International Journal of
Operations Research

International Journal of Operations Research Vol. 10, No. 1, 29—37 (2013)

New Heuristics for No-Wait Flowshops with Performance Measures of Makespan and Mean Completion Time

Harun Aydilek[1,*] **and Ali Allahverdi**[2]

[1]Department of Mathematics and Natural Sciences, Gulf University for Science and Technology P.O. Box 7207, Hawally 32093, Kuwait

[2]Department of Industrial and Management Systems Engineering, Kuwait University, P.O. Box 5969, Safat, Kuwait

Received November 2012; *Revised February 2013*; *Accepted March 2013*

Abstract — The m-machine no-wait flowshop scheduling problem is addressed by taking into account two performance measures of makespan and mean completion time. The objective is to minimize makespan such that mean completion time is less than a certain value. A dominance relation is provided for a special case of the problem. Moreover, two new heuristics are proposed in this paper. Computational analysis indicates that one of the proposed heuristics (eSA) reduces the error of the previously best known heuristic for the problem (HH1) by 70% while the computational time of HH1 is 30% more than that of eSA. Furthermore, the computational analysis also indicates that the other proposed heuristic (eHH) reduces the error of HH1 by 80% while both eHH and HH1 have the same computational time. All the results have been statistically verified.

Keywords — Scheduling, no-wait flowshop, heuristic, makespan, mean completion time, multi-criteria.

1. INTRODUCTION

In many industries, including metal, plastic, and chemical, a no-wait constraint occurs when the operations of a job have to be processed continuously from start to end without interruptions either on or between machines. Therefore, when needed, the start of a job on a given machine is delayed in order that the operation's completion coincides with the start of the next operation on the subsequent machine. This problem is known as no-wait flowshop problem in the literature. The no-wait flowshop problem has attracted the attention of many researchers.

Hall and Sriskandarajah (1996) presented detailed applications and research on this problem. Two commonly used performance measures in the no-wait flowshop scheduling literature are makespan and mean completion time.

For the m-machine no-wait flowshop scheduling problem with makespan minimization objective, many heuristics have been proposed in the literature. For example, Aldowaisan and Allahverdi (2003) proposed several heuristics and showed that their heuristics outperform the previous ones. Grabowski and Pempera (2005) compared the performance of their several algorithms with two local search algorithms proposed by Schuster and Framinan (2003). Framinan and Nagano (2008) proposed a new heuristic based on an analogy between the m-machine no-wait flowshop problem and the well-known Traveling Salesman Problem. Recently, Tseng and Lin (2010) presented a hybrid genetic algorithm; Zhu *et al.* (2009) provided a local search algorithm, Laha and Chakraborty (2009) presented a constructive heuristic based on the principle of job insertion while Qian *et al.* (2009) proposed a hybrid differential evolution algorithm for the same problem.

The m-machine no-wait flowshop scheduling problem has also been addressed with the objective of minimizing total or mean completion time. Fink and Voß (2003) proposed different heuristics including simulated annealing and tabu search. Aldowaisan and Allahverdi (2004) proposed several heuristics and showed that their heuristics are superior to the earlier ones. Shyu *et al.* (2004) proposed an Ant Colony Optimization heuristic and compared their heuristics with the earlier developed heuristics that were originally developed for the case of setup times. Pan *et al.* (2008) presented a particle swarm optimization algorithm for the same problem. Framinan *et al.* (2010) provided a new constructive heuristic for the problem and showed that their heuristic performs better than the earlier existing heuristics.

Allahverdi and Aldowaisan (2002) addressed the no-wait flowshop scheduling problem of with both performance measures of makespan and mean completion time. However, they reduced the problem into a single criterion problem by converting the two performance measures into a weighted sum of the two. In this paper, on the other hand, we consider the problem of minimizing makespan subject to the constraint that mean completion time is not greater than a given value.

Aydilek and Allahverdi (2012) addressed this problem in their recent paper where they proposed several heuristics to solve the problem. They showed that one of their proposed heuristics performs very well. In this paper, we address the same problem of Aydilek and Allahverdi (2012), and propose two new heuristics. We show that our new heuristics perform much

* Corresponding author's email: aydilek.h@gust.edu.kw

better than that of the best performing heuristic of Aydilek and Allahverdi (2012). Furthermore, we develop a dominance relation for the case of 4-machine.

2. PROBLEM DEFINITION AND A DOMINANCE RELATION

Let $C_{\max}$ and MCT denote make span and mean completion time, respectively. Furthermore, let $C_{\max}(\pi)$ and $MCT(\pi)$ represent make span and mean completion time of a given sequence π. The problem can defined as:

Minimize $C_{\max}$

Subject to $MCT \leq M$

In other words, the problem is to find a sequence that minimizes makespan such that the mean completion time is not greater than an M value. M is an upper bound on the value of mean completion time. For a given problem, this M value should be given by scheduler. If the M value is not given, then, this value can be obtained by the algorithm presented by Aydilek and Allahverdi (2012).

The problem is known to be NP-hard. Therefore, the solution of the problem can be estimated by heuristics. First, a dominance relation is presented for a special case of the problem in the next section.

We present a dominance relation for the problem of minimizing $C_{\max}$ for the case of four machines. Dominance relations are very useful for eliminating certain solutions while searching for the optimal solution, and are usually used in implicit enumeration techniques such as a branch-and-bound algorithm or dynamic programming. The objective of this paper is to present heuristics (in the following sections) to find approximate solutions to the problem rather than presenting an enumeration technique. However, the following result can be used when an implicit enumeration technique is developed for a four-machine no-wait flowshop problem. The development of such a technique can be an extension to this paper. It should be noted that there is a limit on the size of the problem that can be solved within a reasonable time when an implicit enumeration technique is used.

We define the main notation used throughout the paper in the following in table, Table 1.

n	Number of jobs
m	Number of machines
$t_{h,m}$	Processing time of job *h* on machine *m*
$t_{[h,m]}$	Processing time of the job in position *k* on machine *m*
$C_{[j]}$	Completion time of the job in position *j*
C_{max}	Makespan

Table 1. Main notation

Theorem: When jobs i and j are adjacent, for a four-machine no-wait flowshop, job i precedes job j in a solution that minimizes $C_{\max}$ if the conditions of either (i) and (iia) or (i) and (iib) are satisfied where

(i) $t_{j,k} \geq t_{i,k} \quad for \ k = 1,2,3,4 \quad and$

(iia) $\max\left(t_{i,3}; t_{j,1} + t_{j,2} - t_{i,2}; t_{j,2}\right) \leq t_{i,3} + t_{i,4} - t_{j,3}$

(iib) $\max\left(0; t_{j,1} + t_{j,2} - t_{i,2} - t_{i,3}; t_{j,2} - t_{i,3}\right) \leq t_{i,3} + t_{i,4} - t_{j,3} - t_{j,4} + \max(0; t_{i,1} + t_{i,2} - t_{j,2} - t_{j,3}; t_{i,2} - t_{j,3})$

Proof: Consider two job sequences π_1 and π_2 such that π_1 has job i in an arbitrary position τ and job j in position $\tau+1$. The sequence π_2 is exactly the same as π_1 except that job j is in position τ and job i in position $\tau+1$.
Let $C_{\max}\left(\pi_r\right)$ denote the makespan of the sequence $\pi_r \left(r = 1,2\right)$.

The completion time of the job in position k in a sequence can be shown to be as following:

$$C_{[k]} = \sum_{r=1}^{k} \max\left\{0; t_{[r,3]} - t_{[r-1,4]} + \max\left(0; t_{[r,1]} + t_{[r,2]} - t_{[r-1,2]} - t_{[r-1,3]}; t_{[r,2]} - t_{[r-1,3]}\right)\right\} + \sum_{r=1}^{k} t_{[r,4]} \qquad (1)$$

Therefore, it follows from equation (1) that the makespan for these two sequences can be written as:

$$C_{\max}(\pi_1) = \sum_{r=1}^{\tau-1} \max\left\{0; t_{[r,4]}; t_{[r,3]} - t_{[r-1,4]} + \max\left(0; t_{[r,1]} + t_{[r,2]} - t_{[r-1,2]} - t_{[r-1,3]}; t_{[r,2]} - t_{[r-1,3]}\right)\right\} + \sum_{r=1}^{\tau-1} t_{[r,4]}$$
$$+ \max\left\{0;\ t_{i,3} - t_{[\tau-1,4]} + \max(0;\ t_{i,1} + t_{i,2} - t_{[\tau-1,2]} - t_{[\tau-1,3]};\ t_{i,2} - t_{[\tau-1,3]})\right\}$$

$$
\begin{aligned}
&+ t_{i,4} \\
&+ t_{i,4} \max\left\{0;\ t_{j,3} - t_{i,4} + \max\left(0;\ t_{j,1} + t_{j,2} - t_{i,2} - t_{i,3};\ t_{j,2} - t_{i,3}\right)\right\} \\
&+ t_{i,4} \\
&+ \max\left\{0;\ t_{[t+2,3]} - t_{j,4} + \max(0;\ t_{[t+2,1]} + t_{[t+2,2]} - t_{j,2} - t_{j,3};\ t_{[t+2,2]} - t_{j,3})\right\} \\
&+ t_{[t+2,4]} \\
&+ \sum_{r=\tau+3}^{n} \max\left\{0;\ t_{[r,3]} - t_{[r-1,4]} + \max\left(0;\ t_{[r,1]} + t_{[r,2]} - t_{[r-1,2]} - t_{[r-1,3]};\ t_{[r,2]} - t_{[r-1,3]}\right)\right\} \\
&+ \sum_{r=\tau+3}^{n} t_{[r,4]} \qquad . \qquad (2)
\end{aligned}
$$

$$
\begin{aligned}
C_{\max}(\pi_2) &= \max\left\{0;\ t_{[r,3]} - t_{[r-1,4]} + \max\left(0;\ t_{[r,1]} + t_{[r,2]} - t_{[r-1,2]} - t_{[r-1,3]};\ t_{[r,2]} - t_{[r-1,3]}\right)\right\} \\
&+ \sum_{r=1}^{\tau-1} t_{[r,4]} \\
&+ \max\left\{0;\ t_{j,3} - t_{[\tau-1,4]} + \max(0;\ t_{j,1} + t_{j,2} - t_{[\tau-1,2]} - t_{[\tau-1,3]};\ t_{j,2} - t_{[\tau-1,3]})\right\} \\
&+ t_{j,4} \\
&+ \max\left\{0;\ t_{i,3} - t_{j,4} + \max\left(0;\ t_{i,1} + t_{i,2} - t_{j,2} - t_{j,3};\ t_{i,2} - t_{j,3}\right)\right\} \\
&+ t_{i,4} \\
&+ \max\left\{0;\ t_{[\tau+2,3]} - t_{\tau,4} + \max(0;\ t_{[\tau+2,1]} + t_{[\tau+2,2]} - t_{i,2} - t_{i,3};\ t_{[\tau+2,2]} - t_{i,3})\right\} \\
&+ t_{[t+2,4]} \\
&+ \sum_{r=\tau+3}^{n} \max\left\{0;\ t_{[r,3]} - t_{[r-1,4]} + \max\left(0;\ t_{[r,1]} + t_{[r,2]} - t_{[r-1,2]} - t_{[r-1,3]};\ t_{[r,2]} - t_{[r-1,3]}\right)\right\} \\
&+ \sum_{r=\tau+3}^{n} t_{[r,4]} \qquad (3)
\end{aligned}
$$

Since both sequences have the same jobs in all positions except τ and $\tau+1$, it follows from equations (2) and (3) that

$$
\begin{aligned}
&C_{\max}\left(\pi_2\right) - C_{\max}\left(\pi_1\right) = \\
&\max\left\{0;\ t_{i,3} - t_{[\tau-1,4]} + \max(0;\ t_{i,1} + t_{i,2} - t_{[\tau-1,2]} - t_{[\tau-1,3]};\ t_{i,2} - t_{[\tau-1,3]})\right\} \\
&+ \max\left\{0;\ t_{j,3} - t_{i,4} + \max\left(0;\ t_{j,1} + t_{j,2} - t_{i,2} - t_{i,3};\ t_{j,2} - t_{i,3}\right)\right\} \\
&+ \max\left\{0;\ t_{[\tau+2,3]} - t_{j,4} + \max(0;\ t_{[\tau+2,1]} + t_{[\tau+2,2]} - t_{j,2} - t_{j,3};\ t_{[\tau+2,2]} - t_{j,3})\right\} \\
&- \max\left\{0;\ t_{j,3} - t_{[\tau-1,4]} + \max(0;\ t_{j,1} + t_{j,2} - t_{[\tau-1,2]} - t_{[t-1,3]};\ t_{j,2} - t_{[\tau-1,3]})\right\} \\
&- \max\left\{0;\ t_{i,3} - t_{j,4} + \max\left(0;\ t_{i,1} + t_{i,2} - t_{j,2} - t_{j,3};\ t_{i,2} - t_{j,3}\right)\right\} \\
&- \max\left\{0;\ t_{[\tau+2,3]} - t_{i,4} + \max(0;\ t_{[\tau+2,1]} + t_{[\tau+2,2]} - t_{i,2} - t_{i,3};\ t_{[\tau+2,2]} - t_{i,3})\right\} \qquad (4)
\end{aligned}
$$

From condition (iia) of the theorem,

$$t_{j,3} + \max\left(t_{i,3}; t_{j,1} + t_{j,2} - t_{i,2}; t_{j,2}\right) \le t_{i,3} + t_{i,4}$$

If we subtract $t_{i,3}$ and $t_{i,4}$ from both sides of the above equation, we obtain

$$t_{j,3} - t_{i,4} + \max\left(t_{i,3}; t_{j,1} + t_{j,2} - t_{i,2}; t_{j,2}\right) - t_{i,3} \le 0$$

By using the property of maximum function, we get

$$t_{j,3} - t_{i,4} + \max(0; t_{j,1} + t_{j,2} - t_{i,2} - t_{i,3}; t_{j,2} - t_{i,3}) \le 0 \qquad (5)$$

It follows from equation (5) that

$$\max\left\{0;\ t_{j,3}-t_{i,4}+\max\left(0;t_{j,1}+t_{j,2}-t_{i,2}-t_{i,3};t_{j,2}-t_{i,3}\right)\right\}=0 \quad (6)$$

Hence, if either equation (6) holds [equivalently condition (iia) of the theorem holds] or condition (iib) of the theorem holds, then

$$\max\left\{0;\ t_{j,3}-t_{i,4}+\max\left(0;t_{j,1}+t_{j,2}-t_{i,2}-t_{i,3};t_{j,2}-t_{i,3}\right)\right\}\leq$$
$$\max\left\{0;\ t_{i,3}-t_{j,4}+\max\left(0;t_{i,1}+t_{i,2}-t_{j,2}-t_{j,3};t_{i,2}-t_{j,3}\right)\right\} \quad (7)$$

Furthermore,

$$max\{0;\ t_{i,3}-t_{[t-1,4]}+max(0;t_{i,1}+t_{i,2}-t_{[t-1,2]}-t_{[t-1,3]};t_{i,2}-t_{[t-1,3]})\}\leq$$
$$max\{0;t_{j,3}-t_{[t-1,4]}+max(0;t_{j,1}+t_{j,2}-t_{[t-1,2]}-t_{[t-1,3]};t_{j,2}-t_{[t-1,3]})\} \quad (8)$$

since $t_{i,k}\leq t_{j,k}$ for k=1,2,3.

Also,

$$\max\{0;t_{[t+2,3]}-t_{j,4}+\max(0;t_{[t+2,1]}+t_{[t+2,2]}-t_{j,2}-t_{j,3};t_{[t+2,2]}-t_{j,3})\}\leq$$
$$\max\{0;t_{[t+2,3]}-t_{i,4}+\max(0;t_{[t+2,1]}+t_{[t+2,2]}-t_{i,2}-t_{i,3};t_{[t+2,2]}-t_{i,3})\} \quad (9)$$

since $t_{i,k}\leq t_{j,k}$ for k=2,3,4.

Hence, it follows from equations (4), (7), (8), and (9) that

$C_{\max}(\pi_2)\leq C_{\max}(\pi_1)$.

3. HEURISTICS

The problem is to find a sequence which minimizes Cmax such that MTC is less than or equal to an M value. It is assumed that the M value is given by scheduler. However, if the M value is not given or known, then, the algorithm given by Aydilek and Allahverdi (2012) can be used to obtain an M value and an initial sequence called π.

Aydilek and Allahverdi (2012) proposed several heuristic for the problem addressed. In this paper, first, a new version of simulated annealing algorithm (eSA) is introduced. In the standard SA, the positions of two randomly selected jobs are exchanged. It is known that in general, insertion operator performs better than exchange operator. Aydilek and Allahverdi (2012) used this idea and showed that one of their heuristics using the insertion operator performs better than the standard SA. However, this is not always the case. Hence, instead of considering only one of the operators, we consider and compute both of the operators. We take the sequence yielding the better result when feasible. More specifically, this is given in steps 4 and 5 of eSA given below.

Step 1. Decide the initial temperature T_i, final temperature T_f, cooling factor cf, the number of repetitions R_n, and the initial sequence s_i, which is the sequence π explained above.

Step 2. Set the temperature $T=T_i$ and the sequence $s=s_i$.

Step 3. Set $j=1$

Step 4. Pick two random integers k and l between 1 and n. Interchange the jobs in position k and l of the sequence s, and call this new sequence $st1$. Insert the job in position k to position l of the sequence s, and call this new sequence $st2$.

Step 5. Evaluate $L=f\left(s\right), Lt1=f\left(st1\right)$ and $Lt2=f\left(st2\right)$ where f is the objective function to be minimized. Define $Lt=\min\left(Lt1,\ Lt2\right)$ and set $st=st2$ if $Lt2\leq Lt1$ and set $st=st1$ otherwise.

Step 6. (Feasibility condition) If $MCT\left(s_t\right)\leq M$, go to Step 7. Else, go to Step 8.

Step 7. If $L_t<L$ then update s with s_t, i.e set $s=s_t$. Else, update s with s_t, with probability $\exp\left(-d/T\right)$, where $d=\left(L_t-L\right)/L$.

Step 8. Set $j=j+1$. If $j=R_n+1$, go to Step 9, else go to Step 4.

Step 9. Set $T=T*cf$

Step 10. If $T<T_f$, go to Step 11, else go to Step 3.

Step 11. s is the sequence for the eSA.

We also propose another heuristic which is a combined version of eSA and HA, where the description of HA is given by Aydilek and Allahverdi (2012). This new heuristic is called eHH. In this new heuristic, the solution is obtained by applying HA to the sequence obtained in Step 11 of eSA.

4. COMPUTATIONAL ANALYSIS

The problem addressed in this paper was earlier addressed by Aydilek and Allahverdi (2012), and they proposed 5 heuristics called SA, HA, mSA, HH1, and HH2. Among these five, the best performing heuristic was shown to be HH1. Therefore, we compare our newly proposed heuristics eSA and eHH with HH1 of Aydilek and Allahverdi (2012). Since HH1 was based on SA, HA, and mSA, we also include these three heuristics in our comparison. In order to have a fair comparison, the same parameters for the simulated annealing algorithm and HA of Aydilek and Allahverdi (2012) are used. The parameters for the Simulated Annealing algorithm are initial temperature, Ti = 0.10, final temperature, Tf = 0.0001, cooling factor, cf = 0.98 and the number of repetitions, Rn = 50. The L value for the heuristic HA was set to 20. In order to have the same computational time, the value of L is set to 16 for the heuristic eHH which is a combination of eSA and HA. Moreover, the processing times are generated similarly which are randomly generated from a uniform distribution U(1,100) as recommended by Hall and Posner (2001). The computer used was a PC with Intel Core 2 Duo CPU T8300 processor of 2.40 GHz running under Windows Vista Business Service Pack 2 operating system with 2GB RAM.

The performances of the heuristics are compared for different values of number of jobs, n, and the number of machines, m. The n values are set to 30, 40, and 50 while those for m are 3, 5, and 10. For each selected combination of n and m, forty replicates are generated. The performances of the heuristics are evaluated by percentage relative error (RE). The percentage relative error is defined as 100*(Average Cmax of the heuristic - Average Cmax of the best heuristic)/ Average Cmax of the best heuristic.

The results of the simulations for RE are given in Table 2 for all combinations of n and m. Each point in Table 2 is the average value of forty replicates. The results are also summarized in Figure 1 which presents the RE versus the number of jobs while Figure 2 indicates the error versus the number of machines. The results in Table 2 and Figure 1 verify the conclusions of Aydilek and Allahverdi (2012). Moreover, the table and the figures clearly indicate that the two newly proposed heuristics eSA and eHH perform much better than the best heuristic HH1 of Aydilek and Allahverdi (2012) for all the considered cases and eHH performs slightly better than eSA. Furthermore, the new heuristic eSA is not only much better than HH1 in terms of performance, but also it is faster than HH1 as seen in Figure 3. On the other hand, the proposed heuristic eHH performs much better than that of eSA while their computational times are almost the same.

Results are verified by ANOVA analysis at a significance level of 0.025. According to the ANOVA results, at least one of the heuristics performs significantly different than the others. Hence, we have further compared the heuristics by using a Tukey honestly significant difference test (HSD) at a significance level of 0.025. According to the results, in general, the performances of eSA and eHH are statistically better than the rest. Moreover, the heuristic eHH performs better than eSA on average; however, this is not supported by statistical analysis. Furthermore, in general, both eSA and eHH perform statistically better than HH1 which is the best performing heuristic known for the problem.

The above results for the Tukey honestly significant difference test can also be observed from the confidence intervals (CI) of the means. For example, the CIs for the heuristics are given in Figure 3 for the case of n=40 and m=5. For the sake of brevity, the CIs for the other cases are not given. Figure 3 shows that the performances of eSA and eHH are statistically better than that of mSA, HH1, HA and SA. Similarly, mSA and HH1 performs significantly better than HA and SA, and HA performs significantly better than SA.

The overall Relative Error of SA, HA, mSA, HH1, eSA and eHH are 2.67, 1.75, 1.30, 1.07, 0.33, 0.22, respectively. Therefore, the newly proposed heuristic eSA reduces the error of the best known heuristic for the problem (HH1) by 69.2% (100*(1.07-0.33)/1.07) while the computational time of eSA is about 30% less than that of HH1, see figure 4. Furthermore, as the number of jobs increases, the gap between the computational times gets larger. On the other hand, the other newly proposed heuristic eHH reduces the error of HH1 by 79.4% while the computational times of the two heuristics are almost the same. When the newly proposed heuristics eSA and eHH are compared, eHH performs 33% better than eSA in terms of RE while the computational time of eHH is about 30% more than that of eSA.

Table 2. Relative errors of proposed heuristics

		n			
m	**Heuristic**	30	40	50	**Average**
3	**SA**	1.48	1.39	1.52	1.46
	HA	1.74	1.61	2.62	1.99
	mSA	1.07	1.28	1.74	1.36
	eSA	0.13	0.05	0.15	0.11
	1HH	1.05	1.11	1.45	1.2
	eHH	0.11	0.01	0.02	0.05
5	**SA**	2.73	2.84	2.85	2.81
	HA	1.56	1.73	2.29	1.86
	mSA	1.05	1.14	1.65	1.28
	eSA	0.26	0.25	0.48	0.33
	1HH	1.04	1.07	1.02	1.05
	eHH	0.25	0.21	0.23	0.23
10	**SA**	3.68	3.78	3.78	3.75
	HA	1.02	0.98	2.2	1.4
	mSA	1.03	1.12	1.59	1.25
	eSA	0.41	0.49	0.8	0.57
	1HH	1.03	0.94	0.89	0.95
	eHH	0.41	0.38	0.34	0.38
Average	**SA**	2.63	2.67	2.72	2.67
	HA	1.44	1.44	2.37	1.75
	mSA	1.05	1.18	1.66	1.3
	eSA	0.27	0.26	0.48	0.33
	1HH	1.04	1.04	1.12	1.07
	eHH	0.26	0.2	0.19	0.22

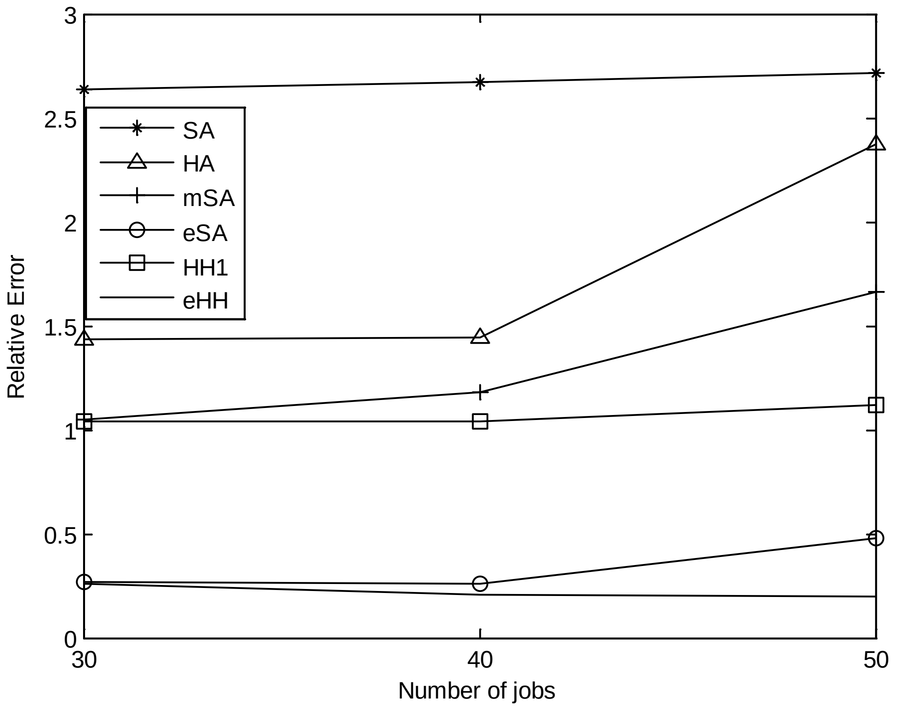

Fig 1. Relative errors versus number of jobs

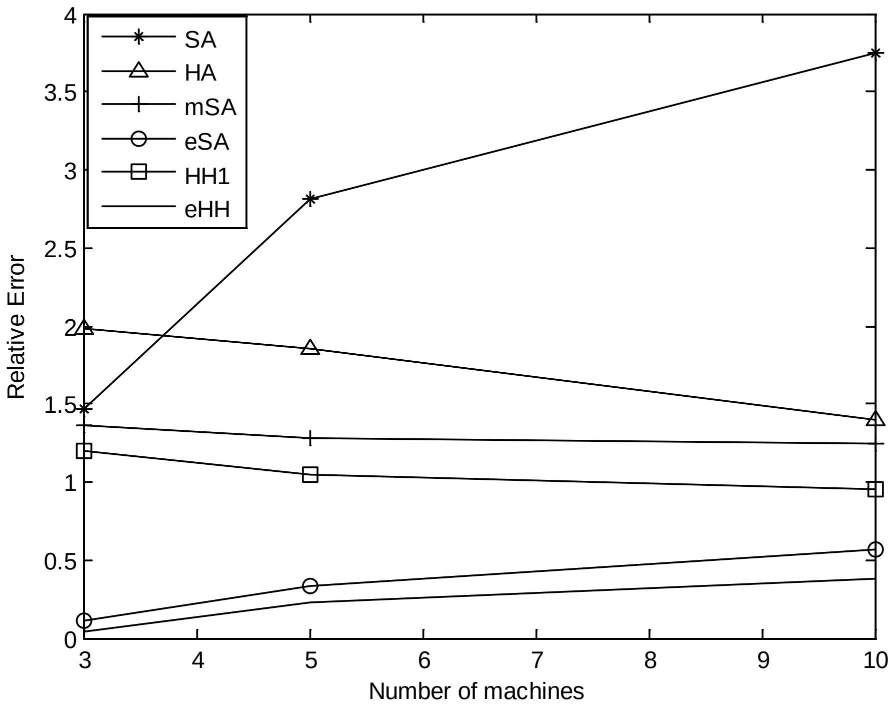

Fig 2. Relative errors versus number of machines

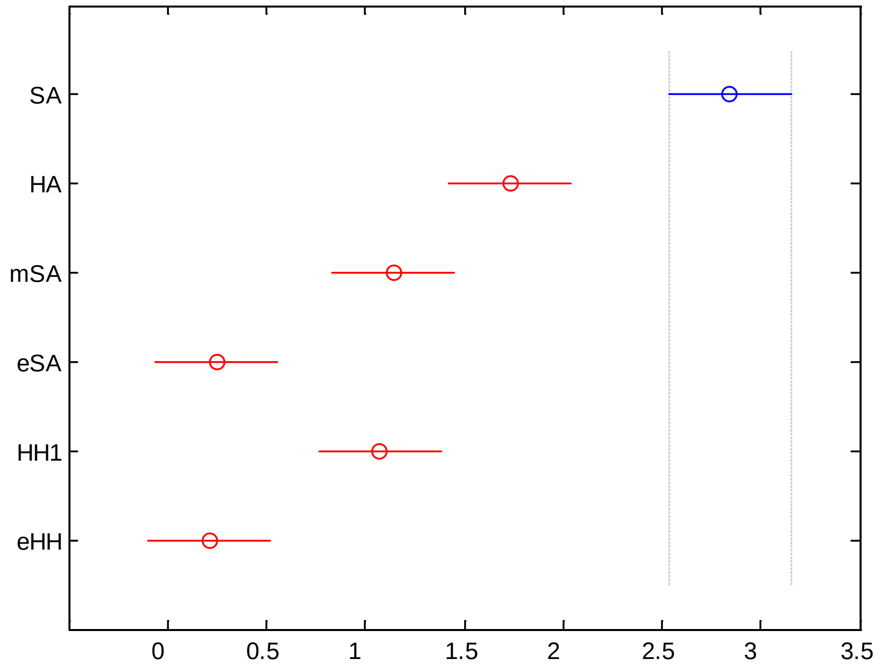

Fig 3. Means and 95 % Confidence Intervals for the different heuristics (n=40, m=5)

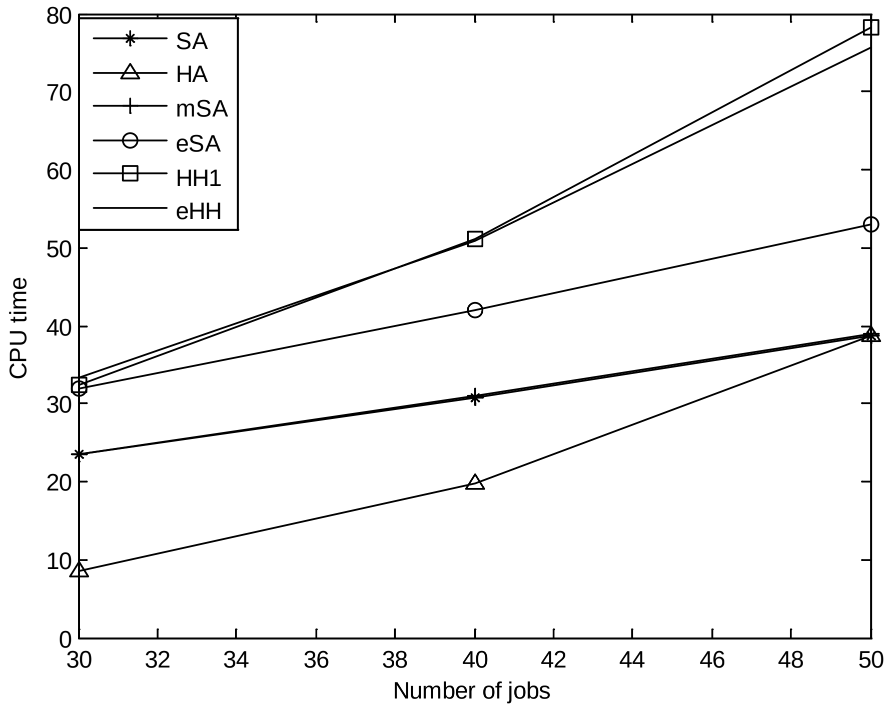

Fig 4. CPU time versus number of jobs

5. CONCLUSIONS

The no-wait m-machine flowshop scheduling problem has been addressed where the objective is to minimize makespan subject to the constraint that mean completion time should not be larger than a certain value. Two new heuristics have been proposed along with a dominance relation. It has been shown by the computational analysis that the new heuristics perform much better than the earlier best known heuristic in terms of the error while the computational time of one of the newly proposed heuristics is about three-fourths of the previously best known heuristic for the problem.

In this paper, a dominance relation has been established for the case of four machines. Dominance relations are very helpful when used in an implicit enumeration technique such as a branch-and-bound algorithm. Therefore, a possible research direction is to develop a branch-and-bound algorithm in which case the dominance relation established in this paper can be utilized. Moreover, the dominance relations can be investigated for 5 or more machines.

In this paper, setup times are ignored or assumed to be included in the processing times. This assumption may be invalid for some scheduling environments, e.g., see Vinod and Sridharan (2009). The importance of setup times, treated as separate from processing times, has been addressed by many researchers and the work of those researchers has been reviewed by Allahverdi *et al.* (2008). Therefore, another extension is to consider the problem addressed in this paper with separate setup times.

ACKNOWLEDGEMENTS

The authors are very grateful to the anonymous referees for their valuable comments and suggestions to improve the paper in the present form. This research was supported by Kuwait University Research Administration Grant No. EI03/11.

REFERENCES

1. Aldowaisan, T. and Allahverdi, A. (2003). New heuristics for no-wait flowshops to minimize makespan. *Computers & Operations Research*, 30: 1219-1231.
2. Aldowaisan, T. and Allahverdi, A. (2004). A new heuristic for m-machine no-wait flowshop to minimize total completion time. OMEGA *The International Journal of Management Sciences*, 32: 345-352.
3. Allahverdi, A. and Aldowaisan, T., (2002). No-wait flowshops with bicriteria of makespan and total completion time. *Journal of the Operational Research Society*, 53: 1004-1015.
4. Allahverdi, A., Ng, C.T., Cheng, T.C.E. and Kovalyov, M.Y., (2008). A survey of scheduling problems with setup times or costs. *European Journal of Operational Research*, 187: 985-1032.
5. Aydilek, H. and Allahverdi, A., (2012). Heuristics for no-wait flowshops with makespan subject to mean completion time.

Applied Mathematics and Computation 219: 351-359.
6. Fink, A. and Voß, S. (2003). Solving the continuous flow-shop scheduling problem by metaheuristics. *European Journal of Operational Research*, 151: 400-414.
7. Framinan, J.M. and Nagano, M.S. (2008). Evaluating the performance for makespan minimization in no-wait flowshop sequencing. *Journal of Materials Processing Technology*, 197: 1-9.
8. Framinan, J.M., Nagano and M.S, Moccellin, J.V. (2010). An efficient heuristic for total flowtime minimization in no-wait flowshops. *International Journal of Advanced Manufacturing Technology*, 46: 1049-1057.
9. Grabowski, J. and Pempera, J. (2005). Some local search algorithms for no-wait flow-shop problem with makespan criterion. *Computers & Operations Research*, 32: 2197-2212.
10. Hall, N.G. and Posner, M.E. (2001). Generating experimental data for computational testing with machine scheduling applications. *Operations Research* 49: 854-865.
11. Hall, N.G. and Sriskandarajah, C. (1996). A survey of machine scheduling problems with blocking and no-wait in process. *Operations Research*, 44: 510-525.
12. Laha, D. and Chakraborty, U.K. (2009). A constructive heuristic for minimizing makespan in no-wait flow shop scheduling. *International Journal of Advanced Manufacturing Technology*, 41: 97-109.
13. Pan, W.K, Tasgetiren, M.F. and Liang, Y.C. (2008). A discrete particle swarm optimization algorithm for the no-wait flowshop scheduling problem. *Computers & Operations Research*, 35: 2807-2837.
14. Qian, B., Wang, L., Hu, R., Huang, D.X. and Wang, X. (2009). A DE-based approach to no-wait flow-shop scheduling. *Computers & Industrial Engineering*, 57: 787-805.
15. Schuster, C.J. and Framinan, J.M. (2003). Approximative procedures for no-wait job shop scheduling. *Operations Research Letters*, 31: 308-318.
16. Shyu, S.J., Lin, B.M.T. and Yin, P.Y. (2004). Application of ant colony optimization for no-wait flowshop scheduling problem to minimize the total completion time. *Computers & Industrial Engineering*, 47: 181-193.
17. Tseng, L.Y. and Lin, Y.T. (2010). A hybrid genetic algorithm for no-wait flowshop scheduling problem. *International Journal of Production Economics*, 128: 144-152.
18. Vinod, V. and Sridharan, R. (2009). Simulation-based metamodels for scheduling a dynamic job shop with sequence-dependent setup times. *International Journal of Production Research*, 47: 1425-1447.
19. Zhu, J., Li, X. and Wang, Q. (2009). Complete local search with limited memory algorithm for no-wait job shops to minimize makespan. *European Journal of Operational Research*, 198: 378-386.