Original research article

Lot streaming Permutation Flow shop with energy awareness

F. D'Amico [a], D. A. Rossit [a,b*], M. Frutos [a,c]

[a] *Departamento de Ingeniería, Universidad Nacional del Sur, Bahía Blanca, Argentina;*

[b] *INMABB UNS-CONICET, Bahía Blanca, Argentina;*

[c] *IIESS UNS-CONICET, Bahía Blanca, Argentina;*

ABSTRACT

In this work, the flow shop scheduling problem with energy awareness is approached with lot-streaming strategies. Energy consumption is modeled within the objective function, together with the makespan, by means of a normalized and weighted sum. Thus, reducing energy consumption guides the optimization process. For lot streaming approaches mathematical models are provided and assessed. The results showed that lot-streaming is an efficient strategy to address this problem, allowing to improve both makespan and total energy consumption compared to the problem without lot-streaming. In turn, the selection of processing speeds for each sublot was incorporated, which improved the strategy yielding the best quality solutions.

1. Introduction

In recent decades, global awareness of the need to change to more sustainable industrial systems has grown [1]. This would reduce the negative impacts of climate change by reducing carbon emissions [2] [3]. One of the most direct ways to reduce these emissions is to reduce energy consumption, since reducing energy consumption reduce the combustion of fossil fuels [4]. Within the production of manufactured products, the reduction of energy consumption can be approached from three different perspectives [5] [6]:

1. At the machinery level: the aim is to develop more efficient machines that require less energy.
2. At the product level: the aim is to design a product from the point of view of energy saving. This is not very feasible in medium and small companies as it requires a lot of investment.
3. At the factory level: you can choose decision models that support and contribute to energy savings.

This work is framed in this last perspective, seeking to develop decision support tools that contribute to energy saving, specifically considering scheduling problems in the production planning process [7]. In-

cluding energy considerations in solving scheduling problems proved to be very effective in reducing energy consumption, achieving reductions of up to 80% [8]. The work of Mouzon et al. [9], was a pioneered work in this field, where the authors developed a study of dispatch rules, and then a multi-objective model that allowed solving scheduling by considerably reducing energy consumption for CNC machines. A critical aspect when trying to reduce energy consumption, it is to minimize the maximum power consumed by the production process, since when the power is increased, the energy consumption increases more than proportionally. Therefore, seeking to minimize the maximum peak of energy consumption contributes to energy efficiency, and consequently to minimize carbon emissions [10]. This approach of seeking to minimize the peak of energy intake was first addressed by Fang et al. [11]. In this work, the authors developed multi-objective mathematical models that sought to minimize the peak of consumption and the makespan in a flow shop manufacturing configuration. These models considered the processing speed of each operation as a decision variable, thus energy consumption depends directly on scheduling solution. In Fang et al. [12] they deepened their study in flow shop configurations with energy consumption considerations, by developing strengthened mathematical formulations, as well as combinatorial approaches that allowed solving the problem optimally. In turn, in that work, they considered two variants of flow shop, one without intermediate storage between successive stages (if there is no intermediate storage, a machine cannot be released until the next machine is released, this is known as blocking flow shop), and another variant with unlimited intermediate storage. Bruzzone et al. [10] addressed the problem of minimizing the peak of energy consumption, but in flexible flow shop systems (more than one machine in parallel per stage). In this article, the authors proposed a resolution approach based on two stages, where the first stage solve scheduling seeking to minimize a traditional objective function such as makespan. Then, in a second stage, the proposed method reschedules the start of activities in order to reduce the peak of energy consumption. On the other hand, Mansouri et al. [13] used a multi-objective approach to solve a 2-stage flow shop system. The objectives that the authors considered were makespan and total energy consumption, and they solved the problem using the □-constraints method.

Given that many of the articles that address this problem do so by making the start and/or end of operations more flexible, as well as the execution of that operation, the study of lot streaming strategies is promising. That is why in this work it is proposed to solve the flow shop problem with energy considerations through lot-streaming strategies. Lot-streaming allows production batches to be divided into sublots, which can advance from one workstation to another without waiting for the entire batch to be processed [14]. In this way, the production of the same batch can be overlapped in more than one production stage, favoring the efficiency of the system by reducing production cycle times [15]. Lot streaming has proven to be a valuable approach to improving scheduling problem solutions. It has been shown that lot streaming allows to efficiently solve rescheduling in problems that are considered machine breakdown [16]. Even in flow shop problems with sequence-dependent setups, the advantages of using an approach based on lot streaming were also evidenced [17]. On the other hand, it allows to reduce makespan times considerably for non-permutation flow shop systems [18], as well as to deal efficiently with complex flow shop systems where one of the stages production works in batch mode [19]. In this work, it proposed to extend these lot streaming applications to solve scheduling in flow shop systems with energy considerations. The intention is to take advantage of the ability to make the processing times of each production lot more flexible by being able to subdivide it, and thus, make scheduling more flexible to minimize energy consumption without penalizing regular objective function, as makespan.

The rest of the work is organized as follows. In section 2 the problem to be studied is described and all the mathematical models are introduced. In section 3 the experiments to be carried out are described. In section 4 the results obtained are presented and discussed, while in section 5 the conclusions of the work are presented.

2. The problem and mathematical models

This section presents the problem in detail, presenting the incorporation of lot-streaming processes into the schedule calculation. Firstly, we introduce Figure 1, where an illustrative example of 3 jobs and 5 machines flow shop scheduling problem with makespan as optimization criterion is depicted. In the top part of Figure 1 is presented the regular solution, meanwhile in the bottom part is presented the solution with lot streaming implemented. It is easy to see the positive impact that lot streaming provides

regarding makespan criterion, since the lot streaming solution (bottom part of Figure 1) has a makespan lower than 1400 and the regular solution (top part of Figure 1) has a makespan of around 2200.

2.1 Problem description

The problem studied in this work is framed within the line of scheduling with energy considerations [9]. These problems seek to solve the classic scheduling problem [7]: assign the jobs or tasks to the available resources seeking to minimize some objective function. In this case we will consider flow shop settings, where the sequence of operations of a given job is the same for all jobs [20]. By incorporating energy considerations, it seeks to minimize the environmental impact generated by production operations through two alternatives, or to minimize total energy consumption [8] or the maximum peak of energy consumption [11]. This work seeks to minimize total energy consumption. This has reason to be in environments where the speed of processing tasks is variable, and the energy consumption is linked to that speed variability [13]. Therefore, the operations considered here are not constant tasks, thus, they can be incorporated as decision variables to the problem, which allows increasing the efficiency of the schedule obtained (there are greater degrees of freedom), but at the same time implies an increase in its complexity

[12] [5].

On the other hand, the innovation of including lot-streaming strategies to the problem is proposed. The lot-streaming consists of the subdivision of the production batches into sublots, so that they can advance in operation even when the complete lot has not finished its processing in the previous operation [21]. In this way, it is possible that two successive machines in the flow shop are processing the same job or lot (divided into sublots) simultaneously [14]. Incorporating lot-streaming into the problem allows increasing flexibility in the job assignment process, taking advantage of the improvements reported by lot-streaming when studying the makespan, leaving greater margin for minimizing total energy consumption.

2.2 Mathematical models

In this section the three mathematical models studied are presented. The first is the most general case, flow shop with energy considerations (FSS+EA), then, the first lot-streaming approach to this problem (SBS+EA) is incorporated. Finally, the lot-streaming is incorporated into the energy consideration by allowing the processing speed to be modified for each sublot (SBSi+EA).
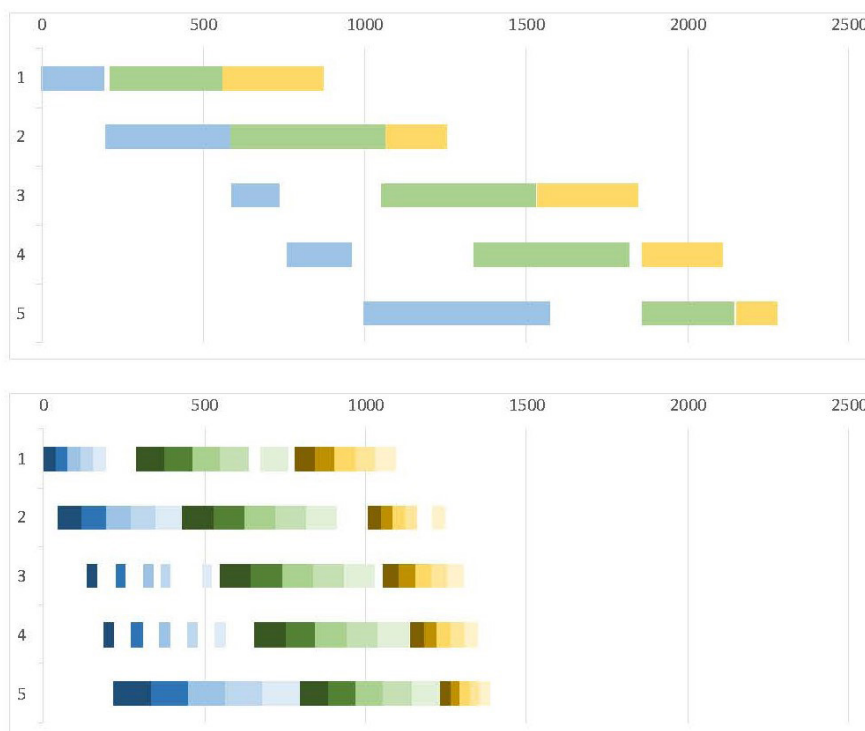
Figure 1. Illustrative example of 5 machines and 3 jobs flow shop scheduling problem with makespan optimization criterion, the top part represents regular flow shop solution and the bottom part represents a lot streaming flow shop solution

2.2.1 Flow Shop with Energy Awareness (FSS+EA)

The first model presented deals with the classic flow shop scheduling problem with energy awareness (FSS+EA). This model seeks to sequence the jobs complying with the production conditions (that is, all the operations that must be carried out in each job are respected), but also, defining the processing speed of each operation. Therefore, both the sequencing and the selection of processing speeds intervene in the calculation of the objective function that contemplates the makespan and energy consumption. These objectives are addressed simultaneously through a normalized weighted sum. Furthermore, as a realistic feature of the problem, preparation and removal times were considered for each job. The mathematical model is presented below.

Sets:

M: machines; $k = 1, 2, ..., m$.

J: Jobs or lots; $j = 1, 2, ..., n$.

V: processing speed; $l = 1, 2, 3$.

Parameters:

G: positive value (large enough).

m: number of machines.

n: number of jobs.

U_j: number of units of product j to be processed (lot size of product j).

$TS_{j,k}$: setup time of job j at machine k.

$P_{j,k}$: standard amount of workload of processing a unit of product j at machine k.

$RT_{j,k}$: time to release machine k after processing one unit or more of product j (independent of the number of units processed).

$Vl1_l$: speed factor for selected speed l, it can take the values 1.2, 1 y 0.8.

$Vl2_l$: energy conversion factor for processing speed l. It takes the values 1.5, 1, 0.6 for fast, normal and slow, respectively.

MO_k: energy conversion factor for machine idle time k.

PM_k: power of machine k.

Variables:

$C_{j,k}$: completion time of job j at machine k.

$X_{j,j'}$: binary variable, which is 1 if job j is processed before j', 0 otherwise ($j \neq j'$).

Mathematical Formulation:

$$\text{Minimize } Z = 0.5 * \left(\frac{c'_{max} - c_{max}^{min}}{c_{max}^{max} - c_{max}^{min}}\right) + 0.5 * \left(\frac{TEC' - TEC^{min}}{TEC^{max} - TEC^{min}}\right) \tag{1}$$

Breakdown of Z (equation 1):

$$C'_{max} \geq C_{j,k} + RT_{j,k}, \forall j, k = m \tag{2}$$

$$TO_k = C_{max} - \sum_j \sum_l \frac{P_{j,k} * U_j}{Vl1_l} * Y_{j,k,l}, \forall k \tag{3}$$

$$TEC' = \sum_k \sum_j \sum_l \frac{PM_k * P_{j,k} * U_j * Vl2_l}{60 * Vl1_l} * Y_{j,k,l} + \sum_k \frac{MO_k * PM_k}{60} * TO_k \tag{4}$$

Subject to:

$$C_{j,k} \geq \frac{P_{j,k} * U_j}{Vl1_l} * Y_{j,k,l} + TS_{j,k}, \forall j, k = 1 \tag{5}$$

$$C_{j,k} \geq C_{j,k-1} + \frac{P_{j,k} * U_j}{Vl1_l} * Y_{j,k,l} + RT_{j,k-1} + TS_{j,k}, k > 1, \forall j \tag{6}$$

$$C_{j',k} \geq C_{j,k} + \frac{P_{jj,k} * U_{j'}}{Vl1_l} * Y_{j',l,k} - G * (1 - X_{j,j'}) + RT_{j,k} + TS_{j',k}, \forall k, j' \neq j \tag{7}$$

$$X_{j,j'} + X_{j',j} = 1, j' \neq j \tag{8}$$

$$\sum_l Y_{j,k,l} = 1, \forall j, k \tag{9}$$

$Y_{j,k,l}$: binary variable, which is 1 if job j is processed at speed l at machine k, 0 otherwise.

$Cmax$: makespan.

TEC: total energetic consumption.

The objective function Z is presented in equation (1), where it is shown that it is made up of makespan (C_{max}) and energy consumption (TEC). Also, in equation (1), the normalization and weights of Z is shown for each of the terms. The objective function weights considered for this work are 0.5. The calculation of C_{max} is carried out according to equation (2). In equation (3) the idle time TOk is calculated. And in equation (4) the energy consumed in the processing is obtained, considering the speeds selected to process each job, as well as the consumption during idle time. Then, the constraints are presented in equations (5) - (9). Constraint (5) sequences the jobs for the first machine. In restriction (6) it is ensured that each job respects its precedence conditions between operations. Constraint (7) sequences the different jobs, if job j is processed before job j' for machine k, then that order is maintained for all machines. Constraint (8) ensures that the reverse ordering (j' before j) is not valid. Finally, restriction (9) ensures that a single speed is selected for each operation.

2.2.2 Lot Streaming Flow Shop with Energy Awareness (SBS+EA)

Based on the FSS+EA model, the lot-streaming strategy is incorporated into the scheduling process, giving rise to the SBS+EA model. In this model, each job j, made up of U_j units, is divided into f sublots of size $s_{i,j}$. When considering sublots, it is also necessary to consider that the transfer times increase their impact, since it does not occur once for the entire batch U_j but rather f times (one for each $s_{i,j}$). On the other hand, incorporating lot-streaming multiplies the number of times the machines must be loaded and unloaded (one for each sublot, instead of one for the entire lot), as well as the transfer or transfer times of the parts. Here are only the indices and parameters that are affected by this addition.

Sets:

I: sublots: $i = 1, 2, ..., f$.

Parameters:

f: number of sublots.

$s_{i,j}$: size of sublot i of job j (number of units contained in the sublot).

FT_j: fixed transfer time of job j (applies to each sublot without depending on the number of units contained in it).

$RT_{j,k}$: unload time associated with machine k after processing a sublot of job j in machine k.

Variables:

$C_{i,j,k}$: completion time of sublot i of job j at machine k.

Breakdown of Z (equation 1) in terms of SBS+EA model:

$$C'_{max} \geq C_{i,j,k} + RT_{j,k}, \forall j, k = m, i = f \tag{10}$$

$$TO_k = C_{max} - \sum_j \sum_i \sum_l \frac{p_{j,k}}{Vl1_l} * s_{i,j} * Y_{j,l,k}, \forall k \tag{11}$$

$$TEC' = \sum_k \sum_j \sum_i \sum_l \frac{PM_k * P_{j,k} * s_{i,j} * Vl2_l}{60 * Vl1_l} * Y_{j,l,k} + \sum_k \frac{MO_k * PM_k}{60} * TO_k \tag{12}$$

Subject to:

$$C_{i,j,k+1} - \frac{p_{j,k+1}}{Vl1_l} * s_{i,j} * Y_{j,k,l} \geq C_{i,j,k} + TS_{j,k+1} + FT_j, \forall j, k < m, i = 1 \tag{13}$$

$$C_{i,j,k+1} - \frac{p_{j,k+1}}{Vl1_l} * s_{i,j} * Y_{j,k,l} \geq C_{i,j,k} + FT_j, \forall j, i > 1, k < m \tag{14}$$

$$C_{i,j,k} - \frac{p_{j,k}}{Vl1_l} * s_{i,j} * Y_{j,k,l} \geq TS_{j,k}, \forall j, i = 1, k = 1 \tag{15}$$

$$C_{i,j,k} - \frac{p_{j,k}}{Vl1_l} * s_{i,j} * Y_{j,k,l} \geq C_{i-1,j,k}, \forall j, k, i > 1 \tag{16}$$

$$C_{i',j',k} - \frac{p_{j',k}}{Vl1_l} * s_{i',j'} * Y_{j',k,l} - TS_{j'k} + G * (1 - X_{j,j'}) \geq C_{i,j,k} + RT_{j,k}, \forall j, j', k, i' = 1, i = f, j \neq j' \tag{17}$$

The C_{max} calculation now considers the completion of the last sublot processed in the last machine m (equation 10). Also the idle time is modified (equation 11), since the operations are dependent on the size of the sublot $s_{i,j}$. The same for the total energy consumed (equation 12). By incorporating lot-streaming, some restrictions of the problem are modified. In constraint (13) the first sublot of job j is sequenced, and the transfer times between two consecutive machines are considered. In restriction (14) it is verified that each sublot respects the precedence of operations of the flow shop. In restriction (15) the capacity required in the first machine for the first sublot is met. The ordering of the sublots of the same job j is obtained by restriction (16). However, by not allowing intermixing, the sequencing between jobs is performed by the same variable as in the previous model, although in this case restriction (17) incorporates the proportionality of the processing according to $s_{i,j}$.

2.2.3 Lot Streaming Flow Shop with Energy Awareness and Processing Speed Indexed in Sublots (SBSi+EA)

In this model, the possibility of varying the processing speed for each sublot is added to the sublot model (previously it was kept fixed for all the sublots of the same job) (SBSi+EA). Based on the SBS+EA model, a new variable is incorporated to define the processing speeds.

Variables:

$Y_{i,j,k,l}$: binary variable, is *1* if sublot i of job j is processed at speed l at machine k, 0 otherwise.

Breakdown of Z (equation 1), in terms of SBSi+EA:

$$TO_k = C_{max} - \sum_j \sum_i \sum_l \frac{p_{j,k}}{Vl1_l} * s_{i,j} * Y_{i,j,l,k} , \forall \, k \tag{18}$$

$$TEC' = \sum_k \sum_j \sum_i \sum_l \frac{PM_k * p_{j,k} * s_{i,j} * Vl2_l}{60 * Vl1_l} * Y_{i,j,l,k} + \sum_k \frac{MO_k * PM_k}{60} * TO_k \tag{19}$$

Subject to:

$$C_{i,j,k+1} - \frac{p_{j,k+1}}{Vl1_l} * s_{i,j} * Y_{i,j,k,l} \geq C_{i,j,k} + TS_{j,k+1} + FT_j, \forall \, j, k < m, i = 1 \tag{20}$$

$$C_{i,j,k+1} - \frac{p_{j,k+1}}{Vl1_l} * s_{i,j} * Y_{i,j,k,l} \geq C_{i,j,k} + FT_j, \forall \, j, i > 1, k < m \tag{21}$$

$$C_{i,j,k} - \frac{p_{j,k}}{Vl1_l} * s_{i,j} * Y_{i,j,k,l} \geq TS_{j,k}, \forall \, j, i = 1, k = 1 \tag{22}$$

$$C_{i,j,k} - \frac{p_{j,k}}{Vl1_l} * s_{i,j} * Y_{i,j,k,l} \geq C_{i-1,j,k}, \forall \, j, k, i > 1 \tag{23}$$

$$C_{i',j',k} - \frac{p_{jj,k}}{Vl1_l} * s_{i',j'} * Y_{i',j',k,l} - TS_{j',k} + G * (1 - X_{j,j'}) \geq C_{i,j,k} + RT_{j,k}, \forall \, j, j', k, i' = 1, i = f, j \neq j' \tag{24}$$

$$\sum_l Y_{i,j,k,l} = 1, \forall \, j, k \tag{25}$$

The idle time TOk is modified (equation 18), by incorporating the variable $Y_{i,j,k,l}$. The same happens with energy consumption (TEC) (equation 19). Then, restrictions (20)-(25) have a similar interpretation to that of the SBS + EA model (equations (13)-(17)), with the difference that now the duration of the processing can be modified for each sublot i of each job j. This last model naturally includes an increase in the number of binary variables, which makes their resolution difficult.

3. Experimental design

The experimental design seeks to compare the models and show empirically if the incorporation of lot-streaming is advantageous when addressing flow shop scheduling problems with energy awareness. For this, the studies that have used similar approaches were taken as a starting point [10] [13]. Regarding the parameterization related to lot-streaming, it was considered similar works from literature [18] [19]. The sizes of the problems studied considered 5 and 10 machines, and 3, 5 and 7 jobs were used for each number of machines. For models SBS+EA and SBSi+EA, f was set in, 2, 3, 4 and 5. Then, for model FSS+EA there are 6 different problem size, while for SBS+EA and SBSi+EA, there are 24 different problem size. On the other hand, for the parameters defined in section 2.2.1 for the different models, the following parameterization was used:

U_j: uniform distribution regarding [4;24] * 5. Thus, jobs lots remain within 20 and 120 units, as in [13].

$s_{i,j}$: is obtained by dividing U_j by f.

$TS_{j,k}$: uniform distribution [1;25].

$P_{j,k}$: uniform distribution [1;5] (it corresponds to processing one each unit of U_j).

FT_j: uniform distribution [1;4].

$RT_{j,k}$: uniform distribution [2;6].

$Vl1_l$: it can take one of (1.2, 1, 0.8). This factor applies to $P_{j,k}$, then, it can be standar time (1), fast (1.2) or slow (0.8).

$Vl2_l$: it can take on of (1.5, 1, 0.6). It depends on $Vl1_l$.

MO_k: 0.05.

PM_k: 60 KW.

For each size of the problem, 5 data sets were generated. Each of these data sets was solved by the three models. The experiments were carried out on an AMD A6-7310 APU with AMD Radeon R4 Graphics computer with 8GB of RAM. All the models were implemented in Pyomo [22] and GUROBI 8.1.1 was used as a solver. SolverStudio [23] was used for data post-processing.

4. Results

In this section we present the results obtained by the experimentation proposed in section 3. The numerical results regarding the objective function are summarized in Table 1. Table 1 shows the relative improvements, expressed in percentages, of the SBS+EA and SBSi+EA models with respect to FSS+EA, based on the value of FSS+EA. The data is grouped by problem size (number of machines and jobs), as well as by elements of the objective function (C_{max} and TEC). In turn, the values for each SBS+EA and SBSi+EA models are distributed according to the number of sublots used in each case. Considering the results of the makespan, it is observed that lot streaming is an effective technique to improve the performance of the system, since all the data in Table 1 are positive (it implies that the C_{max} of SBS+EA and SBSi+EA are lower than those of FSS+EA for the same problem). Even more, if it is analyzed more carefully, the improvements are really significant, since the minimum of the reductions is higher than 22%, in the case of 5J and 5M and 2 sublots. Meanwhile, the maximum reduction of the makespan reaches values higher than 50% (3J and 10M). These results support the idea of analyzing lot streaming as a scheduling strategy. On the other hand, when looking at the last of the rows of Table 1, where the average values for each model are raised, it is observed that SBSi+EA is better than SBS+EA, achieving reductions of 37.4% against 36.8% achieved by SBS+EA, although the difference is not very significant. Regarding the impact of the number sublots, it is observed that, for both models, incorporating a greater number of sublots improves the performance of the model, both for SBS+EA and SBSi+EA. These improvements represent more than 10% reduction in makespan when comparing the result of 2 sublots with that of 5 sublots. Naturally, this incorporation of a greater number of sublots has its counter effect that it adds binary variables, so that the resolution of a problem with 5 sublots is more computationally expensive. This will be discussed in more detail later.

On the other hand, if the TEC is analyzed in Table 1 it is observed that, again, lot-streaming is an effective technique to improve the performance of the system, since all the values in Table 1 are positive, except for a single value. The only value that is negative for the TEC, is in the SBS + EA model in the column of 2 sublots, for the 5J and 5M problem, which is -0.2%. This negative value is associated to the approach used in the modeling of the objective function, since, when modeled through a weighted and normalized sum, the value of Z may be more influenced by the makespan than by the TEC. This can be verified by looking at the value of makespan for the same problem, it is observed that there is a reduction of 22.8% (much higher than -0.2% of TEC).

Thus, when using a single value of Z, the value of TEC is absorbed by the makespan. However, the rest of the values are all positive, which implies that the models based on lot streaming manage to improve energy consumption compared to not implementing them. There is even a more noticeable difference between models than in the case of makespan, for the TEC the SBS + EA obtains an average improvement of 6.7% compared to the FSS + EA, while the SBSi + EA improves the FSS + EA by a 11.7%.

All these results are illustrated in Figure 2. Figure 2 shows how the SBS+EA and SBSi+EA models improve with respect to FSS+EA. Again, the results are grouped by number of sublots used in each experiment. In Figure 2, the higher the bar, the higher the improvement over FSS+EA model (the better the performance of the lot-streaming models over non lot-streaming model). Then, it is observed that the model that enables to set the processing speed for each sublot (SBSi+EA) has higher bars than the model that enable to set the processing speed for each lot, i.e. all sublots share the same processing speed (SBS+EA) for all the cases. This difference is greater for the case of TEC, meaning that indexing processing speed for each sublot has a greater impact in TEC than what it has in makespan. Another observation that can be made is that for small number of sublots the effects of both models, SBS+EA and SBSi+EA, are rather similar, but when the number of sublot increase, the bars are larger (i.e. larger improvements over FSS+EA model) and more diverse between them. This supports the idea that having the possibility to use more sublots in production enables better results in performance. However, lot streaming models needs to model more "production units" (several sublots instead of one lot) requiring more variables than regular flow shop model, which usually affects the computational capacity in solving the problem. The counterpart to the analysis of the results obtained by the different models is to analyze the computational cost of each of the models. That is, to analyze benefits and cost of each of the models. We have already shown the benefits, then, we have to consider the costs of the model. For this, we present Table 2. Table 2 shows for each of the 3 models presented the number of restrictions, variables (continuous and discrete), the average running time and the GAP reached by the software in case of reaching the maximum of 3600 secs (relative to the best relaxed solution). As a first comment, it can be said that the SBSi+EA model is the most computationally expensive, since it is the one that saturates the 3600 second limit the most times. This means that the solver does not reach the optimal solution of the problem (or does not reach the full verification of it). These results reveals that the model with the best performance in objective function, has the greater computational cost. Then, the SBS+EA is the second model in computational cost, and the cheapest is the FSS+EA. This order also coincides with the objective function values, the second in performance rate is SBS+EA and the poorest model is FSS+EA. Naturally, this computational cost increases as the number of jobs increases. But in the case of models with
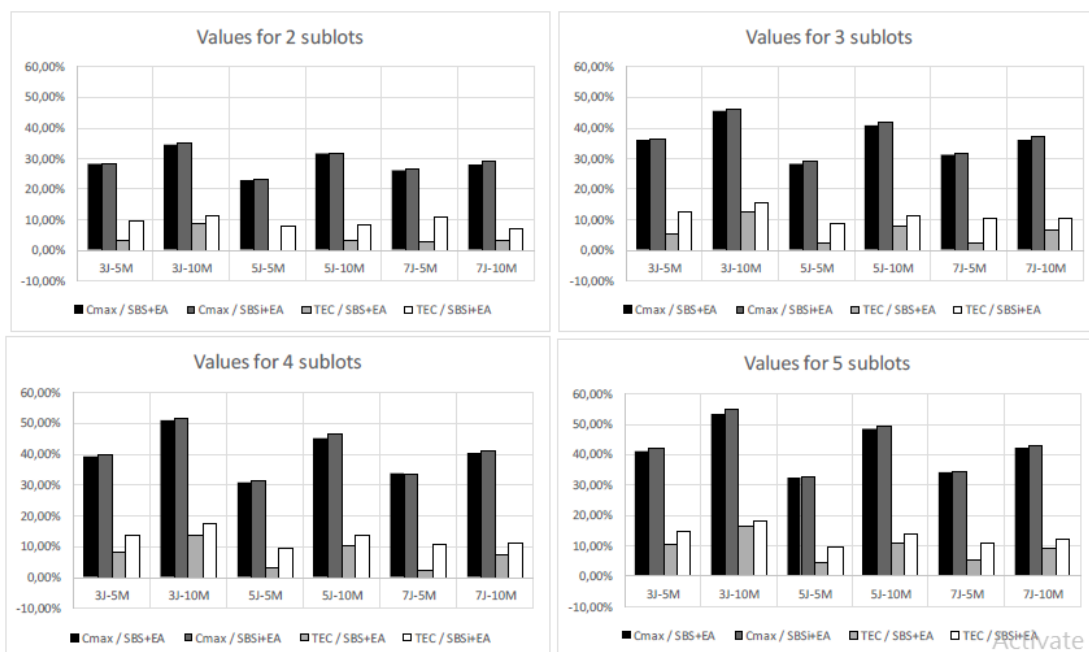

Figure 2. Lot Streaming improvements over FSS+EA model

Table 1. Improvement percentage of the SBS+EA and SBSi+EA models with respect to FSS+EA, for each problem and value of f.
Ref: each problem is identified by the number of jobs (J) and machines (M), e.g. 3J 5M, represents the problem of
3 jobs and 5 machines

	Values for 2 sublots			
	Cmax / SBS+EA	Cmax / SBSi+EA	TEC / SBS+EA	TEC / SBSi+EA
3J 5M	28.40%	28.40%	3.10%	9.50%
3J 10M	34.70%	35.20%	8.90%	11.20%
5J 5M	22.80%	22.90%	-0.20%	8.10%
5J 10M	31.60%	31.70%	3.40%	8.20%
7J 5M	26.10%	26.80%	2.90%	11.10%
7J 10M	28.20%	29.30%	3.40%	6.80%
	Values for 3 sublots			
	Cmax / SBS+EA	Cmax / SBSi+EA	TEC / SBS+EA	TEC / SBSi+EA
3J 5M	36.20%	36.30%	5.40%	12.50%
3J 10M	45.50%	46.30%	12.30%	15.50%
5J 5M	28.30%	29.10%	2.20%	8.90%
5J 10M	40.80%	41.80%	7.70%	11.50%
7J 5M	31.40%	31.70%	2.60%	10.60%
7J 10M	36.10%	37.20%	6.70%	10.50%
	Values for 4 sublots			
	Cmax / SBS+EA	Cmax / SBSi+EA	TEC / SBS+EA	TEC / SBSi+EA
3J 5M	39.50%	39.90%	8.40%	13.90%
3J 10M	51.00%	51.70%	13.60%	17.40%
5J 5M	30.90%	31.50%	3.20%	9.60%
5J 10M	45.30%	46.50%	10.30%	13.50%
7J 5M	33.90%	33.70%	2.50%	10.60%
7J 10M	40.50%	41.10%	7.40%	11.30%
	Values for 5 sublots			
	Cmax / SBS+EA	Cmax / SBSi+EA	TEC / SBS+EA	TEC / SBSi+EA
3J 5M	41.20%	42.00%	10.30%	14.50%
3J 10M	53.50%	55.00%	16.40%	18.20%
5J 5M	32.30%	32.70%	4.20%	9.80%
5J 10M	48.40%	49.50%	10.70%	13.70%
7J 5M	34.40%	34.50%	5.30%	11.00%
7J 10M	42.50%	42.90%	8.90%	12.50%

Table 2. Model comparison in terms of constraints, variables (continuous and discrete), average running time (sec) and GAP (%) (to lower bound).
Ref: each problem is identified by the number of jobs (J) and machines (M), e.g. 3J 5M, represents the problem of 3 jobs and 5 machines. And "Constr." Stands for constraints, "Cont. Var" and "Discr.Var" for continuous and discrete variables, respectively and "Avg. Time" for average time

		FSS+EA	SBS+EA				SBSi+EA			
		1	2	3	4	5	2	3	4	5
3J 5M	Constr.	69	106	133	160	187	121	163	205	247
	Cont. Var.	17	32	47	62	77	32	47	62	77
	Discr. Var.	51	51	51	51	51	96	141	186	231
	Avg. Time	1	1	2	2	2	2	7	43	795
	GAP	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
3J 10M	Constr.	129	206	263	320	377	236	323	410	497
	Cont. Var.	32	62	92	122	152	62	92	122	152
	Discr. Var.	96	96	96	96	96	186	276	366	456
	Avg. Time	1	3	4	8	8	6	36	898	2944
	GAP	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	2.6%
5J 5M	Constr.	175	240	285	330	375	265	335	405	475
	Cont. Var.	27	52	77	102	127	52	77	102	127
	Discr. Var.	95	95	95	95	95	170	245	320	395
	Avg. Time	8	22	35	53	38	52	1185	2413	2939
	GAP	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	2.9%	6.0%
5J 10M	Constr.	325	460	555	650	745	510	655	800	945
	Cont. Var.	52	102	152	202	252	102	152	202	252
	Discr. Var.	170	170	170	170	170	320	470	620	770
	Avg. Time	28	591	1489	1246	1332	1553	2198	3600	3600
	GAP	0.0%	0.0%	0.2%	0.3%	0.3%	0.3%	2.7%	6.0%	9.0%
7J 5M	Constr.	325	460	555	650	745	510	655	800	945
	Cont. Var.	52	102	152	202	252	102	152	202	252
	Discr. Var.	170	170	170	170	170	320	470	620	770
	Avg. Time	28	591	1489	1246	1332	1553	2198	3600	3600
	GAP	0.0%	0.0%	0.2%	0.3%	0.3%	0.3%	2.7%	6.0%	9.0%
7J 10M	Constr.	609	802	935	1068	1201	872	1075	1278	1481
	Cont. Var.	72	142	212	282	352	142	212	282	352
	Discr. Var.	252	252	252	252	252	462	672	882	1092
	Avg. Time	2574	3222	3600	3600	3600	3600	3600	3600	3600
	GAP	3.2%	3.7%	5.7%	5.7%	6.7%	4.3%	9.9%	13.8%	16.7%

lot streaming, as the number of sublots increases, the computational cost also increases. Another indicator of the proportionality between the cost and the size of the problem, in terms of machines and jobs, is the number of variables (particularly the discrete ones). For example, for the case of 5J 10M, the FSS+EA has 170 discrete variables, while the SBS+EA and 5 sublots have the same 170, but the SBSi+EA (with 5 sublots) has 770 discrete variables. In other words, indexing the processing speed in the sublot (SBSi+EA) considerably increases the number of discrete variables, and that is why for the same problem, the FSS+EA takes 28 seconds to reach the optimum, the SBS+EA takes 1332, while SBSi+EA times out of 3600 seconds and fails to converge to the optimum (9% GAP).

5. Conclusions

In this work, a scheduling problem in flow shop systems with energy considerations was approached. This problem, of increasing relevance in recent years, was addressed with lot-streaming strategies, which allows dividing the work lot in such a way that activities can overlap. This strategy makes it possible to exploit the efficiency of production systems by making the duration of operations more flexible. In this work, both makespan and energy consumption were reduced simultaneously. The reductions achieved are more than significant considering that it is only a sequencing strategy, which does not imply an investment in capital to improve installed capacity. In addition, the lot streaming model that allows to vary the processing speed for each sublot was found to deepen the benefits in terms of energy consumption, yielding the best results.

Funding

References

[1] Bach, T. M., & Tortato, U (2018). Cross Country Evidence on the Cointegration and Causality Relationships Between Economic Growth and CO2 Emissions in OECD Countries. International Journal of Industrial Engineering and Management, 9 (1), 31-42

[2] Tonelli, F., Bruzzone, A. A. G., Paolucci, M., Carpanzano, E., Nicolò, G., Giret, A., Salido M. A., & Trentesaux, D. (2016). Assessment of mathematical programming and agent-based modelling for off-line scheduling: Application to energy aware manufacturing. CIRP Annals, 65(1), 405-408.

[3] Briem, A., Betten, T., Held, M., Wehner, D. & Baumann, M. (2019). Environmental Sustainability in the Context of Mass Personalisation – Quantification of the Carbon Foot print with Life Cycle Assessment. International Journal of Industrial Engineering and Management,10 (2), 171-180.

[4] Salman, S. & Alaswad, S. (2020) Mitigating the Impact of Congestion Minimization on Vehicles' Emissions in a Transportation Road Network. International Journal of Industrial Engineering and Management. 11 (1), 40 – 49.

[5] Dai, M., Tang, D., Giret, A., Salido, M. A., & Li, W. D. (2013). Energy-efficient scheduling for a flexible flow shop using an improved genetic-simulated annealing algorithm. Robotics and Computer-Integrated Manufacturing, 29(5), 418-429.

[6] Paolucci, M., Anghinolfi, D., & Tonelli, F. (2017). Facing energy-aware scheduling: a multi-objective extension of a scheduling support system for improving energy efficiency in a moulding industry. Soft Computing, 21(13), 3687-3698.

[7] Pinedo, M. (2012). Scheduling. New York: Springer.

[8] Mouzon, G., & Yildirim, M. B. (2008). A framework to minimise total energy consumption and total tardiness on a single machine. International Journal of Sustainable Engineering, 1(2), 105-116.

[9] Mouzon, G., Yildirim, M. B., & Twomey, J. (2007). Operational methods for minimization of energy consumption of manufacturing equipment. International Journal of production research, 45(18-19), 4247-4271.

[10] Bruzzone, A. A., Anghinolfi, D., Paolucci, M., & Tonelli, F. (2012). Energy-aware scheduling for improving manufacturing process sustainability: A mathematical model for flexible flow-shops. CIRP annals, 61(1), 459-462.

[11] Fang, K., Uhan, N., Zhao, F., & Sutherland, J. W. (2011). A new approach to scheduling in manufacturing for power consumption and carbon footprint reduction. Journal of Manufacturing Systems, 30(4), 234-240.

[12] Fang, K., Uhan, N. A., Zhao, F., & Sutherland, J. W. (2013). Flow-shop scheduling with peak power consumption constraints. Annals of Operations Research, 206(1), 115-145.

[13] Mansouri, S. A., Aktas, E., & Besikci, U. (2016). Green scheduling of a two-machine flow-shop: Trade-off between makespan and energy consumption. European Journal of Operational Research, 248(3), 772-788.

[14] Sarin, S. C., & Jaiprakash, P. (2007). Flow-shop lot streaming. Springer Science & Business Media.

[15] Şen, A., Topaloğlu, E., & Benli, Ö. S. (1998). Optimal streaming of a single job in a two-stage flow-shop. European Journal of Operational Research, 110(1), 42-62.

[16] Han, Y., Gong, D., Jin, Y., & Pan, Q. (2017). Evolutionary multiobjective blocking lot-streaming flow shop scheduling with machine breakdowns. IEEE transactions on cybernetics, 49(1), 184-197.

[17] Sang, H. Y., Pan, Q. K., Duan, P. Y., & Li, J. Q. (2018). An effective discrete invasive weed optimization algorithm for lot-streaming flo shop scheduling problems. Journal of Intelligent Manufacturing, 29(6), 1337-1349.

[18] Rossit, D., Tohmé, F., Frutos, M., Bard, J., & Broz, D. (2016). A non-permutation flow-shop scheduling problem with lot streaming: A Mathematical model. International Journal of Industrial Engineering Computations, 7(3), 507-516.

[19] Ferraro, A., Rossit, D., Toncovich, A., & Frutos, M. (2019). Lot-streaming flow-shop with a heterogeneous machine. Engineering Management Journal, 31(2), 113-126.

[20] Rossit, Daniel Alejandro, Fernando Tohmé, and Mariano Frutos (2018). "The non-permutation flow-shop scheduling problem: a literature review." Omega 77, 143-153.

[21] Trietsch, D., & Baker, K. R. (1993). Basic techniques for lot-streaming. Operations R search, 41(6), 1065-1076.

[22] Hart, W. E., Laird, C. D., Watson, J. P., Woodruff, D. L., Hackebeil, G. A., Nicholson, B. L., & Siirola, J. D. (2017). Pyomo-optimization modeling in python (Vol. 67). Berlin: Springer.

[23] Mason, A. J. (2013). SolverStudio: A new tool for better optimisation and simulation modelling in Excel. **INFORMS** Transactions on Education, 14(1), 45-52.