NO-WAIT FLOWSHOP SCHEDULING PROBLEM WITH TWO CRITERIA

ALI ALLAHVERDI

Department of Industrial and Management Systems Engineering Kuwait University P.O. Box 5969, Safat, Kuwait
E-mail: ali.allahverdi@ku.edu.kw

Abstract- The m-machine no-wait flowshop scheduling problem is investigated with respect to two criteria. The objective is to minimize make span such that mean completion time is less than a certain value. A dominance relation is provided for a special case of the problem, and two new algorithms are presented for the general problem. Extensive computational analysis are conducted to evaluate the performance of the newly proposed two algorithms. The analysis shows that one of the proposed algorithms (eSA) reduces the error of the previously best known algorithm for the problem (HH1) by more than two-thirds while the computational time of HH1 is one-third more than that of eSA. Furthermore, the computational analysis also shows that the other proposed algorithm (eHH) reduces the error of HH1 by more than three-thirds while both eHH and HH1 have the same computational time. All the results have been statistically verified.

Keywords- Scheduling, no-wait flowshop, algorithm, multi-criteria

I. 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.

Allahverdi (2016) 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. The m-machine no-wait flowshop scheduling problem has also been addressed with the objective of minimizing total or mean completion time.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 algorithms to solve the problem. They showed that one of their proposed algorithms performs very well. In this paper, we address the same problem of Aydilek and Allahverdi (2012), and propose two newalgorithms. We show that our new algorithms performmuch better than that of the best performing algorithm of Aydilek and Allahverdi (2012). Furthermore, we develop a dominance relation for the case of 4-machine.

II. PROBLEM DEFINITION ANDA DOMINANCE RELATION

Let C_{max} and MCT denote makespan and mean completion time, respectively. Furthermore, let C_{max}(□) and MCT(□) represent makespan and mean completion time of a given sequence □. The problem is to find a sequence that minimizes makespan such that the mean completion time is not greater than anM 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 algorithms. 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 algorithms (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.

Theorem: When jobs i and j are adjacent, for a four-machine no-wait flowshop, job iprecedes 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) $t_{j,k} \geq t_{i,k}$ for k=1,2,3,4 and
(iia) $\max(t_{i,3}; t_{j,1}+ t_{j,2}- t_{i,2}; t_{j,2}) \leq t_{i,3}+t_{i,4} - t_{j,3}$
(iib) $\max(0; t_{j,1}+ t_{j,2}- t_{i,2}- t_{i,3}; t_{j,2}- t_{i,3})$
$\leq t_{i,3}+ t_{i,4}- t_{j,3} - t_{j,4} + \max(0; t_{i,1}+ t_{i,2}- t_{j,2}- t_{j,3}; t_{i,2}- t_{j,3})$

III. ALGORITHMS (ESAANDEHH)

The problem is to find a sequence which minimizes C_{max} 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 .

Aydilek and Allahverdi (2012) proposed several algorithm 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 algorithms 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 resultwhen feasible. More specifically, this is given in steps 4 and 5 of eSA given below.
The steps of eSA is as follows.

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 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(s), Lt1=f(st1) and Lt2=f(st2) where f is the objective function to be minimized. Define Lt = min (Lt1, Lt2) and set st = st2 if Lt2 ≤ Lt1 and set st = st1 otherwise.
Step 6. (Feasibility condition) If MCT(s_t) ≤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(-d/T), where d = (L_t-L)/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 algorithm which is a combined version of eSA and HA, where the description of HA is given by Aydilek and Allahverdi (2012). This new algorithm is called eHH. In this new algorithm, the solution is obtained by applying HA to the sequence obtained in Step 11 of eSA.

IV. COMPUTATIONAL ANALYSIS

The problem addressed in this paper was earlier addressed by Aydilek and Allahverdi (2012), and they proposed 5 algorithms called SA, HA, mSA, HH1, and HH2. Among these five, the best performingalgorithmwas shown to beHH1. Therefore, we compare our newly proposed algorithms eSA and eHH with HH1 of Aydilek and Allahverdi (2012). Since HH1 was based on SA, HA, and mSA, we also include these three algorithms in our comparison. In order to have a fair comparison, the same parameters for the simulated annealing algorithm and HA ofAydilek and Allahverdi (2012) are used.The parameters for the Simulated Annealing algorithm are initial temperature, T_i = 0.10, final temperature, T_f= 0.0001, cooling factor, cf = 0.98 and the number of repetitions, R_n = 50. The L value for thealgorithmHAwas set to 20.In order to have the same computational time, the value ofL is set to 16for the algorithmeHH which is a combination of eSAand HA. Moreover, the processing times are generated similarly which are randomly generated from a uniform distribution U(1,100).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 algorithms 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 mare 3, 5, and 10. For each selected combination of n and m, forty replicates are generated. The performances of the algorithms are evaluated by percentage relative error (RE). The percentage relative error is defined as 100*(Average C_{max} of the algorithm-Average C_{max} of the best algorithm)/ Average C_{max} of the best algorithm.

The results are also summarized in Figure 1 which presents theRE 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 figuresclearly indicate that the two newly proposedalgorithms eSAand eHH perform much better than the best algorithmHH1 of Aydilek and Allahverdi (2012) for all the considered cases and eHH performs slightly better than eSA. Results are verified by ANOVA analysis at a significance level of 0.025. According to the ANOVA results, at least one of the algorithms performs significantly different than the others. Hence, we have further compared the algorithms 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 algorithmeHH 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 algorithmknown for the problem. The overall Relative Error of SA, HA, mSA, HH1, eSA and eHHare 2.67, 1.75, 1.30, 1.07, 0.33, 0.22, respectively.Therefore, the newly proposed algorithmeSA reduces the error of the best known algorithm 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 algorithmeHH reduces the error of HH1 by 79.4% while the computational times of the two algorithms are almost the same. When the newly proposed algorithms 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.

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 newalgorithmshave been proposed along with a dominance relation. It has been shown by the computational analysis that the new algorithms perform much better than the earlier best known algorithm in terms of the error while the computational time of one of the newly proposed algorithms is about three-fourths of the previously best known algorithm 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 more machines.

ACKNOWLEDGEMENTS

This research was supported by Kuwait University Research Administration Grant No. EI03/15.

REFERENCES

[1] Aldowaisan, T., and Allahverdi, A., 2003. New heuristics for no-wait flowshops to minimize makespan. Computers & Operations Research 30, 1219-1231.

[2] Allahverdi, A., 2016. A survey of scheduling problems with no-wait in process. European Journal of Operational Research255, 665-686.

[3] Allahverdi, A., 2015. The third comprehensive survey on scheduling problems with setup times/costs. European Journal of Operational Research 246, 345-378.

[4] Allahverdi, A., Aldowaisan, T., 2002. No-wait flowshops with bicriteria of makespan and total completion time. Journal of the Operational Research Society 53, 1004-1015.

[5] Aydilek, H., Allahverdi, A., 2012. Heuristics for no-wait flowshops with makespan subject to mean completion time. Applied Mathematics and Computation 219, 351-359.

[6] Framinan, J.M., Nagano, 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.

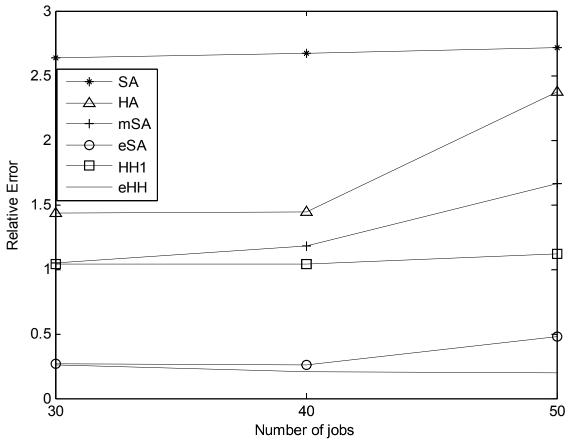

Fig 1. Relative errors versus number of job

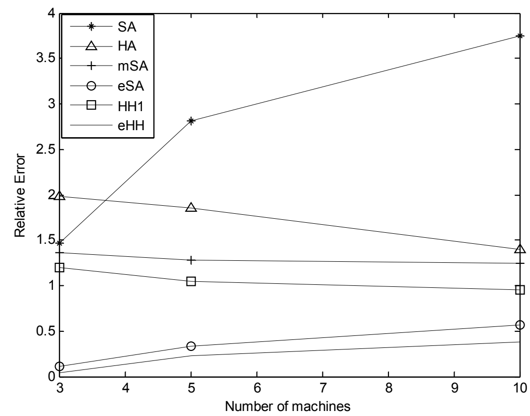

Fig 2. Relative errors versus number of machines

★★★