International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 / https://doi.org/10.14445/22315381/IJETT-V70I8P205
Volume 70 Issue 8, 51-56, August 2022

Original Article

A Multi-Objective Approach to Modelling the Integrated Resource Selection and Operation Sequences Problem in a Production System

Miguel Fernández[1], Avid Roman-Gonzalez[2]

[1] *Department of Engineering, Pontifical Catholic University of Peru, Lima 32, Peru*
[1] *Image Processing Research Laboratory (INTI-Lab), Universidad de Ciencias y Humanidades, Lima, Peru*
[2] *Aerospace Sciences & Health Research Laboratory (INCAS-Lab), Universidad Nacional Tecnologica de Lima Sur, Lima, Peru*

[1]mfernandezp@pucp.edu.pe

Received: 13 May 2022 Revised: 15 June 2022 Accepted: 01 July 2022 Published: 09 August 2022

Abstract *- This paper addresses the integrated resource selection and operation sequences problem. This production scheduling problem is an extension of the flow shop, job shop and flexible job shop problems, its main characteristics being the precedence relationship between operations that are part of customer orders, the lot size of orders and the flexibility of the machines. A mixed-integer programming model is proposed to solve the problem, simultaneously optimising two objectives. This class of problems with more than one objective is known as multi-objective optimization, which consists of obtaining the non-dominated solutions that are part of the Pareto frontier. The problem's first objective is to minimize the makespan or the shortest time to complete all the orders. The second objective of the problem is to balance the workload of the machines, which aims to prevent specific machines from having a low workload and other machines from having an excessive workload. The computational results show that the mathematical model could satisfactorily solve the cases or instances.*

Keywords *- Integrated resource selection operation sequences problem, Mixed-integer programing model, Non-dominated solutions, Pareto frontier.*

1. Introduction

In production systems, customer orders are associated with operations that must be executed within a set of workstations. In this environment, aspects such as variable demand with tight deadlines, acquisition and stock of raw materials, stock of finished products, use of machinery and labour, good use of physical space, etc., can be considered. In this way, a production program must guarantee the shortest delivery time for orders (aimed at meeting customer expectations) and achieve the highest use of workstations (aimed at maximizing efficiency in the factory).

This paper aims to study the integrated resource selection and operation sequences problem (iRS/OS). The iRS/OS problem extends the flow shop, job shop, and flexible job shop problems.

The flow shop problem within a production system consists of programming a set of orders using a set of different machines in series, and the machine's processing order must be determined. If K is the number of orders, the problem is finding a sequence of orders within the $K!$ possible sequences that optimize a performance measure, generally associated with a time factor [1-3].

The job shop problem consists of finding a process sequence of a set of operations (which are part of orders), using a set of different machines to minimize a particular performance measure, with the restriction that each order has its manufacturing plan [2, 4, 5].

The flexible job shop problem is a generalization of the job shop problem and the parallel machine problem. In this environment, it is considered that there is at least one machine capable of performing more than one different operation (flexible machine). This problem consists of assigning each order operation to a specific machine and obtaining a sequence of operations processing, seeking to optimize one or more performance measures [6-8].

The iRS/OS problem is an approximation of a real production system. Each order is associated with a set of operations in this system, which depend on a precedence relationship and its manufacturing plant. Production scheduling considers order lot size, transportation times, and resource flexibility [9].

The problems described in this section are combinatorial optimization problems and are classified as NP-hard [10, 11].

2. Literature review

Several works have been carried out to solve the iRS/OS problem. Moon et al. [12] proposed a genetic algorithm to minimize the sum of the workload of the machines and the total transport time. The chromosome coding only considers the information of the sequence of operations, while a heuristic based on the minimum processing time was used to allocate resources. Similar work, including outsourcing of operations, was developed by Lee et al. [13]. In this work, a genetic algorithm was applied to minimize the makespan, with the restriction of meeting the delivery deadline for each order. Yan et al. [14] proposed a genetic algorithm with a new mutation operator based on disturbance and local search to minimize the makespan while ensuring compliance with the delivery deadline of orders. Shao et al. [15] presented a model that integrates the production planning and scheduling process to improve the performance of a manufacturing system and an approach based on a genetic algorithm to facilitate the said process. Kafashi [16] studied integrating CAD / CAPP / CAM by planning configurations and sequence of operations. A genetic algorithm achieves this objective, showing satisfactory results in flexible manufacturing systems. Al Aquel et al. [17] emphasize the need to improve overall efficiency in modern manufacturing systems; here, the integration of planning and scheduling processes plays an important role. The authors present an optimization algorithm based on priorities and dispatch rules in this work. Saidi and Zarghami [18] presented a mixed integer programming model for a process planning and programming problem that minimizes makespan. The proposed model obtained better results than a genetic algorithm from the literature. Barzanji et al. [19] developed a solution algorithm based on Benders decomposition. This algorithm divides the original problem into two optimization models, a Master problem and a Slave problem. The Master problem determines the process plan and machine allocation, while the Slave problem optimizes the scheduling and sequencing decision.

Regarding works considering more than one objective for the iRS/OS problem, the following can be cited: Zhang et al. [9] proposed a multi-stage genetic algorithm. Here one defines a chromosome represented by two vectors that contain both the information of the sequence of operations and the allocation of machines, considering the minimization of the makespan and the balancing of the workload. Yang and Tang [20] developed a multi-objective genetic algorithm with an adaptive strategy based on adjusting the crossover and mutation rates. Simultaneous minimization of machine downtime and late/early penalty was considered. Dayou et al. [21] presented a hybrid multi-objective genetic algorithm. These authors considered the simultaneous optimization of three objectives: the makespan, the balancing of the workload and the total transport time between machines. The genetic algorithm uses a strategy for classifying non-dominated solutions by agglomerating them with a local search heuristic.

3. Integrated resource selection and operation sequences problem

3.1. Problem Description

The iRS/OS problem can be defined as: given a set of K orders of lot size q_k $(k = 1,\dots, K)$, and a set of N machines, where each order k is composed of a series of J_k operations. Consider that each operation o_{ki} $(k = 1,\dots, K,\ i = 1,\dots, J_k)$ must be processed by some machine M_l $(l = 1,\dots, L)$ within the set of machines M_{ki} capable of processing o_{ki}. Furthermore, the batch size q_k is divided into equal sub-batches, called the unit load u_k, to be processed on the machines. It is desired to find the sequence of operations of the orders and the allocation of the operations in the machines so that the relation of precedence between operations is satisfied, the availability of the machines. It is optimal for minimising the makespan and balancing workload [9].

The assumptions considered in the iRS/OS problem are:

- All orders arrive simultaneously.
- All machines are available from the start.
- The machines cannot stop until they finish processing the batch size.
- The transport time between machines is significant.
- For the same order, processing in one operation begins after the unit load of the predecessor operation has been completed and shipped.
- The setup time is included in the processing time.

3.2. Problem Formulation

The notation used for the indexes, parameters, and decision variables to formulate the mathematical model of the iRS/OS problem is detailed below:

3.2.1. Indexes

k: Order index, $k = 1, 2,\dots, K$

i, j: Operation index, $i, j = 1, 2,\dots, J$

m, n: Machine index, $m, n = 1, 2,\dots, N$

3.2.2. Parameters

J: Number of operations

N: Number of machines

r_{ij}: Precedence relationship between operations

$$r_{ij} = \begin{cases} 1, \text{ if operation } i \text{ precedes operation } j \\ 0, \text{ otherwise} \end{cases}$$

A_{im}: Operation-machine correspondence matrix

$$A_{im} = \begin{cases} 1, \text{ if operation } i \text{ can be processed by machine } m \\ 0, \text{ otherwise} \end{cases}$$

k_i: Returns the order index of operation i

q_i: Returns the order lot size for operation i

p_{im}: Processing time of operation i in machine m

u_{ij}: Unit load size from operation i to operation j
t_{mn}: Transport time between machine m and machine n
M: Huge number

3.2.3. Variables

$y_{ij} = \begin{cases} 1, \text{ if operation } i \text{ is performed immediately before operation } j \\ 0, \text{ otherwise} \end{cases}$

$x_{im} = \begin{cases} 1, \text{ if machine } m \text{ is selected for operation } i \\ 0, \text{ otherwise} \end{cases}$

s_i: Start time of operation i
c_i: Completion time of operation i
c_M: Makespan

w_m: Machine workload m
u_m, v_m: Machine workload deviation m
$\bar{w}$: Average workload of machines
w_D: Average deviation of machines workload

$$w_D = \frac{1}{N}\sum_{m=1}^{N} |w_m - \bar{w}|$$

The mathematical model of the iRS/OS problem is formulated as:

$$\text{Min } c_M \quad (1)$$
$$\text{Min } w_D \quad (2)$$

Subject to:

$$\sum_{m=1}^{N} x_{im} = 1 \quad \forall i \quad (3)$$
$$s_i + q_i \sum_{m=1}^{N} p_{im} x_{im} = c_i \quad \forall i \quad (4)$$
$$y_{ij} = 1 \quad \forall i, j |\, r_{ij} = 1 \quad (5)$$
$$y_{ij} + y_{ji} = 1 \quad \forall i, j \quad (6)$$
$$s_j \geq s_i + u_{ij} p_{im} + t_{mn} - M(3 - x_{im} - x_{jn} - y_{ij}) \quad \forall i, j, m, n | k_i = k_j, m \neq n \quad (7)$$
$$c_j \geq c_i + t_{mn} + u_{ij} p_{jn} - M(3 - x_{im} - x_{jn} - y_{ij}) \quad \forall i, j, m, n |\, k_i = k_j, m \neq n \quad (8)$$
$$s_j \geq c_i - M(3 - x_{im} - x_{jm} - y_{ij}) \quad \forall i, j, m | A_{im} = 1, A_{jm} = 1 \quad (9)$$
$$c_M \geq c_i \;\; \forall i \quad (10)$$
$$w_m = \sum_{i=1}^{J} q_i p_{im} x_{im} \quad \forall m \quad (11)$$
$$\bar{w} = \frac{1}{N}\sum_{m=1}^{N} w_m \quad (12)$$
$$u_m - v_m = w_m - \bar{w} \;\; \forall m \quad (13)$$
$$w_D = \frac{1}{N}\sum_{m=1}^{N} (u_m + v_m) \quad (14)$$
$$x_{im} \in \{0,1\} \quad \forall i, m | A_{im} = 1 \quad (15)$$
$$y_{ij} \in \{0,1\} \quad \forall i, j | i \neq j \quad (16)$$
$$s_i, c_i \geq 0 \quad \forall i \quad (17)$$
$$v_m, u_m, w_m \geq 0 \quad \forall m \quad (18)$$
$$\bar{w}, c_M, w_D \geq 0 \quad (19)$$

This mathematical model considers the simultaneous optimization of two objectives. The objective function (1) represents the depreciation of the makespan, and the objective function (2) represents the minimization of the average deviation of the workload of all machines. Restriction (3) ensures that each operation selects only one machine for processing. Constraint (4) determines the completion time of each operation. Constraints (5) and (6) guarantee the viability of the sequence of operations. Constraints (7) and (8) guarantee the transport of an order between two different machines. Restriction (9) prevents interference between two operations assigned to the same machine. The constraint (10) determines the makespan as the maximum completion time of all operations. The restriction (11) determines the workload of each machine. Constraint (12) determines the average workload of all machines. The restriction (13) determines the deviation from the average workload of each machine. The constraint (14) determines the machines' average deviation of the workload. Constraints (15) - (19) define the domain of the decision variables.

4. Multi-objective optimization

The general model of a multi-objective optimization problem is defined as:

$$Min\ f_1(x), f_2(x), \dots, f_r(x)$$
$$s.t \quad x \in X$$

Given a set of variables of dimension n, $x = \{x_1, \dots, x_n\}$, within the domain X, the problem consists of finding a solution $x^* \in X$ that simultaneously minimizes r objective functions $f(x^*) = \{f_1(x^*), \dots, f_r(x^*)\}$.

Next, the concept of dominance and Pareto-optimal solutions is defined [22]:

4.1. Dominance

A solution x dominates another solution y, if and only if, $f_i(x) \leq f_i(y)$ for $i = 1, \dots, r$ and $f_j(x) < f_j(y)$ in at least one objective function j.

4.2. Pareto-optimal solution

A solution is called Pareto-optimal if it is not dominated by any other solution within domain X. A Pareto-optimal solution x^* cannot be improved concerning any objective without harming at least one other objective, that is:

$$\nexists\ y \in X \text{ such that } f_i(y) \leq f_i(x^*) \quad \text{for all } i = 1, \dots, r$$
$$\text{and} \quad f_j(y) < f_j(x^*) \quad \text{for some } j = 1, \dots, r$$

Pareto-optimal or non-dominated solutions in domain X are called the Pareto-optimal set, and the image of a Pareto-optimal set is called the Pareto frontier.

5. Solution method

In this paper, a multi-start method is applied, which evaluates an adaptation function with several search directions to find the Pareto-optimal set of the problem. The solution method is described below.

To solve a multi-objective optimization problem, it is usual to define an adaptation function *f(x)* given by the weighted sum of all objective functions of the problem [23-25]:

$$Min\ f(x) = \alpha_1 f_1(x) + \alpha_2 f_2(x) + \dots + \alpha_r f_r(x)$$
$$s.a \quad x \in X$$

where $\alpha_1, \dots, \alpha_r$ are the weights associated with each objective function of the problem, which must satisfy the following relationships:

$$\alpha_i \geq 0 \quad \text{for all } i = 1, \dots, r$$
$$\alpha_1 + \alpha_2 + \dots + \alpha_r = 1$$

Note that the search direction is fixed and unique if constant values are used to solve the problem. This last approach is inappropriate for finding all the non-dominated solutions to the problem. A journal article in [24] suggests generating random weights with several search directions. The weights are determined as:

$$\alpha_i = \frac{random_i}{random_1 + \dots + random_r} \quad \text{for } i = 1, \dots, r$$

where $random_1, \dots, random_r$ are random numbers between 0 and 1.

In this way, by generating random weights and solving the problem several times (replicas), it is possible to obtain the candidate solutions to the Pareto-optimal set of the problem.

The following section presents the application of the solution method to a set of cases or instances proposed in the literature.

6. Computer experiments

The AMPL software was used to execute the solution method described in the previous section. The cases or instances of the problem were taken from [12]. Three instances have been considered to evaluate the mathematical model. All these instances have a unit load equal to 10 and a set of 5 machines.

The results obtained by:

a) The makespan minimization (c_M).
b) Minimizing the average deviation of the workload on the machines (w_D).
c) Minimizing the weighted sum of the objectives mentioned in (a) and (b). To resolve the multi-objective problem, 100 replications were considered to search for non-dominated solutions.

6.1. 3 × 12 instance

This instance considers three orders with different lot sizes. These three orders are made up of 12 operations. Table 1 presents the results obtained for this instance.

6.2. 5 × 21 instance

This instance considers five orders with different lot sizes. These five orders are made up of 21 operations. Table 2 presents the obtained results for this instance.

6.3. 7 × 30 Instance

This instance considers seven orders with different lot sizes. These seven orders are made up of 30 operations. Table 3 presents obtained results for this instance.

Table 1. Results obtained from the 3 × 12 Instance

c_M		w_D		$\alpha_1 c_M + \alpha_2 w_D$		
c_M	w_D	c_M	w_D	c_M	w_D	**Frequency**
1046	164.8	1298	104	1046	104	100%

Table 2. Results obtained from the 5 × 21 Instance

c_M		w_D		$\alpha_1 c_M + \alpha_2 w_D$		
c_M	w_D	c_M	w_D	c_M	w_D	**Frequency**
1490	48	2836	28	1490	44	87%
				1554	28	13%

Table 3. Results obtained from the 7 × 30 Instance

c_M		w_D		$\alpha_1 c_M + \alpha_2 w_D$		
c_M	w_D	c_M	w_D	c_M	w_D	**Frequency**
1780	30.4	4596	7.2	1780	15.2	95%
				1800	13.6	1%
				1860	9.6	1%
				1980	7.2	3%

It can be seen in tables 1, 2 and 3 that the mathematical model was able to solve the proposed instances satisfactorily. As the instances increase in size, many Pareto-optimal solutions are obtained. Furthermore, if the depreciation of the makespan is just considered, the software does not return the lowest value of w_D that can be associated with said makespan. Similarly, it happens with the minimization of the average deviation of the workload. This situation implies that the optimal value of an objective function can be associated with different values for the rest of the objective functions. The latter is since the problem treated is formulated with integer variables.

Concerning the frequency observed in tables 1, 2 and 3, it can be concluded that the solution method is inclined to look for the minimum value of the makespan. In these tables, the value of the makespan is high compared to the average deviation of the workload; this difference in ranges influences the search for non-dominated solutions. In the experiments carried out on the three instances, it was found that there is no need to normalize the objectives.

7. Conclusion

This paper addressed the integrated resource selection and operation sequences problem. This problem is formulated as a multi-objective optimization problem given its scope in production systems. To solve this problem, a mixed integer programming model and a solution method capable of reaching the Pareto frontier and the Pareto-optimal or non-dominated solutions that compose it was proposed.

The proposed mathematical model proved to be capable of satisfactorily solving the three instances treated in the computational experiments in a short time. In Instance 3 × 12, the non-dominated solution of the problem was found, demonstrating the existence of cases where the Pareto frontier can be composed of a single solution in which the conflicting objectives converge simultaneously. In Instance 5 × 21, two non-dominated solutions were found. These two solutions, in relative terms, present a significant difference between their objective values ($|c_M| = 64$; $|w_D| = 16$). Here, the decision-maker can assign relative importance or priority to one objective over another and implement one of these solutions. In Instance 7 × 30, four non-dominated solutions were found; the two solutions found at the extremes of the Pareto frontier also show a significant difference ($|c_M| = 200$; $|w_D| = 8$). Here, unlike in the previous instance, having several solutions, the most appropriate is to opt for an intermediate solution based on the relative importance of each objective. Furthermore, in the instances studied, the relevance of the multi-objective approach over the mono-objective approaches is justified since, in the latter, the optimal solution is not necessarily part of the Pareto frontier.

The results showed the advantage of applying a multi-objective approach over a single-objective approach. A multi-objective optimization problem is suitable for decision-making when product managers face conflicting objectives. The problem studied seeks to balance two objectives: (1) guarantee the shortest delivery time for orders, which is aimed at meeting customer expectations; and (2) balancing the workload of the machines, which is oriented so that all the machines present a utilization rate that is close to each other.

References

[1] R. W. Buzzo, and J. V. Moccellin, "Production Scheduling in Flow Shop Systems Using a Hybrid Genetic Algorithm-Simulated Annealing Heuristic Method, Management & Production," vol. 7, no. 3, pp. 364-377, 2000.

[2] M. Pinedo, "*Scheduling Theory Algorithms and System,*" New York: Editorial Prentice Hall, 2008.

[3] A. K.Agarwal, and R. Garg, "Advanced Modelistic Approach of Flowshop Scheduling Problem for 10-Jobs, 10-Machines by Heuristics Models Using Makespan Criterion," *International Journal of Engineering Trends and Technology*, vol. 4, no. 5, pp. 1742-1748, 2013.

[4] G. U. Sankar, and D. Saravanan, "Single Objective Evolutionary Algorithm for Job Shop Scheduling," *International Journal of Mathematics Trends and Technology*, vol. 3, no. 1, pp. 38-42, 2012.

[5] M. Sophia, "Review on Scheduling-Job Scheduling," *International Journal of Engineering Trends and Technology*, vol. 60, no. 1, pp. 41-44, 2018.

[6] I. Kacem, S. Hammadi, and P. Borne, "Approach by Localization and Multiobjective Evolutionary Optimization for Flexible Job-Shop Scheduling," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 32, no. 1, pp. 1-13, 2002a.

[7] I. Kacem, S. Hammadi, and P. Borne, "Pareto-Optimality Approach for Flexible Job-Shop Scheduling Problems: Hybridization of Evolutionary Algorithms and Fuzzy Logic," *Mathematics and Computers in Simulation*, vol. 60, no. 3-5, pp. 245-276, 2002b.

[8] M. Nagamani, E. Chandrasekaran, and D. Saravanan, "Single Objective Evolutionary Algorithm for Flexible Job-Shop Scheduling Problem," *International Journal of Mathematics Trends and Technology*, vol. 3, no. 2, pp. 78-81, 2012.

[9] H. Zhang, M. Gen, and Y. Seo, "An Effective Coding Approach for Multiobjective Integrated Resource Selection and Operation Sequences Problem," *Journal of Intelligent Manufacturing*, vol. 17, no. 4, pp. 385-397, 2006.

[10] J. Du, andJ. Y.-T Leung, "Minimizing Total Tardiness on One Machine is NP-Hard," *Mathematics of Operations Research*, vol. 15, no. 3, pp. 483-495, 1990.

[11] M. R. Garey, D. S. Johnson, and R. Sethi, "The Complexity of Flowshop and Jobshop Scheduling," *Mathematics of Operations Research*, vol. 1, no. 2, pp. 117-129, 1976.

[12] C. Moon, M. Lee, Y. Seo, andY. H. Lee, "Integrated Machine Tool Selection and Operation Sequencing with Capacity and Precedence Constraints Using Genetic Algorithm," *Computers & Industrial Engineering*, vol. 43, no. 3, pp. 605-621, 2002.

[13] Y. H. Lee, C. S. Yeong, and C. Moon, "Advanced Planning and Scheduling with Outsourcing in Manufacturing Supply Chain," *Computers & Industrial Engineering*, vol. 43, no. 1-2, pp. 351-374, 2002.

[14] P. Yan, D. Liu, D. Yuan, and J. Yu, "Genetic Algorithm with Local Search for Advanced Planning and Scheduling," *Third International Conference on Natural Computation*, vol. 3, pp. 781-785, 2007.

[15] X. Shao, X. Li, L. Gao, and Ch. Zhang, "Integration of Process Planning and Scheduling – A Modified Genetic Algorithm-Based Approach," *Computers & Operations Research*, vol. 36, pp. 2082-2096, 2009.

[16] S. Kafashi, "Integrated Setup Planning and Operation Sequencing (ISOS) Using Genetic Algorithm," *The International Journal of Advanced Manufacturing Technology*, vol. 56, no. 5-8, pp. 589-600, 2011.

[17] G. Al Aqel, M. F. Ausaf, X. Li, and L. Gao, "A New Priority-Sort Based Optimization Algorithm for Integrated Process Planning and Scheduling," *International Journal of Modeling and Optimization*, vol. 3, no. 2, pp. 226-231, 2013.

[18] M. Saidi, and S. Zarghami, "Exact Mixed Integer Programming for Integrated Scheduling and Process Planning in Flexible Environment," *Journal of Optimization in Industrial Engineering*, vol. 7, no. 15, pp. 47-53, 2014.

[19] R. Barzanji, B. Naderi, and M. A. Begen, "Decomposition Algorithms for the Integrated Process Planning and Scheduling Problem," *Omega*, vol. 93, pp. 102025, 2020.

[20] J. Yang, andW. Tang, "Preference-Based Adaptive Genetic Algorithm for Multiobjective Advanced Planning and Scheduling Problem," *IEEE International Conference on Industrial Engineering and Engineering Management*, pp. 1935-1940, 2009.

[21] L. Dayou, Y. Pu, and Y. Ji, "Development of a Multiobjective GA for Advanced Planning and Scheduling Problem," *The International Journal of Advanced Manufacturing Technology*, vol. 42, no. 9-10, pp. 974-992, 2009.

[22] A. Konak, D. Coit, andA. Smith, "Multi-Objective Optimization Using Genetic Algorithms: A Tutorial," *Reliability Engineering & System Safety*, vol. 91, no. 9, pp. 992-1007, 2006.

[23] H. Ishibuchi, and T. Murata, "A Multi-Objective Genetic Local Search Algorithm and its Application to Flowshop Scheduling," *IEEE Transactions on Systems, Man, and Cybernetics, Part C* (Applications and Reviews), vol. 28, no. 3, pp. 392-403, 1998.

[24] T. Murata, H. Ishibuchi, and H. Tanaka, "Multi-Objective Genetic Algorithm and its Applications to Flowshop Scheduling," *Computers & Industrial Engineering*, vol. 30, no. 4, pp. 957-968, 1996.

[25] S. G. Ponnambalam, H. Jagannathan, M. Kataria, and A. Gadicherla, "A TSP-GA Multi-Objective Algorithm for Flow-Shop Scheduling," *The International Journal of Advanced Manufacturing Technology*, vol. 23, no. 11-12, pp. 909-915, 2004.