Using Branch and Bound Method to Solve Multiobjective Sequencing Problem

Adawiya Ali Mahmood Al-Nuaimi
University of Diyala
College of Science
Mathematics Department
dradawiyaali@gmail.com
Dr.adawiya@uodiyala.edu.iq

Abstract

This article offers a branch and bound (BAB) solution to the multi-objective composition minimization problem of three objectives of the total completion time ($\sum C_j$), the total lateness ($\sum L_j$) and maximum tardiness (T_{max}). Branch and bound (BAB) is an effective method to discover the optimal solution for composition multiple objective problems in disciplines of machine sequencing. The problem is difficult with just one machine. A heuristic method which is shortest processing time rule was used to find the upper bound. Using the decomposition property of the multi-objective problem, that the lower bound provided by BAB. Based on the results of computer test examples, conclusions on the BAB method's efficiency are taken into account. The BAB gives optimal solution for the problem in high quality.

Keywords: Multi-objective, sequencing, upper bound, lower bound, branch and bound method, machine, optimal solution.

الخلاصة :

هذا البحث يقدم حل بطريقة التفرع والقيد لمسألة تصغير متعددة الهدف تتركب من الاهداف الثلاثة وهي وقت الاتمام الكلي، وقت التفرع والقيد هي طريقة فعالة لاكتشاف الحل الامثل للمسائل متعددة الهدف المركبة في مجالات التأخير الكلي وأعظم تأخير لاسالب. الطريقة التقريبية التي هي قاعدة أصغر وقت تشغيل تم الترتيب على الماكنة. المسألة من النوع الصعب مع حالة الماكنة الواحدة. استخدامها لإيجاد القيد الاعلى لطريقة التفرع والقيد. باستخدام صفة التجزئة للمسألة متعددة الهدف تم تجهيز القيد الادنى لطريقة التفرع والقيد. الاستنتاجات لطريقة التفرع والقيد اخذت بالحسبان بالاعتماد على نتائج اختبار الحاسوب للأمثلة وتبين ان طريقة التفرع والقيد تعطي حل أمثل للمسألة بكفاءة عالية.

الكلمات المفتاحية: تعدد الأهداف، التسلسل، الحد الأعلى، الحد الأدنى، طريقة الفرع والربط، الآلة، الحل الأمثل.

1. Introduction

The machine sequencing issue is important in information systems, manufacturing, and production systems. Find an execution for each task (job) on one or more machines to ensure that the best possible solution is achieved sequencing is the process of minimizing the stated goal function. (Khamees. 2022)

These tasks (jobs) j (j=1,2, ..., n) demand processing time (p_j), assign due dates (d_j), specify completion times ($C_j = \sum_{i=1}^{j} p_i$) for a specific order of tasks (jobs).Researchers who study sequencing have put a lot of effort towards focusing on one consistent performance indicator (objective) that is not minimized once a task (or job) is completed. Even though most real-world sequencing problems involve multiple goals (objectives) (Al-Nuaimi, 2016). on each aim (objective), only individual investigations have typically been conducted. A tiny number of research simultaneously considered many objectives. The broad category for multi-objective sequencing issues is NP-hard (Khamees. 2022). Non-Deterministic Polynomial Time is referred to as NP.

In multiple objectives optimality problems, the aim is finding optimum solution depend on objectives function (Ibrahim, 2019). In White. (1982) study, the multiple objectives linear programming formulation is used for identifying the optimum route within efficient set on effectively policy and routs. Approximately techniques for Pareto-optimum routs in multiple objectives are indicated in the study of Warburton, (1987). A related problem to consecutive one sub matrix that is the consecutive block minimal is suggested in Abo-Alsabeh & Salhi. (2021) study.

Van Wassenhove & Gelders (1980) proposed a pseudo-polynomial algorithm for finding all efficient schedules with respect to $\sum C_j$ and T_{max}. The importance of multicriteria scheduling has been recognized in Franch, (1982) study.

Khamees, (2022) study presents an effective algorithm to locate a roughly complete collection of effective answers to the four objectives $\sum Cj$, $\sum Lj$, L_{max} and $E_{max.}$ In Fattah, (2014) study, hierarchical and composite are solved for the maximum late work and maximum earliness function. Most traditional methods solve optimization problems with small objectives (Abo-Alsabeh & Salhi, 2022). Mathematical programming and approximation methods are used for solving sequencing criteria problems (Kou, et al., 2020).

In Hesham & Abbas (2021) study, multi-criterion in the optimum drug for rheumatoid using the values of drugs as measurements and aggregate map for obtaining real value. There is an important coverage of researches in multi-criterion problems of decision making in Doumpos, et al. (2019) study.

This paper has the following format. Multi-objective sequencing problem is presented in Section 2. A general framework for a branch and bound is described. in Section 3, Includes methods for calculating the upper and lower bounds value. Results of computation-based studies are summarized in Section 4. then conclusions are provided in section 5.

2. Formulation of Multi-Objective Problem in Mathematics

The cost function F:S→ R such that S includes the set of feasible sequence. Indicates to the threeperformance goals(objectives) fi (i= 1,2,3) are combining to create a single set of objective functions. We designated F to a linear composition of the performance objective fi. The issue (problem P) of performance objective optimization of totally completion time ($\sum C_j$), overall Lateness ($\sum L_j$), maximum tardiness (T_{max}) is putted by $1/((\sum C_j + \sum L_j + T_{max})$ and it's called (P). The issue (problem P) can be represented as follows:

$$\left.\begin{array}{ll} z = \underset{\sigma\in S}{Min}\{\sum_{j=1}^{n} C_{\sigma(j)} + \sum_{j=1}^{n} L_{\sigma(j)} + T_{max}(\sigma)\} & \\ s.t. & \\ C_{\sigma(1)} = p_{\sigma(1)} & \\ C_{\sigma(j+1)} = C_{\sigma(j)} + p_{\sigma(j+1)} & j = 1,2,3,\ldots.n-1 \\ L_{\sigma(j)} = C_{\sigma(j)} - d_{\sigma(j)} & j = 1,2,\ldots.n \\ T\sigma(j) = \max\{L_{\sigma(j)}, 0\} & j = 1,2,\ldots.n \end{array}\right\} \ldots(P)$$

Where S is whole of all sequences and σ is the supplied sequence of the tasks (jobs) j, where j=1, 2, n.

Finding a processing order for the tasks (jobs) on one machine is the aim of problem (P) to minimize the sum of total completion time ($\sum C_j$), Total Lateness ($\sum L_j$), maximum tardiness (T_{max}) (i.e. $1//\sum C_j + \sum L_j + T_{max}$). Three sub problems ($SP_1$), ($SP_2$) and ($SP_3$) compound the problem (P) with the following, more basic structure:

$$\left.\begin{aligned} & Z_1 = \min_{\sigma\in S}\{\sum_{j=1}^{n} C_{\sigma(j)}\} \\ & \text{s.t.} \\ & C_{\sigma(1)} = P_{\sigma(1)} \\ & C_{\sigma(j)} = C_{\sigma(j-1)} + P_{\sigma(j)} \qquad j = 2,3,\dots,n \\ & C_{\sigma(j)} \geq 0 \end{aligned}\right\} \dots.(SP_1).$$

(SPT) rule, that is, arranging the tasks (jobs) according to their processing times (p_j) in a non-decreasing sequence. SPT rule is solving the SP_1.

$$\left.\begin{aligned} & Z_2 = \min_{\sigma\in S}\{\sum_{j=1}^{n} L_{\sigma(j)}\} \\ & s.t. \\ & C_{\sigma(1)} = P_{\sigma(1)} \\ & C_{\sigma(j)} = C_{\sigma(j-1)} + P_{\sigma(j)} \qquad j = 2,3,\dots,n \\ & L_{\sigma(j)} = C_{\sigma(j)} - d_{\sigma(j)} \qquad j = 1,2,\dots.n \end{aligned}\right\} \dots.(SP_2).$$

(SPT) rule is solving SP_2 since:

$$\sum_{j=1}^{n} L_{\sigma(j)} = \sum_{j=1}^{n}(C_{\sigma(j)} - d_{\sigma(j)})$$

$$= \sum_{j=1}^{n} C_{\sigma(j)} - \sum_{j=1}^{n} d_{\sigma(j)}$$

$$Z_3 = \min_{\sigma\in S}\{T_{max}\}$$

s.t. $\dots.(SP_3).$

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

$$T_{max} = \max\{T_{\sigma(j)}\} \qquad j = 1,2,\dots.n$$

$$T_{\sigma(j)} \geq 0$$

EDD rule is solving SP3, where EDD rule is sequencing the jobs in non- decreasing order of their due dates.

3. Branch and Bound (BAB) Method (Al-Nuaimi, 2016)

The branch and bound (BAB) method is a strategy for resolving various optimization issues. The most used solution approach in sequencing is the BAB method. An implicit enumeration approach is used in this procedure, which methodically look through several subsets of possible answers to get the best one. Typically, the operation is represented as a search tree with nodes for these subgroups. Every node in a half-finished answer generates a few additional branches that substitute a variety of fresh ones for the original, smaller, mutually incompatible issues. Two are present prevalent forms of branching: the branches in front, which is the sequential order of the tasks (or jobs) starting at the beginning and backward branching, in which the tasks (or jobs) are carried out one by one beginning at the end. For a specific sequencing issue, minimize on objective function Z, using a branching technique, the BAB method incrementally divides the issue into subsets and computes bounds making use of a lower bounds method. These processes exclude any sets of that are discovered to lack an optimal solution. This results in at least one ideal answer. Each produced sub-solution problem's is given a lower bound (LB), which is determined via the bounding technique. LB, which is the price for the randomly arranged tasks (or jobs) and the cost of the sequencing tasks (jobs) for each node, is determined (on the basis of the derived lower bound). If the value of LB for this node is greater than or equal to the upper bound (UB), which is often specified as the minimum value of all recently discovered attainable answers, then this node is ignored. Afterward, the dominance of this node, and the node with the lowest LB among those still present is selected. The branching will stop at a complete set of tasks (jobs), Afterwards, if their value is less than the current UB, they will be assessed, that value will be accepted after the UB is reset. Up till all nodes have been taken into account, the process is completed, i.e., LB≥UB for all nodes in the search tree. An ideal answer to this problem with this UB is one that can be implemented. The following methods were used to determine the lower and upper bounds in order to use the branch and bound (BAB) method to solve the issue (P).

3.1 Heuristic Approach

First, the BAB technique establishes the upper bound (UB) of the issue to be addressed at the outset. The σ = SPT rule yields UB using the heuristic approach, that is, arranging The work (tasks) in a sequence that doesn't decrease of processing times (p_j), where j=1, 2.3,..., n. Calculate the outcome sequence UB= $\sum_{j=1}^{n} C_{\sigma(j)} + \sum_{j=1}^{n} L\sigma(_j) + T_{\max}(\sigma)$.

3.2 Decomposing Technique

"LB" for (P) is the total of the minimum values (SP_1), (SP_2), and (SP_3). Consider Z_1 is the minimum amount value of (SP_1), Z_2 is the minimum amount of (SP_2) and Z_3 is the minimum amount of (SP_3), utilizing theorem (1) to derive LB as well.

3.3 Theorem (Mahmood, 2001)

Suppose Z_1, Z_2, Z_3, and Z are the minimal values for the goal function of (SP_1), (SP_2), (SP_3), and (P) separately, then

$Z_1+Z_2+Z_3 \leq Z$. □

utilizing theorem, the Lower Bound (LB) is influenced by:

LB = $Z_1+ Z_2 +Z_3$.

4. **Finding of Experiments to the Investigated Problem**

Via coding our algorithm (BAB) in Matlab and running it on an HP personal computer with 32 GB of RAM, the BAB method is put to the test. The following is how test issues are generated: every job (task) j, the discrete uniform distribution is used to provide an integer processing time $p_j \in[1,10]$. Furthermore, for every job (task) j, the discrete uniform distribution is used to produce an integer due date ∈ [P(1-TF-RDD/2), P(1-TF+RDD/2], where P= $\sum_{j=1}^{n} p_j$ based on the comparable due date range (RDD) and on the average tardiness factor (TF). TF and RDD are from (Khamees 2022). Each of the parameters, the numbers 0, 2, 4, 6, 8, and 10 are taken into consideration. Two problems are generated for each of the five potential values of the parameters, yielding ten questions (examples) for every specified number of n. where the number of tasks (jobs) n = 5, 10, 15, 20, and 21. The computation results and the problem's solution time (in seconds) (P) are contrasted in the tables below. whenever a challenge could not be satisfactorily resolved in the allocated 1800 seconds, with regard to the issue, computation is given up. The following information can be found in each of these tables:

EX: The Number of Example.: The ideal (optimal) value as equivalent to UB value. .*

Time: Time-based second.

The status $=\begin{cases} 1 & , \textit{If the example is finished in reasonable time} \\ 0, & \textit{Otherwise} \end{cases}$

Table 1: The outcomes for BAB, UB, LB, and time spent computing for n= 3.

EX	Optimal value	UB	LB	Time	The number of node	Status
1	82	82*	82	0.0003786	0	1
2	54	*54	54	0.0001694	0	1
3	38	38*	36	0.0194108	10	1
4	37	37*	37	0.0001873	0	1
5	80	*80	80	0.0001208	0	1
6	46	46*	46	0.0001115	0	1
7	40	40*	40	0.0000986	0	1
8	27	*27	27	0.0000914	0	1
9	56	56*	56	0.0002337	0	1
10	75	75*	75	0.0002318	0	1

ISSN-E: 18735347-02779536

Table 2: The outcomes for BAB, UB, LB, and time spent computing for n= 6.

EX	Optimal value	UB	LB	The number of Nodes	Time	Status
1	69	*69	69	0	0.0001798	1
2	200	*200	193	42	0.0064294	1
3	141	148	139	21	0.0016258	1
4	92	92*	89	52	0.0034494	1
5	175	175*	175	0	0.0001958	1
6	97	97*	97	0	0.0001127	1
7	89	89*	89	0	0.0001118	1
8	120	120*	119	21	0.000589	1
9	160	*160	160	0	0.0001208	1
10	147	147*	147	0	0.0001255	1

Table 3: The outcomes for BAB, UB, LB, and time spent computing for n= 9.

EX	Optimal value	UB	LB	The number of Nodes	Time	Status
1	218	*281	276	178	0.0053375	1
2	166	*166	164	280	0.007015	1
3	244	*244	243	127	0.0043979	1
4	290	292*	276	681	0.0227689	1
5	236	236*	236	0	0.0001868	1
6	289	289*	289	0	0.0001321	1
7	446	446*	446	0	0.000195	1
8	433	433*	433	0	00002066	1
9	238	238*	238	0	0.0002156	1
10	314	314*	312	126	0.0037542	1

Table 4: The outcomes for BAB, UB, LB, and time spent computing for n= 12.

EX	Optimal value	UB	LB	The number of nodes	Time	Status
1	352	356	339	4245	0.0867899	1
2	566	566	552	3641	0.1133465	1
3	588	597	569	11700	0.3022376	1
4	289	290	284	1140	0.0423561	1
5	444	444	440	364	0.0082772	1
6	396	396	383	6091	0.1227879	1
7	402	402	397	2988	0.0599026	1
8	592	602	587	1943	0.0412911	1
9	882	882	882	0	0.0001616	1
10	794	794	794	0	0.0001094	1

Table 5: The outcomes for BAB, UB, LB, and time spent computing for n= 15.

EX	Optimal value	UB	LB	The number of nodes	Time	Status
1	412	413	403	9456	0.187808	1
2	640	651	624	21276	0.4209762	1
3	725	743	704	11623246	235.6249929	1
4	671	671*	665	61308	1.2812634	1
5	918	918*	910	11054	0.230077	1
6	502	505*	487	388766	7.9401201	1
7	628	628*	624	8884	0.1879509	1
8	646	*646	644	1104	0.0241857	1
9	917	*917	917	0	0.0001635	1
10	391	391*	386	96326	1.954751	1

ISSN-E: 18735347-02779536

Table 6: The outcomes for BAB, UB, LB, and time spent computing for n= 18.

EX	Optimal value	UB	LB	The number of nodes	Time	Status
1	968	973	926	42660594	894.684982	1
2	682	688	671	570052	11.8124382	1
3	1048	1069	1020	15518178	321.3020063	1
4	1077	1082	1067	1391768	29.6907895	1
5	1179	1183	1177	41087	0.8468622	1
6	1118	*1118	1112	2130222	45.9811984	1
7	1281	*1281	1281	0	0.0001665	1
8	1458	1458	1450	506065	10.3118318	1
9	1320	1320	13314	12418	0.2532141	1
10	1362	1362	1362	0	0.000162	1

Table 7: The outcomes for BAB, UB, LB, and time spent computing for n= 21.

EX	Optimal value	UB	LB	The number of nodes	Time	Status
1	1463	1476	1442	72027698	1569.115138	1
2	1350	1364	1313	82937538	1800.000346	0
3	700	716	685	81130073	1800.000137	0
4	1175	1185	1157	18086187	389.7670526	1
5	1284	1296	1262	81393241	1800.000228	1
6	1944	1944	1941	221848	4.7557625	1
7	1474	1474	1470	415505	9.0406211	1
8	1644	1644	1641	279261	6.1500436	1
9	1816	1816	1809	112897	2.4086116	1
10	1815	1815	1814	1775644	40.9210929	1

5. Conclusion

The multi-objective sequencing problem $\sum C_j + \sum L_j + T_{max}$ is addressed in this study using the (BAB) method. The complexity of the sequencing problem rises as the number of targets increases. The (BAB) method is used for a substantial number of test tasks. The display of the calculated results demonstrates that the (UB) is helpful and occasionally offers the best solution. The (BAB) method gives the optimal solution of the problem for number of jobs n≤21 effectively. Future study will focus on the use of approximation local search algorithms to multiobjectivesequencing problems.

References:

1. Khamees. I. (2022). ″Solving multiobjective sequencing problem on one machine, ″ M.SC. Thesis. University of Diyala, College of Science, Dept. of Mathematics.
2. Al-Nuaimi, A. A. M. (2016). Optimal solution for simultaneous multicriteria problem. Diyala journal for pure sciences. Vol. 12, No. 2, pp. 18-27.
3. Ibrahim, M. S. (2019). Multi-objective shortest path model for optimal route between commercial cities on america. *Iraqi Journal of Science*, 1394-1403.
4. White, D. J. (1982). The set of efficient solutions for multiple objective shortest path problems. *Computers & Operations Research*, *9*(2), 101-107.
5. Warburton, A. (1987). Approximation of Pareto optima in multiple-objective, shortest-path problems. *Operations research*, *35*(1), 70-79.
6. Abo-Alsabeh R. & Salhi. A. (2021). ″A Metaheuristic Approach to the CIS Problem, ″ Iraqi Journal of Science, Vol. 62, No. 1, pp. 218- 227.
7. Van Wassenhove, L. N., & Gelders, L. F. (1980). Solving a bicriterion scheduling problem. *European Journal of Operational Research*, *4*(1), 42-48.
8. Franch, S. (1982) "Sequencing and Scheduling an Introduction to Mathematics of Job Shop," John Wiley & Sons, New York.
9. Fattah, K. (2014). "Solving multicriteria scheduling problems," M.SC. Thesis, University of Al-Mustansiriyah, College of Science, Dept. of Mathematics.
10. Abo-Alsabeh, R. R., & Salhi, A. (2022). The Genetic Algorithm: A study survey. *Iraqi Journal of Science*, *63*(3).
11. Kou, G., Yang, P., Peng, Y., Xiao, F., Chen, Y., & Alsaadi, F. E. (2020). Evaluation of feature selection methods for text classification with small datasets using multiple criteria decision-making methods. *Applied Soft Computing*, *86*, 105836.
12. Hesham, M., & Abbas, J. (2021). Multi-criteria decision making on the best drug for rheumatoid arthritis. *Iraqi Journal of Science*, 1659-1665.
13. Doumpos, M., Figueira, J. R., Greco, S., & Zopounidis, C. (2019). *New perspectives in multiple criteria decision making*. Springer International Publishing, Cham.
14. Mahmood, A. (2001) "Solution procedures for scheduling job families with setups and due dates," M.SC. Thesis, University of Al-Mustansiriyah, College of Science, Dept. of Mathematics.