Computational Experiments for the Heuristic Solutions of the Two-Stage Chain Reentrant Hybrid Flow Shop and Model Extensions

Lowell Lorenzo[1]

[1]Department of Industrial Engineering and Operations Research, University of the Philippines Diliman 1101 Quezon City, Philippines
e-mail: Lowell.Lorenzo@up.edu.ph

Keywords: scheduling, reentrant hybrid flow shop, flexible job shop, heuristics.

1. Introduction

This is the second part of a research paper for the two-stage chain reentrant hybrid flow shop. In the first part of the research paper which was presented in Lorenzo (2017), this problem was shown to be strongly NP-hard. Lower bounds for the solution were derived and were used to develop heuristic solutions for the problem. For this paper, we now explore the performance of these heuristic solutions against the best derived lower bounds via computational experiments. Then, we develop model extensions namely the reverse two-stage chain reentrant hybrid flow shop and the two-stage flexible job shop with the corresponding heuristic solutions and computational experiments.

The outline of this paper is as follows. A short discussion of the definition of the problem and the derived lower bounds are in Sections 2 and 3. Section 4 presents the base heuristic algorithms and then the discussion on the results of the computational experiments is in Section 5. Finally, model extensions of the problem, modified heuristic algorithms, solutions and computational experiments are then presented in Sections 6-9.

2. Background and Problem Definition

Consider a simple flow shop with m stages. At every stage $i, i=1,\ldots,m$, there is a single machine M_i available to process an operation of a job. Let ϕ_k be the stage visited to perform the kth operation of a job where $\phi_k \in \{1,2,\ldots,m\}$. Then $\phi = (\phi_1, \phi_2, \ldots, \phi_m) = (1,2,\ldots,m)$ is the stage flow sequence for all jobs and consists of m elements or operations. In a simple flow shop, the number of operations a job undergoes is equal to the number of stages. In an m-stage chain reentrant flow shop, its stage flow sequence $\phi = (1,2,\ldots,m,1)$ has now *(m+1)* operations due to an occurrence of a single reentrant operation. The single reentrant characteristic occurs in the *(m+1)*th operation which is performed at stage 1 and is referred to as the finishing operation.

When there are m_i identical parallel machines available in stage i, the resulting system is referred to as a hybrid flow shop. Let this group of m_i machines in stage i be referred to as work center WC_i in stage i.

In the two-stage chain reentrant flow shop, each job $J_j, j = 1, \ldots, n$ has a stage flow sequence $\phi = (1,2,1)$. The processing time of the first operation of job J_j is a_j, its processing time in the second operation is b_j and the reentrant processing time for the finishing operation is c_j. Let the processing time vector for each job be (a_j, b_j, c_j) or simply referred to now as the processing times of J_j in the two-stage chain reentrant flow shop. Since each job is processed in every operation in the chain reentrant flow shop, then $A = (a_1, \ldots, a_n), B = (b_1, \ldots, b_n), C = (c_1, \ldots, c_n)$ are the vectors of processing times for each operation in ϕ respectively.

In the two-stage chain reentrant hybrid flow shop, there are two work centers WC_1 and WC_2 with m_1 and m_2 identical machines in parallel at stages 1 and 2 respectively. There are n jobs that have to be processed and the completion time of J_j occurs when the third or finishing operation at any of the m_1 machines in WC_1 is completed. Let CRF_{m_1,m_2} be a two-stage chain reentrant hybrid flow shop where our objective is to find a schedule that minimizes the maximum completion time. Using the three-tuple convention of defining scheduling problems proposed by Graham *et al.* (1979), minimizing makespan in CRF_{m_1,m_2} can be identified by $F(m_1, m_2)|chain\ reentrant|C_{max}$ for which the optimal objective function value is $C^*_{CRF_{m_1,m_2}}$.

The CRF_{m_1,m_2} system is a general case of the two-stage chain reentrant flow shop studied by Wang *et al.* (1997). In their paper, they study the makespan minimization of $CRF_{1,1}$ and derive a Johnson based heuristic solution with complexity *O(nlogn)* and worst-case error bound of $3/2$ is derived. In Drobouchevitch and Strusevich (1999), another heuristic solution is presented for the same problem with complexity *O(nlogn)* and an improved worst-case error bound of $4/3$.

For the two-stage chain reentrant hybrid flow shop CRF_{m_1,m_2}, we construct two auxiliary two-stage flow shops. These two auxiliary two-stage flow shops are $AF1_{1,1}$ and $AF2_{1,1}$ with their respective processing times $\left(\frac{1}{m_1}a_j, \frac{1}{m_2}b_j\right)$ and $\left(\frac{1}{m_2}b_j, \frac{1}{m_1}c_j\right)$ and their corresponding makespans $C_{AF1_{1,1}}$ and $C_{AF2_{1,1}}$. The AFs just introduced help in the development of lower bounds and this is the focus of the next section.

3. Lower Bounds for $C^*_{CRF_{m_1,m_2}}$

Lower bounds for $C^*_{CRF_{m_1,m_2}}$ can be developed from the constructed auxiliary two-stage flow shops described in the previous section. Since the proofs of these lower bounds were already presented in Lorenzo (2017), these will not be shown here anymore.

Lemma 1. Let $LB_1 = max\left(C_{AF1_{1,1}}, C_{AF2_{1,1}}, \frac{1}{m_1}\sum_{j=1}^{n}(a_j + c_j)\right)$. Then $LB_1 \leq C^*_{CRF_{m_1,m_2}}$.

Lemma 2. There is a permutation 1,2,…,*n* associated with an arbitrary schedule *S* such that for every $1 \leq i \leq n$, there is a $1 \leq k_i \leq i$ such that $\frac{1}{2}\sum_{j=1}^{i}\left(\frac{a_j}{m_1} + \frac{b_j}{m_2}\right) \leq \frac{1}{m_1}\sum_{j=1}^{k_i} a_j + \frac{1}{m_2}\sum_{j=k_i}^{i} b_j$.

Lemma 3. There is a permutation 1,2,…,*n* associated with an arbitrary schedule *S* such that for every $1 \leq i \leq n$, there is a $1 \leq k_i \leq i$, such that $\frac{1}{m_1+m_2}\sum_{j=1}^{i}(a_j + b_j) \leq \frac{1}{m_1}\sum_{j=1}^{k_i} a_j + \frac{1}{m_2}\sum_{j=k_i}^{i} b_j$.

Lemma 4. There is a permutation 1,2,…,*n* associated with an arbitrary schedule *S* such that for every $1 \leq i \leq n$, there is a $1 \leq k_i \leq i$ such that $\frac{1}{m_1+1}\sum_{j=1}^{i}\left(a_j + \frac{b_j}{m_2}\right) \leq \frac{1}{m_1}\sum_{j=1}^{k_i} a_j + \frac{1}{m_2}\sum_{j=k_i}^{i} b_j$.

Lemma 5. There is a permutation 1,2,…,*n* associated with an arbitrary schedule *S* such that for every $1 \leq i \leq n$, there is a $1 \leq k_i \leq i$ such that $\frac{1}{m_2+1}\sum_{j=1}^{i}\left(\frac{a_j}{m_1} + b_j\right) \leq \frac{1}{m_1}\sum_{j=1}^{k_i} a_j + \frac{1}{m_2}\sum_{j=k_i}^{i} b_j$.

Lemma 6. There is an optimal schedule S^* for CRF_{m_1,m_2} such that, for every $1 \leq k_1 \leq k_2 \leq n$, $\frac{1}{m_1}\sum_{j=1}^{k_1} a_j + \frac{1}{m_2}\sum_{j=k_1}^{k_2} b_j + \frac{1}{m_1}\sum_{j=k_2}^{n} c_j \leq C^*_{CRF_{m_1,m_2}}$.

Lemma 7. Let C_{JA_1} be the makespan derived by Johnson's Algorithm (JA) for the auxiliary two-stage flow shop problem with processing times $(a'_j + b'_j, c'_j)$. For any of the following set of values of a'_j, b'_j and c'_j,

$$a'_j = \frac{1}{2m_1}a_j,\ b'_j = \frac{1}{2m_2}b_j,\ c'_j = \frac{1}{m_1}c_j \ \text{ or} \tag{1}$$

$$a'_j = \frac{1}{m_1+m_2}a_j, b'_j = \frac{1}{m_1+m_2}b_j,\ c'_j = \frac{1}{m_1}c_j \ \text{ or} \tag{2}$$

$$a'_j = \frac{1}{m_1+1}a_j, b'_j = \frac{1}{m_2(m_1+1)}b_j,\ c'_j = \frac{1}{m_1}c_j \ \text{ or} \tag{3}$$

$$a'_j = \frac{1}{m_1(m_2+1)}a_j, b'_j = \frac{1}{m_2+1}b_j,\ c'_j = \frac{1}{m_1}c_j, \tag{4}$$

$C_{JA_1} \leq C^*_{CRF_{m_1,m_2}}$.

Let C_{JA_2} be the makespan derived by JA for the auxiliary two-stage flow shop problem with processing times $(b''_j + c''_j, a''_j)$, For any of the following set of values of a''_j, b''_j, and c''_j,

$$a''_j = \frac{1}{m_1}a_j,\ b''_j = \frac{1}{2m_2}b_j,\ c''_j = \frac{1}{2m_1}c_j \ \text{ or} \tag{5}$$

$$a''_j = \frac{1}{m_1}a_j,\ b''_j = \frac{1}{m_1+m_2}b_j,\ c''_j = \frac{1}{m_1+m_2}c_j \ \text{ or} \tag{6}$$

$$a''_j = \frac{1}{m_1}a_j, b''_j = \frac{1}{m_2(m_1+1)}b_j,\ c''_j = \frac{1}{m_1+1}c_j \ \text{ or} \tag{7}$$

$$a''_j = \frac{1}{m_1}a_j, b''_j = \frac{1}{m_2+1}b_j,\ c''_j = \frac{1}{m_1(m_2+1)}c_j, \tag{8}$$

$$C_{JA_2} \leq C^*_{CRF_{m_1,m_2}}.$$

4. Heuristic Algorithms for $F(m_1, m_2)|chain\ reentrant|C_{max}$

Since makespan minimization in CRF_{m_1,m_2} is a general case of makespan minimization in $CRF_{1,1}$, then it is NP-hard as well. This motivates the development of heuristics that will enable in the formulation of a solution to the problem. Towards this end, we also make the following observations. The makespan of CRF_{m_1,m_2} is attained in WC_1, where the first and third operations are processed. The following lemma establishes an optimal property for the scheduling of these operations in WC_1. Proofs of these lemmas and theorems have been presented in the paper of Lorenzo (2017) and will therefore not be discussed.

Lemma 8. To minimize the makespan of CRF_{m_1,m_2}, it is sufficient to consider a schedule wherein all the first operations of all jobs always precede the third operations of all jobs in any of the m_1 machines in WC_1.

In the development of a chain reentrant flow shop heuristic, you first need a sequence to specify the schedule of the jobs. Aside from the sequence you also need to assign each operation on the available machines in the corresponding stage. To do the assignment of operations to machines, we will utilize the first available machine (FAM) and last busy machine (LBM) rules. As the name implies, the FAM rule assigns a job from a sequence based on the first machine that becomes available. The LBM rule on the other hand is a mirror image of the FAM rule. Specifically, the assignment of jobs to machines for example in the second operation using the LBM rule is as follows.

Given a constant $T > 0$ and a sequence S',

Step 1. Set $t_m = T$ for $m = 1, \dots, m_2$.

Step 2. Let J_j be the last unscheduled job in S' and $m = max_{1 \le m \le m_2}\{t_m\}$. Schedule J_j on machine m such that it finishes at time t_m.

Step 3. Set $t_m = t_m - b_j$. $S' = S' - \{j\}$. If $S' = \{\emptyset\}$, stop else go to step 2.

Three heuristics will now be presented for CRF_{m_1,m_2}.

Heuristic H1

Let S_1 be the JA schedule for the AF with processing times $\left(\frac{1}{m_1}a_j, \frac{1}{m_2}b_j\right)$ and S_2 be the JA schedule for the AF with processing times $\left(\frac{1}{m_2}b_j, \frac{1}{m_1}c_j\right)$.

Step 1. Using the sequence S_1,

a. Apply FAM on *A* on the stage 1 machines.

b. Apply LBM on *B* on the stage 2 machines and schedule these tasks as early as possible. Let T' be the largest completion time until the second operation.

c. Apply FAM on *C* on the stage 1 machines from T' and schedule these tasks as early as possible.

d. Calculate the makespan, C_{S_1}.

Step 2. Emulate Step 1 by using the sequence S_2 instead of S_1. Replace *A* with *C* in step 1a, replace *C* with *A* in step 1c and calculate the makespan, C_{S_2}.

Step 3. The makespan of the heuristic is $C_{H1} = min\left(C_{S_1}, C_{S_2}\right)$.

Theorem 1. Let $C^* = C^*_{CRF_{m_1,m_2}}$, then $\frac{C_{H1}}{C^*} \le \frac{3}{2}\left(2 - \frac{1}{m}\right)$, where $m = max(m_1, m_2)$.

Heuristic H2

In Heuristic H2, Step 1c of Heuristic H1 is replaced by an LBM procedure namely:

Step 1c: Apply LBM on *C* on the stage 1 machines from $T' = \sum_{j=1}^{n}\left(a_j + b_j + c_j\right)$ and schedule these tasks as early as possible.

Theorem 2. Let $C^* = C^*_{CRF_{m_1,m_2}}$, then $\frac{C_{H2}}{C^*} \le \frac{3}{2}\left(2 - \frac{1}{m}\right)$, where $m = max(m_1, m_2)$.

Heuristic H3

Heuristics H1 and H2 use two symmetric JA sequences derived from two of the three processing times of the problem. With the lower bounds that have been derived in Lemma 7 based on the three processing times of the problem, we can modify the heuristic's input to now use two symmetric JA sequences based on all three processing times. In Lemma 7, the set of values (1), (2), (3) and (4) are symmetric to (5), (6), (7) and (8) respectively.

Consider the two AF problems with processing times $\left(a_j' + b_j', c_j'\right)$ and its symmetric pair $\left(b_j'' + c_j'', a_j''\right)$. Apply JA to these AF problems to obtain their corresponding schedule σ_k *k*=1,2. Replace S_1 and S_2 with σ_1 and σ_2 in steps 1 and 2 respectively in H1.

We can use any of the following four pairs of three processing time JA schedules in H3. We distinguish them by the following:

1. H3.1 when the pair of JA schedules is based on (1) and (5).
2. H3.2 when the pair of JA schedules is based on (2) and (6).
3. H3.3 when the pair of JA schedules is based on (3) and (7).
4. H3.4 when the pair of JA schedules is based on (4) and (8).

5. Computational Experiments for $F(m_1, m_2)|chain\ reentrant|C_{max}$

A computational experiment using various parameter values was conducted to assess the performance of the heuristic algorithms H1 and H3. H1 uses a two processing time JA schedule input while H3 uses a three processing time JA schedule. For each combination of values below, we generated 10 problem instances using: number of jobs, *n*, number of machines in stage *i*, m_i and the processing times, a_j, b_j and c_j which were randomly generated from U(l,u), which is a discrete uniform distribution in [l,u]. The values considered for these parameters are shown in Table 1.

Table 1 shows the average deviation of the heuristic solution from the lower bound, LB_1 which was established in Lemma 1. Since LB_1 dominates all the lower bounds established in Lemma 7, it is used in the calculation of the average deviation. The average deviation *AD* is given by the formula, $AD = (C((H_i) - LB_1) * 100\%/LB_1$ where $C(H_i)$ is the makespan obtained in heuristic H_i.

The following can be observed from the computational experiment.

1. The average deviation increases as m_1 increases. This can be explained by the behavior of the lower bound LB_1. Recall that $LB_1 = max\left(C_{AF1_{1,1}}, C_{AF2_{1,1}}, \frac{1}{m_1}\sum_{j=1}^{n}(a_j + c_j)\right)$ where $C_{AF1_{1,1}}$ is the makespan from a JA schedule with processing times $\left(\frac{1}{m_1}a_j, \frac{1}{m_2}b_j\right)$ and $C_{AF2_{1,1}}$ is the makespan from a JA schedule with processing times $\left(\frac{1}{m_2}b_j, \frac{1}{m_1}c_j\right)$. When m_1 is small, the lower bound is dominated by $\frac{1}{m_1}\sum_{j=1}^{n}(a_j + c_j)$. As the value of m_1 increases, the lower bound now gets dominated by either $C_{AF1_{1,1}}$ or $C_{AF2_{1,1}}$.
2. The H3 heuristic generated a better solution than H1 based on the observed lower average deviation. In all the problem cases, each variant of H3 yielded a lower average deviation versus H1. Among the variants of the H3 heuristic, H3.3 generated the smallest overall average deviation of 0.81%. However, H3.3 did not generate the smallest average deviation per problem scenario.
3. From Table 1, we can see that as the number of jobs increases, the average deviation decreases. When *n* = 40, the average deviation is 1.82% and when *n* = 80, it decreases to 0.89%. This can be explained by the higher utilization of the machines in the work centers when there are more jobs.
4. As the variability of processing time increases, the average deviation in H3 also increases. When the processing times are uniformly distributed in the interval [1,20], the average deviation is 0.78% and when the processing times are uniformly distributed in the interval [10,50] it increases to 1.24%. This cannot be extended for H1 as seen in Table 1.

Table 1: % Average Deviation from Lower Bound

$n = 40$	$a_j, b_j, c_j \sim$ U(10,50)					$a_j, b_j, c_j \sim$ U(1,20)				
	$H1$	$H3.1$	$H3.2$	$H3.3$	$H3.4$	$H1$	$H3.1$	$H3.2$	$H3.3$	$H3.4$
m_1=2, m_2=6	0.49	0.09	0.15	0.16	0.18	0.28	0.08	0.08	0.08	0.08
m_1=4, m_2=4	1.59	0.69	0.69	0.48	0.70	1.46	0.25	0.25	0.20	0.20
m_1=6, m_2=2	7.94	4.71	4.10	3.98	4.84	8.66	3.38	2.24	1.96	4.49
Average	3.34	1.83	1.64	1.54	1.91	3.47	1.24	0.86	0.75	1.59
$n = 80$										
m_1=2, m_2=6	0.18	0.06	0.07	0.06	0.07	0.27	0.02	0.02	0.02	0.02
m_1=4, m_2=4	1.01	0.19	0.19	0.22	0.18	1.03	0.10	0.10	0.10	0.10
m_1=6, m_2=2	4.04	1.99	1.85	1.89	2.32	5.72	1.36	0.73	0.61	2.29
Average	1.75	0.75	0.70	0.72	0.86	2.34	0.49	0.28	0.24	0.80

6. Model Extensions

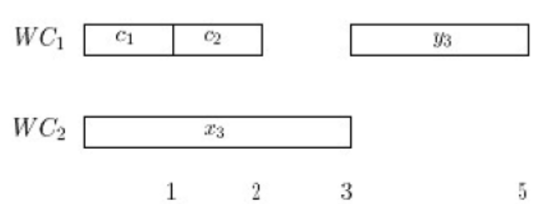
Consider the two-stage flexible job shop problem $FJ(m_1, m_2)|o = 3|C_{max}$ where $o_1, \ldots, o_n = 3$ is the number of operations of job J_j. In this problem, there are two work centers WC_1 and WC_2 with m_1 and m_2 identical machines in parallel respectively. In the two-stage flexible job shop, each job J_j can either start in WC_1 or WC_2 and must complete three operations wherein no two consecutive operations are done on the same work center. There are thus two types of jobs, type 1 jobs which start in WC_1 and type 2 jobs which start in WC_2. There are n_1 type 1 jobs and n_2 type 2 jobs. Type 1 jobs are therefore processed in a CRF_{m_1,m_2} system with processing times (a_j, b_j, c_j) for operations one, two and three respectively. Type 2 jobs have processing times (x_j, y_j, z_j) for operations one, two and three respectively wherein the *x* and *z* tasks are performed in WC_2 and the y task is processed in WC_1. Let $X = (x_1, \ldots, x_n), Y = (y_1, \ldots, y_n), Z = (z_1, \ldots, z_n)$ be the vectors of processing times for each operation of the type 2 jobs. Let us refer to this two-stage flexible job shop as FJ_{m_1,m_2} where our objective is to minimize makespan which is attained at $C^*_{FJ_{m_1,m_2}}$. Whereas the makespan of CRF_{m_1,m_2} is always attained in WC_1, the makespan of FJ_{m_1,m_2} can either occur in WC_1 or WC_2. Lemma 8 established an optimal property of the arrangement of the first operations and third operations of the type 1 jobs in WC_1 for CRF_{m_1,m_2}. In FJ_{m_1,m_2}, the work centers WC_1 (WC_2) aside from processing the first operations and third operations of the type 1 (2) jobs also process the second operations of the type 2 (1) jobs. The following lemma establishes an order between the three operations of the jobs in an optimal solution.

Lemma 9. To minimize the makespan of FJ_{m_1,m_2}, it is sufficient to consider an optimal schedule wherein all the *A (X)* tasks precede the *B (Y)* and the *C (Z)* tasks.

Proof: The proof is similar to Lemma 8.■

This lemma however does not establish the optimality of the precedence of a *Y (B)* task before a *C (Z)* task. Without loss of generality, consider WC_1 and WC_2 where each work center consists of one machine. Consider also two identical type 1 jobs with processing times (0,0,1) and one type 2 job with processing times (3,2,0). The optimal solution is shown in Figure 1. The optimal solution shows the *C* tasks precede the *Y* task. Making the *Y* task precede the *C* tasks will result to an increase of the optimal makespan. Thus, we cannot establish the precedence of a second operation before a third operation in an optimal solution.

Figure 1: Optimal Solution



Although the previous lemma does not establish the optimality of the precedence of a second operation before a third operation, we can utilize this property in the development of a heuristic algorithm for this problem. But first, we develop some lower bounds for $C^*_{FJ_{m_1,m_2}}$.

7. Lower Bounds For $C^*_{FJ_{m_1,m_2}}$

Type 1 jobs are symmetric to type 2 jobs because their stage flow sequence for each operation and the number of machines at each stage are interchanged. Since type 1 jobs are processed in a CRF_{m_1,m_2} system, a symmetric system must likewise be defined for the type 2 jobs. Let RRF_{m_2,m_1} be the reverse two-stage chain reentrant hybrid flow shop which is applicable for the type 2 jobs. Since the type 2 jobs are processed first in WC_2, the RRF_{m_2,m_1} system therefore has a stage flow sequence $\phi = (2,1,2)$.

Consider RRF_{m_2,m_1} where the processing time vector for a job J_j is (x_j, y_j, z_j). Two auxiliary two-stage flow shops $AF3_{1,1}$ and $AF4_{1,1}$ can be constructed with their respective processing times $\left(\frac{1}{m_2}x_j, \frac{1}{m_1}y_j\right)$ and $\left(\frac{1}{m_1}y_j, \frac{1}{m_2}z_j\right)$. By applying JA to these auxiliary flow shops, their corresponding makespans are $C_{AF3_{1,1}}$ and $C_{AF4_{1,1}}$. By appropriately substituting $C_{AF3_{1,1}}$ and $C_{AF4_{1,1}}$ in Lemma 1, we have $max\left(C_{AF3_{1,1}}, C_{AF4_{1,1}}\right) \leq C^*_{RRF_{m_2,m_1}}$.(32)

Lemma 10. Let $LB_2 = max\left(\frac{1}{m_1}\sum_{j=1}^{n}(a_j + y_j + c_j)\,,\ \frac{1}{m_2}\sum_{j=1}^{n}(x_j + b_j + z_j)\right)$. Then $LB_2 \leq C^*_{FJ_{m_1,m_2}}$.

Proof: If the makespan is attained in WC_1, then $\frac{1}{m_1}\sum_{j=1}^{n}(a_j + y_j + c_j) \leq C^*_{FJ_{m_1,m_2}}$ and if the makespan is attained in WC_2, then $\frac{1}{m_2}\sum_{j=1}^{n}(x_j + b_j + z_j) \leq C^*_{FJ_{m_1,m_2}}$.■

Lemma 11. $C^*_{RRF_{m_2,m_1}} \leq C^*_{FJ_{m_1,m_2}}$ and $C^*_{CRF_{m_1,m_2}} \leq C^*_{FJ_{m_1,m_2}}$.

Proof: The proof is obvious.■

Consider again RRF_{m_2,m_1} and the auxiliary two-stage flow shop now with processing times $(x'_j + y'_j, z'_j)$. Let C_{JA_3} be the makespan derived by JA for this AF. By appropriately substituting the following set of values of x'_j, y'_j and z'_j in Lemma 7, we have $C_{JA_3} \leq C^*_{RRF_{m_2,m_1}} \leq C^*_{FJ_{m_1,m_2}}$.

$$x'_j = \frac{1}{2m_2}x_j,\ y'_j = \frac{1}{2m_1}y_j,\ z'_j = \frac{1}{m_2}z_j \text{ or} \tag{9}$$

$$x'_j = \frac{1}{m_1+m_2}x_j, y'_j = \frac{1}{m_1+m_2}y_j,\ z'_j = \frac{1}{m_2}z_j \text{ or} \tag{10}$$

$$x'_j = \frac{1}{m_2+1}x_j, y'_j = \frac{1}{m_1(m_2+1)}y_j,\ z'_j = \frac{1}{m_2}z_j \text{ or} \tag{11}$$

$$x'_j = \frac{1}{m_2(m_1+1)}x_j, y'_j = \frac{1}{m_1+1}y_j,\ z'_j = \frac{1}{m_2}z_j, \tag{12}$$

By symmetry, let C_{JA_4} be the makespan derived by JA for the AF with processing times $(y''_j + z''_j, x''_j)$.By appropriately substituting the following set of values of x''_j, y''_j and z''_j in Lemma 7, we have, $C_{JA_4} \leq C^*_{RRF_{m_2,m_1}} \leq C^*_{FJ_{m_1,m_2}}$.

$$x''_j = \frac{1}{m_2}x_j,\ y''_j = \frac{1}{2m_1}y_j,\ z''_j = \frac{1}{2m_2}z_j \text{ or} \tag{13}$$

$$x''_j = \frac{1}{m_2}x_j,\ y''_j = \frac{1}{m_1+m_2}y_j,\ z''_j = \frac{1}{m_1+m_2}z_j \text{ or} \tag{14}$$

$$x''_j = \frac{1}{m_2}x_j, y''_j = \frac{1}{m_1(m_2+1)}y_j,\ z''_j = \frac{1}{m_2+1}z_j \text{ or} \tag{15}$$

$$x''_j = \frac{1}{m_2}x_j, y''_j = \frac{1}{m_1+1}y_j,\ z''_j = \frac{1}{m_2(m_1+1)}z_j, \tag{16}$$

8. Heuristic Algorithms for $FJ(m_1, m_2)|o = 3|C_{max}$

Since CRF_{m_1,m_2} and RRF_{m_2,m_1} are special cases of the more general problem FJ_{m_1,m_2}, the heuristics H1 and H3 can be modified accordingly to solve it. Since it was observed that as the number of jobs increased in CRF_{m_1,m_2}, the performance of the heuristics H1 and H3 also improved. With this insight, a modification of these heuristics can be constructed to solve FJ_{m_1,m_2}.

Lemma 9 does not establish the precedence of the second operation before the third operation in an optimal solution. However, in order to establish some structure in the heuristic, we will make the second operation always precede the third operation. The following heuristic illustrates the use of this property together with the optimal property stated in Lemma 9. The modified versions of H1 and H3 are now presented as H1a and H3a respectively.

Heuristic H1a

Let $S_{1.1}$ be the JA schedule for the AF with processing times $\left(\frac{1}{m_1}a_j, \frac{1}{m_2}b_j\right)$ and $S_{1.2}$ be the JA schedule for the AF with processing times $\left(\frac{1}{m_2}x_j, \frac{1}{m_1}y_j\right)$. Let $S_{2.1}$ be the JA schedule for the AF with processing times $\left(\frac{1}{m_2}b_j, \frac{1}{m_1}c_j\right)$ and $S_{2.2}$ be the JA schedule for the AF with processing times $\left(\frac{1}{m_1}y_j, \frac{1}{m_2}z_j\right)$.

Step 1. Using the sequence $S_{1.1}$ for A, B and C and $S_{1.2}$ for X, Y and Z:

a. Apply FAM on A and X in WC_1 and WC_2 respectively.

b. Apply LBM on B and Y in WC_2 and WC_1 respectively and schedule them as early as possible.

c. Apply FAM on C and Z in WC_1 and WC_2 respectively.

d. Calculate the makespan CS_f.

Step 2. Emulate Step 1 by using the sequences $S_{2.1}$ and $S_{2.2}$ instead of $S_{1.1}$ and $S_{1.2}$ respectively. Replace A with C and replace X with Z in step 1a. Replace C with A and Z with A in step 1c and calculate the makespan, CS_b.

Step 3. The makespan of the heuristic $C_{H1a} = \min(CS_f, CS_b)$.

Heuristic H3a

Similar to H3, consider the two AF problems with processing times $(x_j' + y_j', z_j')$ and its symmetric pair $(y_j'' + z_j'', x_j'')$. Apply JA to these AF problems to obtain their corresponding schedule *k*, *k* = 1,2. Recall that in H3, σ_1 and σ_2 replaced S_1 and S_2 respectively in H1. Similarly, replace $S_{1.1}$ and $S_{1.2}$ with σ_1 and $\sigma_2$respectively and $S_{2.1}$ and $S_{2.2}$ with τ_1 and τ_2 respectively in H1a.

We can use any of the following four pairs of three processing time JA schedules in H3a. We distinguish them by the following:

1. H3.1a when the two pairs of JA schedules are based on (1) and (5) for A, B and C and (9) and (13) for X, Y and Z.
2. H3.2a when the two pairs of JA schedules are based on (2) and (6) for A, B and C and (10) and (14) for X, Y and Z.
3. H3.3a when the two pairs of JA schedules are based on (3) and (7) for A, B and C and (11) and (15) for X, Y and Z.
4. H3.4a when the two pairs of JA schedules are based on (4) and (8) for A, B and C and (12) and (16) for X, Y and Z.

9. Computational Experiments for $FJ(m_1, m_2)|o = 3|C_{max}$

Similarly as in Section 5, a computational experiment using various parameter values was conducted to assess the performance of the heuristic algorithms H1a and H3a. H1a uses a two processing time JA schedule input while H3a uses a three processing time JA schedule. For each combination of values below, we generated 10 problem instances using: number of jobs, n_1 and n_2, number of machines in stage i,m_i and the processing times, a_j, b_j, c_j, x_j, y_j and z_j which were randomly generated from U(l,u), which is a discrete uniform distribution in [l,u]. The values of these parameters are shown in Table 2.

Table 2 shows the average deviation of the heuristic solution from the lower bound, LB_2 which was established in Lemma 10. From the computational experiments, we find that LB_2 dominates all the lower bounds established for FJ_{m_1,m_2} and is thus used in the calculation of the average deviation AD.

Since the utilization of the machines in the two work centers increases when processing both types 1 and 2 jobs, it is expected that the average deviation of the heuristic solutions will improve over the values seen in Table 1. This is validated by the results shown in Table 2. The following additional observations can be made from the computational experiment.

1. The H3a heuristic generates a better solution than H1a based on the observed lower average deviation. The average deviation for H3a was 0.21% while it was 0.96% for H1a. In all the problem cases, each variant of H3a yielded a lower average deviation versus H1a. Among the variants of the H3a heuristic, H3.3a generated the smallest overall average deviation of 0.18%. However, H3.3a did not generate the smallest average deviation per problem scenario.
2. As the number of jobs increases, the average deviation decreases. When n = 40, the average deviation of H1a is 1.27% and it decreases to 0.65% when n = 80. For H3a, the average deviation is 0.30% when n = 40 and it decreases to 0.11% when n = 80. This can be explained by the higher utilization of the machines in the work centers when there are more jobs.
3. As the variability of processing times increases, the average deviations in H3a also increase. This is observed when the average deviation of 0.15% for the interval [1,20] increases to 0.27% for the interval [10,50].

Table 2: % Average Deviation from Lower Bound

	$a_j, b_j, c_j, x_j, y_j, z_j \sim$ U(10,50)					$a_j, b_j, c_j, x_j, y_j, z_j \sim$ U(1,20)				
	$H1a$	$H3.1a$	$H3.2a$	$H3.3a$	$H3.4a$	$H1a$	$H3.1a$	$H3.2a$	$H3.3a$	$H3.4a$
$n_1 = 10, n_2 = 30$										
m_1=2, m_2=6	0.52	0.43	0.50	0.25	0.52	0.88	0.22	0.26	0.23	0.46
m_1=4, m_2=4	2.01	0.51	0.51	0.60	0.42	2.86	0.27	0.27	0.28	0.34
m_1=6, m_2=2	0.62	0.20	0.15	0.11	0.25	0.42	0.06	0.06	0.06	0.06
Average	1.05	0.38	0.39	0.32	0.40	1.39	0.18	0.20	0.19	0.28
$n_1 = 20, n_2 = 20$										
m_1=2, m_2=6	0.37	0.10	0.21	0.19	0.19	0.79	0.11	0.11	0.11	0.14
m_1=4, m_2=4	2.98	0.93	0.93	0.61	0.85	3.51	0.41	0.41	0.41	0.47
m_1=6, m_2=2	0.61	0.13	0.14	0.12	0.25	1.01	0.11	0.15	0.11	0.11
Average	1.32	0.39	0.43	0.31	0.43	1.77	0.21	0.22	0.21	0.24
$n_1 = 30, n_2 = 10$										
m_1=2, m_2=6	0.44	0.18	0.13	0.08	0.22	0.84	0.06	0.06	0.06	0.06
m_1=4, m_2=4	1.63	0.46	0.46	0.63	0.64	2.35	0.25	0.25	0.31	0.25
m_1=6, m_2=2	0.67	0.35	0.50	0.26	0.67	0.38	0.30	0.59	0.15	0.52
Average	0.91	0.33	0.36	0.33	0.51	1.19	0.20	0.30	0.17	0.28
$n_1 = 30, n_2 = 50$										
m_1=2, m_2=6	0.32	0.10	0.10	0.08	0.08	0.26	0.06	0.06	0.06	0.06
m_1=4, m_2=4	1.16	0.23	0.23	0.32	0.29	1.11	0.15	0.15	0.12	0.12
m_1=6, m_2=2	0.23	0.07	0.04	0.14	0.05	0.34	0.04	0.04	0.04	0.04
Average	0.57	0.13	0.12	0.18	0.14	0.57	0.08	0.08	0.08	0.08
$n_1 = 40, n_2 = 40$										
m_1=2, m_2=6	0.28	0.08	0.07	0.05	0.14	0.26	0.02	0.02	0.02	0.02
m_1=4, m_2=4	1.43	0.37	0.37	0.36	0.29	1.41	0.13	0.13	0.16	0.13
m_1=6, m_2=2	0.29	0.08	0.07	0.10	0.13	0.39	0.06	0.06	0.06	0.06
Average	0.66	0.18	0.17	0.17	0.19	0.69	0.07	0.07	0.08	0.07
$n_1 = 50, n_2 = 30$										
m_1=2, m_2=6	0.30	0.11	0.07	0.07	0.05	0.31	0.02	0.02	0.02	0.02
m_1=4, m_2=4	1.24	0.21	0.21	0.21	0.41	1.55	0.08	0.08	0.05	0.11
m_1=6, m_2=2	0.31	0.09	0.09	0.13	0.19	0.55	0.07	0.07	0.07	0.07
Average	0.62	0.14	0.13	0.14	0.22	0.80	0.06	0.06	0.05	0.07

10. Conclusion

This paper evaluated the performance of the heuristic solutions developed by Lorenzo (2017) for the $F(m_1, m_2)|chain\ reentrant|C_{max}$ problem against the best established lower bound through computational experiments. The results showed that the heuristic solutions gave very good approximations to the optimal solution. The CRF_{m_1,m_2} model is then extended to include features of the RRF_{m_2,m_1} model which leads to the formulation of the FJ_{m_1,m_2} model. The previous heuristic solutions and lower bounds developed for the CRF_{m_1,m_2} model are then modified for the $FJ(m_1, m_2)|o = 3|C_{max}$ problem and the computational experiments again yield very good approximations to the optimal solution. Future research shall be directed towards the improvement of the heuristic solutions for CRF_{m_1,m_2} which may yield better worst-case error bounds.

Acknowledgements

The author would like to thank the Paul Dumol PCA from the UP Diliman College of Engineering and the UP Diliman ERDT/FRDG program for providing funding for this research paper and attendance to the 17th International Conference on Project Management and Scheduling respectively.

References

Aldakhilallah K A., Ramesh R., 2001, “Cyclic scheduling heuristics for a re-entrant job shop manufacturing environment”, International Journal of Production Research, Vol.12, pp. 2635-2657.

Bispo C., Tayur S., 2001, “Managing simple reentrant flow lines: theoretical foundation and experimental results”, Vol. 33, IIE Transactions, pp. 609-623.

Buten R E., Shen V Y., 1973, “A scheduling model for computer systems with two classes of processors”, Proc. 1973 Sagamore Computer Conference on Parallel Processing, pp. 130-138.

Drobouchevitch I G., Strusevich VA., 1999, “A heuristic algorithm for two-machine re-entrant shop scheduling”, Annals of Operations Research, Vol. 86, pp. 417-439.

Graham R L., Lawler, E L. and Lenstra J K., 1979, “Optimization and approximation in deterministic sequencing and scheduling: a survey”, Ann. Discrete Math, Vol. 5, pp. 287-326.

Guinet A G., Solomon M M.. 1996, “Scheduling hybrid flowshops to minimize maximum tardiness or maximum completion time”, Vol. 34, pp. 1643-1654.

Gupta J N D., Tunc E A., 1994, “Scheduling a two-stage hybrid flowshop with separable setup and removal times”, Vol. 77, pp. 415-428.

Hall N G., Lee T E., Posner M E., 2002, “The complexity of cyclic shop scheduling problems”, Journal of Scheduling, Vol. 5, pp. 307-327.

Hoogeveen J A., Lenstra J K., Veltman B., 1996, “Preemptive scheduling in a two-stage multiprocessor flow shop is NP-hard”, European Journal of Operational Research, Vol. 89, pp. 172-175.

Hwang H, Sun J U., 1997, “Production sequencing problem with reentrant work flows and sequence dependent setup times” Vol. 33, pp. 773-776.

Johnson S M., 1954, “Optimal two and three-stage production schedules with setup times included”, Naval Research Logistics Quarterly, Vol. 1, pp. 61-68.

Koulamas C., Kyparisis G J., 2000, “Asymptotically optimal linear time algorithms for two-stage and three-stage flexible flow shops”, Naval Research Logistics, Vol. 47, pp. 259-268.

Kubiak W., Lou S X., Wang Y., 1996, “Mean flow time minimization in reentrant job shops with a hub”, Operations Research, Vol. 44, pp. 764-776.

Lee C Y., Vairaktarakis G L., 1994, “Minimizing makespan in hybrid flowshops”, Operations Research Letters, Vol. 16, pp. 149-158.

Lev V., Adiri I., 1984, “V-Shop scheduling”, European Journal of Operational Research, Vol. 18, pp. 51-56.

Lorenzo, L., 2017, “Minimizing Makespan in a Class of Two-Stage Chain Reentrant Hybrid Flow Shops”, Proc. of the World Congress on Engineering (WCE) 2017, Vol. I, pp. 50-56.

Lu S, Kumar P. R., 1991, “Distributed scheduling based on due dates and buffer priorities”, IEEE Trans. on Automatic Control, Vol. 12, pp. 1406-1416.

Wang M Y., Sethi S P. and van de Velde S L., 1997, “Minimizing makespan in a class of reentrant shops”, Operations Research Vol. 45, pp. 702-712.