Iraqi Journal of Statistical Sciences

Heuristic Approaches for Solving bi-objective Single Machine Scheduling Problems

Hawzain. A. Salh and **Ayad Mohammed Ramadan**

Department of Mathematics, College of Science, University of Sulaimania , Sulaimania, Iraq

Article information

Correspondence:
Hawzain.A.Salh
hawzin.math@gmail.com

Abstract

In this research paper, n jobs have to be scheduled on one-machine to minimize the sum of maximum earliness and maximum tardiness. We solved a series of bi-criteria scheduling problems that related to minimize the sum of the maximum earliness and tardiness. Three new algorithms were presented, two for hierarchical objective and one for the simultaneous objective. Using the results of these algorithms, we minimize the sum of maximum earliness and maximum tardiness. This objective considered as one of the NP-hard problem, and it is also irregular, so this objective missed some helpful properties of regularity. The proposed algorithms had simple structures, and simple to implement. Lastly, they tested for different n.

1. Introduction

Scheduling (sequencing) in the theoretical and practical is an important area of operational research. Since most of the scheduling problems have large sizes, it is necessary to find a desirable algorithm for these problems (Werner et al., 2018). Many researchers considered single objective function, where in the practical the decision maker has to choose only a single objective from many objectives (T'kindt and Billaut, 2002). Recently, multi-criteria scheduling problems have increased. In general, two structures are existed hierarchical minimization and the simultaneous minimization (Hoogeveen, 1992). In the hierarchical the objectives are not in same important, where in the simultaneous the objectives have the same important.

In this paper, scheduling n jobs on single machine are considered without interruption. Each job has positive processing time and due-date. The objective function to be is the sum of maximum earliness and maximum tardiness which first is introduced by Amin-Nayeri and Moslehi (Nayeri and Moslehi, 2000). The two criteria were considered by many authors; here we introduce some recent works that related to them. Each criterion has been studied in different environments individually or composite by many authors. (Jawad et al., 2020) introduced some heuristic approaches to the total completion time and the total earliness. Also, (Ali and Jawad, 2020) used local search methods for the total completion time and the total earliness. (Cheachan and Kadhim, 2021) applied a branch and bound technique to the problem of the total completion time and the maximum earliness where the due dates are triangular fuzzy numbers. (Baker, 2014) addressed the stochastic scheduling problem to minimize total expected earliness and tardiness costs.

(Chachan and Hameed, 2019) used branch and bound method to solve four criteria, completion time, the tardiness, the earliness, and the late work. With unequal release date (Aneed, 2013) found the optimal solution by branch and bound method of the sum of completion times, maximum earliness and maximum tardiness.

Here, a new algorithm was presented to minimize the sum of maximum earliness and maximum tardiness $1//(E_{max} + T_{max})$. A new algorithm is presented which depends on an optimal solution of some related problems as shown in Figure (1). First we solved problem (1) and showed that this solution was the first efficient solution for the problem (3). Also by solving problem (2) we obtained the last efficient solution for the problem (3). The next step was finding all the efficient solutions for problem (3), and then found the sum of each efficient solution obtained the optimal solution of problem (4).

2. Notations and Definitions

In this section the following notations are used

N = the set {1, 2, 3,…, n}.

Π = the set of permutation schedules.

π = a permutation schedule.

p_j = processing time for job j.

d_j = due date for job j.

C_j = completion time for job j.

E_j = Max $\{d_j – C_j , 0\}$; the earliness of job j.

E_{max} = Max $\{E_j\}$; the maximum earliness.

T_i = Max $\{C_j – d_j, 0\}$; the tardiness of job j.

T_{max} = Max $\{T_j\}$; the maximum tardiness.

MST: (minimum slack times) jobs are sequenced in non-decreasing order of minimum slack times s_j, where $s_j = d_j - p_j$.

SPT: (shortest processing time) they are in non-decreasing order of p_j.

EDD- rule: (Early due date) they are sequenced in non-decreasing order of d_j.

Definition (1) (Jouni, 2000): Consider a problem P , a schedule $\pi \in \Pi$ (where Π is the set of all schedules) is said to be feasible, if it satisfies the constraints of P.

Definition (2) (Aneed, 2013): A feasible schedule π^* is efficient, with respect to the criteria (f and g) if there is no any feasible schedule π such that $f(\pi) \leq f(\pi^*)$ and $g(\pi) \leq g(\pi^*)$, and at least one of the inequalities is strict.

3. Mathematical Formulations

Here, some problems are considered which constructed by two objectives as shown in the Figure (1).
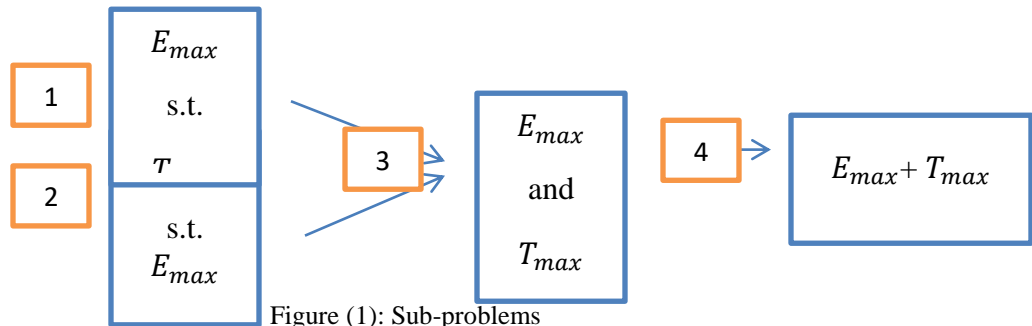

Figure (1): Sub-problems

The problem can be stated as: n jobs are available for processing at time 0, and their parameters are known in advance. The key parameters in the model include the processing time for job j (p_j) and the due date (d_j). In the actual schedule, job j completes at time C_j (Baker, 2014). The two objectives under consideration are the maximum tardiness and maximum earliness. For a sequence π of the jobs, earliness $E_{\pi(j)}$, and the tardiness $T_{\pi(j)}$ are given by:

$C_{\pi(1)} = p_{\pi(1)},$

$C_{\pi(j)} = C_{\pi(j-1)} + p_{\pi(j)},$ j= 2, 3, ..., n,

$E_{\pi(j)} = \max\{d_{\pi(j)} - C_{\pi(j)}, 0\}, \qquad\qquad j = 1, 2, ..., n,$

$T_{\pi(j)} = \max\{C_{\pi(j)} - d_{\pi(j)}, 0\}, \qquad\qquad j = 1, 2, ..., n.$

So the mathematical form of this problem can be formulated as:

$$Z = \min\{E_{max}(\pi) + T_{max}(\pi)\}$$

s. t.

$E_{\pi(j)} \geq 0, \qquad\qquad\qquad j = 1, 2, ..., n.$

$T_{\pi(j)} \geq 0, \qquad\qquad\qquad j = 1, 2, ..., n.$ ... (P)

$E_{\pi(j)} \geq d_{\pi(j)} - C_{\pi(j)}, \qquad\qquad j = 1, 2, ..., n.$

$T_{\pi(j)} \geq C_{\pi(j)} - d_{\pi(j)}, \qquad\qquad j = 1, 2, ..., n.$

4. Combine Conflicting Criteria

To solve problem (P), two structures are available to combine conflicting criteria, namely hierarchical and simultaneous (Hoogeveen, 1992). For the hierarchical case, the criteria are ranked according to their importance, where for the simultaneous case the criteria have the same importance (Hoogeveen and Velde, 1996).

4.1. Problems (1) and (2)

The problem (1) and (2) can be written as:

1/ / Lex (Tmax, Emax) and 1/ / Lex (E_{max}, T_{max}) respectively and they are not solved yet. The optimal schedules of E_{max} and T_{max} are given by MST-rule and EDD-rule respectively (Abdul-Razaq and Mohammed, 2016), but there are no any direct method to solve the composition of them. So, we will present two new algorithms.

An Algorithm (1)

Algorithm (1) for problem (1) is a modification of Smith's algorithm (Smith, 1956), and Lawler's algorithm (Lawler, 1973).

Step (1): Set $R = \sum p_j$, $N = \{1, 2, ..., n\}$, $k = n$, $j = 1, 2, ..., n$.

Step (2): Find a job j^* such that $R - d_{j^*} \leq T_{max}$ (EDD). If there exist a tie, order these jobs by MST rule. Assign job j^* in position k.

Step (3): Set $R = R - p_j$, $N = N - \{j^*\}$, $k = k-1$. If $k = 0$, stop. Else, go to step (2).

Proposition (1): Algorithm (1) gives optimal schedule for problem (1).

Proof: The job j^* in step (2) of algorithm (1) satisfies the condition $T_{j^*} \leq T_{max}$ (EDD). Also, if there exists a tie, then choose the job j^* with largest s_{j^*}. Hence the constructed schedule gives optimal for problem (1). ∎

An Algorithm (2)

Algorithm (2) for problem (2) is a modification of the algorithm of Hoogeveen and Van de Velde (Hoogeveen and Velde, 1990).

Step (1): Find E_{max} (MST) = E*.

Step (2): $k = 1$, $r_j = \max\{s_j - E^*, 0\}$ $\forall_j \in N$, $\sigma = (\varphi)$, σ be the schedule jobs.

Step (3): Find a job $j^* \in N$ with minimum r_j such that $r_j^* \leq C_{k-1}$ (if a tie exists, choose the job j^* with smallest d_{j^*}.

Step (4): $N = N - \{j^*\}$, $\sigma = (\sigma, \sigma(k))$. If $N = \varphi$ go to step (5), Else $k = k+1$, go to step (3).

Step (5): stop.

Proposition (2): Algorithm (2) gives optimal schedule for problem (2).

Proof: The job j^* in step (3) of algorithm (2) satisfies the condition $E_{j^*} \leq E_{max}$ (MST). Also, if there exists a tie, then choose the job j^* with smallest d_{j^*}. Hence the constructed schedule gives optimal for problem (2). ∎

38

4.2. Problems (3)

This problem is a simultaneous case, and it can be written as follows: $1//F(E_{max}, T_{max})$,
i.e., minimize a function F of two criteria with the same important. So, the optimal solution doesn't make sense, because there exists no such solution that minimizes both criteria simultaneously. Here, we will present a new algorithm to find all the efficient solutions. MST-rule doesn't give efficient solution for the problem (3) (Kawi and Abdul-Razaq, 2017) because there exists $\pi \in \Pi$ such that $E_{max}(\pi) = E_{max}$ (MST) and $T_{max}(\pi) < T_{max}$ (MST) as illustrate in the following example

Example (1): Consider 4- jobs problem

i	1	2	3	4
p_i	1	5	12	19
d_i	18	21	25	30

MST- sequence (4, 1, 2, 3) gives $E_{max} = 11$, $T_{max} = 12$, where the sequence (4, 3, 2, 1) gives $E_{max} = 11$, $T_{max} = 19$. Therefore, the sequence (4, 3, 2, 1) dominates the sequence is (4, 1, 2, 3). To find such π (say π^*), it is exactly the optimal solution of problem (2).

Proposition (3): π^* is the first efficient solution of problem (3).
Proof: Since $E_{max}(\pi^*) = E_{max}$ (MST) from algorithm (2), and if a tie exist during the algorithm, choose the job j* with smallest d_{j*}. So π^* is efficient solution for problem (3).

An Algorithm (3)

This algorithm finds all the efficient solutions of the problem (3) which depends on the known algorithm (Van Wassenhove and Gelders, 1980).

Step (1): Find $\pi \in \Pi$ by using algorithm (2). π is the first efficient solution of problem (3),
$\Delta = T_{max}(\pi)-1$.
Step (2): Set $R = \sum p_j$, N= {1, 2, ..., n}, k = n, j = 1, 2, ..., n.
Step (3): Find a job j* such that $R - d_{j*} \leq \Delta$. If there exist a tie, order these jobs by MST rule. Assign job j* in position k.
Step (4): Set $R = R - p_{j*}$, N = N - {j*}, k = k-1. If k = 0, π^* is a solution, go to step (5). Else, go to step (3).
Step (5): $\Delta = T_{max}(\pi^*)-1$, go to step (2).
Note that the algorithm stops whenever T_{max} reached T_{max} (EDD). This gives the following proposition.
Proposition (4): The solution of problem (1) is the last efficient solution of problem (3).
Proof: Let $\sigma \in \Pi$ be the optimal solution of problem (1), i.e., $T_{max}(\sigma) = T_{max}$ (EDD). To prove that σ is efficient for problem (3). For σ, $T_{max}(\sigma) = T_{max}$ (EDD) $< T_{max}(\sigma^*) \ \forall \ \sigma^* \in \Pi$. Also, gives minimum $E_{max}(\sigma)$. ∎

4.3. Optimal Schedule for Problem (4)

After characterized all the efficient solution for problem (3). Now it is clear that one of the sum of the efficient solutions is optimal schedule for problem (4). To illustrate the algorithms consider again the example (1)

j	1	2	3	4
p_j	1	5	12	19
d_j	18	21	25	30

Using algorithm (2) to solve problem (2), we have E_{max} (MST) = 11, and $s_1 = 17$, $s_2 = 16$, $s_3 = 13$, and $s_4 = 11$; $r_1 = 17$, $r_2 = 16$, $r_3 = 13$, and $r_4 = 11$. From r_j (j=1, 2, 3, 4), choose r_4 and assign it in position 1, then job 1, 2, 3. So, get the schedule (4, 1, 2, 3) with $E_{max} = 11$ and $T_{max} = 12$. This solution is the first efficient solution for the problem (3). Using algorithm (1) to solve problem (1), we have $R = \sum p_j = 37$, N= {1, 2, 3, 4}, k = 4, j

= 1, 2, 3, 4, T_{max} (EDD) = 7. Find that job 4 satisfies the condition, so assign job 4 in position 4, R= 37- p_4= 18. Continue to choose jobs 1, 2, 3. So the sequence (3, 2, 1, 4) is optimal with E_{max}= 13 and T_{max} = 7. It is also the last efficient solution for problem (3).

To find the efficient solutions for the problem (3), use algorithm (3) stating with the schedule (4, 1, 2, 3) with E_{max}= 11 and T_{max} = 12. Δ = 12-1=11. Job 4 satisfies the condition, R=37-19=18. Continue with ordering the jobs according to MST-rule if there exist a tie. We get the schedule (3, 2, 1, 4) with E_{max}= 13 and T_{max} = 7. Optimal solution for problem (4) is one of the efficient solutions namely (3, 2, 1, 4) with value 20.

5. Computational Results

Here, the results are introduced via computational tests to show the ability of the proposed algorithm. The problems were generated as follows: p_j are generated from the uniform distribution $[1, 10]$. Also, d_j is generated from the uniform distribution $[0, \sum_{j=1}^{n} p_j]$. The algorithm was coding by Matlab 8.1 (R2013a) and implemented on Intel (R) Pentium (R) CPU 2117U@ 1.80 GHZ, with RAM 4.00 GB personal computer.

6. Conclusion

In this paper, three algorithms were presented to find optimal schedule for the problem of minimizing the sum of maximum earliness and maximum tardiness during solving some problems related to these objectives E_{max} and T_{max}. This objective is an NP-hard and it is irregular, so some properties of regular objective are missed. Therefore, we can conclude that this objective is difficult to solve, especially in the case where the range between T_{max} (MST) and T_{max} (EDD) is large or between E_{max} (MST) and E_{max} (EDD). The proposed algorithms were easy to run and had a simple structure. To evaluate the ability of the algorithm, it was tested on some examples with different n.

References

1- Abdul-Razaq, T. S., and Mohammed, H. A. (2016). Exact and Approximation Approaches for Solving Multiple Objective Function. *Al-Mustansiriyah Journal of Science*, 27 (3), 89-97.

2- Ali, F. H., and Jawad, A. A. (2020). Minimizing The Total Completion Time and Total Earliness Time Functions for a Machine Scheduling Problem using Local Search Methods. *Iraqi Journal of Science, Special* Issue, 126-133, doi: 10.24996/ijs.2020.SI.1.17.

3- Amin-Nayeri. M.R. and Moslehi. G. (2000). Optimal Algorithm for Single Machine Sequencing to Minimize Early/ Tardy Cost. *ESTEGHLAL Journal of Engineering*, 19, 35–48. https://jcme.iut.ac.ir/article-1-180-en.html

4- Aneed, J. S. (2013). Minimizing Three Simultaneous Criteria in Machine Scheduling Problem. *Thesis, University of Thi-Qar.*

5- Baker, K. R. (2014). Minimizing Earliness and Tardiness Costs in Stochastic Scheduling. *European Journal of Operational Research*, 236 (2), 445-452.

6- Chachan, H. A. and Hameed, A. S. (2019). Exact Methods for Solving Multi-Objective Problem on Single Machine Scheduling. *Iraqi Journal of Science*, 60 (8), 1802-1813 doi: 10.24996/ijs.2019.60.8.17.

7- Cheachan, H. A. and Kadhim, M.T. (2021). Exact Method for Solving Single Machine Scheduling Problem under Fuzzy Due Date to Minimize Multi-Objective Functions. *Ibn Al-Haitham International Conference for Pure and Applied Sciences (IHICPS)*, *Journal of Physics: Conference Series* 1879, 022126 IOP Publishing doi:10.1088/1742-6596/1879/2/022126.

8- Hoogeveen, J and Van de Velde, S. (1995). Minimizing Total Completion Time and Maximum Cost Simultaneously is Solvable in Polynomial Time. *Operations Research Letters,* 17 (5), 205-208, Available: 10.1016/0167-6377(95)00023-d.

9- Hoogeveen, J.A. (1992). Single-Machine Bicriteria Scheduling. *Doctoral Thesis, CWI, Amsterdam.*

10- Hoogeveen J. A. and Van de Velde, S. (1990). Polynomial Time Algorithms for Single Machine Multicriteria Scheduling, CWI, BS-R9008. file:///C:/Users/shelan/Downloads/5747D.pdf

11- Jawad, A. A., Ali, F. H., and Hasanain, W. S. (2020). Using Heuristic and Branch and Bound Methods to Solve a Multi-Criteria Machine Scheduling Problem. *Iraqi Journal of Science*, 61 (8), 2055-2069, doi: 10.24996/ijs.2020.61.8.21.

12- Jouni. L. (2000). Multi-objective Nonlinear Pareto-Optimization. Lappeenranta University of Technology*, Laboratory of Information Processing*
.

13- Lawler E.L. (1973). Optimal Sequencing 0f a Single Machine Subject to Precedence Constraints. *Management Science*, 19 (5), 544-546.

14- Smith, W.E. (1956). Various Optimizers for Single Stage Production, *Naval Research Logistics Quarterly*, *3*(1), 59-66.

15- T'kindt, V. and Billaut, J.-C. (2002). Multi-criteria Scheduling. *Theory, Models and Algorithms Springer, Berlin*.

16- Van Wassenhove. L. N. and Gelders, F. (1980). Solving a Bicriterion Scheduling Problem. *European Journal of Operational Research.* 4, 42-48.

17- Werner, F. , Larysa, B. and Sotskov, N. (2018). Special Issue on Algorithms for Scheduling Problems. *Algorithms*, 11 (6).

18- Younis kawi, H. and Abdul-Razaq, T. S. (2017). Single Machine Scheduling To Minimize a Function of Square Completion Time and Maximum Earliness Simultaneously. *Journal of Al-Qadisiyah for Computer Science and Mathematics*, 2 (1), 79-95.

مناهج إرشادية لحل مشاكل جدولة آلة مفردة ثنائية الهدف

هاوزين احمد صالح   و   اياد محمد رمضان

قسم الرياضيات، كلية العلوم ، جامعة السليمانية، السليمانية، العراق

في ورقة البحث هذه ، يجب جدولة عدد n من الوظائف على آلة واحدة لتقليل مجموع الحد الأقصى من التبكير والتأخير الأقصى. لقد حللنا سلسلة من مشاكل الجدولة ثنائية المعايير التي تتعلق بتقليل مجموع الحد الأقصى من التبكير والتأخير. تم تقديم ثلاث خوارزميات جديدة ، اثنتان للهدف الهرمي وواحدة للهدف المتزامن. باستخدام نتائج هذه الخوارزميات ، نقوم بتقليل مجموع التبكير الأقصى والتأخير الأقصى. يعتبر هذا الهدف أحد مشكلات NP – hard ، وهو أيضًا غير منتظم ، لذلك فقد هذا الهدف بعض الخصائص المفيدة للانتظام. تحتوي الخوارزميات المقترحة على هياكل بسيطة وسهلة التنفيذ. أخيرًا ، قاموا باختبار n مختلفة.

الـكلمـات الـمفتـاحـية : جدولة المشكلات ، الوظيفة متعددة الأهداف ، الهدف الهرمي ، الهدف المتزامن