



HAL
open science

An Optimization Framework for Job-shop with Energy Threshold Issue With consideration of machining operations with consumption peaks

Sylverin Kemmoe Tchomte, Damien Lamy, Nikolay Tchernev

► **To cite this version:**

Sylverin Kemmoe Tchomte, Damien Lamy, Nikolay Tchernev. An Optimization Framework for Job-shop with Energy Threshold Issue With consideration of machining operations with consumption peaks. 7th Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA 2015), Aug 2015, Prague, Czech Republic. hal-03556368

HAL Id: hal-03556368

<https://hal.science/hal-03556368>

Submitted on 4 Feb 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An Optimization Framework for Job-shop with Energy Threshold Issue With consideration of machining operations with consumption peaks

S. Kemmoe-Tchomte • D. Lamy • N. Tchernev

Abstract In this paper the problem of the Job-shop is extended to support energy constraints. The objective is to propose scheduling tools for manufacturing systems considering consumption threshold that must not be exceeded. The operations are supposed to consume more energy at beginning and thus representing a consumption peak that is often present in machine tools. This assumption results in considering that an operation is divided into two sub-operations. The goal is then to propose the best schedule considering the energy threshold, the consumptions of operations and duration of consumption peaks as given data. A Mixed Integer Linear Model (MILP) for the problem solving is proposed; it is based on flow approach to take into account the energy threshold. Since it is difficult to find exact solutions for medium and large size problems, a metaheuristic based on a GRASPxELS is proposed. Small scale instances for the problem have been generated, and results expose the relevance of the metaheuristic approach.

1 Introduction

For several years, environmental and energy constraints have become essential criteria in decision making inside enterprises or laboratories [11]. This is also one of the major issues for governments, who care of the ecological impact of the industrial sector on the planet and on society. Thus, constraints such as footprint carbon and energy (fossil or not) consumption are now seriously taken into account. According to the International Energy Outlook proposed by the U.S. Energy Information Administration, companies consumed more than 50% of the global

Sylverin Kemmoe-Tchomte
CRCGM EA 3849, Université d'Auvergne, Clermont-Ferrand, France
E-mail: sylverin.kemmoe_tchomte@udamail.fr

Damien Lamy
LIMOS UMR CNRS 6158, Aubière, France
E-mail: lamy@isima.fr

Nikolay Tchernev
LIMOS UMR CNRS 6158, Aubière, France
E-mail: tchernev@isima.fr

delivered energy in 2010 [8]. However they tend to apply their good practices to their different sites and to their suppliers by banishment of some materials for example, which could lead to the ISO14000 certification [11]. More than 64% of the allocated energy to the industrial sector was used by non-OECD industry. However, companies' answers to this problem are still limited. Without a doubt, according to Rager et al. [23] to provide solutions to this problem, two types of measures could be taken: technological and/or organizational. Technological measures are quite expensive since they emphasize on new machines or production process, whereas organizational measures focus on improvements for the existing system, thus leading to energy efficiency. A lot of enhancements can be made on this last point in order to obtain better results, meeting industrial expectations and without investing in new machines. According to [7] three decisions could be made to build an energetically responsible production system. At first, the energy used by inactive machines could be minimized (by switching them off/on or by reducing lazy times). Another solution consists in moving activities from On-peak hours to Off-peak hours. Finally, it is possible to avoid consumption peak which could lead to an overbilling. In this paper, this last point is treated in a Job-shop like manufacturing environment where a near optimal schedule should be proposed while an energy consumption threshold should not be exceeded. From the best of our knowledge, it is one of the first researches focusing on this problem.

In the next section, a literature review of articles concerning scheduling problems with energy constraints is proposed. In the third section, assumptions used in this study are presented. In the fourth section, the mathematical model of the problem is given. In the fifth section, a GRASPxELS metaheuristic is introduced in order to obtain near optimal solutions of the problem. Results for small scale instances are presented in section six. Finally, the last section consists in a conclusion and research perspectives

2 Related work

The literature is full of industrial problems consisting of minimizing the total treatment time, also called makespan, and other objective functions. Until recently, only a few works dealt with energy optimization as an important constraint in scheduling. However, the "Green Manufacturing" field of research is increasingly studied and hence a non-exhaustive review is proposed in this section.

[19] proposed methods and operational tools to minimize the energy consumption of a single machine. A mathematical formulation to minimize simultaneously the total completion time of a set of operations and the total energy consumption is proposed. Their model handles the different states a machine can be in: idle, running, switch ON or OFF. Later the same authors [20] extend their previous work and proposed a Greedy Randomized Adaptive Search Procedure (GRASP) which objective is to find a solution minimizing both the total energy used and the total tardiness.

[6] used a Genetic Algorithm associated to a Simulated Annealing in order to provide approach solutions to a Flow-shop problem with consideration of total energy consumption. The power used by the machines according to their state is taken into account. They suggest to turn on/off machines according to the need of the production system and respecting conditions. Thus, a machine will not be turned off if the next operation to be scheduled starts earlier than the duration of the turn off/on process. Their results are given in a Pareto graphic. However, they did not use industrial data and underlined that their model does not handle the possible breakdowns. Considering this last point, [25] noticed that an energy efficient system provides robustness and is less sensitive to breakdowns and thus they worked on the correlation between makespan, energy and robustness. In their model a machine may have variable speeds for processing operations. A machine which is processing a task quickly will consume more energy and the treatment time will be reduced, however if the machine is processing the operation more slowly, the consumption will be reduced. In this context if a breakdown occurs, the lost time

could be caught up by increasing processing speed of the machine. Their work is one of the first including robustness in the optimization of production systems under energy constraints. The model proposed by [9] consists in minimizing the carbon footprint, the makespan and the consumption peaks in a Flow-shop context with variable speeds allowed on machines. The market tool they used did not permit to obtain a single point on the Pareto frontier in a calculation day even in the case of a two stage Flow-shop with two possible speeds per machine and 36 jobs.

[7] proposed a state of the art of different practices concerning the manufacturing systems with energy constraints. They also stated that it is very difficult to obtain data concerning energy consumption. As stressed in their study it is not easy to optimize recently constructed manufacture industries because they are energetically well designed; however, even a smooth optimization in less recent manufactures could lead to a strong improvement. In their work [4] proposed a solution which consists in avoiding consumption peaks on a Flexible Flow-shop. They used an Energy Aware Scheduling (EAS) module on the existing schedule obtained with an APS (Advanced Planning and Scheduling) system. The EAS does not modify the given schedule, but optimizes it from the viewpoint of energy consumption by defining a new timetable for the operations. Two approaches are used: a mix-integer linear program (MILP) and a Randomized Neighborhood Search (RANS). They noticed that the MILP can be used when a large consumption peak is allowed and quickly overtaken when the energy threshold is lowered. The possibility to link the given EAS to an existing APS without changing the system in place for a solution handling both applications at the same time is a strong advantage of their approach. They finally noted the fact that their model could be strengthened, by integration of other objectives such as variability of costs and energy need of machines which is constant in their study (unitary).

[18] took into account the variability of electricity pricing during a day, by inclusion of Time-of-Use (TOU) rates in a Flexible Flow-shop. They noticed that it is better in a flexible production system to have parallelized a fast machine with high energy consumption with a slow but economic one, and high energy machine with a slow and economic one, rather than two medium machines from the viewpoint of speediness and energy consumption. [26] used a similar approach by taking into account TOU and transitions between machine states in a single machine process. The genetic algorithm they implemented could be used in extension of an MRPII (Manufacturing Resource Planning System). However their model does not modify the sequence given in input in order to find a better makespan, and consideration of machines with variable speeds is mentioned as a future study. [17] Studied a Job-shop where both the total tardiness and the total energy consumption are minimized by reducing the idle times of machines. [32] Proposed a time indexed linear program which objective is to minimize the energy spending and the carbon footprint under a TOU pricing in a height level Flow-shop.

Finally, [21] pointed out that the industrial mind-set is still focused on the fact that optimizing energy consumption is time-consuming and too expensive which is a barrier to improvements in energy efficiency that could be made in enterprises. In addition, the recent state of the art proposed by [27] underlines the lack of decisional tools relative to energy efficiency of production systems. Their review shows that most of the studies are input-oriented and quite recent and they stress the necessity to develop more output-oriented or mixed methods.

In order to clarify the approaches proposed in the literature review, the different systems under study, the way energy is optimized and the objective functions are centralized in Table 1. While reviewing the literature a lack of study concerning the Job-shop problem with consumption peaks consideration appeared, since all studies conducted so far mainly concern the total energy minimization as seen in [17], [12] or [25]. In the next section are presented the assumptions used in this paper concerning the Job-shop with an Energy Consumption Threshold (JSECT) where the operations to be scheduled present an electricity consumption peak at their start.

Table 1: Different studies concerning energy in production systems

References	Single Machine	Flow-shop	Flexible Flow-shop	Job-shop	Flexible Job-shop	Carbon footprint	Consummation Pcs	Total Energy	Energy Price (TOU)	Makespan	Total Tardiness	Robustness
Balogun and Mativenga, 2013 [2]								x				
Bruzzone et al., 2012 [4]			x				x			x	x	
Dai et al., 2013 [6]			x					x		x		
Fang et al., 2011 [9]		x				x	x			x		
He et al., 2005 [12]				x				x		x		
He et al., 2015 [13]					x			x		x		
Liu et al., 2014 [17]				x				x			x	
Luo et al., 2013 [18]			x						x	x		
Mouzon et al., 2007 [19]	x							x		x		
Salido et al., 2013 [25]				x				x		x		x
Shrouf et al., 2014 [26]	x								x			
Xu et al., 2014 [31]			x				x			x	x	
Zhang et al., 2014 [32]		x				x			x			

3 Assumptions

3.1 Job-shop assumptions

This study is based on the well-known Job-shop theoretical model (A review of the Job-shop problem is given in [14]) representing workshops with multiple paths. The Job-shop problem consists in scheduling a set of n jobs that have to be sequenced on m machines. Each job J_i (i.e.: $i \in [1, n]$) is composed of m operations O_{ij} (i.e.: $j \in [1, m]$) which are sequenced in a predefined order noted $G_i = \{O_{i1}, O_{i2}, \dots, O_{im}\}$. Each operation has to be processed on a given machine μ_{ij} during a processing time p_i and no preemption is allowed (a started operation must not be stopped before its end). The Job-shop problem consists in finding a feasible schedule by managing machine disjunctions as the machines are mutualized between jobs. One of the commonly used objectives is then to find such a schedule with a minimal global duration, also called makespan. The Job-shop problem has received a huge attention over the years, with several extensions such as transportation constraints, time-lags constraints, or financial constraints. Using the disjunctive graph introduced by [24] the operations can be modeled by vertices. Precedence constraints between operations of the same job are represented by an arc. Disjunctive constraints between two operations which require the same machine are modeled by an edge. The cost of an outgoing arc is equal to the duration of the operation. An example of a non-oriented disjunctive graph for a Job-shop scheduling problem where three jobs must be scheduled on three machines is presented on Fig. 1 where dashed edges represent disjunction constraints and each operation is performed by a given machine M_i .

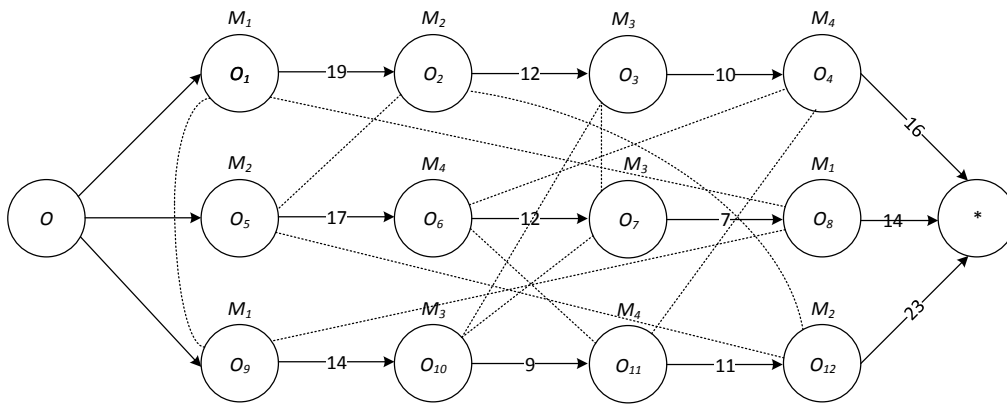


Fig. 1 : A non-oriented disjunctive graph

In the study, the graph representation proposed by [24] is extended with new arcs modelling precedence between operations when a task is delayed because of the energy used by previous and non-finished operations.

3.2 Energy assumptions

In the literature concerning energy optimization on production systems, the operations are generally represented as a simple energy block [29] which representation does not fit the reality. An energy block consists in considering an operation from the energy viewpoint, leading to a rectangle where two contiguous sides respectively represent the duration of the operation and its maximum energy consumption as shown on Fig. 2.

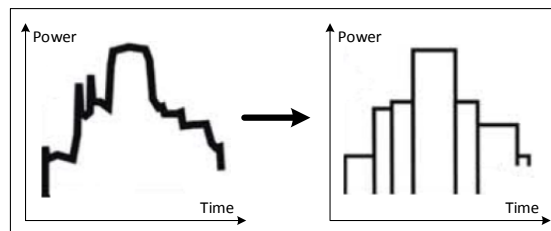


Fig. 2 : Energy Block Representation proposed by Weinert et al., 2011 [29]

Furthermore, in the literature most of the operations are considered from three points of view: basic, cutting, and idle/ready as stressed in [1]. These assumptions are used in the work of [17] when minimizing the total energy consumption. [4] considered that every operation had a unitary consumption because of the lack of industrial data, and thus represented them as a simple energy block, not taking into account the different states of the machine. The time indexed linear model they proposed showed that the lower the energy threshold is the harder it is to find an exact solution for such a problem with a linear solver. Still, their model permitted to lay the foundation of the work presented in this study. However several differences between our approaches must be enlightened. First, the basis of our problem is the Job-shop problem which is more general than a Flow-shop. But the main difference to be noticed concerns the energy assumptions. Indeed, it can be seen in the literature that machine operations have a complex energy behaviour and thus should not be only represented as a single block, especially while considering energy threshold. As stressed in [4] it is quite difficult to obtain real and precise data from the industrial sector. However, some useful information can be found concerning power profiles of machines on works related to green manufacturing, both from the viewpoint of machine or process. Thus, it can be stressed that most of the time an extra energy consumption peak is occurring at the start of the operation as shown in [7; 15] or [16].

Since the objective of this study is to minimize the makespan while considering an energy threshold constraint, it appears that a simple block representation would imply to loose time during the schedule. Indeed, based on the power profile of a 2kW fibre laser proposed in [16], it can be stressed that the laser cutting process consumes almost 37kW whereas it needs less than 20kW after the peak. Thus, if a threshold is fixed at 40kW and another operation that consumes less than 15kW must be scheduled, it means that such an operation should start at the end of the laser cutting process since both operations would consume more than 40kW if treated simultaneously ($15+37=52\text{kW} > 40\text{kW}$). If the laser cutting process has a long duration (approximately 400s in the given example), a huge amount of time is lost whereas the operation could have been scheduled just after the 37kW peak. Hence, in this paper are considered operations that are divided into two sub-operations in order to represent the consumption peak at the beginning, and a lower consumption during the remaining processing time. However, since operations cannot be stopped in the Job-shop, two sub operations relative to the same global operation must be linked in order to keep the integrity of the given operation. This constraint is modelled using no-wait arcs between such sub-operations, or maximal time-lags between two consecutive operations which values are equal to the opposite duration of the peak's length (i.e.: if peak duration is equal to a , then the time-lag max is equal to $-a$; A study on time-lags can be found in [5]). A graphical representation of such a problem is presented in the Fig. 3.

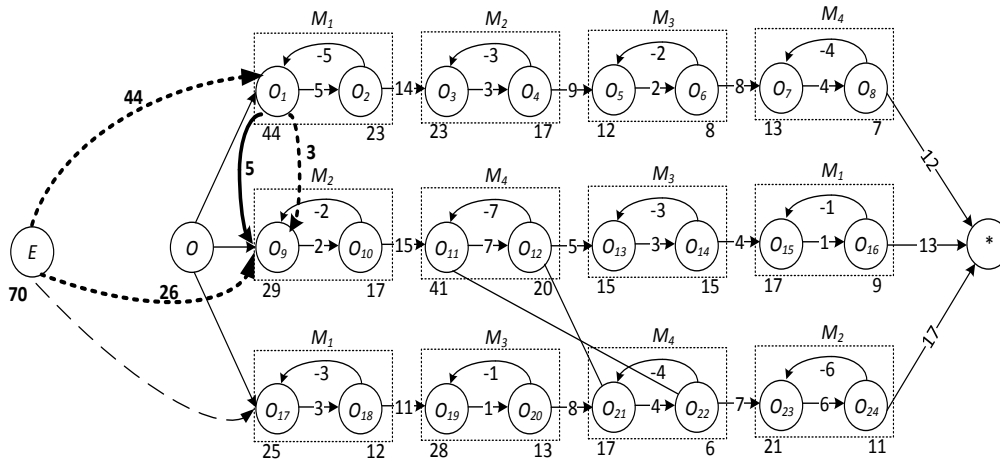


Fig. 3 : Graphical representation of a JSECT

On the Fig. 3 some extra rectangles have been added to delimitate the global operations presented on Fig.1 and to provide the energy needed for each operation. Furthermore, it can be stressed that only two disjunctive edges relative to disjunctive constraints on machines between two operations are presented on the graph, in order to ease its reading (the former operations O_6 and O_{11} in the Fig. 1 which have been transformed into operations $O_{11}-O_{12}$ and $O_{21}-O_{22}$). Indeed, since operations cannot be stopped once started, these edges represent the fact that the first sub-operation (i.e.: representing the consumption peak) must be scheduled after the end of a previous operation treated by the same machine, and hence must be scheduled after the second sub-operation of such an operation. In this example, it means that the operation O_{11} should be scheduled after the end of the operation O_{22} or the operation O_{21} should come after completion of the operation O_{12} . To complete this graph representation, the energy threshold is modelled as an extra vertex with a value corresponding to the available energy that must not be exceeded. Thus, one of the objective is to efficiently allocate the energy to the machines in order to obtain the smallest possible makespan. When there is not enough energy allocable to an operation, because of running operations extra conjunctive arcs are added to model the new precedence between operations due to the energy threshold. These extra conjunctive arcs represent the duration that an operation must wait before enough energy could flow from a finished operation to the one that must be scheduled (arc in bold on Fig. 3). Thus, a conjunctive arc may be present

between operations that are not directly related to the same machine. In Fig. 3, it can be easily understood that the available energy will not be enough to schedule all possible operations as the threshold is set to 70. Thus O_1 on machine M_1 and O_9 on machine M_2 cannot be simultaneously scheduled as the sum of their energy is equal to 73, which is greater than 70. Hence, if a sequence of operations is given, where the first operation (O_1) of job J_1 appears before the first operation (O_9) of job J_2 then it will not be possible to start operation O_9 at 0. Indeed, if O_1 starts at 0, since the operation O_2 is linked to the operation O_1 with a maximal time lag equal to -5, the operation O_2 must start directly after the end of the operation O_1 . To start operation O_2 , 23 energy units are transferred at the end of the operation O_1 to the vertex modelling operation O_2 . Because of the energy threshold, the operation O_9 will start at 5 since operation O_1 (consumption peak of the operation on Machine M_1) has a duration equal to 5. In this configuration, the operation O_9 would start at 5. Thus a conjunctive arc (represented in bold) links the operation O_1 and O_9 . When the operation O_1 is finished, a flow of 3 energy unit (dashed edge) comes in addition to the 26 units left in the Energy Vertex in order to start the operation O_9 (the values could be different depending on the other operations to be scheduled as mentioned before). By doing this repartition, the energy threshold cannot be exceeded during the schedule and thus the constraint is respected.

In the next section a mathematical formalization of the Job-shop with Energy Consumption Threshold is given. This model includes operations with a consumption peak at their beginning.

4 Linear programming

The model representing the problem has been built to obtain exact solutions not exceeding a given Energy Threshold by repartition of energy resources among the different machines. It relies on a flow added to the incumbent Job-shop problem.

4.1 Parameters

- M : set of machines;
- V : set of all the sub-operations ($|V|=2.|M|$);
- i,j,k,l : indexes representing the different sub-operations to schedule, $i,j,k,l \in \llbracket 1;|V| \rrbracket$;
- O_i : global operation of the sub-operation i ;
- J_i : job of the operation i ;
- p_i : duration of sub-operation i ;
- μ_i : machine required to process sub-operation i , $\mu_i \in M$;
- H : a large positive number.
- E_{max} : energy threshold that must not be exceeded;
- E_i : energy required for processing the sub-operation i ;

4.2 Variables

- C_{max} : completion date of all operations also referred as the makespan of the schedule;
- s_i : starting time of sub-operation i ;
- $x_{i,j}$: binary variable equal to 1 if sub-operation i is realized before sub-operation j and equal to 0 otherwise;
- $y_{i,j}$: binary variable equal to 1 if there is a non-null energy flow from sub-operation i to power the sub-operation j and equal to 0 otherwise;
- $\varphi_{i,j}$: denotes the number of energy units directly transferred from sub-operation i to sub-operation j ;

4.3 Linear Formulation

The first line (1) of the linear program refers to the objective of the problem which is the minimization of the completion time of all operations (makespan):

$$\text{Min } C_{\max} \quad (1)$$

The second set of constraints (2) gives the expression of the makespan, which must be greater or equal to the end date of all the operations:

$$s_i + p_i \leq C_{\max}, \forall i \in V \quad (2)$$

The third set of constraints (3) represents the disjunctions constraints for operations occurring on the same machines. In these constraints, if two operations i and j of different jobs must be treated by the same machine, then i is treated before j , or j is treated before i :

$$\begin{aligned} x_{j,k} + x_{l,i} &= 1, \\ \forall (i, j, k, l) \in V / J_k \neq J_i, k < l, i < j, O_i = O_j, O_k = O_l, \mu_i = \mu_k \end{aligned} \quad (3)$$

The fourth set of constraints (4) defines the starting dates of operations of a job according to its sequence of operations.

$$s_j - s_i \geq p_i, \forall (i, j) \in V / j > i, J_i = J_j \quad (4)$$

The fifth set of constraints (5) ensures that, if i and j are two sub-operations referring to the same global operations, then j is processed directly after the end of the sub-operation i (i.e. no-wait constraints).

$$s_j - s_i = p_i, \forall (i, j) \in V / i \neq j, O_i = O_j \quad (5)$$

The next constraints (6) adjust the starting dates of operations that belong to different Jobs but need the same machine, as they can't be processed simultaneously.

$$\begin{aligned} s_j - s_i - Hx_{i,j} &\geq p_i - H, \\ \forall (i, j) \in V / O_i \neq O_j, \mu_i = \mu_j, \\ \text{mod}(i, 2) &= 0, \text{mod}(j, 2) = 1 \end{aligned} \quad (6)$$

The constraint (7) avoids to exceed the Energy Threshold when processing the operations as it can't be allocated more energy to the operations than E_{\max} .

$$\sum_{j \in V} \varphi_{0,j} \leq E_{\max} \quad (7)$$

Constraints (8) ensure that the sum of energy flows from sub-operations and initial energy threshold is equal to the energy needed for the sub-operation j .

$$\sum_{i \in V} \varphi_{i,j} = E_j, \forall j \in V / i \neq j \quad (8)$$

Constraints (9) ensure that the sum of energy flows from the considered sub-operation i to the other ones never exceeds the energy that was used for its processing.

$$\sum_{j \in V} \varphi_{i,j} \leq E_i, \forall i \in V / i \neq j \quad (9)$$

Constraints (10) ensure that if there is an energy flow from i to j $y_{i,j}=1$ (this variable is then used in Constraints (12)). If $y_{i,j}=0$ then no flow is possible from i to j .

$$\varphi_{i,j} \leq Hy_{i,j}, \forall (i, j) \in V / i \neq j \quad (10)$$

Constraints (11) stipulate that if there is no need of a flow from i to j ($\varphi_{i,j}=0$), then necessarily $y_{i,j} = 0$, if $y_{i,j} = 1$, then $\varphi_{i,j} \geq 1$:

$$y_{i,j} \leq \varphi_{i,j}, \forall (i, j) \in V / i \neq j \quad (11)$$

Constraints (12) fix the energy flow between sub-operations of a same global operation according to the energy available:

$$\varphi_{i,j} = \min(E_i, E_j), \forall (i, j) \in V / i < j, O_i = O_j \quad (12)$$

Constraints (13) adjust the starting dates of sub-operations which need to wait before the end of previous operations in order to not exceed the energy threshold and receive an energy flow from a previously scheduled operation:

$$s_j - s_i - Hy_{i,j} \geq p_i - H, \forall (i, j) \in V / J_i \neq J_j \quad (13)$$

Constraints (14) stipulate that no flow is possible between two sub-operations i and j , if i and j belong to the same job and if i is processed before j .

$$\varphi_{j,i} = 0, \forall (i, j) \in V / i < j, J_i = J_j \quad (14)$$

Constraints (15) imply that if there is a flow from i to j then there must not be flows from j to the predecessors of i in the corresponding product line.

$$y_{j,k} + y_{i,j} \leq 1, \forall (i, j, k) \in V / k \leq i, J_i = J_k, J_i \neq J_j \quad (15)$$

Constraints (16) imply that if operation i and j need the same machine, but i is scheduled before operation j then no flows are allowed from j to any predecessors of i in its product line.

$$x_{i,j} + y_{j,k} \leq 1, \forall (i, j, k) \in V / k \leq i, J_i = J_k, i \neq j, \mu_i = \mu_j \quad (16)$$

Finally, the set of constraints (17) avoids cycles between operations occurring on the same machine.

$$x_{i,j} + x_{j,k} + x_{k,i} \leq 2, \forall (i, j, k) \in V / J_i \neq J_j \neq J_k, \mu_i = \mu_j = \mu_k \quad (17)$$

As stressed by the results presented in Table 3, the resolution of such a problem as the JSECT is difficult for a solver even for small scale instances, thus an approach using a metaheuristic is proposed in the next section.

5 GRASP_xELS

5.1 Principles of the GRASP_xELS

The GRASP_xELS is a multi-start metaheuristic proposed by [22] and is relying on a Greedy Randomized Adaptive Search Procedure (GRASP) proposed by [10] and an Evolutionary Local Search (ELS) proposed by [30]. This metaheuristic helped to quickly bring very good results to several problems. Furthermore, the combination of the GRASP and the ELS, aims to propose a better suited metaheuristic which will explore a wider range of solutions. A template algorithm of the GRASP_xELS is proposed below. As stressed in the Algorithm 1, a GRASP_xELS is divided into three phases: the construction phase, the local search phase and the ELS phase. During the ELS phase, neighborhood of the previous local optimum solution is explored through mutations and then ameliorated thanks to the local search. The mutation consists in permuting elements in the repetition vector used by [3] if they belong to different jobs. Finally, the different specificities corresponding to the construction and local search phase are exposed in the next sub-section as they are important part of the metaheuristic.

Algorithm 1: GRASP_xELS

Procedure name GRASP_xELS

Begin

```

1.  $S^* \leftarrow \emptyset$ 
2. for  $p := 1$  to  $np$  do
3.    $S \leftarrow$  Construction_Phase
4.    $S \leftarrow$  Local_Search_Phase
5.   if  $(f(S) < f(S^*))$  then
6.      $S^* \leftarrow S$ 
7.   endif
8.   for  $i := 1$  to  $nb\_ELS$  do
9.      $f(nS^*) :=$  INFINITY;
10.    for  $j := 1$  to  $nb\_N$  do
11.       $nS :=$  neighbor of  $S$  obtained by permutation in sequence;
12.       $nS :=$  Local_Search_Phase;
13.      if  $(f(nS) < f(nS^*))$  then
14.         $nS^* := nS$ ;
15.      endif
16.    endfor
17.     $S := nS$ ;
18.    if  $(f(S) < f(S^*))$  then
19.       $S^* := S$ 
20.    endif
21.  endfor
22.  if  $(f(S) < f(S^*))$  then
23.     $S^* \leftarrow S$ 
24.  endif
25. endfor
26. return  $S^*$ 
end

```

5.2 Specificities

Construction phase:

As the main objective is to propose a solution with minimal makespan, a construction rule based on the duration of the activities is chosen. At each construction step, an activity is randomly chosen from a list of activities with small durations. The constructed sequence is then evaluated.

Local search phase:

The construction phase rarely produces local optimum thus it is useful to explore neighborhood of the constructed solution to obtain a better one. The chosen local search is relying on the neighborhood of [28] which is really fast evidenced by the results obtained in term of computation time (Table 2 and 3).

Finally, as one of the most important algorithm of this study is the evaluation of a sequence of operation for the JSECT, this procedure is presented on the next sub-section.

5.3 Evaluation of a sequence of operations

As mentioned before, a sequence of the operations relies on a repetition vector. The evaluation of such a sequence is handling both physical and energy constraints. The starting date of an operation is modified inside the evaluation function in order to respect the energy threshold. A principle algorithm of such an evaluation function is given in Algorithm 3. This algorithm returns the makespan, the starting date and the father of each operation.

Algorithm 3: Evaluation

Procedure name Evaluation

Input/output

λ : sequence to evaluate

Input

n : number of operations

Variables

i : loop index

$op_M[]$: last operation on machine

$t_Job[]$: time job has been treated

$t_S[]$: table for energy change dates

$t_E[]$: table for energy available according to dates

job : job treated

op : operation to schedule

$machine$: machine for the operation

$father, fatherD$: predecessor and disjunctive predecessor

d, dPD : end date of predecessor and disjunctive predecessor

Begin

```

1.   Init  $op\_M, t\_Job, t\_S$ ; Init  $t\_E$  with energy threshold allowed;
2.   FOR  $i := 0$  to  $n$  DO
3.      $job := \lambda.sequence[i]$  ;
4.      $op :=$  operation corresponding to  $job$ 's occurrence;
5.      $machine :=$  machine for  $op$ ;
6.      $d := 0; dPD := 0$ ;
7.      $father := -1; fatherD := -1$ ;
8.     IF  $t\_Job[job] < 0$  THEN
9.       //Conjunctive father
10.       $father := vertex - 1$  ;
11.       $d := End[father]$ ;
12.    END IF
13.    IF ( $op\_M[machine] < -1$ ) THEN
14.      //Disjunctive father
15.       $fatherD := op\_M[machine]$  ;
16.       $dPD := End[fatherD]$  ;
17.    END IF
18.    IF ( $dPD > d$ ) THEN update father and  $d$ ; END IF
19.    Call Adjust_Energy_Date( $d, op, father, t\_S, t\_E$ );
20.    Save  $d$  and  $father$  into  $\lambda$ ;
21.    Increment  $t\_Job[job]$ ;
22.     $op\_M[machine] := op$ ;
23.  END
24.  Compute final makespan and store it into  $\lambda$ ;

```

End

Algorithm 3 is focused on evaluating a sequence of operations according to machine disjunctions. The energy aspect of the schedule is handled during the Adjust_Energy_Date algorithm, where the starting date of an operation is updated in order to respect the given energy threshold. The principles of adjusting dates according to available energy is given in algorithm 4.

Algorithm 4: Adjust_Energy_Date

Procedure name **Adjust_Energy_Date**

Input/output

d: theoretical starting date of the operation
father: theoretical father of the operation
t_S[:]: table for energy change dates
t_E[:]: table for energy available according to dates

Input

op: operation to schedule

Variables

s₁, *s₂*, *s₃*: indexes for placing operation according to energy
op_placed: boolean

Begin

```

1.   s1:=0, s2:=0, s3:= 0;
2.   op_placed := false;
3.   WHILE not op_placed DO
4.     s1:= index in t_E with enough energy available for first part of op starting from s1;
5.     Check if there is enough energy during first part of op;
6.     s2:= index for end of first part of op if possible;
7.     IF the first part of op is schedulable THEN
8.       Check if there is enough energy during second part of op;
9.       s3:= index for end of second part of op if possible;
10.    IF the second part is schedulable THEN
11.      op_placed := true;
12.    ELSE
13.      s1:= s3
14.    END
15.  ELSE
16.    s1:= s2;
17.  END
18. END
19. IF t_S[s1] <> d THEN update father according to energy disjunction; END
20. d:= t_S[s1];
21. insert d, end date of first and second part of op in t_S;
22. deduce energy used between s1 and s3 in t_E;

```

End

6 Computational evaluation

At first, the linear program has been tested using the CPLEX 12.4 solver on a machine embedding a Xeon E7-8870 processor. The previous algorithms have been implemented in C++ and have been executed on a computer with an i7-4800MQ processor, running Windows 7 (Linpack Benchmark: 2277.01 MFLOPS). Parameters used in the GRASP_xELS for the number of restart, the number of ELS and the number of neighbors are respectively 60, 30, 15. The instances used for the JSECT are composed by four to six jobs and four machines. These instances have been randomly generated by considering that the basic energy consumption and the consumption peak of an operation is between 1 and half of the duration of the operation, and

the duration of the consumption peak is taken between 0 and a third of the duration of the operation. An example of an instance is given on Fig. 4. All instances tested in Tables 2 and 3 are available online (see : <http://damienlamy.com>).

```

Number of jobs
4 4 ←
Number of machines
1 21 0 53 3 55 2 34 ← Machine 2 for
0 21 3 52 2 26 1 71 ← Operation 4
3 39 1 42 2 31 0 12 ← Duration of operation
1 77 0 55 2 66 3 77
// part 1 : Basic energy for operations
1 2 0 7 3 3 2 9 ← Operation needs 9 Energy
0 2 3 18 2 7 1 27 ← unit during the process
3 14 1 6 2 10 0 3
1 29 0 12 2 7 3 14
// part 2 : Peak energy for operations
1 5 0 21 3 12 2 17 ← Operation needs 17 more Energy
0 3 3 22 2 2 1 16 ← unit at the start of process
3 9 1 5 2 7 0 4
1 36 0 6 2 12 3 19
// part 3 : Peak duration
1 1 0 10 3 17 2 8 ← Duration of consumption peak
0 6 3 13 2 6 1 7 ← for operation
3 3 1 7 2 3 0 1
1 16 0 10 2 11 3 4
    
```

Fig. 4: An example of an instance for JSECT

For each instance, five replications have been made. The results are presented on Table 2 and Table 3. In these tables the column *Jobs x Machines* represents the number of jobs and machines of the instance. Concerning the CPLEX part of the tables, the *Energy Threshold* column represents the energy allowed for the schedule. The *BKS* column refers to the best solution found by the solver. When these solutions are proven optima, an asterisk has been added to the result. The *UB-LB* columns represent the upper bound and lower bound provided by the solver – a dash in *UB* column means that optimal solution has been found. The *Gap* column represents the percentage distance between *UP* and *LB*. The *TT* column corresponds to the time needed for the solver to found the *BKS* – When the computation time reaches 10800s the execution is stopped. In the GRASPxEELS part of the tables, the *AVG_S* column represents the average solution over 5 runs. *TT_S* refers to the total execution of the metaheuristic while *TTB_S* refers to the average computation time requested to obtain the best solution found, both in seconds. Finally, the *DEV_LB* column corresponds to the deviation to the *LB* and the *DEV_UB*, in the Table 3, corresponds to the deviation to the best solution found by the solver.

Table 2: Results obtained with CPLEX and GRASPxEELS on 45 instances – Part I

Jobs x Machines	Instances	CPLEX					GRASPxEELS				
		Energy Threshold	BKS	UB - LB	Gap	TT	AVG_S	TT_S	TTB_S	DEV_LB	
4 x 4	Inst_1	85	296*	- 296	0	3,26	296	0,15	0,000	0	
		75	301*	- 301	0	3,58	301	0,17	0,004	0	
		65	317*	- 317	0	13,48	317	0,21	0,000	0	
	Inst_2	83	404*	- 404	0	4,95	404	0,18	0,000	0	
		73	448*	- 448	0	5,45	448	0,23	0,002	0	
	Inst_3	63	492*	- 492	0	62,13	492	0,28	0,004	0	
		62	300*	- 300	0	5,41	300	0,23	0,002	0	
		52	307*	- 307	0	11,17	307	0,38	0,004	0	
	Inst_4	42	345*	- 345	0	90,16	345	0,64	0,012	0	
		85	309*	- 309	0	4,49	309	0,13	0,000	0	
	Inst_5	75	336*	- 336	0	27,22	336	0,15	0,006	0	
		65	343*	- 343	0	15,72	343	0,19	0,026	0	
		77	318*	- 318	0	2,56	318	0,14	0,000	0	
			67	329*	- 329	0	7,47	329	0,19	0,000	0
			57	350*	- 350	0	26,02	350	0,32	0,002	0
Average:						18,87		0,24	0,004		

Table 3 : Results obtained with CPLEX and GRASP_xELS on 45 instances – Part II

Jobs x Machines	Instances	Energy Threshold	CPLEX				GRASP _x ELS					
			BKS	UB - LB	Gap	TT	AVG_S	TT_S	TTB_S	DEV_LB	DEV_UB	
5 x 4	Inst_1	85	344*	-	344	0	7,42	344	0,44	0,010	0	0
		75	344*	-	344	0	13,02	344	0,53	0,004	0	0
		65	370*	-	370	0	40,4	370	0,73	0,004	0	0
	Inst_2	83	425*	-	425	0	13,75	425	0,55	0,002	0	0
		73	448*	-	448	0	1755,19	448	0,69	0,002	0	0
		63	510*	-	510	0	8606,19	510	0,89	0,014	0	0
	Inst_3	66	366*	-	366	0	70,74	366	0,64	0,028	0	0
		56	375*	-	375	0	274	375	0,99	0,004	0	0
		46	468	468	395	15,6	10800	442	1,70	0,052	11,9	-5,56
	Inst_4	85	309*	-	309	0	32,94	309	0,39	0,008	0	0
		75	336*	-	336	0	389,07	336	0,47	0,016	0	0
		65	343*	-	343	0	227,76	343	0,60	0,230	0	0
	Inst_5	77	320*	-	320	0	6,53	320	0,43	0,022	0	0
		67	331*	-	331	0	83,29	331	0,60	0,008	0	0
		57	367*	-	367	0	591,34	367	0,92	0,042	0	0
Average:						1527,44		0,71	0,03	0,79	-0,37	
6 x 4	Inst_1	100	406	406	405	0,25	10800	406	0,85	0,076	0,25	0
		90	406*	-	406	0	112,07	406	1,08	0,088	0	0
		80	426*	-	426	0	6022,18	426	1,48	0,054	0	0
	Inst_2	83	477	477	476	0,25	10800	477	0,95	0,010	0,21	0
		73	519	519	499	3,85	10800	519	1,24	0,168	4,01	0
		63	582	582	474	18,56	10800	572	1,71	0,234	20,68	-1,72
	Inst_3	95	433*	-	433	0	44,2	433	0,93	0,018	0	0
		85	463	463	435	6,05	10800	454	1,21	0,082	4,37	-1,94
		75	465*	-	465	0	855,29	465	1,66	0,204	0	0
	Inst_4	85	362*	-	362	0	2140,83	362	0,79	0,006	0	0
		75	411	411	340	17,27	10800	385	0,90	0,098	13,24	-6,33
		65	429	429	363	15,38	10800	429	1,34	0,016	18,18	0
	Inst_5	82	403	403	388	3,72	10800	395	0,88	0,076	1,80	-1,99
		72	408*	-	408	0	2455,89	408	1,24	0,072	0	0
		62	471	471	403	14,44	10800	441	1,92	0,104	9,43	-6,37
Average:						7255,36		1,21	0,09	4,81	-1,22	

The results show that the GRASP_xELS provides sound solutions. For the first instances (Table 2) the metaheuristic always found the solution provided by the solver, in less than half of a second, whereas it took more than 18 seconds in average for the solver to prove the convergence. Concerning the instances with 5 jobs and 4 machines, the solver always found the optimal solution, unless for the Inst_3 when considering the lowest energy threshold possible. On this set of instances the metaheuristic always provides a solution which quality is better or equal to the solution proposed by the solver. Concerning computation time, the metaheuristic is quite competitive since it stops in less than a second in average, when the solver needs more than 1500s in average to prove the convergence. When increasing the number of jobs to 6, the solver starts to be overtaken with only 6 proven optima. With these instances, the metaheuristic finds solutions which are always better or equal to the ones provided by the solver (more than 1% of improvement in average) in less than 2 seconds for each replication, which is approximately 6000 times faster than the time needed by the solver in average. The results show that the metaheuristic is really helpful when searching for good solutions rapidly. Even if the results are not proven to be optima, their quality are always better or equal to these provided by the CPLEX solver which validates this work.

7 Conclusion

Nowadays, commonly used objective functions in scheduling problems, such as makespan or total tardiness cannot be considered as the only objectives. Environmental issues and economic reasons lead to also take into account other objectives such as minimisation of greenhouse gas emission or electricity consumption. There exists different ways of increasing energy efficiency of a production system by minimizing the total energy needed, or the consumption peaks. This study focuses on the last point, leading to the formulation of the Job-shop with an energy

threshold constraint. In this problem each operation presents two type of energy consumption, thus a model considering sub-operations that correspond to the different energy consumption is proposed. The energy threshold is handled by respecting a maximum energy capacity over the machine-network. The instances generated representing small production systems, with few jobs and machines, are not easily solved exactly. Hence a metaheuristic based on a GRASP_xELS has been implemented in order to obtain faster solutions to these small scale instances. The results show that the approach is effective since the GRASP_xELS returns really good results which are always equal or better to the ones provided by CPLEX, in a competitive duration. However, exact methods should be further studied, with the use of Lagrangian relaxation for example. In a future work, medium and large scale instances will be addressed in a bi-objective context thus leading to a Pareto graphical representation. It could also be interesting to add other objectives such as decreasing the total energy consumption or include TOU pricing in order to reduce the cost of the production. The possibility to have different energy behaviour for the machines should also be studied as the accuracy of the energy discretization [21]. Finally, combining the different approaches seen in the literature concerning the energy-efficient production systems could lead to an interesting and complete problem.

8 Acknowledgements

This work was financially supported by the French Public Investment Bank (BPI) and granted by the ECOTHER project.

References

1. Aramcharoen A. and Mativenga P.T., Critical factors in energy demand modelling for CNC milling and impact of toolpath strategy, *Journal of Cleaner Production*, 78, 63-74 (2014).
2. Balogun V.A. and Mativenga P.T., Modelling of direct energy requirements in mechanical machining processes, *Journal of Cleaner Production*, 41, 179-186 (2013).
3. Bierwirth C., A generalized permutation approach to Job-shop scheduling with genetic algorithms. *OR spektrum*, 17, 87-92 (1995).
4. Bruzzone A.A.G., Anghinolfi D., Paolucci M. and Tonelli F, Energy-aware scheduling for improving manufacturing process sustainability: A mathematical model for flexible flow shops, *CIRP Annals – Manufacturing Technology*, 61, 459-462 (2012).
5. Caumont A., Lacomme P., Tchernev N., A memetic algorithm for the job-shop with time-lags, *Computers and Operations Research*, 35, 2331-2356 (2008).
6. Dai M., Tang D., Giret A., Salido M.A., Li W.D., Energy-efficient scheduling for a flexible flow shop using an improved genetic-simulated annealing algorithm, *Robotics and Computer-Integrated Manufacturing*, 29, 418-429 (2013).
7. Dufloy J.R., Sutherland J.W., Dornfeld D., Hermann C., Jeswiet J., Kara S., Hauschild M. and Kellens K., Towards energy and resource efficient manufacturing: A processes and systems approach, *CIRP Annals – Manufacturing Technology*, 61, 587-609 (2012).
8. Energy Administration Information (EIA), *International Energy Outlook*, <http://www.eia.gov/>, (2013).
9. Fang K., Uhan N., Zaho F., Sutherland J.W., A new approach to scheduling in manufacturing for power consumption and carbon footprint reduction, *Journal of Manufacturing Systems*, 30, 234-240 (2011).
10. Feo T.A., Resende M.G.C. and Smith S.H. A greedy randomized adaptive search procedure for maximum independent set, *Operations Research*;42: 860–878 (1994).
11. Gutowski T., Murphy C., Allen D., Bauer D., Bras B., Piwonka T., Sheng P., Sutherland J.W., Thurston D., and Wolff E., Environmentally benign manufacturing: Observations from Japan, Europe and the United States, *Journal of Cleaner Production*, 13, 1-17 (2005).
12. He Y., Liu F., Cao H. and Li C., A bi-objective model for job-shop scheduling problem to minimize both energy consumption and makespan, *Journal CSUT*, 12(2), 167-171 (2005).

13. He Y., Li Y., Wu T. and Sutherland J.W., An energy-responsive optimization method for machine tool selection and operation sequence in flexible machining job shops, *Journal of Cleaner Production*, 87, 245-254 (2015).
14. Jain A.S. and Meeran S., Deterministic Job-shop scheduling: Past, present and future, *European Journal of operations research*, 113(2), 390-434 (1999).
15. Kara S. and Li W., Unit Process energy consumption models for material removal processes, *CIRP Annals – Manufacturing Technology*, 60, 37-40 (2011).
16. Kellens K., Costa-Rodrigues G., Dewulf W., Dufloy J.R., Energy and Resource Efficiency of Laser Cutting Processes, 8th International Conference on Photonic Technologies, *Physics Procedia*, 56, 854-864 (2014).
17. Liu Y., Dong H., Lohse N., Petrovic S., Gindy N. An investigation into minimising total energy consumption and total weighted tardiness in job shops. *Journal of Cleaner Production*, 65, 87-96 (2014).
18. Luo H., Du B., Huang G.Q., Chen H. and Li X., Hybrid flow shop scheduling considering machine electricity consumption cost, *International Journal of Production Economics*, 146, 423-439 (2013).
19. Mouzon G.C., Yildirim M.B. and Twomey J., Operational methods for minimization of energy consumption of manufacturing equipment, *Wichita State University Libraries* (2007).
20. Mouzon G.C. and Yildirim M.B., A Framework to Minimize Total Energy Consumption and Total Tardiness on a single Machine, 4th Annual GRASP Symposium, *Wichita State University*, (2008).
21. O’Rielly K. and Jeswiet J. (2014). Strategies to improve industrial energy efficiency. 21st CIRP Conference on Life Cycle Engineering, *Procedia CIRP*, 15, 325-330.
22. Prins C., A GRASP x evolutionary local search hybrid for the vehicle routing problem. In: Pereira FB, Tavares J, editors. *Bio-inspired algorithms for the vehicle routing problem*, *Studies in computational intelligence*, 161, 35–53 (2009).
23. Rager M., Gahm C. and Denz F., Energy-oriented scheduling based on Evolutionary Algorithms, *Computers & Operations Research*, 54, 218-231 (2015).
24. Roy B. and Sussman B., Les problèmes d'ordonnancement avec contraintes disjonctives, *Note DS No. 9 bis, SEMA Paris* (1964).
25. Salido M.A., Escamilla J., Barber F., Giret A., Tang D. and Dai M., Energy-aware Parameters in Job-Shop Scheduling Problems, *GREEN-COPLAS 2013; IJCAI 2013 Workshop on Constraint Reasoning, Planning and Scheduling Problems for a Sustainable Future* (2013).
26. Shrouf F., Ordieres-Meré J., García-Sánchez A. and Ortega-Mier M., Optimizing the production scheduling of a single machine to minimize total energy consumption costs, *Journal of Cleaner Production*, 67, 197-207, (2014).
27. Trentesaux D. and Prabhu V., Sustainability in Manufacturing Operations Scheduling: Stakes, Approaches and Trends, *International Federation for Information Processing*, 106-113 (2014).
28. Van Laarhoven P.J.M., Aarts E.H.L. and Lenstra J.K., Job-shop scheduling by simulated annealing; *Operations Research*; 40(1), 113-125 (1992).
29. Weinert N., Chiotellis S. and Seliger G., Methodology for planning and operating energy-efficient production systems, *CIRP Annals – Manufacturing Technology*, 60, 41-44, (2011).
30. Wolf S. and Merz P., Evolutionary local search for the super-peer selection problem and the p-hub median problem, *Lecture notes in computer science*, Berlin, Springer, 4771, 1–15 (2007).
31. Xu F., Weng W. and Fujimura S., Energy-Efficient Scheduling for Flexible Flow Shops by Using MIP, *Proceedings of the 2014 Industrial and Systems Engineering Research Conference*.
32. Zhang H., Zhao F., Fang K. and Sutherland J.W., Energy-conscious flow shop scheduling under time-of-use electricity tariffs, *CIRP Annals – Manufacturing Technology* (2014).