Minimizing The Number of Tardy Jobs in Hybrid Flow Shops with Non-Identical Multiple Processors

Daryl L. Santos and Ivan Roa

Department of Systems Science and Industrial Engineering
Binghamton University
Vestal Parkway East
Binghamton, NY 13902-6000

Corresponding author's Email: Santos@binghamton.edu

Abstract: Two-stage hybrid flow shops (a.k.a., flow shops with multiple processors (FSMPs)) are studied wherein the multiple processors at a stage are non-identical, but related (a.k.a., uniform) in their processing speeds. The impact of ten different dispatching procedures on a due-date based criterion (specifically, the number of tardy jobs) is analyzed over a set of 1,800 problems of varying configurations wherein the number of jobs per problem is between 20 and 100 and their due dates are randomly assigned. Results indicate that the modified due date (MDD), earliest due date (EDD), slack (SLK), shortest processing time (SPT), and least work remaining (LWR) rules are statistically inseparable but yield superior performance to the other rules included in this study. The longest processing time (LPT) and most work remaining (MWR) rules provide the poorest performance.

Significance: FSMPs (with applications widespread throughout industry) are studied utilizing a more realistic assumption (non-identical multiple processors) than most of the extant FSMP literature (identical processors).

Keywords: Hybrid Flow Shops, FSMPs, Scheduling, Non-identical Processors, Number of Tardy Jobs.

1. Introduction

When a flow shop environment is modified by adding one or more additional processors at a stage of operation (e.g., to relieve a bottleneck), a new scheduling environment is created. This environment is known by various terms: hybrid flow shop (a hybrid of pure flow shop and parallel processor environment), flexible flow lines, flexible flow shop, and flow shop with multiple processors (FSMP). FSMPs appear in a variety of applications including, but not limited, to the following (Roa, 2000): petrochemical industries; printed circuit board (PCB) assembly, see Figure 1; automotive assembly; food processing, and pharmaceutical industries.
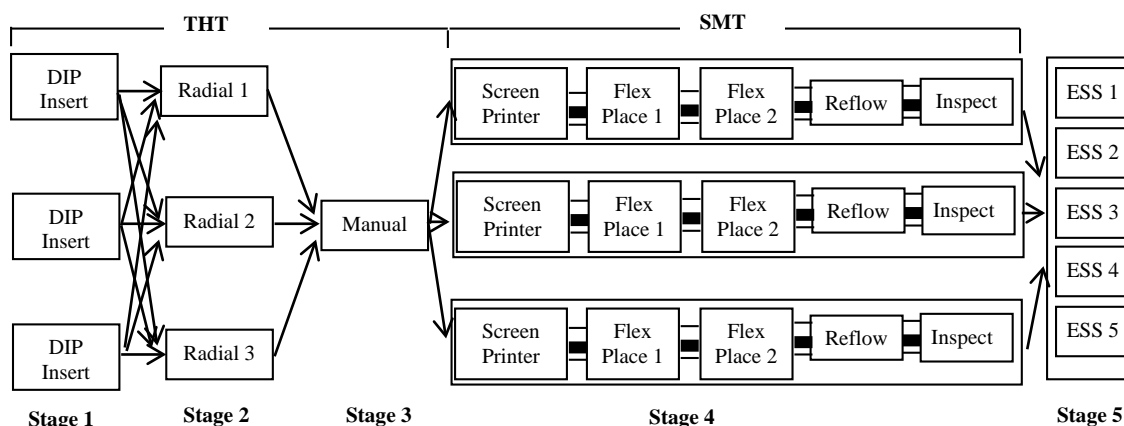

Figure 1. Schematic of the Hybrid Flow Shop in Printed Circuit Board Assembly

Figure 1 demonstrates the FSMP in a typical PCB facility. Herein, PCBs have through-hole technology (THT) components inserted through the board, followed by surface mount technology (SMT) components being soldered onto the board, and then

the PCBs are subjected to environmental stress screening (ESS). At any of the stages, one or more processors may exist. As aforementioned, the FSMP is manifested in a variety of enterprises (from petrochemical to food processing) but in this PCB assembly example alone, the FSPM is widespread: the US and Japanese electronics manufacturers produce more than 50% of total world's output and the global electronics industry has been estimated to be $1.5 trillion, www.ipc.org.

While study of the FSMP is, compared to other scheduling environments, relatively young – with the first reported results in the 1970s (Salvador, 1973) – there has been a considerable amount of attention paid to this environment in the past two decades. With rare exception (e.g., Roa and Santos (1999), Roa (2000), Jenabi et al. (2007), Santos and Roa (2007), Rabiee et al. (2014)), however, most of the extant literature in this environment utilizes an assumption that is often not realistic. The assumption that is often used in these studies is that the multiple processors at a stage are identical in their processing speed. It is rarely the case that a newly added processor will be exactly identical in speed to the one currently in the shop because of the following reasons: the newly added equipment may be of a newer(older) generation and be faster(slower) in its processing; the newly added equipment may be from a different manufacturer than the machine(s) already in place (while they may have the same capabilities, they will likely be of differing speed); and even "identical" machines (same manufacturer, same generation) will have some natural variation between them causing them to not be equal in their processing speeds.

The purpose of this paper is to relax the assumption of identical processing speeds in a study of a two-stage FSMP with due-dates. An experiment is performed wherein 1,800 problems of varying configurations are studied where the scheduling is performed using ten different dispatching rules and the objective is to minimize the number of tardy jobs. This work is intentional in selecting dispatching rules that are easy to implement on the shop floor, much like those presented in Brah (2007), but that work considered the multiple processors at a stage to be identical.

1.1 Non-Identical Parallel Processors

Non-identical parallel processors can be categorized into two types – uniform (related) and unrelated. For the unrelated case, the processing time of a job is arbitrary, and depends on the choice of machine on which it is processed and there is no job-to-job similarity in processing times in moving from one processor to another. In this study, the non-identical processors are uniform, meaning that their processing speeds are different, but the time taken for a job to be processed on one machine is proportional to the time it would take to be processed on another machine.

In the uniform processors case, a pre-specified speed factor (s_k) is the proportionality between them (Friesen and Langston, 1983 and Dessouky et al., 1990). In determining the processing time of a job on a machine, we use

$p_{jk} = p_j / s_k$.

Where, in the above,
  p_{jk} is the processing time of job j on machine k with speed factor;
  p_j is the processing time of job j without speed factor; and
  s_k is the *speed factor* for machine k.

For example, having three jobs (J1, J2, and J3) with base (e.g., on slowest machine) processing times 4, 7, and 9, respectively, and three machines (m_1, m_2, and m_3) with speeds 1.0, 1.2, and 1.4, respectively, their resulting processing times are shown in Table 1.

Table 1. Example of Uniform Processing Times

m_k \ p_j	J1	J2	J3
m_1 ($s_1 = 1.0$)	4.00	7.00	9.00
m_2 ($s_2 = 1.2$)	3.33	5.83	7.50
m_3 ($s_3 = 1.4$)	2.85	5.00	6.42

1.2 Scheduling Assumptions

Before beginning the analysis, particular assumptions about the structure of the scheduling problem are necessary to set certain boundaries to the research. In the assumptions stated below, the following notation is used: n is the number of jobs, s is the number of stages in the system, and M_j is the number of machines at each stage j.

Santos & Roa

The assumptions used in this work are consistent with general studies in scheduling (French 1982) and particular investigations in other FSMP studies (Santos, Hunsucker, and Deal (2001), Brah and Loo (1999), Hunsucker and Shah (1994), among others) and are as follows:

1. A job may not be processed on more than one stage at a time;
2. The number of jobs, *n*, is known and fixed;
3. All jobs arrive at time zero;
4. The processing times of the jobs at each of the stages are known and are constant;
5. Set-up time is considered part of the processing time;
6. Set-up is independent of the job sequence;
7. All jobs follow the same sequence;
8. A job may only visit one machine at a stage that has multiple processors;
9. No job may be pre-empted;
10. The flow shop has $s \geq 2$ stages in series;
11. Each stage has $M_j \geq 1$ machines ($j=1…s$);
12. All machines are available at time zero and are fully functional during the scheduling period;
13. No machine may process more than one job at a time;
14. Machines may be idle; and
15. In-process inventory is allowed.

While these assumptions are not always realistic (e.g., there are certainly some problems where set-up times are dependent upon job sequence and there are situations where in-process inventory is not allowed), this work is restricted to the above conditions. In addition to the above generalized FSMP assumptions, the following assumptions are specific to this study:

1. The number of stages is two ($s = 2$);
2. The due dates are known at time zero;
3. The speeds of the multiple processors at a stage are known and fixed at time zero; and
4. The performance measure investigated in this work is the number of tardy jobs (n_T).

2. Experimental Design

Even when the identical processor case of the FSMP is considered, the scheduling environment is NP-Hard (Brah (1988 and 1992)). When the processors are non-identical, the complexity increases. The intractability of this environment lends itself to study via empirical methods. As such, a study is carried out in which the number of jobs is varied from 20 to 100, the number of machines per stage is varied from 1 to 4, and the processing speeds (of the non-identical machines at a stage) varies from 1.0 to 1.5 (for references regarding the speed parameters, see Guinet (1995) and Tamimi and Rajan (1997)).

With five different job sizes (20, 40, 60, 80, and 100), twelve different environment configurations (varying the number of machines at a stage and their processing times), and with 30 problems generated per job/shop configuration, there are 1,800 problems analyzed in this study. The exact problem configurations studied in this work appear in the experimental matrix in Table 2.

Table 2. Experimental Matrix

Jobs	m_1	$s_{k,1}$	m_2	$s_{k,2}$	Problems
20 40 60 80 100	1	1	2	1.0, 1.15	30
20 40 60 80 100	1	1	3	1.0, 1.15, 1.3	30
20 40 60 80 100	1	1	4	1.0, 1.15, 1.3, 1.4	30
20 40 60 80 100	2	1.0, 1.2	1	1.0	30
20 40 60 80 100	2	1.0, 1.2	3	1.0, 1.15, 1.3	30
20 40 60 80 100	2	1.0, 1.2	4	1.0, 1.15, 1.3, 1.4	30

Jobs	m_1	$s_{k,1}$	m_2	$s_{k,2}$	Problems
20 40 60 80 100	3	1.0, 1.2, 1.4	1	1.0	30
20 40 60 80 100	3	1.0, 1.2, 1.4	2	1.0, 1.15	30
20 40 60 80 100	3	1.0, 1.2, 1.4	4	1.0, 1.15, 1.3, 1.4	30
20 40 60 80 100	4	1.0, 1.2, 1.4, 1.5	1	1.0	30
20 40 60 80 100	4	1.0, 1.2, 1.4, 1.5	2	1.0, 1.15	30
20 40 60 80 100	4	1.0, 1.2, 1.4, 1.5	3	1.0, 1.15, 1.3	30

The number of jobs (from 20 through 100) is similar to other studies (see Younes, Santos and Maria (1998) for example). The method of obtaining the baseline processing times (e.g., times on the slowest machine) is random generation from U[1, 10] and is similar to other FSMP studies (see Hunsucker and Shah (1994) and Santos, Hunsucker, and Deal (1996 and 2002) among others).

2.1 Due-Date Assignment Methods (DDMs)

There are two categories of due-date assignment methods, namely, exogenous and endogenous (Conway, Maxwell, and Miller (1967)). As suggested by some researchers (e.g., Cheng, 1986), the exogenous DDMs can be associated with situations where the determination of the due date is done by a customer or internal marketing department. This determination is done without significant (if any) regard to the impact (or the relationship) of that due date to the shop environment. The exogenous DDMs yield due dates that are disassociated from the interrelationships of the jobs and the shop environment. The situation that most commonly falls under this case is presented when a customer prescribes a due date to a company.

Endogenous DDMs (sub-categorized into job-based and shop-based DDMs) are more in keeping with the integration and pacing purposes of the shop floor. They directly incorporate some aspect of the job or shop status into the computation of the due date and are typically determined by a shop manager or similar person with a comparable understanding of the shop. Furthermore, some techniques applied to problems that are not due-date based, like the shifting bottleneck method (SBM) for minimizing the makespan, utilize a constructed due-date (see Pinedo (2002)) for the embedded single machine scheduling problem. This constructed due-date is, therefore, endogenous

A review of exogenous and endogenous (both job-based and shop-based) due-date assignment methods can be found in Martinez (1992) and Roa (2000). For the purposes of this work, we have opted to assign individual due dates to each job using a random method. The reason for this selection is that the random DDM is found to be more representative of realistic situations in which completely arbitrary deadlines are assigned to customers (Conway, Maxwell, and Miller (1967)). Furthermore, in Conway's (1964) seminal work in the "RAND [Corporation] study", the random DDM was utilized.

Table 3, below, augments Table 2 by including the values of the parameters used in generating the random due-dates for the jobs. The due-dates are generated from a uniform distribution, namely, U[0, *max*] where *max* varies depending upon the number of jobs (*max* increases with number of jobs) and shop floor configuration (*max* tends to decrease as more machines are available for processing).

Table 3. Experimental Matrix with Parameters for Randomized Due-Date Assignments

m_1	$s_{k,1}$	m_2	$s_{k,2}$	Due-Dates: U[0, *max*]				
				Jobs				
				20	40	60	80	100
1	1	2	1.0, 1.15	0,142	0,262	0,375	0,517	0,617
1	1	3	1.0, 1.15, 1.3	0,145	0,257	0,417	0,497	0,606
1	1	4	1.0, 1.15, 1.3, 1.4	0,146	0,270	0,396	0,488	0,616
2	1.0, 1.2	1	1.0	0,142	0,268	0,381	0,519	0,622
2	1.0, 1.2	3	1.0, 1.15, 1.3	0,72	0,127	0,181	0,250	0,298

m_1	$s_{k,1}$	m_2	$s_{k,2}$	Due-Dates: U[0, *max*]				
				Jobs				
				20	40	60	80	100
2	1.0, 1.2	4	1.0, 1.15, 1.3, 1.4	0,70	0,127	0,181	0,228	0,276
3	1.0, 1.2, 1.4	1	1.0	0,135	0,265	0,379	0,492	0,602
3	1.0, 1.2, 1.4	2	1.0, 1.15	0, 68	0,129	0,180	0,241	0,299
3	1.0, 1.2, 1.4	4	1.0, 1.15, 1.3, 1.4	0,50	0,81	0,116	0,151	0,191
4	1.0, 1.2, 1.4, 1.5	1	1.0	0,146	0,266	0,370	0,490	0,616
4	1.0, 1.2, 1.4, 1.5	2	1.0, 1.15	0,68	0,134	0,182	0,229	0,289
4	1.0, 1.2, 1.4, 1.5	3	1.0, 1.15, 1.3	0,49	0,82	0,116	0,152	0,187

The values of the parameters for the uniform distribution in this paper are selected as shown in Table 3 for two reasons. First of all, they provide a good spread among the due dates assigned to various jobs in each simulation run. Secondly, these values of the parameters yielded a sufficient number of tardy jobs necessary to evaluate the performance of the dispatching procedures for the criteria chosen within all system configurations. In addition, each configuration has unique values of the parameters for the uniform distribution to produce an important number of tardy jobs.

2.2 Dispatching Procedures

The dispatching procedures considered in this study include those that are easy to understand and, therefore, often implemented in practice (and often with disregard to the performance measure). Of the ten included in the study, seven of these dispatching procedures require no information about the due dates of the jobs, while three of the procedures are due-date based dispatching methods. Of course, this list is certainly not exhaustive, however, several simulation studies on the identical-processors case of the FSMP have been performed with these procedures (Brah (1996), Hunsucker and Shah (1992), and Martinez (1992)) and their inclusion will allow comparisons between this study (non-identical processors) and the prior ones (identical processors). Furthermore, most of these rules are also used in due-date scheduling of job-shop scheduling environments (see Sule (1996), Morton and Pentico (1993), and Conway, Maxwell, and Miller (1967), among many others). The ten dispatching procedures studied in this work are now presented:

1. FIFO (First In First Out). Select the operation of the job that was first to enter the queue at that stage.
2. LIFO (Last In Last Out). Select the operation of the job that last entered the queue for service.
3. LPT (Largest Processing Time First). Select the operation with the largest processing time.
4. SPT (Shortest Processing Time First). Select the operation with the shortest processing time.
5. MWR (Most Work Remaining First). Select the operation associated with the job having the most work remaining.
6. LWR (Least Work Remaining First). Select the operation associated with the job having the least work remaining.
7. RAN (Random). Select the operation randomly.
8. EDD (Earliest Due Date). Select the operation of the job associated with the earliest due date.
9. MDD (Modified Due Date). Select the operation associated with the least maximum due date or all remaining processing time (as based on the slowest machine(s)).
10. SLK (Slack). Select the operation of the job associated with the least due date minus all remaining processing time (as based on the slowest machine(s)).

All the procedures are non-optimal methods that seek to obtain good results in cases (such as this environment) where an optimal solution is not practical to obtain. The dispatching rules will now be examined to determine their performance with regards to the experimental matrix of 1,800 problems generated with the objective of minimizing the number of tardy jobs, n_T.

3. Scheduling Results

As the environment under study is NP-Hard, and as the number of jobs is large for most of the problems generated in the matrix, we do not have the luxury of optimal solutions for the problems in our study. An obvious lower bound of zero tardy jobs ($LB(n_T) = 0$) can therefore be set as a reference for comparison. For problems smaller than those included in this work, the practitioner can find optimal solutions via the use of an MILP presented by Santos and Roa (2007).

Typical statistics utilized in empirical comparisons of scheduling procedures use either relative errors or absolute errors with regards to the optimal (if known) or lower bound result. Since our lower bound is $LB(n_T) = 0$, we cannot use relative errors as a metric because of a division by zero result in all cases. Furthermore, since our lower bound is zero, absolute error simplifies to the results obtained by the scheduling procedures, anyway. Consequently, the statistic that will be used is simply the number of tardy jobs for comparison between the procedures.

3.1 Mean Number of Tardy Jobs (%) by Job Size

Five tables (Table 4 through Table 8), one for each job size, are now presented that provide the mean number of tardy jobs, in percentage form, for the different dispatching procedures and further categorized by shop floor configuration.

The number of tardy jobs is converted to percentage form to allow comparisons among the different job sizes (tables). The last row summarizes the statistics for all configurations for that procedure. Amalgamated results of these tables are discussed in Section 3.2.

Table 4. Mean Number of Tardy Jobs, in Percentage, for 20-Job Problems

Jobs	m_1	m_2	FIFO	LIFO	LPT	SPT	MWR	LWR	EDD	MDD	SLK	RAN
20	1	2	8.30	8.37	9.93	6.83	9.87	7.00	5.63	5.53	5.77	8.37
20	1	3	8.33	9.03	10.93	6.27	10.70	7.03	7.00	6.73	7.13	8.93
20	1	4	8.97	9.53	10.97	7.37	10.63	7.87	8.67	8.40	8.60	9.00
20	2	1	8.63	9.30	10.40	6.53	11.13	6.47	7.30	7.20	7.63	8.43
20	2	3	9.43	9.10	11.57	7.27	11.03	7.27	8.30	7.93	8.37	9.17
20	2	4	8.37	8.37	10.40	7.30	9.77	7.40	7.03	6.77	7.07	8.27
20	3	1	9.03	9.13	11.47	7.23	11.23	7.00	8.40	8.13	8.47	9.07
20	3	2	9.37	9.43	11.23	7.07	12.07	7.03	8.43	8.30	8.60	8.87
20	3	4	8.67	8.80	10.43	7.30	10.53	7.57	6.97	6.40	7.27	8.50
20	4	1	7.53	7.93	10.07	5.77	10.13	5.57	6.80	6.77	6.97	7.73
20	4	2	8.63	8.10	11.00	7.17	11.33	6.90	7.57	7.30	7.87	8.10
20	4	3	9.00	8.93	11.10	7.60	11.37	7.50	7.77	7.50	8.57	8.83
20	G	G	8.69	8.84	10.79	6.98	10.82	7.05	7.49	7.25	7.69	8.61

Table 5. Mean Number of Tardy Jobs, in Percentage, for 40-Job Problems

Jobs	m_1	m_2	FIFO	LIFO	LPT	SPT	MWR	LWR	EDD	MDD	SLK	RAN
40	1	2	16.90	17.27	22.00	12.90	20.70	14.17	11.63	11.47	12.03	17.53
40	1	3	18.77	18.93	23.70	14.00	22.63	15.43	15.13	15.00	15.43	19.20
40	1	4	17.17	17.10	21.27	12.40	20.10	13.30	10.00	9.97	9.80	17.53
40	2	1	17.33	16.83	21.20	13.40	21.87	12.67	14.03	14.00	14.70	17.37
40	2	3	17.47	18.50	22.37	13.97	21.43	14.87	13.87	13.53	13.90	18.30
40	2	4	17.43	17.80	22.47	13.87	20.97	14.77	11.80	11.43	11.80	17.77
40	3	1	18.13	17.17	22.27	13.57	22.77	13.27	16.77	16.63	17.13	17.50
40	3	2	16.57	17.10	21.87	14.30	22.37	13.83	14.60	14.07	14.93	17.00
40	3	4	19.63	18.93	23.53	15.07	23.37	15.90	17.50	16.60	17.77	18.53

Jobs	m_1	m_2	FIFO	LIFO	LPT	SPT	MWR	LWR	EDD	MDD	SLK	RAN
40	4	1	16.77	16.57	21.70	12.60	21.97	12.40	16.00	15.80	16.00	17.07
40	4	2	17.80	16.53	20.93	13.70	21.57	13.23	15.73	15.40	15.37	17.77
40	4	3	17.63	17.33	21.70	15.27	22.70	14.57	15.47	14.33	15.17	17.83
40	G	G	17.63	17.51	22.08	13.75	21.87	14.03	14.38	14.02	14.50	17.78

Table 6. Mean Number of Tardy Jobs, in Percentage, for 60-Job Problems

Jobs	m_1	m_2	FIFO	LIFO	LPT	SPT	MWR	LWR	EDD	MDD	SLK	RAN
60	1	2	27.90	27.63	34.67	20.67	32.47	21.80	18.70	18.50	18.57	27.20
60	1	3	27.00	27.50	34.07	20.23	32.07	21.97	17.53	17.20	17.33	27.37
60	1	4	25.60	26.90	33.67	19.60	32.00	21.50	15.47	15.33	15.10	26.47
60	2	1	27.30	27.00	33.77	21.00	35.07	19.77	22.07	22.07	22.30	27.53
60	2	3	27.93	27.23	35.13	21.90	33.17	21.87	20.17	19.60	20.33	27.40
60	2	4	25.70	25.27	32.37	19.63	31.30	21.27	16.17	15.90	16.13	26.13
60	3	1	27.17	25.90	33.83	19.97	34.57	19.27	24.23	24.10	24.30	26.70
60	3	2	28.17	27.33	33.53	22.50	36.57	20.77	22.10	21.90	22.13	27.37
60	3	4	27.33	26.83	34.80	21.53	32.73	21.53	22.33	21.17	22.60	27.33
60	4	1	27.40	27.50	35.30	20.77	35.90	20.03	25.90	25.90	26.20	27.83
60	4	2	27.37	27.23	34.07	22.47	34.83	21.53	23.63	22.93	24.00	27.20
60	4	3	27.90	28.33	34.27	22.63	36.40	21.60	22.83	22.63	23.97	27.83
60	G	G	27.23	27.06	34.12	21.08	33.92	21.08	20.93	20.60	21.08	27.20

Table 7. Mean Number of Tardy Jobs, in Percentage, for 80-Job Problems

Jobs	m_1	m_2	FIFO	LIFO	LPT	SPT	MWR	LWR	EDD	MDD	SLK	RAN
80	1	2	34.90	35.87	44.13	26.33	41.70	27.83	21.23	21.00	21.00	35.10
80	1	3	36.23	36.93	46.07	26.70	44.27	28.70	19.83	19.63	20.13	36.33
80	1	4	37.17	37.93	47.23	28.13	45.03	29.47	29.13	28.93	29.07	36.73
80	2	1	34.23	34.63	43.63	26.67	44.57	25.43	28.23	27.67	28.80	34.77
80	2	3	33.17	33.90	43.07	26.70	40.00	27.50	15.63	14.83	15.77	33.63
80	2	4	35.67	37.63	47.50	28.20	45.07	30.23	25.57	25.03	25.77	37.80
80	3	1	37.27	36.30	46.93	27.03	46.87	26.77	34.50	34.33	34.70	37.10
80	3	2	35.03	35.60	43.57	27.83	46.07	26.83	26.37	25.90	27.07	35.40
80	3	4	35.43	36.33	46.37	28.70	43.47	29.73	24.83	23.80	24.87	35.90
80	4	1	36.77	35.87	46.53	27.80	46.90	27.30	34.50	34.57	35.17	37.13
80	4	2	37.70	38.33	47.90	29.57	48.63	27.97	32.27	31.80	33.00	37.70
80	4	3	36.90	35.70	44.07	29.57	47.63	28.03	26.97	26.17	27.37	37.23
80	G	G	35.87	36.25	45.58	27.77	45.02	27.98	26.59	26.14	26.89	36.24

Table 8. Mean Number of Tardy Jobs, in Percentage, for 100-Job Problems

Jobs	m_1	m_2	FIFO	LIFO	LPT	SPT	MWR	LWR	EDD	MDD	SLK	RAN
100	1	2	44.83	45.53	57.20	33.17	54.47	36.37	27.10	26.77	27.00	44.90
100	1	3	46.97	47.40	59.27	34.70	57.13	38.23	31.67	31.30	31.50	45.83
100	1	4	45.03	44.30	56.90	32.40	55.03	35.63	22.40	22.17	22.53	45.77
100	2	1	45.00	45.90	54.43	35.40	57.40	33.30	36.10	35.70	36.03	44.70
100	2	3	44.87	43.73	56.33	34.70	53.10	35.37	26.13	25.80	26.53	44.00
100	2	4	47.73	46.90	60.43	35.27	57.17	38.80	33.90	33.67	34.03	47.10
100	3	1	48.10	48.47	60.03	35.77	61.13	35.37	44.10	43.80	44.40	47.33
100	3	2	45.13	45.43	53.90	36.80	57.43	34.03	34.03	33.23	34.47	43.80
100	3	4	44.00	44.63	57.00	34.27	52.17	35.80	23.97	23.00	24.53	44.27
100	4	1	45.40	45.73	56.53	33.20	57.10	32.17	43.13	42.93	43.33	44.60
100	4	2	45.87	46.53	58.33	36.17	59.77	34.50	39.00	38.40	39.73	46.50
100	4	3	46.23	46.43	56.03	37.67	58.47	34.67	32.17	32.30	33.63	46.63
100	G	G	45.76	45.92	57.20	34.96	56.70	35.35	32.81	32.42	33.14	45.45

3.2 Summary of Overall Performance

Summarized results of the optimality percentage and mean number of tardy jobs, among all 1,800 problems, according to each procedure are provided in Table 9. After each rule, the first column lists the lower bound on percentage optimality, and the second column lists the mean number of tardy jobs, expressed in percentage form (since this table summarizes all job sizes). This last column is, in fact, a conglomeration of the results in Tables 4 through 8.

As aforementioned, we do not know the optimal solutions to these 1,800 problems. One thing we do know, however, is that if no tardy jobs are found for a scheduling instance, then we have obtained an optimal solution. Of course, we do not know the cases where we may have found an optimal solution if the optimal solution has $n_T > 0$. As such, we only know the minimum number of optimal solutions found and this number corresponds to the cases where the procedures have found $n_T = 0$; thus, our percentage optimality is expressed in "$\geq$ x%". In parentheses next to the percentage is the number of times in which that procedure found $n_T = 0$.

Table 9. Summary of Overall Performance.

Procedure	Optimality %, (Count of $n_T = 0$)	Mean Number of Tardy Jobs %
FIFO	$\geq$ 0.00%, (0)	27.04%
LIFO	$\geq$ 0.00%, (0)	27.11%
LPT	$\geq$ 0.00%, (0)	33.96%
SPT	$\geq$ 0.00%, (0)	20.91%
MWR	$\geq$ 0.00%, (0)	33.66%
LWR	$\geq$ 0.00%, (0)	21.10%
EDD	$\geq$ 2.05%, (37)	20.44%
MDD	$\geq$ 2.00%, (36)	20.09%
SLK	$\geq$ 1.72%, (31)	20.66%
RAN	$\geq$ 0.00%, (0)	27.06%

It perhaps should not be surprising that the rules that include some information about the due dates (namely, EDD, MDD, and SLK) seem, at first glance, to be superior to the others. For example, when looking at the mean percentage of number of tardy jobs, the overall results for these three rules are superior to the other procedures considered in this study. Furthermore,

these three rules are the only ones in which we can guarantee to have found optimal results and these rules also have the three lowest mean number of tardy jobs (%).

However, these results are the raw data results. When looking at SPT, we can see that its mean number of tardy jobs is larger, but not appreciably different, from the three due-date based dispatching rules. The next section will provide a comparison between the ten procedures using statistical techniques to determine if there are statistically significant differences between the techniques.

4. Statistical Analysis of the Data

While the three due-date based rules (EDD, MDD, and SLK) seem to be superior in terms of basic data analysis – they were the only ones to arrive at known optimal solutions and have smaller mean numbers of tardy jobs – it is important to see if their performance is statistically separable from the other methods included in this study.

To test the performances of the procedures in a statistically significant manner, analysis-of-variance (ANOVA) and two multiple-comparison procedures, Least Significant Difference (LSD) test and Duncan's Multiple Range test, are performed.

4.1 ANOVA Results

An ANOVA examination is conducted with the following hypothesis test:

$$H_o: \mu_1 = \mu_2 = \mu_3 = \mu_4 = \mu_5 = \mu_6 = \mu_7 = \mu_8 = \mu_9 = \mu_{10}$$

vs. $H_1: \mu_i \neq \mu_j$ for at least one pair (i, j) .

Where, in the above, μ_1, μ_2, μ_3, μ_4, μ_5, μ_6, μ_7, μ_8, μ_9, and μ_{10} are the mean number of tardy jobs (expressed in percentage) among all 1,800 problems for the FIFO, LIFO, LPT, SPT, MWR, LWR, EDD, MDD, SLK, and RAN procedures, respectively. In the ANOVA results, the value of $F_o = 11.15$ is calculated. Since 11.15 > 1.90 (the value of $F_{0.05,9,590} = 1.90$, working at $\alpha = 0.05$ as a confidence level), the hypothesis that the mean number of tardy jobs is equal must be rejected.

4.2 Multiple-Comparison Procedures

Table 10 and Table 11 present the results for the LSD test and Duncan's Multiple Range test, respectively.

$\alpha = 0.05$ $df_E = 590$ $MS_E = 157.55$ *Critical Value = 4.50*

Table 10. Least Significant Difference (LSD) Multiple-Comparison Test

MEAN		27.04	27.11	33.96	20.91	33.66	21.10	20.44	20.09	20.66	27.06
	PROC	FIFO	LIFO	LPT	SPT	MWR	LWR	EDD	MDD	SLK	RAN
27.04	FIFO	0.00	0.08	**6.92**	**6.13**	**6.63**	**5.94**	**6.60**	**6.95**	**6.38**	0.02
27.11	LIFO		0.00	**6.84**	**6.21**	**6.55**	**6.01**	**6.68**	**7.03**	**6.45**	0.06
33.96	LPT			0.00	**13.05**	0.29	**12.86**	**13.52**	**13.87**	**13.29**	**6.90**
20.91	SPT				0.00	**12.76**	0.19	0.47	0.82	0.24	**6.15**
33.66	MWR					0.00	**12.57**	**13.23**	**13.58**	**13.00**	**6.61**
21.10	LWR						0.00	0.66	1.01	0.44	**5.96**
20.44	EDD							0.00	0.35	0.22	**6.62**
20.09	MDD								0.00	0.58	**6.97**
20.66	SLK									0.00	**6.39**
27.06	RAN										0.00

A bold value at an intersection identifies where that pair of rules perform significantly differently (i.e., the values are greater than the critical value of 4.50).

$\alpha = 0.05$ $df_E = 590$ $MS_E = 157.55$

Critical Ranges: R2 = 4.49, R3 = 4.73, R4 = 4.85, R5 = 5.01, R6 = 5.10, R7 = 5.17, R8 = 5.23, R9 = 5.28,
R10 = 5.33

Table 11. Duncan's Multiple Range Test

MEAN		33.96	33.66	27.11	27.06	27.04	21.10	20.91	20.66	20.44	20.09
	PROC	LPT	MWR	LIFO	RAN	FIFO	LWR	SPT	SLK	EDD	MDD
20.09	MDD	**13.87**	**13.58**	**7.03**	**6.97**	**6.95**	1.01	0.82	0.58	0.35	0.00
20.44	EDD	**13.52**	**13.23**	**6.68**	**6.62**	**6.60**	0.66	0.47	0.22	0.00	
20.66	SLK	**13.29**	**13.00**	**6.45**	**6.39**	**6.38**	0.44	0.24	0.00		
20.91	SPT	**13.05**	**12.76**	**6.21**	**6.15**	**6.13**	0.19	0.00			
21.10	LWR	**12.86**	**12.57**	**6.01**	**5.96**	**5.94**	0.00				
27.04	FIFO	**6.92**	**6.63**	0.08	0.02	0.00					
27.06	RAN	**6.90**	**6.61**	0.06	0.00						
27.11	LIFO	**6.84**	**6.55**	0.00							
33.66	MWR	0.29	0.00								
33.96	LPT	0.00									

Both multiple comparison tests are in agreement: while MDD, EDD, SLK, SPT, and LWR are the best performing procedures, they cannot be statistically separated. The statistically separable worst performing procedures are LPT and MWR and the other procedures in the middle of the pack (FIFO, LIFO, and RAN) are statistically similar, but differ from the best and worst groups.

5. Conclusions and Recommendations

These new results are now compared with similar studies that have been performed with FSMPs with the identical-processors assumption.

5.1 Prior Results in FSMPs with Identical Multiple Processors

Hunsucker and Shah (1992) studied constrained FSMPs (where $s \geq 2$); the constraint is the number of jobs concurrently allowed in the system. They utilized the same basic dispatching rules as those in this study (but did not include EDD, MDD, and SLK). In their studies, with $s = 2$ and with normal congestion levels, they found that SPT yielded the best results for minimizing the number of tardy jobs, whereas LPT yielded the worst.

Martinez (1992) conducted studies on various dispatching procedures and DDMs in FSMPs where $s \geq 5$. The rules he considered were SPT, EDD, SLK, Operation Due Date (ODD), and Operation Slack (OPNSLK). However, while he evaluated different performance measures, n_T was not among them; of the due-date based performance measures, he did evaluate mean tardiness (Tbar), mean lateness (Lbar), and percent tardiness (T%). If some similarities can be drawn (since the number of stages is greater than two (2) and since the performance measures, though due-date based, are not the same), it is important to mention that SPT performed the best with all three of these performance measures (in some cases not statistically separable from OPNSLK).

Brah (1996) investigated the performance of ten dispatching rules with tardiness criteria in FSMPs . He found that the best procedures are MDD, CR + SPT, EDD, and SLK. He extended this work later (Brah, 2007).

5.2 Comparisons to Prior Studies

As compared to the prior work in FSMPs with identical processors, the experiments performed here are in agreement with those studies.

Furthermore, Conway, Maxwell, and Miller (1967) concluded that, in the job shop environment, when due dates are assigned by an exogenous method (as in this study), SPT yields good performances. Morton and Pentico (1993) and Sule (1996) reported that SPT has been successfully used for job shop scheduling to minimize tardiness of jobs in a due date environment.

5.3 Suggestions and Comments for the Practitioner

All of the rules studied in this work are both practical and easy to implement and was a significant reason for their inclusion. The hybrid flowshop environment is one that has widespread applications and markets are now driven by customers that are stressing due-dates for their products. This study has shown that hybrid flow shops, with non-identical multiple processors, can be scheduled effectively for the number of tardy jobs criterion.

Furthermore, while MDD, EDD, and SLK are not statistically separable from SPT and LWR, they do have the lowest mean number of tardy jobs (expressed in percentage) and are the only rules that we can guarantee to have found optimal solutions in our matrix of 1,800 problems. As such, we would recommend any of these three rules. Also, as earlier mentioned, for smaller problems the practitioner may rely upon an optimal technique such as the MILP developed for this problem by Santos and Roa (2007).

The interested reader may be concerned that this study presents a large amount of simulated data wherein the processing times were generated from one type of probability distribution, the uniform and one range (1-10). To reiterate, the uniform was chosen in this work because it has been utilized in many other scheduling studies as cited earlier. It is also important to note that an even earlier study was performed to determine the effect of varying the underlying processing times in the single stage multiprocessor scheduling environment (a building-block to the FSMP), see Santos (1990). In that work, different scheduling heuristics were compared while changing the underlying distribution and ranges of the processing times utilized to generate the problem instances. In that work, the uniform, Erlang, exponential, and normal distributions were studied. A conclusion was found that by changing the distributions and ranges, only minor variations were found in the performance of the heuristics and furthermore, the comparative rankings of the heuristics did not change. Hypothesizing that this finding would be extended to this work, the practitioner interested in applying the methods herein should have little worry if the processing times are not uniformly distributed.

5.4 Recommendations for Future Work

FSMP research, as previously stated, is still in its nascent stages as compared with most other scheduling environments. This is particularly true for FSMPs with non-identical multiple processors. This work is just beginning to scratch the surface. Future work can incorporate more stages (s > 2), the use of different due-date assignment methods, and can focus on different performance measures (e.g., Lbar, Tbar, or others). Furthermore, for more computer integrated shop floor control systems wherein we need not restrict focus to easily implementable dispatching rules, evolutionary-based heuristics and other metaheuristics may be considered. Correspondingly, some interesting approaches to extend this work may be based on those found in Rabiee and Jolai (2014), Behnamian and Zandieh (2011), and Jenabi et al. (2007), among others.

6. References

Behnamian, J. and Zandieh, M. (2011). A Discrete Colonial Competitive Algorithm for Hybrid Flowshop Scheduling to Minimize Earliness and Quadratic Tardiness Penalties. *Expert Systems with Applications, 38(12)*, pp. 14490-14498.

Brah, S.A. (1988). *Scheduling in a Flow Shop with Multiple Processors,* Unpublished Ph.D. Dissertation, University of Houston, Houston, TX.

Brah, S.A. and Loo, L.L. (1999). Heuristics for Scheduling in a Flow Shop with Multiple Processors. *European Journal of Operational Research, 113*, pp. 113-122.

Brah, S.A. (2007). A Comparative Analysis of Due Date Based Job Sequencing Rules in a Flow Shop with Multiple Processors. *Production Planning and Control, 7(4)*, pp. 362-373.

Cheng, T.C. (1986). Optimal Due Date Assignment in a Job Shop. *International Journal of Production Research, 24,* pp. 503-515.

Conway, R.W. (1964). An Investigation of Priority Assignment in a Job Shop. RAND Corporation Memorandum RM-3789-PR.

Conway, R.W., Maxwell, W.L., and Miller, L.W. (1967), *Theory of Scheduling***,** Addison-Wesley, Reading, Massachusetts.

Dessouky, M.I., Lageweg, B.J., Lenstra, J.K., and Van de Velde, S.L. (1990). Scheduling Identical Jobs on Uniform Parallel Machines. *Statistical Neerlandica, 44*, pp.115-123.

Friesen, D.K. and Langston, M,A. (1983). Bounds for Multifit Scheduling on Uniform Processors. *SIAM Journal on Computing, 12*, pp. 60-70.

Guinet, A. (1995). Scheduling Independent Jobs on Uniform Parallel Machines to Minimize Tardiness Criteria. *Journal of Intelligent Manufacturing, 6*, pp. 95-103.

Hunsucker, J.L. and Shah, J.R. (1992). Performance of Priority Rules in a Due Date Flow Shop. *Omega, 20*, pp. 73-89.

Hunsucker, J.L. and Shah, J.R. (1994). Comparative Performance Analysis of Priority Rules in a Constrained Flow Shop with Multiple Processors Environment. *European Journal of Operational Research, 72*, pp. 102-114.

Jenabi, M. et al. (2007). Two Hybrid Meta-Heuristics for the Finite Horizon ELSP in Flexible Flow Lines with Unrelated Parallel Machines. *Applied Mathematics and Computation, 186(1)*, pp. 230-245.

Martinez, J. (1992). *Due Date Assignment Methods and Dispatching Rules in a Flow Shop with Multiple Processors*. M.S.I.E. Thesis, University of Houston, Houston, TX.

Morton, T.E and Pentico, D.W. (1993). *Heuristic Scheduling Systems: with Applications to Production Systems and Project Management*, John Wiley and Sons, New York.

Pinedo, M. (2002). *Scheduling: Theory, Algorithms, and Systems, 2nd Ed*.**,** Prentice Hall, Englewood Cliffs, New Jersey.

Rabiee, M. et al. (2014). A Biogeography-based Optimization Algorithm for a Realistic No-Wait Hybrid Flow Shop with Unrelated Parallel Machines to Minimize Mean Tardiness. *International Journal of Computer Integrated Manufacturing, 29(9)*, pp. 1007-1024.

Roa (2000). *A Study of Flow Shops with Non-Identical Multiple Processors*. Ph.D.S.S. Dissertation. Binghamton University, Binghamton, NY.

Roa and Santos (1999). Lower Bounds for Two-Stage Hybrid Flow Shops with Non-identical Multiple Processors. *Proceedings of the 4th International Conference on Industrial Engineering, Theory, and Applications*, San Antonio, Texas.

Salvador, M.S. (1973). A Solution to a Special Case of Flow Shop Scheduling Problems. In S.E. Elmaghraby (ed.), *Symposium of the Theory of Scheduling and Applications***,** Springer-Verlag, New York.

Santos, D.L. (1990). *The Effect of Varying the Underlying Processing Times in the Single Stage Multiprocessor Scheduling Environment*. M.S.I.E. Thesis, University of Houston, Houston, Tx.

Santos, D.L. (1993). *A Study of Makespans in Flow Shops with Multiple Processors*. Ph.D.I.E. Dissertation, University of Houston, Houston, TX.

Santos, D.L., Hunsucker, J.L., and Deal, D.E. (1996). An Evaluation of Sequencing Heuristics in Flow Shops with Multiple Processors. *Computers and Industrial Engineering, 30*, pp. 681-692.

Santos, D.L., Hunsucker, J.L., and Deal, D.E. (2001). Makespan Improvement in Flow Shop with Multiple Processors via an Exchange Heuristic. *Production Planning and Control, 12*, pp. 283-295.

Santos, D.L. and Roa, I. (2007). Mixed-Integer Linear Programming Models for Managing Hybrid Flow Shops with Uniform, Non-Identical Multiple Processors. *Asia Pacific Management Review*, 12(2), pp. 95-100.

Schellhase, J.C. (1996). *The Placement of an Additional Processor in a Flow Shop with Multiple Processors*. Ph.D.I.E. Dissertation, University of Houston, Houston, TX.

Shah, J.R. (1989). *Heuristic Programming Study of a Constrained Flow Shop with Multiple Processors*. MS.I.E. Thesis, University of Houston, Houston, TX.

Sule, D.R. (1996). *Industrial Scheduling*, PWS Publishing Company, Boston, MA.

Tamimi, S.A. and Rajan, V.N. (1997). Reduction of Total Weighted Tardiness on Uniform Machines with Sequence Dependent Setups. *6th Industrial Engineering Research Conference Proceedings*, Washington, DC, pp. 181-185.

Younes, N., Santos, D., and Maria, A. (1998). A Simulated Annealing Approach to Scheduling in a Flow Shop with Multiple Processors. *7th Industrial Engineering Research Conference Proceedings*, Banff, Canada, 8pp.