PAPER • **OPEN ACCESS**

Problems of production lines in parallel machines with lapses: Mathematical modelling

To cite this article: R Prada *et al* 2019 *J. Phys.: Conf. Ser.* **1160** 012017

IOP Conf. Series: Journal of Physics: Conf. Series 1160 (2019) 012017 doi:10.1088/1742-6596/1160/1/012017

Problems of production lines in parallel machines with lapses: Mathematical modelling

R Prada[1], **R Ruiz**[2] **and C Hernández**[1]
[1] Universidad Francisco de Paula Santander, San José de Cúcuta, Colombia
[2] Polytechnic University of Valencia, Valencia, Spain

E-mail: raulprada@ufps.edu.co, cesaraugusto@ufps.edu.co

Abstract. Production programming consists of assigning, sequencing and timing tasks leading to the acquisition of physical assets in productive resources. Normally, we talk about "jobs" to be sequenced in "machines" so that the result of the production programming is shown in the form of a Gantt chart where each job is divided into as many tasks as necessary and each task is executed in the machines of the factory as technologically necessary. In the productive reality several machines are observed in each stage of the production process, which suggests a problem of hybrid flow workshop type (HFS). However, in some companies a machine of each stage is joined by conveyor belts. All these machines would behave like a single virtual machine since the works enter in order in the first machine and that order is maintained until the last machine, without allowing neither stops nor changes in the sequence. Then in a modelling approach, it is suggested to reduce the problem of (HFS) to a problem with several production lines in parallel where each line would behave would be basically like a virtual machine. The problem is that since the line is so long, once the last element of a job leaves the first stage, it can start the processing of a new job while the first sequenced work moves through the production line. Effectively, the "parallel machine" modelling the line would be processing more than one job at the same time, that is, overlaps between the processed jobs are evident. The model is evaluated with the help of two of the commercial solvers and to be able to evaluate it, it was necessary to create a set of instances that would represent the situation under study.

1. Introduction

The increasing involvement of production companies in international markets is becoming increasingly demanding because of competitiveness to offer of a wide range of products, with short life cycles and low profit margins; This situation requires companies to have great flexibility in the production process in order to comply with increasingly smaller orders, with personalized characteristics a required delivery date.

The characteristics mentioned before demand from companies the efficient and rational use of all their available resources in order to guarantee customer satisfaction, the profitability of their products and business consolidation.

Regardless of what production system used, as mentioned in [1-3], every company production must plan to consider the time horizon, and from this activity the Master Production Plan is generated. Long-term production policy such as product lines to be elaborated or production capacity. In the medium term, aspects such as the need for raw materials and labor, among others, are considered; already in the short term, the control of the production takes place, which is based on the allocation phases (the necessary resources are determined over time to guarantee the realization of the various products), sequencing (the order of execution of the different manufacturing orders identifying the start and end moments of each of them) and programming (it is the unification of the two previous phases), and all together form the production program [4].

As evidence as mentioned previously, it is a complex process to generate a good production order that optimizes the company's available resources (which always will be scarce). This production order would demand adequate tools that contribute to the optimization of the process that is why all the investigations that can be carried out in this field provide greater knowledge on the subject.

2. Parallel machines

The present investigation focused on the problems of parallel machines where it is necessary to carry out the assignment without interruption of n independent jobs $\mathrm{N} = \{\mathrm{n}_1, \dots, \mathrm{n}_\mathrm{n}\}$ with positive processing times $\mathrm{p}_1, \mathrm{p}_2, \dots, \mathrm{p}_\mathrm{n}$ in m machines parallel $\mathrm{M} = \{\mathrm{m}_1, \dots, \mathrm{m}_\mathrm{m}\}$ to minimize the completion time of the last work completed in the workshop $\mathrm{C}_{\mathrm{máx}} = \mathrm{máx}\{\mathrm{C}_1, \dots, \mathrm{C}_\mathrm{m}\}$ which is also known as makespan, where C_i is the sum of the processing times of the jobs assigned to the machine i. This problem can be abbreviated as $\mathrm{P}\ //\mathrm{C}_{\mathrm{máx}}$ using the triple field notation of [5].

For the particular case of the unrelated parallel machines, the following assumptions are added: a) each job j, $\mathrm{j} = 1, \dots, \mathrm{n}$ must be processed in exactly one of the m machines in parallel; b) no machine can process more than one job at a time and once a job has been processed in the machine, it cannot be interrupted until it is finished; c) the processing time of a job is known, certain , finite and a fixed positive number that depends on the speed of the machine in which it is made. With the above, the model can be denoted as $\mathrm{R}\ //\mathrm{C}_{\mathrm{máx}}$.

To solve combinatorial problems, the traditional method is the definition of a mathematical model that guarantees obtaining the optimal solution. This solution scheme is only possible with problems of relatively small size, because when the problems are complex or large; this solution paradigm is not recommended since it demands considerably large amounts of time and memory. Using mixed-integer linear programming (MILP), the generic problem of non-related parallel machines is defined mathematically. The binary variable x_{ij}adds the value of 1 if job j is assigned to machine i and zero in any other case. The minimization of the makespan $\min \mathrm{C}_{\mathrm{máx}}$is defined as an objective function and is subject to the following restrictions,

$$\sum_{i=1}^{m} x_{ij} = 1\,, \forall j \in N \tag{1}$$

$$\sum_{j=1}^{n} p_{ij} \,.\, x_{ij} \leq C_{\mathrm{máx}}, \forall i \in M \tag{2}$$

$$x_{ij} \in \{0,1\}, \forall j \in N, \forall i \in M \tag{3}$$

$$C_{\mathrm{máx}} \geq 0 \tag{4}$$

The first set of constraints Equation (1) is processed on only a single machine. The second set of constraints Equation (2) ensures that the maximum sum of all jobs assigned to each machine is the value of $\mathrm{C}_{\mathrm{máx}}$. The restriction Equation (3) is used to define $\mathrm{C}_{\mathrm{máx}}$ as a binary variable and the restriction Equation (4) defines the $\mathrm{C}_{\mathrm{máx}}$ as a non-negative variable that can take any real value greater than or equal to zero.

The contributions of [6] allow us to know a little more about the operation offered by this MILP model since it is not the only model that has been addressed in the literature. In the same year, the work carried out by [7] was highlighted, arguing that the problem of programming with MIP models can be described based on the decision variables on which it is based. They considered the following types of decision variables: a) Variables of the completion time of a job, which is a key metric to measure the quality of a production schedule and which are sometimes called "variables of natural date"; b)

Assignment and date position variables, specify what work is planned below and at what time the processing of this work will begin; c) Variables of linear ordering, through them the relations of precedence between the works are described and they are denominated "variables of the sequence"; d) Indexed time variables, usually assign jobs to time periods; e) Network variables, or variables of the traveling salesperson.

They proposed four different paradigms of the MILP model for the problem of optimization in the programming of parallel machines. Specifically, the decision variables considered were indexed time variables, network variables, and date assignment and positional variables. The model that used time-indexed variables (called M1) took less time, was able to provide optimal solutions more frequently, and if it did not find the optimal solution, it provided lower limits closer to the optimum.

In [8] a mathematical model of mixed integer programming (MIP) is developed for the problem of parallel machines no related to configuration times dependent on the sequence denoted as $R\ /s_{ijk}/C_{máx}$. By adding the configuration times to the generic problem, this is no longer just an assignment problem, since depending on how different jobs are sequenced among those assigned to a machine, the end time of that machine will be different, so the model to solve this problem should consider allocation and sequencing operations.

2.1. Proposed mathematical model

In the MILP formulation for the case of a company in the ceramic sector that presents several machines in each of the stages of production, but with the particularity that in each stage there is a machine linked to another of the next stage by the band system conveyor; therefore, despite being a typical hybrid flow workshop situation [9], it can be approached as a case of parallel virtual machines, considering in the model the configuration times depending on the machine and the sequence, without leisure time in the machines and the overlaps between jobs since has already said, considering each production line as a single virtual machine with a single process it is unproductive to wait for the last ceramic tile to come out of a job to start the next sequenced work. The proposed model takes as background the mathematical models generated in the works of doctoral theses of [10-12].

Let $N = \{1, \dots, n\}$ be the set of all the jobs to be performed in $M = \{1, \dots, m\}$ parallel virtual machines not related to processing time in each virtual machine i for each job j of $P_{ij}$and time of configuration between consecutive jobs on the machine denoted as S_{ijk}. It is considered a parameter q_{ij} that contains the maximum level from which a job j can overlap in the virtual machine i with its successor k. This time depends on the speed at which the press works (bottleneck in the production process) since in the company the fastest presses have been connected to the slower ovens. Based on information provided by the production manager, it is concluded that the levels generated from the 30%, 25%, 20% and 15% of the processing time of the j and k jobs in the virtual machines 1, 2 are generated after the elapsed time 3 and 4, respectively.

Additionally, the positive whole constant V is considered. The model aims to minimize the makespan and involves the following decision variables:

- X_{ijk} It is worth 1 if job j precedes work k in virtual machine i and 0 otherwise.
- C_{ij} Job completion time j in the virtual machine i.
- lag_{ijk} It is the overlap between the consecutive jobs j and k in the virtual machine i, being $j \neq k$.
- $C_{máx}$ Maximum completion time.

It is subject to the following restrictions:

$$\sum_{\substack{j=0 \\ j\neq k}}^{n} \sum_{i=1}^{m} X_{ijk} = 1, k = \{1, \dots, n\} \tag{5}$$

$$\sum_{\substack{k=1\\j\neq k}}^{n}\sum_{i=1}^{m} X_{ijk} \leq 1, j = \{1,..,n\} \tag{6}$$

$$\sum_{\substack{h=0\\h\neq j,h\neq k}}^{n} X_{ihj} \geq X_{ijk}, j,k = \{1,..,n\}, j \neq k, i = \{1,..,m\} \tag{7}$$

$$\sum_{i=1}^{m}\left(X_{ijk} + X_{ikj}\right) \leq 1, j = \{1,..,n\}, k = \{j+1,..,n\}, j \neq k \tag{8}$$

$$\sum_{k=1}^{n} X_{i0k} \leq 1, i = \{1,..,m\} \tag{9}$$

$$C_{i0} = 0, i = \{1,..,m\} \tag{10}$$

$$C_{ik} + V\left(1 - X_{ijk}\right) = C_{ij} + P_{ik} + S_{ijk} - lag_{ijk}, j = \{0,..,n\}, k = \{1,\ldots,n\}, j \neq k, i = \{1,..,m\} \tag{11}$$

$$lag_{ijk} \leq q_{ij}, i = \{1,\ldots,m\}, j = \{1,\ldots,n\}, k = \{1,\ldots,n-1\} \tag{12}$$

$$lag_{ijk} \leq q_{ik}, i = \{1,\ldots,m\}, j = \{1,\ldots,n\}, k = \{1,\ldots,n-1\} \tag{13}$$

$$C_{max} \geq C_{ik}, k = \{1,..,n\}, i = \{1,\ldots,m\} \tag{14}$$

$$X_{ijk} \in \{0,1\}, j,k = \{1,\ldots,n\}, i = \{1,\ldots,m\} \tag{15}$$

The set of restrictions Equation (5) ensures that each job is processed in a single virtual machine. The set of restrictions Equation (6) ensures that each job has at most one successor. The set of restrictions Equation (7) ensures that if a job is processed in a virtual machine, that job has a predecessor. The set of restrictions Equation (8) ensures that between two consecutive jobs only one condition of being predecessor or successor of one another is fulfilled. The set of restrictions Equation (9) limits to a single successor of the dummy jobs in each virtual machine. The set of constraints Equation (10) defines the completion time at zero for the dummy jobs in each virtual machine. The set of constraints Equation (11) controls the completion time in each virtual machine, in case the job k is the successor of job j on line i ($X_{ijk} = 1$) then the completion time C_{ik} must be equal to C_{ij}, plus the processing time of work k, plus the configuration time between jobs j and k, minus the level of overlap between jobs j and k within the virtual machine. In case that ($X_{ijk} = 0$ then the constant V represents the redundant constraint. The set of constraints Equation (12) and Equation (13) establishes the minimum permissible overlap level between two consecutively sequenced jobs. For example, if $q_{ij} = 6$ and $q_{ik} = 9$ the overlap between jobs would be $lag_{ijk} = 6$, which indicates that the lowest bound will always be accepted and the other restriction would be redundant. The set of constraints Equation (14) defines the makespan and Equation (15) defines the binary variables.

3. Computational evaluation

The characteristic elements of the model considered are described in detail in Table 1. As you can see, the smallest instance has 7 jobs to be done in 2 machines considering a preparation time equivalent to $10\% * U[1.99]$; and the largest, 15 jobs to be carried out on 4 machines considering a configuration time equivalent to $125\% * U[1.99]$. It is important to emphasize that the values of N and M were considered equidistant, thus facilitating the statistical analyzes to be carried out. In addition, as can be seen in later sections, the size of the instances is adequate for the exact models that are proposed.

For the evaluation of the proposed mathematical model, tests were performed on a computer with an Intel Core i5 processor at 2.39 GHz and 8 GB of RAM. For each instance under evaluation, a maximum execution time of 60 minutes has been allowed.

The main intention is to identify if there are differences in the functioning of these two solvers with the proposed model using the created instances. For this, one of the most powerful methodologies is used, such as the Design of Experiments (DOE) [13]. The DOE is a structured and organized method to determine the relationship between the factors that possibly affect a process. For the case of study in this situation, we will study the effect on the response variable, execution time and the factors: Jobs, Virtual Machines, Solvers and Preparation Times.

Table 1. Description of the factors considered in the instances created defines the binary variables.

Feature	Description
Number of jobs (n)	$\{7, 8, 9, 10, 11, 12, 13, 14, 15\}$
Number of virtual machines (m)	$\{2, 3, 4\}$
Processing times of each job (p_{ij})	U [1,99]
Preparation times between jobs (s_{ijk})	$\{10\% * U[1,99], 50\% * U[1,99],$ $100\% * U[1,99], 125\% * U[1,99]\}$
Overlaps (lag_{ijk})	$\{70\%p_{ij}, 75\%p_{ij}, 80\%p_{ij}, 85\%p_{ij}\}$
Solvers	$CPLEX$ 12.6.2.0 $-$ $Gurobi$ 6.0.4
Number of replicates by instance type	5
Total instances	540

4. Results

To determine which of the two solvers works best in this situation studied, the number of resolved instances was optimally analyzed for each of them in the execution time considered (maximum 60 minutes). In Table 2 the percentage of unresolved instances is very similar for both solvers, then one could believe that there are no significant differences in the functioning of the two.

Table 2. Instances resolved optimally according to the Solver.

	Instances Solved	Instances No Solved
Cplex	35.64%	14.37%
Gurobi	36.20%	13.79%
Total	71.84%	28.16%

In order to determine the representative characteristics in the model, an analysis of variance of the aforementioned factors was performed for the time (in seconds) required to find the solution. Only resolved instances were included in the analysis(774 optimal solutions). From the ANOVA (see Table 3) it can be concluded that the solver used does not have a statistically significant effect on the execution time, that is, it is indifferent to use Cplex or Gurobi in our study situation.

Table 3. Analysis of Variance for Time considering all the factors.

Source	Sum of squares	gl	Medium square	Reason F	Value P
Main Effects					
A: n	9.88875E7	8	1.23609E7	50.90	0.0000
B: m	1.73539E7	2	8.67696E6	35.73	0.0000
C: Setup	8.64734E6	3	2.88245E6	11.87	0.0000
D: Solver	342.629	1	342.629	0.00	0.9700
Residue	1.84577E8	760	242864.0		
Total (After correcting)	2.89954E8	744			

Also, from Table 3 it is concluded that there is a statistically significant effect on time with the factors: number of jobs (n), number of machines (m) and preparation time (Setup) with a confidence level of 95%. In order to graphically visualize the effects on each factor that was significant, the Tukey HSD Intervals are presented below.

As shown in Figure 1 the amount of work to be done could be grouped into four clusters with significant differences between them, which would be: a) Cluster #1: for 7, 8 and 9 jobs; b) Cluster #2: for 10 job; c) Cluster #3: for 11 and 12 jobs; d) Cluster #4: for 13, 14 and 15 jobs. Likewise, in Figure 2 it can be observed that there are no significant differences in time when there are 2 or 3 machines, but with the addition of one more machine to the production process, the times are significantly reduced.

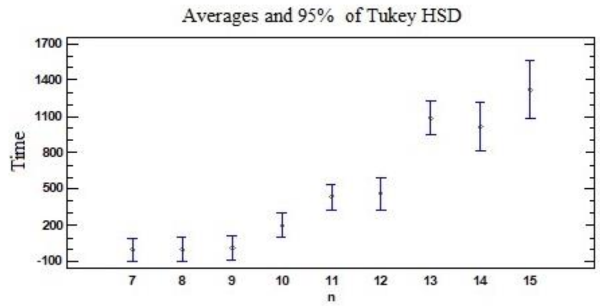

Figure 1. Tukey Intervals at 95% for the Number of jobs (n).

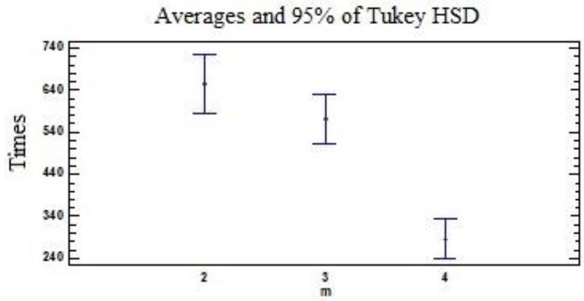

Figure 2. Tukey Intervals at 95% for the Number of machines (m).

Figure 3 shows the possible existence of an inverse relationship between the increase in the percentage of the (setup time) of the machines between jobs and the search time of the optimal solution.

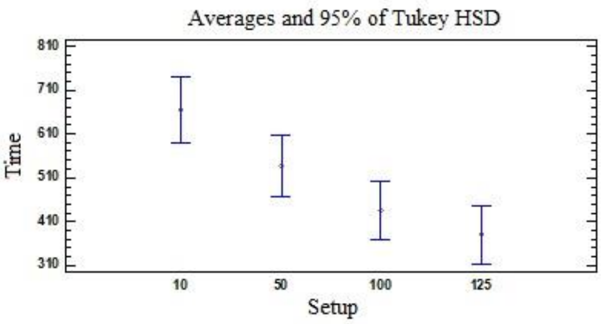

Figure 3. Tukey Intervals at 95% for Setup Time.

5. Conclusions

It began by addressing a problem already studied, the problem of the hybrid flow workshop that is the production environment closest to the ceramic tile production sector. Considering the production environment of a particular ceramics company, it was observed that the company, using conveyor belts, connected all the machines of the production process from the pressing phase to the packaging phase. Additionally, the company does not present material storage buffers in progress and the line must always be producing.

With the mentioned characteristics, it was analyzed that in spite of having a hybrid flow workshop problem, the fact that a machine of each phase of the productive process were connected to each other, behaved with a single virtual machine, then in the end it was concluded that the problem would be transformed into a problem of unrelated parallel virtual machines, which demand optimal allocation and sequencing processes, guaranteeing the incorporation of some operating characteristics such as: the preparation and adjustment times of the machines depending on the machine and sequence; the absence of idle times between sequenced jobs within each virtual machine and in order to always ensure the continuity of the production process, overlaps between jobs were included.

With the, we started with a problem that was already known until we came to formulate a real productive situation that has never been studied before, according to a review of the investigative background. Part of the background information consulted was the starting point for the construction of an exact mathematical model that would optimally solve the problem posed. Finally, certain instances were created that served as a test bench for the proposed model, which led us to conclude that there were no significant differences between the solvers evaluated, but that the number of jobs, the number of machines and the time of preparation were significant with respect to the time it took the model to achieve the optimal solution.

References

[1] Framinan JM, Leisten R and García RR 2014 *Manufacturing scheduling systems: An Integrated view on models, methods and tools* (London: Springer-Verlag) Chapter 3 pp 51-63
[2] Johnson LA and Montgomery DC 1974 *Operations research in production planning, scheduling, and inventory control* (New York: Wiley)
[3] Pinedo M L 2009 *Planning and scheduling in manufacturing and services* second edition (New York: Springer-Verlag)
[4] Núñez RP 2016 *Problemas de líneas de producción como secuenciación en máquinas paralelas con solapes. Revisión del estado del arte y modelado matemático* (Valencia: Universidad Politécnica de Valencia)
[5] Graham RL, Lawler EL, Lenstra JK and Kan AR 1979 Optimization and approximation in deterministic sequencing and scheduling: a survey *Annals of Discrete Mathematics* **5** 287-326
[6] Fanjul-Peyro L and Ruiz R 2010 Iterated greedy local search methods for unrelated parallel machine scheduling *European Journal of Operational Research* **207** 55-69
[7] Unlu Y and Mason SJ 2010 Evaluation of mixed integer programming formulations for non-preemptive parallel machine scheduling problems *Computers & Industrial Engineering* **58** 785-800
[8] Vallada E and Ruiz R 2011 A genetic algorithm for the unrelated parallel machine scheduling problem with sequence dependent setup times *European Journal of Operational Research* **211** 612-22
[9] Jiménez AP 2012 *Solución del problema de programación de flow-shop flexible empleando el algoritmo genético de Chu-Beasley* (Pereira: Universidad Tecnológica de Pereira)
[10] Ruiz R 2003 *Metaheuristic techniques for flexible production scheduling* (Valencia: Universidad Politécnica de Valencia)
[11] Urlings T 2010 *Heuristics and metaheuristics for heavily constrained hybrid flowshop problems* (Valencia: Polytechnic University of Valencia)
[12] Vallada E 2007 *Sequencing in flow workshops with delivery dates. New metaheuristic methods and parallel computing* (Valencia: Polytechnic University of Valencia)
[13] Montgomery DC 2012 *Design and analysis of experiments* eighth edition (New Jersey: John Wiley & Sons)