Production scheduling with a piecewise-linear energy cost function

Mustapha Haouassi*†, Chloé Desdouits*†, Rodolphe Giroudeau* and Claude Le Pape†

* MAORE team (algorithms for scheduling and networks)
LIRMM, 34095 Montpellier Cedex 5, France
Email: {firstname.lastname}@lirmm.fr

† Analytics for Solutions team
Schneider Electric Industries SAS, 38 TEC, 38050 Grenoble Cedex 09, France
Email: {firstname.lastname}@schneider-electric.com

***Abstract*—We tackle a production scheduling problem in a manufacturing system. The aim is to design an efficient exact method and find a trade-off between tardiness, storage and energy costs described by a piecewise-linear function. Therefore, we propose two time-based MILP formulations. The first one is precedence-oriented and the second is storage-oriented. These two formulations are compared and tested on realistic instances in order to show their limits. Good solutions are obtained, even on the biggest instance.**

I. Introduction

Nowadays, the electricity price increases and the demand-response market emerges [1]. This incites factories, inter alia, to take into account the electricity tariff when building their production plans. There are two main ways to reduce the electricity bill: reduce the consumption by improving machines efficiency, or shift consuming activities when electricity is cheap. This paper is focused on the latter. The electricity cost function can be constant over time, a peak/off-peak tariff, or even a more complex cost function. In this paper, a general form of electricity tariff is studied. Time buckets are considered, and in each time bucket, a piecewise-linear function of the instantaneous power consumed gives the electricity cost per time unit. This kind of electricity tariff penalizes power peaks above given values, by increasing the variable cost of the energy and adding a fixed cost. This cost function would allow grid operators to better regulate the grid by penalizing power consumption peaks at crucial time periods. Reciprocally, it might in some contexts be interesting to concentrate energy consumption, which can be done by decreasing the variable cost as power grows. Aggregators could also use this kind of cost function to feed the demand-response market. In order to evaluate the potential impact of taking into account such an electricity tariff in a production scheduling problem, a real manufacturing plant is studied.

The studied Schneider Electric plant produces electrical cabinets. There are two parallel production lines, one for the bodies, one for the doors. At a certain point, they are painted, then put together. The current scheduling strategy in the plant is led by the will to minimize the stock, and to produce in due time, with the right quality level. This "zero stock policy" is inherited from kanban [2], one of the current production management strong principles. In a manufacturing plant, one of the only ways to get some flexibility without compromising the delivery date is to introduce some storage and to change the inventory policy. In this paper, the potential of challenging the "zero stock policy" by generating energy-aware production plans is studied. A more detailed description of the plant process, as well as a discussion about energy measurement and energy consumption characterization are given in [3].

In this paper, the challenge is to develop efficient exact methods to solve the problem as accurately as possible, while modeling efficiently the piecewise-linear energy costs. In Section II, a formal description of the scheduling problem is given. Then, Section III is dedicated to an overview of existing works about production scheduling with energy costs. In Section IV, two different Mixed-Integer Linear Programming formulations are proposed. Whereas Section V is devoted to the description of the data-set, computation and analysis of the results.

II. Modeling

In this section, the problem data are first described, then an optimal solution is characterized, finally the corresponding Graham's notation is given and complexity is discussed.

A set of two types of materials is considered: final materials (different references of electrical cabinets), and intermediate materials (cabinet bodies and doors, at different production states). Each material has an associated storage cost that models the holding cost. For instance, the storage cost of a body is higher than the storage cost of the corresponding door since the body is more difficult to transport and to store and its economical value (quantity of steal) is higher.

Definition 1. Let M be the set of all those materials. $\forall mat_m \in M$, let $sc_m \in \mathbb{R}^+$ be the storage cost of one unit of the material mat_m.

The plant receives a set of customer demands, each of them associated to a desired quantity of a given material and a due date. A tardiness cost for delivering a demand after the due date is decided, depending on the kind of demand.

Definition 2. Let D be the set of customer demands, and $\forall dem_d \in D$, let: $duedate_d \in \mathbb{N}$ be the due date of the demand, $fmat_d \in M$ be the final product to satisfy the demand, $q_d \in \mathbb{N}$ be the needed quantity of product, $tc_d \in \mathbb{R}^+$ be the cost of delivering one time unit after $duedate_d$.

Each customer demand has to be satisfied by a job, composed of six activities. For each job, the precedence graph between activities is the same in-tree.

Definition 3. Let: J be the set of jobs; $j_d \in J$ be a job such that $j_d = \{a_1[d], \ldots, a_6[d]\}$; G_d be the precedence in-tree associated to a job j_d; A be the set of all activities associated to every demand: $A = \bigcup_{j_d \in J} j_d$; G be the precedence graph between all activities, a collection of in-trees: $G = (A, \mathcal{A}) = \bigcup_{j_d \in J} G_d$.

The plant has a set $R = \{r_1, \ldots, r_5\}$ of machines with unitary capacities. Each activity $a \in A$ has to be scheduled on a given machine. Moreover, for each job j_d, the same assigment of the activities of j_d on the machines is given.

Definition 4. Let $R = \{r_1, \ldots, r_5\}$ be the set of considered machines. $\forall j_d \in J$, the assignment of the activities on the machines is: $(a_1[d], r_1)$, $(a_2[d], r_2)$, $(a_3[d], r_3)$, $(a_4[d], r_2)$, $(a_5[d], r_4)$, and $(a_6[d], r_5)$.

Besides, each precedence arc $(a_i[d], a_j[d]) \in \mathcal{A}$, holds the intermediate material reference and quantity and a minimum delay (time lag) between activities. Since the precedence graph is a disjoint collection of in-trees, every activity has at most one successor. Thus, data related to an arc (a_i, a_j) are considered as activity a_i data.

Definition 5. $\forall a_i \in A$, let: vpt_i be the processing time for every unit of material a_i produces (called variable processing time); pt_i be the total minimum processing time of a_i; res_i be the machine on which a_i has to be processed; p_i be the power consumption of the activity a_i; $imat_i$ be the intermediate material produced by a_i and consumed by a_j; q_i be the number of units of this $imat_i$ activity a_i has to produce for each unit of final material needed; $dmin_i$ be the minimum time lag between the production of each unit of $imat_i$ and the consumption of the same unit.

Figure 1 shows the precedence graph of a job j_d. The graph is an oriented in-tree with six activity nodes. Each activity is scheduled on a different machine, except for the painting activities that share the same painting machine (represented by an exclusive or). Each arc of the graph corresponds to an intermediate material, a quantity, and a minimum delay between activities. For example, in order to produce one unit of the final material, Activity a_6 has to consume one painted body and two painted jointed doors. In order to consume a painted jointed door, a drying delay of the joint of six hours has to be respected.

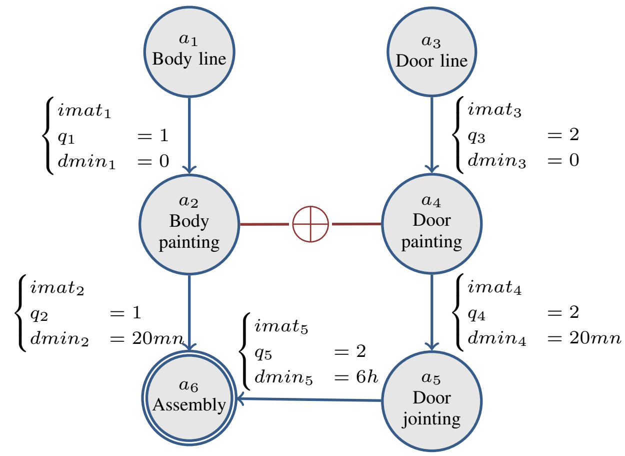


Figure 1. Precedence graph of a job j_d

An admissible solution to this scheduling problem is a schedule of the activities of every job. Both preemptive and non-preemptive solutions can be considered. Non-preemptive solutions are more easily applicable in a real factory, but preemptive solutions can really improve objective value.

Let st_i (resp. et_i) be the start time (resp. completion time) of the activity a_i. Let st_i^k be the time when activity a_i starts the production of the piece of material corresponding to the k^{th} piece of final material. The following constraints define the problem:

- the activities duration has to be respected:

$$\forall a_i[d] \in A, \qquad st_i = st_i^1 \tag{1}$$

$$et_i = st_i^{q_d \times q_i} + vpt_i \tag{2}$$

$$\forall k \in \{1, \ldots, q_d \times q_i\}, \quad st_i^{k+1} \geq st_i^k + vpt_i \tag{3}$$

- at most one activity is scheduled on the same machine at the same time:

$$\forall (a_i, a_j) \in A \times A | res_i = res_j \wedge i \neq j, \quad st_i \geq st_j \Rightarrow st_i \geq et_j \tag{4}$$

- every piece of material $imat_i$ consumed by a_j must have been produced by a_i at least $dmin_i$ sooner:

$$\forall a_j[d] \in A, \forall k \in \{1, \ldots, q_d \times q_j\}, \quad st_j^k \geq \max_{a_i \in A | \exists (a_i, a_j) \in \mathcal{A}} \left(st_i^{\lceil k \times q_i / q_j \rceil} + vpt_i + dmin_i \right) \tag{5}$$

Moreover, an optimal solution minimizes:

the storage costs

$$\sum_{mat_m \in M} (sc_m \times sq_m) \tag{6}$$

where sq_m is the number of units of mat_m stored, multiplied by the duration they were held in stock,

the tardiness costs

$$\sum_{d \in D} (tc_d \times l_d) \tag{7}$$

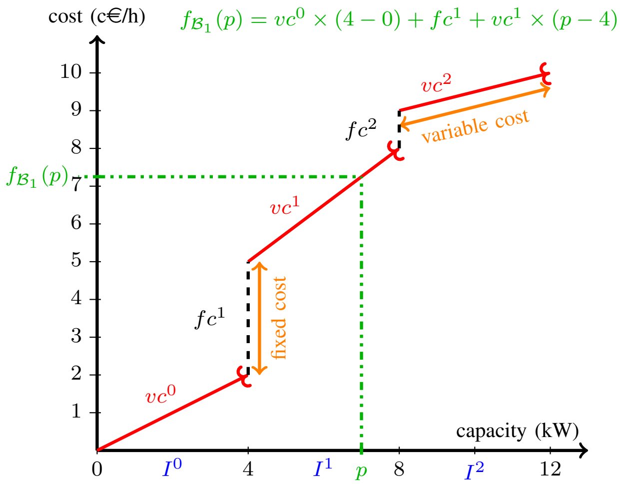


Figure 2. Energy cost function of the time bucket $\mathcal{B}_1$.

where l_d is the delivery tardiness of Demand dem_d,

the energy costs

Definition 6. Let T be the time interval considered for the scheduling problem. Let p_{max} be the maximum power that can be consumed by the studied plant and $P = [0, p_{max}]$ be the related interval of definition of possible power values. Then, let us define $power : T \to P$ a function of time that gives the overall power consumption of the plant at time t, such that: $power(t) = \sum_{a_i \in A | a_i \text{ is running at } t} p_i$.

We consider a set of buckets, that are time intervals where the energy cost function does not change. For each time t in a bucket $\mathcal{B}_l$, the electricity cost is given by the same piecewise-linear cost function $f_{\mathcal{B}_l}$, depending on $power(t)$. The power capacity, for a fixed bucket $\mathcal{B}_l$, is divided into capacity intervals (non necessarily uniform), such that in each interval the function $f_{\mathcal{B}_l}$ is linear.

Definition 7. Let: $\mathcal{B}$ be the set of buckets; $\mathcal{B}_l \in \mathcal{B}$ be a bucket such that: $\mathcal{B}_l = [\mathcal{B}_l^{inf}, \mathcal{B}_l^{sup}[$; I_l be the ordered set of power capacity intervals for a given bucket $\mathcal{B}_l$; $I^h \in I_l$ be a power capacity interval such that: $I^h =]cmin^h, cmax^h]$; fc^h be the fixed cost that occurs when: $power(t) \geq cmin^h$; vc^h be the variable cost that occurs when: $cmin^h \leq power(t) \leq cmax^h$; $f_{\mathcal{B}_l} : P \to \mathbb{R}^+$ be the piecewise-linear cost function for a given bucket $\mathcal{B}_l$, such that:

$$f_{\mathcal{B}_l}(p) = \sum_{I^h \in I_l | p > cmin^h} [fc^h + vc^h \times (\min(cmax^h, p) - cmin^h)]$$

Figure 2 gives an illustration of the piecewise-linear cost function $f_{\mathcal{B}_1}$ for a fixed bucket $\mathcal{B}_1$. In this example, power capacity is partitioned in a uniform way, every 4 kW. The value of $f_{\mathcal{B}_1}(p)$ is computed by looking over each power capacity interval, such that p is lower or equal the maximum capacity of the interval: I^0 and I^1. For each of those capacity intervals, the related cost is given by a fixed cost plus a linear function of the power exceeding the minimal capacity of the interval. For example, the cost induced by the interval I^1 is given by $fc^1 + vc^1 \times (p-4)$. Finally, the obtained value $f_{\mathcal{B}_1}(p)$ is a cost in Euro cents per hour, that has to be integrated over time in order to obtain the total cost in Euros over a given time period.

The total energy cost over a time period T is given by:

$$\mathcal{E} = \sum_{\mathcal{B}_l \in \mathcal{B}} \int_{\mathcal{B}_l^{inf}}^{\mathcal{B}_l^{sup}} f_{\mathcal{B}_l}(power(t))\,\mathrm{d}t \qquad (8)$$

This problem is a generalized flow-shop problem with precedence, due dates, time-lags, and an energy resource. The objective considered is a trade-off between tardiness cost, storage (earliness) cost and energy cost. A convenient and classical notation for scheduling problems is the Graham's notation scheme introduced in [4]. In [5], a survey of complexity results for scheduling problems with time-lags is given, and the notation l_j for time lags in the β field of the Graham's notation is introduced. Thus, our problem Π may be described as follows:

$$\Pi : F|res1, \text{in-tree}, d_j, l_j| \sum_j (w_j^E E_j + w_j^T T_j) + \mathcal{E}$$

We also address the preemptive case and note Π' the problem Π with preemption allowed.

III. Related Work

Flow-shop scheduling and lot-sizing problems have been broadly studied in the literature. In [6], a review of production scheduling is given, covering many aspects of our problem Π. More recent results are given in [7], $\mathcal{NP}$-hardness is proven for several previously opened flow-shop problems. In particular, $\Pi_2 : F2|p_{ij} = 1, chains| \sum_j(T_i)$ is proven being $\mathcal{NP}$-hard. Since Π is more general than Π_2, both are $\mathcal{NP}$-hard. Notice that Π', where the preemption is allowed for Π, remains hard (see [8]). [1]

Electricity-aware scheduling is an emerging trend. If one wants to participate in the demand-response market, one can consider maximum power-peak capacity, or electricity cost functions, both varying over time. In the first case, the key point is to manage electricity consumption to respect capacity constraints.

In [10], a scheduling problem close to ours (except for assignment, precedence and objective function) is presented: the unrelated parallel machine scheduling problem with electricity costs. The electricity cost function studied has time buckets with linear cost functions.

In [11], the authors introduce the Energy Scheduling Problem (EnSP) where activities are defined by their required energy and minimum and maximum resource requirements. The problem consists in finding, for each activity, a starting and completion times but also a power allocation that can vary over time. The objective considered is the sum of the linear energy costs and power overrun costs. A two-step approach to

[1] One could also consider Π as a RCPSP with unitary capacity resources (machines), a in-tree precedence graph and a more complex objective function. Point out [9] for a survey about RCPSP.

solve the problem is given, composed of a time-based MILP model and a constraint propagation technique.

In [12] and [13], this problem is extended with time windows and linear efficiency functions. Mixed Integer Linear Programs, satisfiability tests and a hybrid branch-and-bound method are proposed to solve the problem.

Another extension of the EnSP, given in [14], consists in considering non-linear efficiency functions. The authors introduce a piecewise-linear lower and upper bounding framework, as well as two MILP formulations, and a branch and price procedure to solve the problem. They also show that the preemptive scheduling problem with piecewise-linear energy cost is $\mathcal{NP}$-complete.

When the key point is no more the electricity capacity, but the electricity cost, complex electricity cost functions have to be considered.

In [15], a short overview of energy-related objectives in lot-sizing and flow-shops problems is given, but only peak/off-peak electricity cost functions are considered.

In the same way, in [16], a slightly different version of problem Π is studied: the time horizon is divided into several time buckets with a linear electricity cost function associated with each bucket. The authors also use a local search method to solve the problem and give a MIP formulation used as one of the local search operators.

On our side, a piecewise-linear electricity cost function, varying over time is considered, as explained in Section II.

IV. Resolution Methods

This section is devoted to the presentation of two time-based Mixed-Integer Linear formulations: the first one is non-preemptive and precedence-oriented, and the second one is preemptive and storage-oriented.

Definition 8. In both formulations, the considered time horizon T is discretized such that:

- τ is the time step (ex: 10mn)
- H is the number of time steps considered (ex: 60)
- $T = \{0, \tau, \ldots, (H-1) \times \tau\}$ is the set of instants considered
- $t \in T$ is a given time instant considered (ex: 6h45)

A. Time-Based Precedence-Oriented Formulation

This formulation is based on a classical time-based scheduling formulation (see [17]) that uses time-indexed boolean variables giving the starting time of each activity. This formulation does not allow preemption and thus addresses Problem Π. There are classical: precedence constraints with minimum delays, tardiness constraints and tardiness cost. Moreover, the storage constraints used give an approximation of the storage cost. Indeed, for each precedence arc (a_i, a_j), the whole material quantity is considered stored between the st_i and st_j. This is exact if and only if the production rate and consumption rate of both activities are the same. Otherwise this is an approximation of the storage costs that allows us to store material only if it is profitable regarding the others costs.

Finally, modeling piecewise-linear energy costs with a linear program was a challenge and is the biggest contribution of this paper. Note that the electricity cost depends on the power consumption of all activities at the same time, thus this is a coupling constraint.

Let the decision variables of this formulation be:

$\mathbf{y_{i,t}}$	$\forall a_i \in A$, $\forall t \in T$	a boolean variable that is worth 1 if the activity a_i starts at t, 0 otherwise
$\mathbf{et_i}$	$\forall a_i \in A$	the completion time of the activity a_i
$\mathbf{stock_m}$	$\forall mat_m \in M$	the total amount of time during which one unit of material mat_m is stored
$\mathbf{tard_d}$	$\forall dem_d \in D$	the (positive) amount of time between the due date of demand dem_d and its delivery date
$\mathbf{power_t}$	$\forall t \in T$	the power consumption of the whole production plan at time t
$\boldsymbol{\beta_{t,h}}$	$\forall t \in T$ $\forall I^h \in I_l \mid t \in \mathcal{B}_l$	a boolean variable equals to 1 if the power capacity interval I^h is used at time t
$\boldsymbol{\alpha_{t,h}}$	$\forall t \in T$ $\forall I^h \in I_l \mid t \in \mathcal{B}_l$	a real variable equals to the power value used in the capacity interval I^h at time t

A feasible non-preemptive solution, regarding $\mathbf{y_{i,t}}$ variables, is defined by the following equations:

$$\sum_{t \in T} \mathbf{y_{i,t}} = 1 \qquad \forall a_i \in A \tag{9}$$

$$\sum_{t \in T} t \times (\mathbf{y_{j,t}} - \mathbf{y_{i,t}}) \geq dmin_i \qquad \forall (a_i, a_j) \in \mathcal{A} \tag{10}$$

$$\sum_{a_i \in A \mid res_i = r} \sum_{t' = t^{inf}}^{t} \mathbf{y_{i,t'}} \leq 1 \qquad \forall r \in R, \forall t \in T \tag{11}$$

$$\mathbf{et_i} = \sum_{t \in T} (\mathbf{y_{i,t}} \times t + \mathbf{y_{i,t}} \times pt_i) \qquad \forall a_i \in A \tag{12}$$

Equation (9) is a classical assignment constraint that ensures activities can start only once. Equation (10) defines start-to-start precedence constraints with minimum delay. Equation (11) is the disjunctive constraint that ensures every machine is used by at most one activity at the same time. In this constraint, $t^{inf} = \lfloor \frac{t - pt_i}{\tau} \rfloor \times \tau + 1$ is the first time step where a_i can have begun and still being running at time t. Finally, Equation (12) sets the completion time of each activity, regarding their starting time.

Then, the following constraint is used to compute how much storage is used over the whole time horizon.

$$\mathbf{stock_m} = \sum_{(a_i, a_j) \in \mathcal{A} \mid mat_i = mat_m} \sum_{t} \big[t \times \tau \times (\mathbf{y_{j,t}} - \mathbf{y_{i,t}}) \times q_i \big], \forall mat_m \in M \tag{13}$$

Equation (13) allows us to recover: the idle time between the starting time of activity a_i (that produces material mat_m) and the starting time of activity a_i (that consumes it), multiplied by the quantity of material mat_m produced. Note that this is an approximation of the real storage duration of mat_m, since that supposes activities have the same consumption and production rate.

The following classical tardiness constraints ensure that $\mathbf{tard_d}$ is the positive tardiness of demand dem_d, regarding its due date.

$$\mathbf{tard_d} \geq \mathbf{et_6}[\mathbf{d}] - duedate_d \qquad \forall dem_d \in D \qquad (14)$$

$$\mathbf{tard_d} \geq 0 \qquad \forall dem_d \in D \qquad (15)$$

Equation (14) ensures that the tardiness of a given demand dem_d is greater than the amount of time between its due date and the completion time of the last activity of the job j_d. Equation (15) ensures that this tardiness is positive.

Equation (16) computes the instantaneous power used by the whole set of activities, at every time step.

$$\mathbf{power_t} = \sum_{a_i \in A} \sum_{t'=t^{inf}}^{t} p_i \times \mathbf{y_{i,t'}} \qquad \forall t \in T \qquad (16)$$

Previous time-steps have to be looked over in order to determine if an activity is running at time t, following the same principles as Equation (11).

Now, let us introduce the constraints defining the piecewise-linear energy cost function, as described in Section II.

$$\forall t \in T, \quad \forall I^h \in I_l | t \in \mathcal{B}_l,$$

$$\boldsymbol{\beta_{t,h}} \geq \frac{\mathbf{power_t} - cmin^h}{p_{max}} \qquad (17)$$

$$\boldsymbol{\alpha_{t,h}} \leq (cmax^h - cmin^h) \times \boldsymbol{\beta_{t,h}} \qquad (18)$$

$$\boldsymbol{\alpha_{t,h}} \geq (cmax^h - cmin^h) \times \boldsymbol{\beta_{t,h+1}} \qquad (19)$$

$$\forall t \in T, \quad \mathbf{power_t} = \sum_{I^h \in I_l | t \in \mathcal{B}_l} \boldsymbol{\alpha_{t,h}} \qquad (20)$$

Equation (17) forces $\boldsymbol{\beta_{t,h}}$ to 1 when the power consumed at time t is greater than the minimal capacity of the interval. Equation (19) ensures that if the capacity interval I^{h+1} is (partially) covered, then the capacity interval I^h is entirely covered. Equation (18) ensures that if the capacity interval I^h is (partially) covered, then $\boldsymbol{\beta_{t,h}}$ is worth 1. Equation (20) ensures that the sum of capacity interval covering equals to the power consumed at the same time.

Finally, the objective function of this time-based precedence-oriented formulation is defined below.

Energy objective

$$\sum_{t \in T} \sum_{I^h \in I_l | t \in \mathcal{B}_l} fc^h \times \boldsymbol{\beta_{t,h}} + vc^h \times \boldsymbol{\alpha_{t,h}} \qquad (21)$$

Tardiness objective

$$\sum_{d} \mathbf{tard_d} \times tc_d \qquad (22)$$

Storage objective

$$\sum_{m} \mathbf{stock_m} \times sc_m \qquad (23)$$

Then, the overall MILP formulation is given by:

$$\begin{array}{ll} \text{min} & (21) + (22) + (23) \\ \text{s.t.} & (9) - (20) \end{array}$$

B. Time-Based Storage-Oriented Formulation

This second formulation allows preemption and is driven by the amount of materials in storage. The key point is that a precedence link between activities is equivalent to a shared storage of material. Indeed, the stock of mat_i has to be filled by a_i before a_j could be scheduled. In fact, managing stock levels is somehow the dual point of view of ensuring precedence, as the materials graph is the dual graph of the precedence graph. Now, let us show how precedence constraints are replaced by storage constraints in this formulation.

The decision variables given below are the ones that are different from the previous formulation. Let:

$\mathbf{x_{i,t}}$	$\forall a_i \in A$, $\forall t \in T$	be a boolean that is equal to 1 if a_i is running between t and $t+\tau$
$\mathbf{dur_{i,t}}$	$\forall a_i \in A$, $\forall t \in T$	be the duration of a_i that covers $[t, t+\tau]$
$\mathbf{z_{d,t}}$	$\forall dem_d \in D$, $\forall t \in T \cup \{H \times \tau\}$	be a boolean that is equal to 1 if demand dem_d is supplied at time t
$\mathbf{stock_{m,t}}$	$\forall mat_m \in M$, $\forall t \in T \cup \{H \times \tau\}$	be the amount of material mat_m stored at time t

Moreover, $\mathbf{tard_d}$, $\mathbf{power_t}$, $\boldsymbol{\alpha_{t,h}}$ and $\boldsymbol{\beta_{t,h}}$ are defined in the same way as in Subsection IV-A.

As this formulation allows preemption, consistency between x boolean variables and dur amounts of time is ensured by the following equations.

$$\mathbf{dur_{i,t}} \leq \tau \times \mathbf{x_{i,t}} \qquad \forall a_i \in A, \forall t \in T \qquad (24)$$

$$\mathbf{dur_{i,t}} \geq \mathbf{x_{i,t}} \qquad \forall a_i \in A, \forall t \in T \qquad (25)$$

$$\sum_{t \in T} (\mathbf{dur_{i,t}}) = pt_i \qquad \forall a_i \in A \qquad (26)$$

Equation (24) ensures that $\mathbf{x_{i,t}} = 0$ implies $\mathbf{dur_{i,t}} = 0$ and bound dur by τ, while Equation (25) ensures that $\mathbf{dur_{i,t}}$ is not null if $\mathbf{x_{i,t}}$ is not null. Equation (26) ensures that an activity is executed during its required processing time. Note that a solution of this formulation does not give starting and completion time of each activity: given the duration of an activity over a time-step, its starting time has to be decided.

Disjunctive constraints relative to machines unitary capacity are stated in Equation (27).

$$\sum_{a_i \in A | res_i = r} \mathbf{x_{i,t}} \leq 1 \qquad \forall r \in R, \forall t \in T \qquad (27)$$

Note that the disjunction concerns the whole time-step even if two activities could be scheduled one after the other in the same time-step. A relaxation of this constraint using $\mathbf{dur_{i,t}}$ variables is currently investigated.

Delivery and tardiness constraint are re-formulated as below.

$$\forall dem_d \in D,$$
$$\sum_{t \in T \cup \{H\}} \mathbf{z_{d,t}} = 1 \tag{28}$$
$$\mathbf{tard_d} = \sum_{\forall t \in T \cup \{H\}} (t \times \mathbf{z_{d,t}}) - duedate_d \tag{29}$$
$$\mathbf{tard_d} \geq 0 \tag{30}$$

Equation (28) ensures each demand is delivered once. Equations (29) and (30) ensures delivery is after due date and tardiness is consistent.

The key point of this formulation is the following flow-like equations that regulates material storage. The idea is that the quantity of a material mat_m stored at time t is given by the quantity of the same material stored at time $t - \tau$, minus the quantity consumed at time $t - \tau$, plus the quantity produced at time $t - \tau$. In order to compute those quantities, one has to look over each precedence arc related to mat_m. Then the quantity of material produced by an activity a_i is given by: $vpt_i/\mathbf{dur_{i,t-\tau}}$. This reasoning works for every precedence arc without a minimum delay requested. But when there is a minimum delay $dmin_i$ on a precedence arc (a_i, a_j), another virtual material mat_i^{fict} has to be created to ensure the delay. Indeed, for a given time-step t, the stock of mat_i, as well as the stock of mat_i^{fict} are influenced by what was produced at time-step $t^{delay} = \lfloor \frac{t - dmin_i}{\tau} \rfloor \times \tau$.

$$\begin{aligned}
\forall mat_m \in M, \forall t \in T \cup \{H \times \tau\}, \mathbf{stock_{m,t}} &= \mathbf{stock_{m,t-\tau}} \\
&+ \sum_{a_i \in A | mat_i = mat_m} \left(\frac{\mathbf{dur_{i,t-\tau}}}{vpt_i} \right) \\
&- \sum_{dem_d \in D | fmat_d = mat_m} (\mathbf{z_{i,t}} \times q_d) \\
&- \sum_{(a_i,a_j) \in \mathcal{A} | mat_i = mat_m \wedge dmin_i = 0} \left(\frac{\mathbf{dur_{j,t-\tau}}}{vpt_j} \right) \\
&- \sum_{a_i \in A | mat_i = mat_m \wedge dmin_i > 0} \left(\frac{\mathbf{dur_{i,t^{delay}-\tau}}}{vpt_i} \right)
\end{aligned} \tag{31}$$

Equation (31) regulates the stock of each material mat_m. Every activity a_i increases $stock_{m,t}$ of the quantity of mat_m it can produce in $\mathbf{dur_{i,t-\tau}}$. Every demand that consumes mat_m decreases $stock_{m,t}$ of q_d units if it is delivered at time t. Every activity a_j decreases $stock_{m,t}$ of the quantity of mat_m it can consume in $\mathbf{dur_{j,t-\tau}}$, if there is no minimum delay on the arc (a_i, a_j). If there is a minimum delay on arc (a_i, a_j), then the material mat_m is automatically consumed $dmin_i$ time units after having being produced. The quantity automatically consumed increases another virtual stock introduced below.

$$\forall (a_i, a_j) \in \mathcal{A} | dmin_i > 0, \forall t \in T \cup \{H \times \tau\},$$
$$\mathbf{stock^{fict}_{i,t}} = \mathbf{stock^{fict}_{i,t-\tau}} + \frac{\mathbf{dur_{i,t^{delay}-\tau}}}{vpt_i} - \frac{\mathbf{dur_{j,t-\tau}}}{vpt_j} \tag{32}$$

Equation (32) regulates the stock of each fictive material mat_i^{fict} induced by every precedence arc (a_i, a_j) producing mat_i with a not null minimum delay.

Finally, Equation (33) computes the instantaneous power used by the whole set of activities, at every time step, while Equations (17) - (20) ensure the electricity cost is computed as in the previous formulation.

$$\mathbf{power_t} = \sum_{a_i \in A} p_i \times \mathbf{x_{i,t}} \qquad \forall t \in T \tag{33}$$

Note that this equation induces a pessimistic computation of the power consumed during a time-step as all activities are considered being simultaneous.

Then, the overall MILP formulation is given by:

$$\begin{array}{ll}
\min & (21) + (22) + \\
& \sum_{mat_m \in M} \sum_{t \in \{\tau, \ldots, H \times \tau\}} \mathbf{stock_{m,t}} \times \tau \times sc_m \\
\text{s.t.} & (17) - (20) \ \& \ (24) - (33)
\end{array}$$

V. Experiments and Results

This section reports on a computational comparison of both formulations described in Section IV and the formulation given in [16] that addresses the same problem.

The algorithms compared have been implemented in Java using the Concert library of IBM ILOG CP optimizer 12.6.1 on an Intel(R) Core(TM) i7-4810MQ CPU (2.80GHz).

The experiments have been conducted on three different instances, whose data have been built using the real plant data. The first one has two demands for a time horizon of two days. The second one has six demands for a time horizon of five days. The third one has no precedence (there are only the door line data), and two hundreds and ten demands for a time horizon of seven days.

The painting process is a carousel on which doors and bodies are hung, painted, then unhung. In those instances, the painting process is modeled by two activities and resources and a painting delay between them.

The electricity cost functions are the same in every instance: there are three time buckets for each day and two intervals of capacity in each bucket. For the sake of the example, the second capacity interval costs less than the first one. Moreover, only the door line and the body line require electricity.

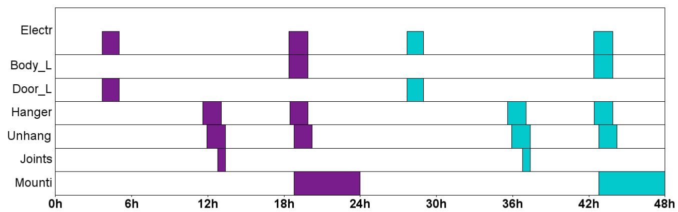
Our experimental study aims to compare resolution methods in term of solution quality, run time and memory usage. For the time-indexed formulations, several time steps are tried in order to evaluate their impact on run time and solution quality. Maximum run time is also tuned to see how the methods behave through their resolution process.

Before presenting results, let us comment the size of the each formulation, depending on input data size. Table I gives the number of variables and constraints for each formulation. We can see that the storage-oriented formulation has more variables and constraints than the precedence-oriented formulation, while both have $O\left(|T| \times \left[|J| + \sum_{\mathcal{B}_l \in \mathcal{B}} |I_l|\right]\right)$ variables. On the other hand, the overlaps formulation used in [16] has $O\left(|J|^2 + |J| \times |\mathcal{B}|\right)$ variables and constraints.

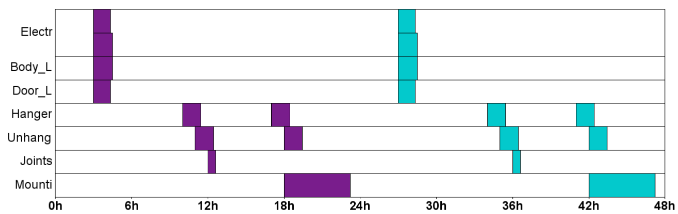
Table I
DIMENSION OF THE STUDIED MILP FORMULATIONS

Formulations	Overlaps	Precedence-oriented	Storage-oriented
Variables	$4\|J\|^2 + 16\|J\| + 18\|J\| \times \|\mathcal{B}\|$	$\|T\| \times (6\|J\| + 2\sum_{\mathcal{B}_l \in \mathcal{B}} \|I_l\| + 1) + 13\|J\|$	$\|T\| \times (19\|J\| + 2\sum_{\mathcal{B}_l \in \mathcal{B}} \|I_l\| + 1) + 8\|J\|$
Including binaries	$4\|J\|^2 - 3\|J\| + 6\|J\| \times \|\mathcal{B}\|$	$\|T\| \times (6\|J\| + \sum_{\mathcal{B}_l \in \mathcal{B}} \|I_l\|)$	$\|T\| \times (7\|J\| + \sum_{\mathcal{B}_l \in \mathcal{B}} \|I_l\|) + \|J\|$
Constraints	$8\|J\|^2 + 23\|J\| + 30\|J\| \times \|\mathcal{B}\|$	$\|T\| \times (3\sum_{\mathcal{B}_l \in \mathcal{B}} \|I_l\| + 7) + 25\|J\|$	$\|T\| \times (22\|J\| + 3\sum_{\mathcal{B}_l \in \mathcal{B}} \|I_l\| + 7) + 19\|J\|$

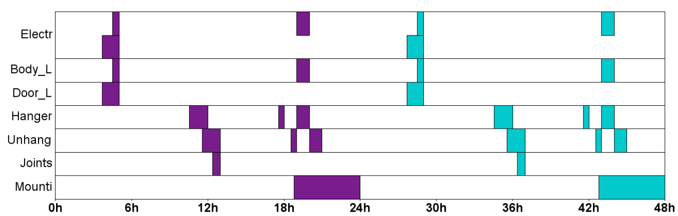
Now, let us present Figure 3 that holds Gantt diagrams obtained with each method on the same given instance. Each solution is optimal for its formulation and one-hour time-steps were chosen for the sake of the example. The X-axis represents the time horizon while the Y-axis represents the resources (electricity and machines). Considered resources are (top-down): electricity, body line, door line, a worker who hangs materials to be painted, another worker who unhangs those materials, door jointing machine, workers who assembly bodies and doors.



(a) Overlaps formulation



(b) Time-based precedence-oriented formulation



(c) Time-based storage-oriented formulation

Figure 3. Gantt diagrams obtained on a two-days instance.

Figure 3a shows the overlaps formulation results. In this formulation, linear cost functions are taken into account instead of piecewise-linear cost functions. Indeed, only the first power capacity interval of each bucket is considered. Thus, scheduling energy consuming activities at the same time is not significant in this formulation. The door line activities are scheduled early to benefit of the cheap electricity tariff. The body activities are scheduled just before the mounting activities, when the electricity is cheap again, in order to pay a small storage cost.

However, the precedence-oriented formulation schedules body activities at the same time as door activities. That allows the schedule to benefit from the second power capacity interval where the electricity cost is much lower than every other costs. In this case, paying more storage costs is interesting to gain even more on electricity cost.

On the other hand, the storage-oriented formulation uses preemption to refine the trade-off between storage costs and electricity costs. Indeed, body activities are preempted to benefit from the low electricity costs in the first bucket as well as reducing storage costs by scheduling the last part of body activities at the latest time possible. Moreover, hanging and unhanging tasks are preempted to save storage cost because of the low consumption rate of the mounting activities.

Now, let us look at numeric results on Table II. The first three columns give the context of the experiment: the instance, the time-step used and the maximum run-time allowed. Instances are denoted by the couple: (number of jobs, number of days). The other columns give, for each formulation: the gap between the best integer objective value and the best relaxed objective value, the energy cost, the storage cost and the tardiness cost of the best solution obtained.

Now, let us analyze the results. On the smallest instance (2,2), all formulations give an optimal solution in less than ten minutes. The time-indexed formulations obtain a lower cost than the Overlaps formulation because of the linear energy cost function considered by the latter. The storage-oriented formulation obtains better results than the others, because the preemption is allowed.

For the (6,5) instance, the overlaps formulation is closer to the optimal than the time-indexed formulations. Its tardiness cost is the smallest among all solutions. However, the energy cost obtained (and thus the objective value) is worse than those of the other formulations because of the linear energy cost function used.

For the (210,7) instance, the overlaps formulation does not give a feasible solution after ten minutes, nor after one hour, unlike the time-indexed formulations. The storage-oriented formulation has a better gap than the precedence-oriented one and gives a better objective function result.

Increasing the time step should decrease the run-time needed by the time-indexed formulations to get feasible solutions. On the other hand, an optimal solution with a big time-step would be worse than (or equal to) an optimal solution with a small time-step. The experiments show that increasing the time-step globally improves the solution found by the storage-oriented formulation. That is not the case for the precedence-

Table II
NUMERIC RESULTS

	Formulations		Overlaps				Precedence-oriented				Storage-oriented			
Instances	τ (mn)	run-time	gap	ener.	stor.	tard.	gap	ener.	stor.	tard.	gap	ener.	stor.	tard.
(2, 2)	10	10 mn	0 %	40,8	36,7	0	0 %	22	37	0	0 %	24,4	18,2	0
(6, 5)	10	10 mn	0.31 %	111,87	24,4	5,52	7.85 %	66,24	47,1	23,6	12.11 %	69,58	17,93	19,08
(6, 5)	15	10 mn	"	"	"	"	8.55 %	67,99	46,74	23,6	2.34 %	66,24	11,03	21,06
(6, 5)	15	1 hour	"	"	"	"	6.96 %	67,41	45,12	9,16	1.37 %	66,24	11,39	11,16
(210, 7)	10	10 mn	∞	∞	∞	∞	98,23 %	4074,2	60,76	5129,5	63.77 %	3892,42	1,34	795,9
(210, 7)	15	10 mn	"	"	"	"	98,25 %	4566	95,64	5937,16	5.00 %	4070,85	1,02	4,5
(210, 7)	15	1 hour	"	"	"	"	95,78 %	4332,6	44,84	1183,36	3.07 %	4156,05	0.89	0,04

oriented formulation.

Finally, the size of the formulations, discussed in Section V, has only partially the expected influence on the solution quality. Indeed, although the overlaps formulation is event-based, the fact it depends on $|J|^2$ results in finding no solution for the biggest instance. On the other hand, albeit the storage-oriented formulation has a biggest number of variables and constraints than the precedence-oriented one, allowing the preemption is sufficient to give better results in the same conditions. A trade-off has to be made between the quantity of preemption allowed and the objective value.

VI. CONCLUSION

In this paper, a scheduling problem taken from a real manufacturing plant is formalized. Two MILP formulations are given in order to solve the problem efficiently with some modeling approximations. The emphasis is put on the electricity cost modeling and a set of piecewise-linear cost functions is considered. We show that good solutions can be found for realistic instances, with both the proposed formulations.

In the real manufacturing plant context, those formulations are aimed to be integrated in a decision-aid tool. Such a tool could be used by the plant manager to generate and compare possible production plans, as well as their impact on each of the cost functions. A good way to build a decision-aid tool would be to use these formulations as local search operators, on accurately chosen time-windows.

Finally, from a fundamental research point of view, the following tracks are considered. Adding setup costs on activities in the storage-oriented formulation would discourage preempting too many times activities if the gain is not so big. Modeling in an exact way the instantaneous power consumed could be investigated, using [11]. An event-based formulation taken into account piecewise-linear energy cost functions is currently under study. Experiments with benchmark instances will follow.

ACKNOWLEDGMENT

This work has been conducted as part of the Arrowhead European project and has been partially funded by the Artemis/Ecsel Joint Undertaking, supported by the European Commission and French Public Authorities, under grant agreement number 332987.

REFERENCES

[1] M. H. Albadi and E. El-Saadany, "A summary of demand response in electricity markets," *Electric power systems research*, vol. 78, no. 11, pp. 1989–1996, 2008.

[2] Y. Sugimori, K. Kusunoki, F. Cho, and S. Uchikawa, "Toyota production system and kanban system materialization of just-in-time and respect-for-human system," *International Journal of Production Research*, vol. 15, no. 6, pp. 553–564, 1977.

[3] C. Desdouits, J.-L. Bergerand, P.-A. Berseneff, C. L. Pape, and D. Yanculovici, "Energy study of a manufacturing plant," in *ECEEE Industrial Efficiency Summer Study*, Berlin, Germany, in press.

[4] R. L. Graham, E. L. Lawler, J. K. Lenstra, and A. R. Kan, "Optimization and approximation in deterministic sequencing and scheduling: a survey," *Annals of discrete mathematics*, vol. 5, pp. 287–326, 1979.

[5] X. Zhang, "Scheduling with time lags," Ph.D. dissertation, Erasmus University Rotterdam, 2010.

[6] S. C. Graves, "A review of production scheduling," *Operations research*, vol. 29, no. 4, pp. 646–675, 1981.

[7] P. Brucker and S. Knust, "Complexity results for single-machine problems with positive finish-start time-lags," *Computing*, vol. 63, no. 4, pp. 299–316, 1999.

[8] T. Gonzalez and S. Sahni, "Flowshop and jobshop schedules: complexity and approximation," *Operations research*, vol. 26, no. 1, pp. 36–52, 1978.

[9] P. Brucker, A. Drexl, R. H. Möhring, K. Neumann, and E. Pesch, "Resource-constrained project scheduling: Notation, classification, models, and methods," *European Journal of Operational Research*, vol. 112, no. 1, pp. 3–41, 1999.

[10] J.-Y. Ding, S. Song, R. Zhang, R. Chiong, and C. Wu, "Parallel machine scheduling under time-of-use electricity prices: New models and optimization approaches," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 1138–1154, 2016.

[11] C. Artigues, P. Lopez, and A. Hait, "The energy scheduling problem: Industrial case-study and constraint propagation techniques," *International Journal of Production Economics*, vol. 143, no. 1, pp. 13–23, 2013.

[12] M. Nattaf, C. Artigues, and P. Lopez, "A hybrid exact method for a scheduling problem with a continuous resource and energy constraints," *Constraints*, vol. 20, no. 3, pp. 304–324, 2015.

[13] M. Nattaf, C. Artigues, P. Lopez, and D. Rivreau, "Energetic reasoning and mixed-integer linear programming for scheduling with a continuous resource and linear efficiency functions," *OR Spectrum*, vol. 38, no. 2, pp. 459–492, 2016.

[14] S. U. Ngueveu, C. Artigues, and P. Lopez, "Scheduling under a non-reversible energy source: An application of piecewise linear bounding of non-linear demand/cost functions," *Discrete Applied Mathematics*, vol. 208, pp. 98–113, 2016.

[15] O. Masmoudi, A. Yalaoui, Y. Ouazene, and H. Chehade, "Lot-sizing in flow-shop with energy consideration for sustainable manufacturing systems," *IFAC-PapersOnLine*, vol. 48, no. 3, pp. 727–732, 2015.

[16] G. German, C. Desdouits, and C. Le Pape, "Energy optimization in a manufacturing plant," in *ROADEF annual conference*. ROADEF, 2015.

[17] A. A. B. Pritsker, L. J. Waiters, and P. M. Wolfe, "Multiproject scheduling with limited resources: A zero-one programming approach," *Management science*, vol. 16, no. 1, pp. 93–108, 1969.