static SPR are time independent such as FIFO rule. Dynamic SPR rules are time dependent such as MS rule which are explained below:

i. First In First Out rule: FIFO rule

In this rule, the job that arrives first at the machine is selected and processed first. FIFO is the mostly used rule because it is a simple, easily to use and clear fair rule.

ii. Minimum Slack rule: MS rule

Slack time is defined as time remaining between the processing time of the job and scheduling time (or delivery due date). Slack time is obtained by subtracting the current time and the processing times from the scheduling time. Thus, a job with zero slack would have just enough time to be completed. The jobs are processed in the sequence of non-decreasing slack time. MS rule is a dynamic priority rule in which the priority of each job is a function of time. It is easy to determine.

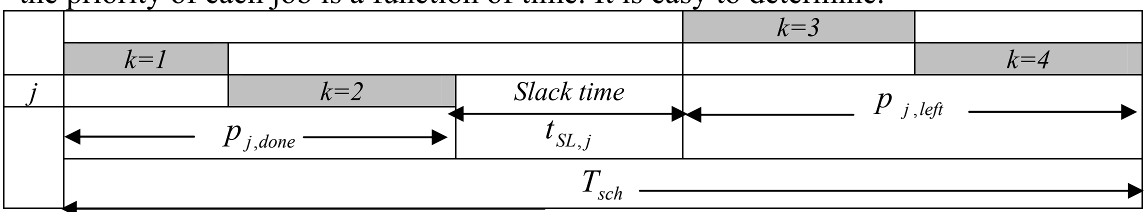


Figure 10: Slack time diagram.

MS rule can be used for both systems FSS and JSS as follows: (see Figure 10)

- Search the competition between the jobs to define which job to be processed first. Based on the competition, there are two cases:

a. If there is no competition, the job which is ready for processing is scheduled first.

b. If there is a competition. Specify the current time which is the time of operation *k* of job *j* on machine *m*, t_j. Then, find the slack time using the formula:

$$t_{SL,j} = T_{sch} - p_{j,left} - t_j \qquad (33)$$

Where, $t_{SL,j}$ is the slack time of job j

T_{sch} is scheduling time (or delivery due date).

$p_{j,left}$ is the remaining processing time of job j. It can be computed by

$$p_{j,left} = P_j - p_{j,done} \tag{34}$$

Where, P_j is the total processing time of the job j

$$P_j = \sum_{k=1}^{K_j} \tau_{j,m,k} \, n_j \tag{35}$$

$p_{j,done}$ is the processing time of the job j which is done. It can be computed by

$$p_{j,done} = \sum_{k=1}^{k_m - 1} \tau_{j,m,k} \, n_j \tag{36}$$

k_m is the actual number of operations of processing.

$$p_{j,left} = \sum_{k=1}^{K_j} \tau_{j,m,k} \, n_j - \sum_{k=1}^{k_m - 1} \tau_{j,m,k} \, n_j \tag{37}$$

$$p_{j,left} = \sum_{k=k_m}^{K_j} \tau_{j,m,k} \, n_j \tag{38}$$

In the initial condition, usually, the current time of first job on first machine is started with zero. $t_j = 0,\ p_{j,left} = P_j \rightarrow t_{SL,j} = T_{sch} - P_j$ (39)

- Arrange the jobs to be scheduled according to the non-decreasing ascending sequence. Job which has the minimum slack is the first.

b) Johnson's Rule (JR)

JR is a heuristic method to solve two machines flow-shop scheduling problem in order to reduce makespan. JR steps:

1. List processing time of each job on both machines.
2. Select the Shortest Processing Time (SPT) as follows:

- If SPT at the machine (1), schedule the job first.
- If SPT at the machine (2), schedule the job last.
- If SPT at both (tie), select randomly (arbitrarily).

3. Eliminate the job selected from further steps.
4. Repeat steps 2, 3 for each remaining jobs until the schedule is complete.

In case there is significant idle time at the second machine (from waiting for the job to be finished at the first machine), then lot splitting may be used.

A.9.3 Simulation methods

Simulation is a descriptive rather than an optimization approach that involves developing a model of real system and then performing experiments on that model.

In the simulation procedure, the model of the system is experimented by inserting decision variables values into the model under assumed conditions and then observing their effect on the criterion variable.

The objective of simulation model is to analyze and understand the behavior of the real system by running and testing the model under a variety of operating conditions.

Simulation models can be classified into the following two main categories:

a) Computer Simulation Approach (CSA)

With the increasing sophistication of computer hardware and software, one area which has grown rapidly for manufacturing systems is Computer Simulation

Approach (CSA). CSA generally refers to use a computer to perform experiments on a model of a real system. These experiments may be undertaken before the real system is operational.
The goal of use of CSA for manufacturing scheduling is to achieve the following objectives: First, represent visually manufacturing process behavior of a chosen schedule. Second, represent the manufacturing structure in high details. Third, evaluate the performance of manufacturing system.
CSA is able to perform a wide range of simulations for different types of scheduling problems such as dynamic, static, probabilistic and deterministic.
CSA is the most adaptable and flexible approach to study the behavior of the FMS under different schedules. By CSA, testing and validation of schedules can carried out to select the most proper schedule which can be successfully implemented.
In the increasingly competitive world of manufacturing, CSA has been accepted as a powerful tool for FMS scheduling.
Scheduling problems can be characterized as complex queuing problems which can be studied extensively via CSA. CSA can also be used in conjunction with statistical and mathematical programming techniques.
CSA particularly helps an engineer to generate the right schedule of given objective.
The FMS scheduling problems usually have two attributes:
Complexity: due to combinatorial behavior of FMS scheduling process which consists of different machines to process different jobs of different volumes in order to achieve usually multi-objective (large number of variables, constraints and objectives)
Uncertainty: due to stochastic nature such as jobs arrival and unexpected stoppages.
The analytical methods are difficult or almost impossible to solve FMS scheduling problem whereas CSA is an effective approach
Furthermore, the performance of a large number of scheduling priority rules (SPR) can be studied extensively using CSA. CSA can provide answers to the question: which rule should you choose? Many investigations can be carried out by CSA to determine the proper rule which meets certain performance measure.
Simulation models can be classified as continuous or discrete.
There are many types of simulation: first, *Continuous simulation* models are based on continuous mathematical equations with values for all points in time. Second, *Discrete simulation* occurs only at specific points in time. Discrete simulation can be triggered to run by units of time. This is called *Discrete Event Simulation*; points in between either have no value or can not be computed because of the lack of some sort of mathematical relationship to link the succeeding events.
In this research, Deterministic, Heuristic, and Discrete Event Simulation is used.

b) Artificial Intelligence (AI)

AI refers to that branch of simulation models that attempts to imitate purposeful behavior. *AI* is a simulation of human behaviors on the computer.
AI is that part of computer science and engineering of making intelligent computer programs. It is related to the similar task of using computers to understand human intelligence or behavior. The goal of *AI* is to simulate such human behaviors on the computer. The art of application of *AI* is known as knowledge engineering.
AI is essentially an extension of heuristic simulation. It is simultaneously the boldest and most difficult branch of simulation since it attempts to simulate human thinking.
AI is having a major effect on the scheduling of manufacturing operations.
In general, AI applications in manufacturing encompass the following approaches:
Generate and Test *(GT),* Constraint Propagation *(CP),* Simulated Annealing *(SA),* Tabu Search *(TS),* Genetic Algorithms *(GA),* Neural Network *(NN),* Fuzzy Logic

(FL), Machine Learning *(ML),* Expert Systems (*ES*), Natural Language (*NL*), Beam Search (*BS*).

The *AI* approaches could give effective solutions for complicated scheduling problems but these approaches are outside of the scope of this research.

A.9.4 Hybrid Dynamical Approach (HDA)

HDA gives an excellent opportunity to solve FMS scheduling problems, having the nice feature that the overall planning procedure can be reduced to the determination of one parameter, called demand rate coefficient. The proper choice of this parameter provides the fulfillment of production tasks. When using HDA it is supposed that the parts deliveries are (virtually) distributed and the parts arrive to (virtual) input buffers according to the given demand rates (determined with the above coefficient). The production is organized by control laws of switching nature. The production of parts is represented as reducing the content of (virtual) buffers, i.e. when a buffer becomes empty; a switch to produce other part is performed. Which part to produce next, is determined by the so called switching law. The most well-known switching laws are Periodic-Switching; Clear-the-Largest Buffer First; Clear-the-Largest-Work.

A.10 Some important definitions

Job chart is a chart to display the processing time required to process certain jobs.

Load chart is a chart to display the processing times of the machine groups.

Infeasible schedule is a schedule in which the jobs to be processed are conflicted due to unsatisfied some constraints.

Feasible schedule is a schedule in which the jobs to be processed are satisfied the specified one or more constraints.

Optimal schedule is a schedule in which the jobs to be processed are satisfied one or more constraints and achieved one or more objectives.

Appendix B

B.1 Productivity

Productivity, generally, is defined as the mathematical ratio of output to input of the production system.

European Productivity Agency (EPA) has defined productivity as "Productivity is an attitude of mind. It is the mentality of progress, of the constant improvements of that which exists. It is the certainty of being able to do better today than yesterday and continuously. It is the constant adaptation of economic and social life to changing conditions. It is the continual effort to apply new techniques and methods. It is the faith in human progress."

Productivity is an approach to creativity and ingenuity. It is a way to make manpower work smarter not harder.

Productivity improvement is required everywhere and every-time because it is an objective and a method. It is a desired objective to be achieved and, at the same time, it is a proper method which must be used to improve the performance of system.

Generally, productivity is a method to improve the quality of manufacturing activities such as plant design, product design, process design, layout, material handling and transporting, physical working condition.

Productivity value is absolutely different from production value. It is interrelated to each other. Production is the output of the system whereas productivity is the relative term of output to input. Increasing production may or may not increase productivity.

According to Relation (1) productivity can be increased by many ways, one of the most important way is by "Decrease input with same output".

B.1.1 Types of Productivity:

Based on the type of input used, the productivity can be classified as follows:

a) Total Productivity:

Total productivity is defined as the ratio of output to total inputs. It is expressed as:

$$Total\ productivity = \frac{Output}{Total\ Input} \tag{2}$$

This definition has some difficulties; it is difficult to measure the total productivity because the inputs (material, machine, energy…) have different values and different physical units. It is also difficult to specify the effect of each input individually. To overcome these difficulties we use another measure. It is called Partial productivity.

b) Partial productivity:

Partial productivity is defined as the ratio of output to individual input. It is expressed as:

$$Partial\ productivity = \frac{Output}{Individul\ input} \tag{3}$$

many partial productivity measures are used in the literature. Because FMS machines are generally considered as main resources, Furthermore, machine cost of FMS involves a high capital investment, we use machine productivity. It is expressed as follows:

$$Machine\ productivity = \frac{Production\ Quantity}{Machine\ Input} \quad (4)$$

There are many criteria to measure the machine productivity based on the type of machine input such as number of machines or total processing time.

B.2 Profitability

Profitability is defined as the ratio of the sales value to the cost spent. It can be expressed as

$$Profitability = \frac{Sales}{Cost} \quad (5)$$

The American Productivity Centre (APC) developed a mathematical model to express the relationship between profitability and productivity as follow:

$$Sales = output * unit\ price$$

$$Cost = input * unit\ cost$$

$$Profitability = \frac{Output}{Input} * \frac{unit\ price}{unit\ cost}$$

$$\frac{unit\ price}{unit\ cost} = price\ recovery \quad (6)$$

From equation (5) one gets

$$Profitability = Productivity * price\ recovery \quad (7)$$

According to relation (7), APC model demonstrates that the Profitability is a function of productivity and price recovery. Profitability can be increased by increasing productivity. There is a direct positive proportion between productivity and profitability. Profitability highly depends on the productivity.

At maximum makespan productivity, the makespan utilization is maximum and the production is accelerated faster. This usually leads to lower cost and larger sales, consequently, higher profitability.

Low makespan productivity indicates that the system has idle times. i.e. low utilization. The lower utilization leads to higher cost. Idle machines mean idle capital investment, consequently, lower profitability.

Also, low makespan productivity indicates usually long production time and slow production. This leads to lose a new production (i.e. sales), consequently, lower profitability.

Although FMSs have the highest machine productivity among the current manufacturing systems but, they still often have low makespan productivity and inefficient schedule.

To find efficient schedule, minimization of maximum production time (makespan) is a critical desirable objective. This objective is one of the main objectives of the present research.

B.3 Utilization

Utilization is defined as the degree of available capacity of the machine(s) under certain condition. There are many types of utilization based on the capacity type. Some types of capacity are:

- *Design capacity* is the maximum output per period under ideal conditions.
- *Planned capacity* is the maximum output per period under real conditions.
- *Actual capacity* is the maximum output per period under operating conditions.

There are many utilization criteria used in manufacturing systems such as:

$$Design\ utilization = \frac{Planned\,Capacity}{Design\ Capacity} \tag{8}$$

$$Planned\ utilization = \frac{Actual\,Capacity}{planned\,Capacity} \tag{9}$$

Appendix C

C.1. Why Lot Streaming?

"For the current competitive manufacturing environment, more and more emphasis is placed on reducing lead times for fulfilling orders of existing products and bringing new products to market. Therefore, compressing the manufacturing lead times has become an important research problem in planning and scheduling for a production environment. A technique known as *lot streaming* has become an important scheduling tool to help reduce the makespan (i.e. manufacturing lead time) in a batch production environment" [26].

"Although there have been advances in production control techniques for a multi-stage manufacturing environment, Material Requirements Planning (MRP) continues to remain popular in industry. However, MRP generally does not consider capacity constraints and assumes fixed production lead times that are not realistic. One can minimize the makespan by creating sub-lots and by overlapping the operations using frequent material movement. This process is known as *lot streaming* and is also helpful in reducing the total lead time and the total work-in-process (WIP)." [27].

C.2. Advantages of Lot Streaming

Lot streaming is used as a practical method to improve the performance of the manufacturing systems. A number of modern manufacturing systems, such as GT, JIT, and philosophy of Optimized Production Technology (OPT)/Synchronous Manufacturing /Theory Of Constraints (TOC) use lot streaming theory as integral part of them. Lot streaming concept was researched independently of JIT and *OPT* as extensions of MRP which may have the same idea of lot streaming under different names. Currently, it has been integrated and used in practical to JIT and OPT.

Also, lot streaming is used to reduce WIP and finished goods inventories (lower cost).

"Lot streaming is a technique which may improve customer service; each sub-batch can be delivered to the customer immediately upon finish, without waiting for the remaining sub-batches of the same job to be processed." [28].

Lot streaming increases the speed of the production flow, i.e. it accelerates the flow of the batch through a production process.

Currently, In the global competitive manufacturing environment, *lot streaming* is used as an effective technique to apply the Time-Based competition (TBC) Strategy and Make To Order (MTO) strategies so that to achieve the following advantages [8]: Reduce lot size, Reduce cycle time, Deliver on time and Increase productivity.

Generally, lot streaming in FMS is extremely important to realize the maximum utilization and speed up the production.

For these reasons, this technique has been studied over the past few years extensively.

This research is one of these studies in which a new direction is developed to solve complicated lot streaming FMS scheduling problems.

C.3. Lot Streaming and Preemption

There are two types of lot splitting processes that are conceptually worth distinguishing as follows:

- *Preemption* is a split process in which the batch is interrupted (broken off) during the production run so that to produce one or more urgent batches having a higher priority and no overlapping is allowed.
- *Lot streaming* is a split process in which the batches are broken into sub-batches for overlapping production.

In lot streaming, all orders can be considered as competitive urgent orders.

C.4. Lot Streaming, Process Batch and Transfer Batch

There are interrelations between lot streaming, process batch and transfer batch

- *Process batch* is defined as the production quantity of a product to be processed in a particular length of time.
- *Transfer batch* is the sublot to be transferred to next machine after processed on previous machine.
- *Lot streaming* is dividing the process batch into transfer batches for overlapping production.

C.5. Lot Streaming, Lot-Sizing, and Scheduling

Lot streaming, lot sizing, and scheduling are strongly interconnected by each other. Lot streaming is termed sometimes in the literature [30] as lot sizing. But, they are definitely absolute different.

- *Lot sizing* can be considered as one of production planning function of finding the optimal lot sizes to be produced at minimum cost (trade-off optimization between holding and setup costs) to meet the forecast demands before scheduling the work.
- *Lot streaming* can be considered as one of production scheduling function of breaking the lot size into smaller sub-lots for overlapping the sub-batches in order to meet one or more objectives, for example, at minimum production time.

In lot sizing, there is no sequencing of lots on the machines whereas in lot streaming, the routing, operation sequencing, job sequencing, and scheduling are essential.

In scheduling problems, jobs can be considered to represent an "order" of one or more items, they are treated as wholly indivisible (non-preemptable) or continuously divisible (preemptable). Furthermore, a common job constraint is that a job of multiple items can be processed by at most one machine at a time,

Lot streaming is used to describe the decomposition of jobs into smaller sub-jobs so that sub-jobs of the same job can be processed on different machines simultaneously that to improve schedule performance dramatically.

It is obvious that we can simply overcome some restrictive scheduling constraints such as job constraint. The disadvantage of lot streaming is that the scheduling process of jobs (sub-jobs) will become more complicated and difficult to solve. If established scheduling algorithms are used to solve the resulting lot streaming scheduling problems (and even many simple ones), the execution times of the algorithms may well become unacceptable. This is because algorithm running times *must* grow exponentially with the number of jobs considered.

Typically, lot-sizing problems are significantly different from scheduling problems. In a lot sizing problem, the customers and forecasted demands that appear are used as the primary input to the problem. A cost estimation and optimization lead to reduce the number of lots scheduled. The demands, in terms of due dates and quantities, may provide an opposing force for the creation of multiple lots. Lots (the outputs of a lot-sizing problem) are very similar in definition to jobs in a scheduling problem (the inputs to a scheduling problem). In lot sizing, there is no sequencing of lots on the

machines whereas in scheduling, the routing, operation sequencing and job sequencing are essential.
As noted by Potts and Van Wassenhove [25], scheduling literature nearly always assumes that lot-sizing decisions have already been taken. Lot-sizing literature does not usually consider sequencing issues; yet, in industrial scheduling environments, these two types of decision are strongly interrelated. The interrelationship is reinforced; the setup times are modeled directly in scheduling problems, while setup costs are commonly modeled in lot sizing problems. Literature on problems involving relevant elements from both scheduling and lot-sizing is surprisingly uncommon. Potts and Van Wassenhove [25] have assisted in the study of such problems by presenting the common job model in scheduling, providing a classification scheme for combined lot-sizing/scheduling, and surveying relevant literature on the topic.

C.6. Lot Streaming Classification

Based on the classification presented in [32], the lot streaming models can be classified as follows:

C.6.1 Lot streaming classification according to the sub-batch size

According to the sub-batch size which is defined as the number of products involved in the sub-batch, the lot streaming can be classified as follow:
a) *Variable lot streaming (VLS)*. The sub-batches sizes are different at all machines.
b) *Consistent lot streaming (CLS)*. The sub-batches sizes are different but each sub-batch has the same size at all machines.
c) *Equal lot streaming (ELS)*. The sub-batch sizes are equal at all machines.

C.6.2 Lot streaming classification according to the sub-batch number

The lot streaming models can be classified into two models as follows:
a) *Continuous lot streaming (CLS)*. The sub-batch sizes are real numbers.
b) *Discrete lot streaming (DLS).* The sub-batch sizes are integer numbers.
Generally, the size of sub-batch of a batch can take a real number, but for manufacturing systems where the production is a discrete-item production, the integrality constraint makes a lot streaming problem difficult to solve. By rounding off rule, the optimal discrete sub-batch size for continuous model can be found but the optimality is not guaranteed.

C.6.3 Lot streaming classification according to the idle time form

Lot streaming can be classified into two cases:
a) *No-idling lot streaming case (NI)*. There is no idle time between the processing of the sub-batches of a batch on a machine, i.e., once machine starts processing a batch, it must do so until all its sub-batches have been processed.
b) *Intermittent idling lot streaming case (II)*. There is idle time between the processing of two successive sub-batches of a batch on a machine.

C.6.4 Lot streaming classification according to setup times

Lot Streaming can be classified according to setup times into two classes:
a) Zero-setup time lot streaming
Theoretically, lot streaming in which the setup time is very small is called zero setup time lot streaming. This setup time is included in the processing time of the job. It is easy to obtain the optimal LS (as in the given example 1, 2) by breaking the batches into maximum number of sub-batches (product per sub-batch).
b) Non-Zero setup time lot streaming

Lot streaming in which the setup time takes a significant value is called non-zero setup time lot streaming. In this type of lot streaming, the optimization which will be performed is a trade-off optimization between the time saved (decreased production time) by overlapping processing and the extra-time (increased production time) due to additional sub-batches setups.

C.7. Setup time classification

C.7.1 Setup time classification according to process sequence dependency

Non-zero setup time can be classified according to process sequence dependency into two classes:
a) Sequence-independent setup time: If the setup times depend only on the current and the next sub-batch (or batch) to be processed, the setup time is called sequence-independent setup tim*e*. It is usually included in the processing time
b) Sequence-dependent setup time: If the setup times depend on the current, next and on the preceding sub-batch (or batch), the setup time is called Sequence-dependent setup tim*e*. It is usually found in situations where the facility is a multipurpose machine.

C.7.2 Setup time classification according to the batch attaching

The setups times in the scheduling problem can be classified also according to the batch attaching, there are two kinds:
a) Attached setup time: If the setup time can be performed as soon as the batch is available (arrived), the setup time is called attached setup time.
b) Detached setup time: If the setup time can be performed as soon as the machine is available (free), the setup time is called detached setup time.
Setup time can be classified into two types:
1. *Equal setup time*: if the setup times of all machines are equal.
2. *Variant setup time*: if the setup times are different.

C.8 Setup time estimation

Setup time values can be estimated by one of two following methods:
a) Sum value estimation if the actions of the elements are performed in sequential.
b) Maximum value estimation if the actions are performed in parallel.
In this research, the setup time is assumed as equal attached independent-sequence setup time and the setup time value is estimated by maximum value estimation.

References

1. Serope K., Steven R. S., 2000, "**Manufacturing Engineering and Technology**"4th Edition. Prentice Hall.
2. Harold t., .John a., Oliver s., 1975. “Manufacturing Organization and Management” , PP. 588.Prentice-Hall
3. Groover P., 1987, “Automation, Production Systems, and Computer Integrated Manufacturing”, PP.808.Prentice-Hall.
4. Pinedo M., 2005. “Planning and Scheduling in Manufacturing and Services” Springer series in Operations Research.
5. Rembold, B.O.Nnaji, A. Storr, 1993, “Computer Integrated Manufacturing and Engineering”, PP.640.Addison-Wesley.
6. Chase R. B., Jacobs F. R., Aquilano N. J., 2004, "Operations Management" 10th International Edition. McGraw Hill. USA.
7. Katsundo Hitomi, 1996, "Manufacturing Systems Engineering" 2nd edition, Taylor & Francis Ltd. London, Great Britain, UK