Research Journal of Applied Sciences 3 (2): 147-152, 2008
ISSN: 1815-932X

Heuristics for Minimizing Total Completion Time and Number of Tardy Jobs Simultaneously on Single Machine with Release Time

[1]E.O. Oyetunji and [2]A.E. Oluleye
[1]Department of Applied Mathematics and Computer Science,
University for Development Studies, Ghana
[2]Department of Industrial and Production Engineering, University of Ibadan, Ibadan, Nigeria

Abstract: In this study, the problem of simultaneously minimizing the total completion time and number of tardy jobs with release dates on a single machine is investigated. Three heuristics (called HR4, HR5 and HR6) were proposed for the bicriteria problem and were compared with the branch and bound (BB) procedure in order to evaluate effectiveness. Computational experiments, focusing on both the effectiveness (a measure of the closeness of the value of the performance measure) and efficiency (a measure of the execution time or speed) of the solution methods, were presented.

Key words: Single machine scheduling, simultaneous minimization, total completion time, number of tardy jobs, Heuristics

INTRODUCTION

Simultaneous minimization of scheduling criteria involves aggregating the criteria into a single function (called the composite objective function) which is then minimized. This approach, unlike hierarchical minimization approach, places equal importance on the scheduling criteria being considered. The problem of simultaneously minimizing 2 or more criteria has been studied by many researchers (Nagar *et al*., 1995; Oyetunji, 2006; Mazzini and Armentano, 2001; Sayin and Karabati, 1999; Cliff and Jole, 1997; Patchrawat, 2000; Shnits *et al*., 2004).

Van Wassenhove and Gelders (1980) studied the problem of simultaneously minimizing the average completion time and lateness of a schedule on one machine. They proposed a polynomial time algorithm that found the set of bicriterion schedules which are Pareto optimal. A set of schedules is said to be "Pareto optimal" if no schedule exists that is simultaneously better, in terms of both criteria, than any of the schedules in that set. Hoogeveen and Velde (1995) extended these results to show that the number of distinct schedules in the set of Pareto optimal schedules is polynomial.

Chakrabarti *et al*. (1966) considered and proposed general techniques for creating algorithms to optimize the makespan and average weighted completion time simultaneously. Also, Stein and Wein (1997) were able to improve these results by showing that for a general class of scheduling problems; bicriterion schedules exist that are simultaneous constant factor approximations for the problem of minimizing the average weighted completion time and makespan. Specifically, Stein and Wein (1997) were able to show that for many scheduling problems there exist schedules which are at most 1:88-approximations for both criteria. Rasala (1999) extended the work of Stein and Wein (1997) and proved the existence of bicriteria schedules for several pairs of common scheduling criteria. However, the results apply to a very general class of scheduling problems.

Wang (1997) studied the single machine scheduling problem of minimizing the makespan and average weighted completion time with release dates. Hoogeveen (1992, 2005) considered a number of scheduling criteria by aggregating the criteria into a single function called composite objective function. Oyetunji (2006) and Oyetunji and Oluleye (2008) built on the work of Hoogeveen (1992, 2005) by proposing a basis for evaluating the performance of solution methods to bicriteria scheduling problems.

In practice, decision makers usually have to consider multiple criteria before arriving at a decision (Ehrgott and Grandibleux, 2000). Also, the total cost of a schedule is usually a complex combination of processing costs, inventory costs, machine idle-time costs and lateness penalty costs, amongst others. A single performance

Corresponding Author: E.O. Oyetunji, Department of Applied Mathematics and Computer Science, University for Development Studies, Ghana

measure, therefore, represents only a component of the total cost of a schedule. Thus, considering scheduling problems with more than one criterion is more relevant in the context of real life scheduling problems (Nagar *et al*., 1995). In this study, the single machine scheduling problem of simultaneously minimizing the total completion time and number of tardy jobs criteria is explored. Three heuristics were proposed and compared with the branch and bound method.

The problem: Given the following for a single machine problem:

- A set of n jobs; $J_1, J_2, \ldots J_n$.
- Processing time of each job; P_i.
- Release or ready time of each job; r_i.
- Due dates of each job; d_i.

The time at which the processing of each jobs completes is C_i. A job J_i is said to be tardy if it is completed after its due date (i.e., $C_i > d_i$).
We define

$$U_i = \begin{cases} 1; C_i > d_i \\ 0; \text{otherwise} \end{cases} \tag{1}$$

The aim is to simultaneously minimize the total completion time and number of tardy jobs criteria on a single machine with release time. We assume that pre-emption is not allowed and that the problem is static and deterministic i.e., number of jobs, their processing times and ready times are all known and fixed. Our assumption seems reasonable as many real life scheduling problems can be so modeled. For an example, in a manufacturing setting, no manager would want to commence operations (running expensive machines) when uncertain about the number of jobs, their processing requirements, due dates and their ready times.

The total completion time criterion takes care of the manufacturer's concern while the customer's concern is taken care of by the number of tardy jobs criterion. Hence, the bicriteria problem being explored addresses both the manufacturer's and customer's concerns simultaneously.

Using the notations of Graham *et al*. (1979), the problem is represented as:

$$1 \left| r_i \right| \left(\sum_{i=1}^{n} C_i,\ \sum_{i=1}^{n} U_i \right)$$

Even though many combinations of scheduling criteria have been studied (Hoogeveen, 1992, 2005; Oyetunji, 2006; Mazzini and Armentano, 2001; Sayin and Karabati, 1999; Cliff and Wein, 1997; Patchrawat, 2000; Shnits *et al*., 2004; Wassenhove and Gelders, 1980; Hoogeveen and Velde, 1995; Chakrabarti et al., 1996; Wang, 1997; Rasala, 1999), it appears that the literature on simultaneous minimization of total completion time of jobs and number of tardy jobs is sparse when compared to other problem types.

SOLUTION METHODS

The solution methods are classified into the proposed and selected solution methods.

Proposed solution methods: The following solution methods are proposed for simultaneously minimizing the total completion time of jobs and number of tardy jobs with release time on a single machine. The rationale for the 3 heuristics being proposed followed the idea of Stein and Wein (1997). Stein and Wein (1997) worked on "On the existence of schedules that are near-optimal for both makespan and total weighted completion time". They showed that given an optimal makespan schedule and an optimal total weighted completion time schedule, a valid schedule can be constructed from both schedules through a process called truncation and composition of schedules.

Therefore, based on performance, existing good solution methods for the problems of minimizing total completion time of jobs with release time on a single machine and minimizing number of tardy jobs with release time on a single machine were selected from Oyetunji (2006) and combined in a systematic way to form new heuristics for the bicriteria problem.

HR4 heuristic: The HR4 heuristic made use of both HR1 and HR3 heuristics (Oyetunji, 2006) in a systematic manner. The HR1 heuristic was proposed for the problem of minimizing the total completion of jobs on a single machine with release time while the HR3 heuristic was proposed for the problem of minimizing number of tardy jobs on a single machine with release time.

HR4 heuristic steps:

Step 1	:	Obtain a schedule (S1) using HR1.
Step 2	:	Obtain a schedule (S2) using HR3.
Step 3	:	Remove all late jobs from S2 and call the remaining schedule $S2^1$
Step 4	:	Obtain schedule $S1^1$ by removing $S2^1$ from S1
Step 5	:	Form a schedule HR4 by appending $S1^1$ to the end of $S2^1$. Note: $S1^1$ should be appended in the ascending order of the processing times of the jobs in $S1^1$ schedule.

Note that HR4 cannot be worse than HR3 in terms of the number of tardy jobs. Also step 5 offers the prospect of a better total completion time of jobs.

HR5 heuristic: The HR5 heuristic is a combination of the HR1 (Oyetunji, 2006) and DAU (Dauzere-Perez, 1995) heuristics. The HR1 method was proposed for minimizing the total completion of jobs on a single machine with release time while the DAU method was proposed for minimizing the number of tardy jobs on a single machine with release time.

HR5 heuristic steps:

- Step 1 : Obtain a schedule (T1) using HR1.
- Step 2 : Obtain a schedule (T2) using DAU.
- Step 3 : Remove all late jobs from T2 and call the remaining schedule $T2^1$.
- Step 4 : Obtain schedule $T1^1$ by removing $T2^1$ from T1
- Step 5 : Form a schedule HR5 by appending $T1^1$ to the end of $T2^1$. Note $T1^1$ should be appended in the ascending order of the processing times of the jobs in $T1^1$ schedule.

HR6 heuristic: The HR6 heuristic is a combination of the AL1 (Oyetunji, 2006) and DAU (Dauzer-Perez, 1995) methods. The AL1 method was proposed for minimizing the total completion time of jobs on a single machine with release time while the DAU method was proposed for minimizing the number of tardy jobs on a single machine with release time.

HR6 heuristic steps:

- Step 1 : Obtain a schedule (R1) using AL1.
- Step 2 : Obtain a schedule (R2) using DAU.
- Step 3 : Remove all late jobs from R2 and call the remaining schedule $R2^1$.
- Step 4 : Obtain schedule $R1^1$ by removing $R2^1$ from R1
- Step 5 : Form a schedule HR6 by appending $R1^1$ to the end of $R2^1$. Note that $R1^1$ should be appended in the ascending order of the processing times of the jobs in $R1^1$ schedule.

Selected solution method: Because the literature on the metrics (criteria) studied in this paper appears sparse, there was no known polynomial time solution method(s) to compare our heuristics with. Therefore, the heuristics were compared with the branch and bound (BB) method which is known to provide optimal results.

Branch and bound procedure: The branch and bound solution method for the bicriteria problem was implemented following the procedure of Sayin and Karabati (1999). They implemented the branch and bound for the problem of simultaneously minimizing makespan and sum of completion time in a two machine flow shop by applying 2 branch and bound methods (one for makespan called BB_{MS} the other for sum of completion time called BB_{SC}) to obtain the values of makespan and sum of completion time at each node (Sayin and Karabati, 1999). Therefore, we also applied 2 branch and bound methods to obtain the values of total completion time and number of tardy jobs at each node.

The value of the normalized composite objective function at each node was computed. The node that gave the best solution (that is the smallest value of the normalized composite objective function) was used to determine branching. At the terminal node, when all the jobs have been fully assigned, the node was noted and became the solution to the considered problem.

DATA ANALYSIS

In order to assess the performance of the proposed heuristic, 50 problems each were randomly generated for 22 different problem sizes ranging from 3 to 500 jobs. In all, a total of 1100 randomly generated problems were solved. The processing time of jobs were randomly generated with values ranging between 1 and 100 inclusive. The ready time of jobs were also randomly generated with values ranging between 0 and

$$\sum_{i=1}^{n} P_i$$

inclusive. The due dates were also randomly generated with values ranges between $(r_i + p_i)$ and $(r_i + 2^*p_i)$ inclusive.

A program was written in Microsoft visual basic 6.0 to apply the solution methods (HR4, HR5, HR6 and BB) to the problems generated. The program computes the value of the normalized linear composite objective function obtained by each solution method for each problem. The data was exported to Statistical Analysis System (SAS version 9.1) for detailed analysis. SAS is a very versatile statistical package and was employed to enable credible conclusions to be drawn from the results. The hardware used for the experiment is a 2.4 GHz Pentium IV with 512 MB of main memory.

The general linear model (GLM) procedure in SAS was used to compute the mean value of the normalized linear composite objective function for each problem size (50 problem instances were solved under each problem size) and by solution methods. The test of means was also carried out using the GLM procedure so as to determine whether or not the differences observed in the mean value of the normalized linear composite objective function obtained by various solution methods are statistically significant.

RESULTS

The results obtained when the mean value of the normalized composite objective function was computed for each solution method and problem size are shown in Table 1. Based on the minimum mean value of the normalized composite objective function (effectiveness), a ranking order BB, HR4, HR6 and HR5 was obtained for $3 \leq n \leq 8$ problems while a ranking order BB, HR6, HR5 and HR4 was obtained for $10 \leq n \leq 500$ problems (Table 1).

Figure 1 shows the approximation ratio HR4/BB, HR5/BB, HR6/BB, HR6/HR4 and HR6/HR5. It was observed that the performance of HR6 and HR5 get better as the number of jobs increase whereas that of HR4 gets worse as the number of jobs increase (Fig. 1). This means that HR6 and HR5 heuristics have the potential for giving good results for large sized problems while HR4 has the potential for giving good results for small sized problems. The HR6 also appears to outperform HR5 (Fig. 1).

Table 2 shows the mean time taken (seconds) to solve an instance (efficiency) of a bicriteria problem under various problem sizes and by the solution methods evaluated. There was no solution method that was consistently better between HR5 and HR4 across the problem sizes (Table 2). However, both HR5 and HR4 were faster than HR6 and BB for 3 = n = 500. Also, the HR6 heuristic was faster than BB (Table 2). For example, HR4, HR5, HR6 and BB took about. 1,. 1, 2 and 481 sec, respectively to solve an instance of problem involving 200 jobs (Table 2).

The results shown in Table 1 were subjected to further statistical test to determine whether or not the differences observed in the mean values of the normalized composite objective function obtained from the solution methods are significant. The results obtained are summarized in Table 3-6.

Table 3 shows that the difference in the performance of BB and HR4 are not significant ($p \leq 0.05$) when $3 \leq n \leq 5$. However, the performance of both BB and HR4 were significantly different from (better than) that of HR6 and HR5 at the same level and under the same problem loading (Table 3).

When, $6 \leq n \leq 10$, the performance of BB is significantly different from (better than) the performance of HR4, HR5 and HR6 ($p \leq 0.05$). Under the same problem loading and level, the performance of HR4, HR5 and HR6 is not significantly different (Table 4). For $12 \leq n \leq 30$ problems, the performance of BB was significantly different from (better than) that of HR4, HR5 and HR6 ($p \leq 0.05$). Also, under the same problem loading and level, the performance of HR5 and HR6 are significantly different from (better than) that of HR4 (Table 5).

For $40 \leq n \leq 500$ problems, the performance of BB is significantly different from (better than) that of HR4, HR5 and HR6 ($p \leq 0.05$). Also, under the same problem loading

Table 1: Mean of the normalized composite objective function obtained from the solution methods

Problem size	Mean of the normalized composite objective function			
	BB	HR4	HR5	HR6
3×1	0.86	0.93	1.23	1.23
4×1	1.54	1.67	1.81	1.80
5×1	1.81	2.02	2.48	2.47
6×1	2.34	2.66	2.83	2.83
7×1	2.78	3.29	3.45	3.44
8×1	3.20	4.15	4.22	4.22
9×1	3.79	4.54	4.63	4.61
10×1	4.22	5.08	5.09	5.06
12×1	5.02	6.32	5.97	5.92
15×1	6.83	8.51	7.86	7.79
20×1	9.51	12.18	10.30	10.14
25×1	11.73	15.82	12.84	12.59
30×1	13.84	19.05	15.39	15.03
40×1	18.97	26.48	21.02	20.43
50×1	24.28	34.32	26.55	25.65
100×1	48.50	71.62	52.01	49.89
120×1	58.23	86.91	63.40	60.83
140×1	67.89	101.71	72.64	69.53
200×1	96.08	146.83	103.67	99.14
300×1	-	221.72	155.60	148.66
400×1	-	296.92	206.67	197.53
500×1	-	371.71	258.07	246.47

Sample size = 50

Table 2: Mean of time taken to obtain the normalized composite objective function by solution methods

Problem Size	Mean of time taken (sec)			
	BB	HR4	HR5	HR6
3×1	0.8791	0.0002	0.0003	0.1409
4×1	0.6761	0.0004	0.0001	0.1406
5×1	0.8009	0.0002	0.0003	0.1418
6×1	1.0627	0.0006	0.0001	0.1424
7×1	1.5514	0.0016	0.0010	0.1390
8×1	1.8374	0.0006	0.0013	0.1462
9×1	2.2335	0.0011	0.0018	0.1283
10×1	2.7916	0.0004	0.0002	0.1296
12×1	3.0853	0.0019	0.0016	0.1296
15×1	3.422	0.0016	0.0016	0.1315
20×1	7.4707	0.0024	0.0034	0.1349
25×1	8.9151	0.0033	0.0039	0.1512
30×1	12.4608	0.0044	0.0048	0.1515
40×1	18.6614	0.0081	0.0087	0.1721
50×1	34.6055	0.0108	0.0109	0.1721
100×1	102.5728	0.0421	0.0453	0.3615
120×1	140.3210	0.0581	0.0634	0.3956
140×1	270.2090	0.0787	0.0840	0.4681
200×1	480.5320	0.1533	0.1646	1.7628
300×1	-	0.3119	0.3234	1.7071
400×1	-	0.5443	0.5631	2.4587
500×1	-	0.8434	0.8984	4.2134

Sample size = 50

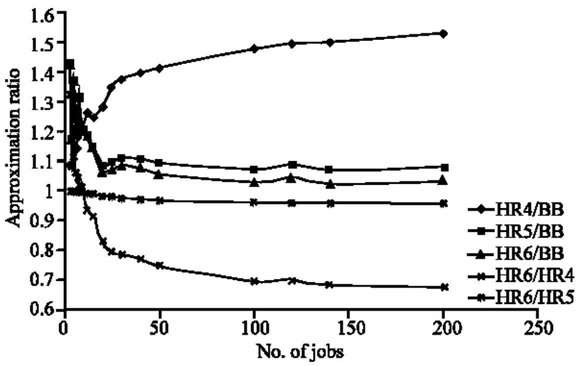

Fig. 1: Approximation ratio of the solution methods

Table 3: Test of means of normalized composite objective function for $3 \le n \le 5$ problems

	Heuristics			
Heuristics	BB	HR4	HR5	HR6
BB	-	X	*	*
HR4	X	-	*	*
HR5	*	*	-	X
HR6	*	*	X	-

Note * indicate significant result at 5% level; Sample size = 50, X, indicate non significant result at 5% level, - indicate not necessary

Table 4: Test of means of normalized composite objective function for $6 \le n \le 10$ problems

	Heuristics			
Heuristics	BB	HR4	HR5	HR6
BB	-	*	*	*
HR4	*	-	X	X
HR5	*	X	-	X
HR6	*	X	X	-

Note * indicate significant result at 5% level; Sample size = 50, X, indicate non significant result at 5% level, -indicate not necessary

Table 5: Test of means of normalized composite objective function for $12 \le n \le 30$ problems

	Heuristics			
Heuristics	BB	HR4	HR5	HR6
BB	-	*	*	*
HR4	*	-	*	*
HR5	*	*	-	X
HR6	*	*	X	-

Note *indicate significant result at 5% level; Sample size = 50, X, indicate non significant result at 5% level, -indicate not necessary

Table 6: Test of means of normalized composite objective function for $40 \le n \le 500$ problems

	Heuristics			
Heuristics	BB	HR4	HR5	HR6
BB	-	*	*	*
HR4	*	-	*	*
HR5	*	*	-	*
HR6	*	*	*	-

Note *indicate significant result at 5% level; Sample size = 50, - indicate not necessary

and level, the performance of HR6 is significantly different from (better than) that of HR4 and HR5. HR5 is also significantly different from (better than) HR4 ($p \le 0.05$) (Table 6).

CONCLUSION

The single machine bicriteria scheduling problem of simultaneously minimizing the total completion time (C_{tot}) and number of tardy jobs (NT) with release time (the

$$1 \left| r_i \right| (\sum_{i=1}^{n} C_i, \sum_{i=1}^{n} U_i)$$

problem) was explored. Three heuristics (HR4, HR5 and HR6) were proposed for this problem. Based on performance (with focus on both effectiveness and efficiency), the HR6 heuristic is recommended for the bicriteria scheduling problem of simultaneously minimizing the total completion time (C_{tot}) and number of tardy jobs (NT) with release time on a single machine.

REFERENCES

Chakrabarti, S., C.A. Phillips, A.S. Schulz, D.B. Shmoys, C. Stein and J. Wein, 1996. Improved Scheduling Algorithms for Minsum Criteria. In: F. Meyer auf der Heide and B. Monien (Eds.). Automata, Languages and Programming, number 1099 in Lecture Notes in Computer Science. Springer, Berlin. Proceedings of the 23rd International Colloquium (ICALP).

Dauzere-Perez, S., 1995. Minimizing late jobs in the general one machine scheduling problem. Eur. J. Operational Res., 81: 134-142.

Ehrgott, M. and X. Grandibleux, 2000. An Annotated Bibliography of Multiobjective Combinatorial Optimization. Report in Wirtschaftsmathematik, No. 62, Fachbereich Mathematik-Universitat Kaiserslautern.

Graham, R.L., E.L. Lawler, J.K. Lenstra and A.H.G. Rinnooy Kan, 1979. Optimization and approximation in deterministic sequencing and scheduling: A survey. Ann. Discrete Math., 5: 287-326.

Hoogeveen, J.A. and S.L. van de Velde, 1995. Minimizing total completion time and maximum cost simulatneously is solvable in polynomial time. Operations Res. Lett., 17: 205-208.

Hoogeveen, J.A., 1992. Single-Machine Bicriteria Scheduling. PhD thesis, Technische Uni-versiteit Eindhoven, The Netherlands.

Hoogeveen, J.A., 2005. Multicriteria Scheduling. Eur. J. Operational Res., 167 (3): 592-623.

Mazzini, R. and V.A. Armentano, 2001. A heuristic for Single Machine Scheduling with Early and Tardy Costs. Eur. J. Operational Res., 128: 129-146.

Nagar, A., J. Haddock and S. Heragu, 1995. Multiple and bicriteria scheduling: A literature survey. Eur. J. Operational Res., 81 (1): 88-104.

Oyetunji, E.O. and A.E. Oluleye, 2008. Evaluating Solution Methods to Bicriteria Scheduling Problems. Accepted for presentation at the 2nd International conference on engineering research and development.

Oyetunji, E.O., 2006. The development of scheduling heuristics for the bicriteria problem on a single machine. Ph.D. Thesis, University of Ibadan, Nigeria.

Rasala, A., 1999. Existence theorems for scheduling to meet two objectives. Technical report, Dartmouth College. Computer Science Technical Report PCS-TR99-347.

Sayin, S. and S. Karabati, 1999. A Bicriteria Approach to the 2-machine flowshop Scheduling problem. Eur. J. Operational Res., 113: 435-449.

Shnits, B., J. Rubinovitz and D. Sinreich, 2004. Multicriteria dynamic scheduling methodology for controlling a flexible manufacturing system. Int. J. Prod. Res., 42 (17): 3457-3472.

Stein Cliff and Wein Joel, 1997. On the Existence of Schedules that are Near-Optimal for both Makespan and Total Weighted Completion Time. Operations Res. Lett., 21 (3): 115-122.

Uthaisombut and Patchrawat, 2000. New Directions in Machine Scheduling. Ph.D Thesis, Michigan State University, USA.

Van Wassenhove, L.N. and F. Gelders, 1980. Solving a bicriterion scheduling problem. Eur. J. Operations Res., 4: 42-48.

Wang, Y., 1997. Bicriteria job sequencing with job release dates. Technical report, Max Plank Institute, Germany. MPI-I-97-1-005.