



**HAL**  
open science

## Scheduling under non-reversible energy sources

Sandra Ulrich Ngueveu, Christian Artigues, Pierre Lopez

► **To cite this version:**

Sandra Ulrich Ngueveu, Christian Artigues, Pierre Lopez. Scheduling under non-reversible energy sources. *Discrete Applied Mathematics*, 2016, 208, pp. 98-113. hal-01163925

**HAL Id: hal-01163925**

**<https://hal.science/hal-01163925v1>**

Submitted on 15 Jun 2015

**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.

# Scheduling under non-reversible energy sources

Sandra Ulrich Ngueveu<sup>1,2</sup>, Christian Artigues<sup>1,3</sup> and Pierre Lopez<sup>1,3</sup>

<sup>1</sup>CNRS, LAAS, 7 avenue du Colonel Roche, F-31400 Toulouse, France

<sup>2</sup>Univ de Toulouse, INP, LAAS, F-31400 Toulouse, France

<sup>3</sup>Univ de Toulouse, LAAS, F-31400 Toulouse, France

Tel.: +33-(0)5.61.33.62.90

Fax: +33-(0)5.61.33.69.36

{ngueveu,artigues,lopez}@laas.fr

Received: date / Accepted: date

## Abstract

In this paper, we address a preemptive scheduling problem involving multiple non-reversible energy sources. To the classical scheduling issue, an additional decision level is added regarding the selection of the energy source used to satisfy the total power demand of tasks processed at each instant. Different non-reversible energy sources are available, with different characteristics in terms of efficiency and power range. The objective is to identify the best combination between scheduling and energy resource utilization that minimizes the total energy cost of the project. Non-linear efficiency functions used to compute energy costs are bounded from above and below by two piecewise-linear curves, yielding two instances of a scheduling problem with a piecewise-linear objective that can be solved separately. For the piecewise-linear scheduling problem, we show that the problem involving multiple sources is equivalent to a single-source problem, the particular case of a linear function is polynomially solvable, and the case with a piecewise-linear function with two pieces is NP-hard. A pseudo-polynomial size time-indexed mixed-integer linear formulation of the problem and its Dantzig-Wolfe decomposition yielding an extended formulation are presented. A branch-and-price procedure is proposed to solve the extended formulation. The formulations are compared on a set of scheduling instances, considering partly realistic efficiency functions.

**Keywords:** energy-aware scheduling, efficiency functions, combinatorial optimization, mathematical programming, column generation

**Mathematics Subject Classification (2000):** MSC 49M27, MSC 65K05, MSC 90C27

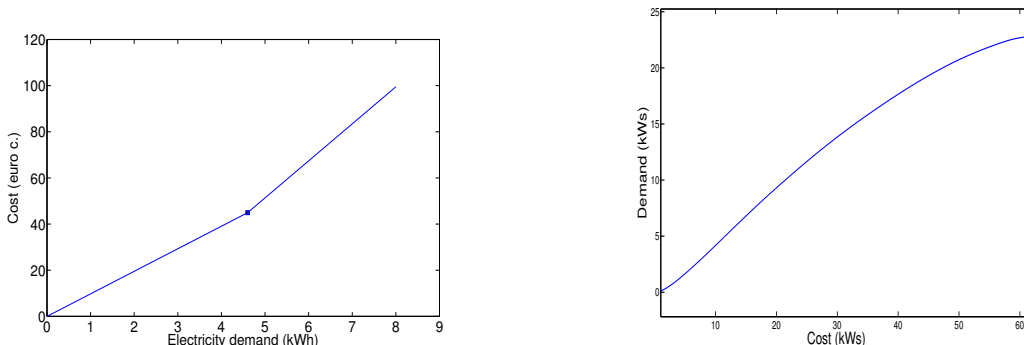
## 1 Introduction

Energy considerations are becoming paramount for real-world applications. Moreover optimization issues are at the core of many industrial systems. A rising combinatorial optimization challenge is then the integration of energy constraints in deterministic scheduling

and resource allocation models. We consider in this paper a scheduling problem where the objective is to minimize the total energy cost of a set of preemptive tasks subject to time windows. Under a discrete time model, the set of tasks that are in process at a given time period generates a demand for this period. We assume that there are several non-reversible energy sources that can be used to cover this demand, such that the part of the demand covered by a source is converted in cost via a non-linear function.

For a given source, the conversion function represents physical, technological, performance and/or billing characteristics. Contrary to reversible energy sources such as batteries and supercapacitors, non-reversible energy sources such as fuel cells, electric grid and combustion engines can produce energy but are unable to recover it (at least during the considered scheduling horizon).

We first give two practical examples of such non-linear demand/cost conversion function. The first example is the inclining block rate for electricity tariff, which is popular in many countries including the U.S. [24]. A customer having an energy consumption of  $Q$  kWh in a certain time period, will pay a bill  $B = P_1 \min(Q, K) + P_2 \max(Q - K, 0)$  on that period where  $K$  is a threshold up to which the rate is  $P_1$  while consumption above the threshold is billed with another rate  $P_2 > P_1$ . In Fig. 1(a), the expression of  $B$  in function of  $Q$  for a threshold of 4.6 kWh and prices  $P_1 = 9.767\text{€c}$  and  $P_2 = 16.0485\text{€c}$ . The function is piecewise linear. Note that the consumption is always measured on a period basis (classically 15 min, see the case study in the foundry industry in [14]). For other types of energy sources, obtaining the energy required by a set of tasks during a certain period can require a physical conversion subject to energy loss. For example, a Fuel Cell follows the scheme: *source*  $\rightarrow$  *converter*  $\rightarrow$  *usable energy* and the “energy cost” refers to the energy consumed from the source. In Fig. 1(b), the usable energy (demand) is expressed as a non-linear function of the consumed energy (cost) for a typical fuel cell used in hybrid electric vehicles [11, 12, 13].



(a) Conversion function for electricity tariff      (b) (Inverse) conversion function for a fuel cell

Figure 1: Non-linear energy costs.

This work is part of an ongoing effort aiming at solving explicitly and in an integrated fashion energy resource allocation problems and energy-consuming activity scheduling problems with such non-linear energy costs, yielding the following contributions.

First, together with a recent contribution [21], this paper contributes to the definition of a novel and efficient methodology for the integration of energy characteristics in combinatorial optimization problems via piecewise-linear lower and upper bounding of the non-linear energy conversion functions. Second, we show that the preemptive scheduling problem with piecewise-linear energy cost from multiple non-reversible sources, can be transformed into an equivalent single-source problem. Third, we exhibit a polynomially-solvable case of the problem and we show its NP-hardness in general. Fourth, we provide mixed-integer linear programming (MILP) formulations, among which an extended formulation. A branch-and-price procedure is proposed to solve the extended formulation. Computational experiments on a set of scheduling instances, considering partly realistic efficiency functions show the efficiency of our approach.

The remainder of the paper is organized as follows. Section 2 presents a relevant literature review, then the definition, a first MILP formulation and the complexity analysis of the problem are provided in Section 3. Section 4 proposes an extended formulation issued from the Dantzig-Wolfe decomposition of the first formulation. A branch-and-price procedure is described in Section 5. Finally, a computational evaluation of the propositions on scheduling instances with realistic efficiency functions is provided in Section 6, before stating the Conclusions.

## 2 Literature review and presentation of the piecewise-linear lower and upper bounding framework

Over the years the stakes of resource allocation problems and production scheduling applications have evolved towards a more responsible management of resources. In particular new models in production scheduling were considered, where the energy demand can be modulated, mainly to avoid peaks of electrical consumption. A state-of-the-art review of energy concerns in production scheduling and a method for minimizing total energy cost on a single machine scheduling problem can be found in [22]. In this paper, as well as in many other related studies, the energy costs fluctuate over time, depending on the market and on the processing state of the machine but are seldom subject to non-linear variations depending on demand. For more complex job-shop or resource-constrained project scheduling problems, the inclining block rate for electricity tariff yielding the piecewise-linear function with two pieces presented in Fig. 1(a) was considered in [14] for a scheduling problem in a foundry where a hybrid constraint-programming/MILP approach was proposed. The problem was further studied in [3] but although the non-linear electricity costs were considered, other non-linearities coming from energy modulation were ignored, as explained below. A metal is melted in induction furnaces and the electrical power of the furnaces can be adjusted at any time to avoid exceeding a maximum prescribed power limit. The electrical power can be seen as a continuous function of time to be determined, with the constraint that it must lie within minimum and maximum power levels that must be satisfied for the melting operation. More precisely, let  $P_i(t)$  be the power used by operation  $i$  at time  $t$ , then the considered energy constraints state that the total energy consumed equal to  $\int_{t_i}^{f_i} P_i(t)dt$ , where  $t_i$  and

$f_i$  are the start and end times of the melting operation  $i$ , has to be equal to the operation energy demand  $W_i$ , which amounts to ignore the efficiency functions of the furnace. For the same type of adjustable power problems, constraint propagation algorithms on the basis of a continuous setting were proposed in [2] but energy efficiency functions were also ignored. A related work has also been carried out by Kis [16] for a discretized time problem with variable-intensity tasks, who established polyhedral results and proposed a branch-and-cut procedure. Besides time discretization, the problem does not involve efficiency functions. Recently, the constraint propagation algorithms were extended to linear efficiency functions in [19].

Other studies consider explicitly non-linear energy constraints and costs. In scheduling, several authors considered different variants of the problem where the resource (not necessarily an energetic resource) usage may vary continuously and such that the amount of resource required by a task may vary over time. Węglarz *et al.* [23] call this issue the processing rate *vs.* resource amount model, as the processing rate of the activity is a continuous increasing function of the allotted resource amount at a time, which corresponds to the efficiency function (see also [4, 5]). Providing a general framework for solving mixed discrete/continuous problems with concave processing rate (efficiency) functions, Józefowska *et al.* [15] show that once the sequence of sets of tasks to be scheduled in parallel is determined, the continuous resource allocation can be made by a convex non-linear optimization problem. In the literature on parallel processor scheduling, the malleable task model also considers the possibility of changing the number of processors assigned to a task over time, with non-linear processing rate functions but this yields a non-linear relation between the number of used processors and task duration without consideration of energy costs [6]. Another family of studies concern non-linear energy consumption models for power-aware scheduling in computing devices [1]. This consists in the possibility to scale the speed  $s(t)$  of the processor at time  $t$  so that the power consumed by the processor is generally approximated by  $s(t)^\alpha$ , with  $\alpha$  a constant. The objective is to find a compromise between the total energy consumption of the processor given by  $\int_0^\infty s(t)^\alpha dt$  and the duration  $f_i - t_i$  of a task scheduled in an interval which must be sufficient to bring the required amount of work, i.e.  $W_i = \int_{t_i}^{f_i} s(t) dt$ . The major outcomes obtained in this area consist in complexity results for offline problems and competitiveness analysis of online algorithms. In contrast with these studies, we aim at rather proposing mathematical decomposition methods to solve (relatively) general problems.

Furthermore, these references are generally considering that resource/energy is assigned individually to each task with a modulation of the amount yielding variable task duration and energy consumption with possibly non-linear relations. Our study takes place in different setting where the schedule generates a global period-wise energy demand that has to be satisfied with different non-reversible energy sources with non-linear energy/cost conversion functions. This concerns for example the above-presented electricity tariff application in the foundry industry [14]. Another example is the global energy demand that is generated from the schedule of task consuming activities in a smart home where different energy sources are available. In [9], a time-indexed MINLP formulation is proposed to deal with energy-consuming task scheduling, energy source selection from renewable resources (e.g. solar energy) based production forecasts, dispatchable energy generators (e.g. fuel cells) using a linear approximation

of the production cost, storage devices with linear efficiency functions and the electric grid for which a quadratic cost is assumed. A daily scheduling problem is solved in different scenarios involving 8 energy-consuming non-preemptable tasks. In this work, we do not consider the storage devices but we aim at considering more general efficiency functions.

Previous works have focussed on the optimization of the allocation of multiple sources of energy to predefined demand curves in hybrid electric vehicles. In such real-world applications, non-linearities coming from energy efficiency functions, make the allocation problem difficult to solve. In this context, Guemri *et al.* [13] showed that flaws in existing modeling hypotheses led to significant gaps between state-of-the-art solutions and optimal ones. Global optimization-based heuristics have been designed to outperform the prior state-of-the-art [12, 13]. As an alternative to non-linear modeling yielding suboptimal solutions and important computation time, a new and efficient combinatorial modeling was proposed [11]. Although these studies considered the allocation of multiple energy sources and general non-linear efficiency functions, the scheduling of energy consuming activities, that would allow more flexibility for energy management was not considered.

A few promising mathematical programming-based approaches on similar problems can be found [7, 8], either based on MINLP or transformations into approximate MILP. We now present an alternative approach based on piecewise-linear lower and upper bounding, rather than approximating, the non-linear efficiency functions (see also [17]). Successfully applied by Ngueveu *et al.* [21] as a proof of concept for the solution of a water production optimization problem, the solution framework is decomposed into two stages: (i) the bounding of the non-linear energy efficiency function, then (ii) the reformulation of the problem, which originally is a mixed-integer non-linear problem (MINLP), into two mixed integer linear problems (MILP) using the pair of bounding functions already mentioned above. The piecewise-linear bounding of a function  $f$  of  $m$  variables within a tolerance value  $\epsilon$  consists in identifying two piecewise-linear functions denoted  $\bar{f}^\epsilon$  and  $\underline{f}^\epsilon$  that verify equations (1) to (3). The two MILPs, denoted  $\overline{\text{MILP}}$  and  $\underline{\text{MILP}}$  respectively, are obtained by substituting  $f$  with  $\bar{f}^\epsilon$  and  $\underline{f}^\epsilon$ , respectively.

$$\underline{f}^\epsilon(x) \leq f(x) \leq \bar{f}^\epsilon(x), \quad \forall x \in \mathbb{R}^m \quad (1)$$

$$f(x) - \underline{f}^\epsilon(x) \leq \epsilon f(x), \quad \forall x \in \mathbb{R}^m \quad (2)$$

$$\bar{f}^\epsilon(x) - f(x) \leq \epsilon f(x), \quad \forall x \in \mathbb{R}^m \quad (3)$$

Solving a  $\overline{\text{MILP}}$  generates solutions that are feasible for the original MINLP, and that have a total cost less than  $\epsilon\%$  higher than the optimal solution cost. Solving a  $\underline{\text{MILP}}$  generates solutions that may not be feasible for the original MINLP, but whose total cost is less than  $\epsilon\%$  lower than the optimal solution cost and can help proving the optimality of a solution. However, if the non-linear function appears only in the objective function, as it will be in our case, then any solution of  $\underline{\text{MILP}}$  is also feasible for the original problem. Obtaining a lower bound and an upper bound with a controlled gap is an advantage of this approach against a single piecewise-linear approximation. Note that both problems share the exact same structure and only differ in terms of the numerical data of their respective piecewise-linear functions. Therefore, a single dedicated resolution method needs to be applied to solve both problems. Furthermore, recall that we approximate a non-linear energy demand/cost

conversion function, which either gives the energy cost in function of the energy demand such as in the billing application (Fig. 1(a)) or the energy demand in function of the energy cost such as for the fuel cell efficiency function (Fig. 1(b)). It follows that to obtain an integrated model of these two cases, inversion of some source functions can be necessary, which may be not an easy task for some non-linear functions. This is an argument in favor of the piecewise-linear bounding scheme, rather than a direct usage of the non-linear function. As we will see when studying the structure of our scheduling problem, there is another advantage of this framework.

In Ngueveu *et al.* [21], the resulting  $\overline{\text{MILP}}$  and  $\underline{\text{MILP}}$  were solved with a MILP solver. The cases where the resulting problems cannot be efficiently solved with a black box solver remain to be studied.

### 3 Problem statement and complexity analysis

The problem definition requires a discrete time horizon  $\mathcal{T} = \{1, \dots, |\mathcal{T}|\}$ , a set of preemptive activities  $\mathcal{A} = \{1, \dots, n\}$ , each activity  $i$  having a duration  $p_i$ , an energy demand per time unit of  $b_i$  and a time window that starts at  $r_i$  and ends at  $d_i$ . A constant term  $a_{it}$  with  $i \in \mathcal{A}$  and  $t \in \mathcal{T}$  is equal to 1 if  $t \in [r_i, d_i[$  and 0 otherwise. The set of non-reversible energy sources available is  $\mathcal{S} = \{1, \dots, m\}$ . Remind that a non-reversible source is only able to produce energy, but not to recover it. Let  $\rho^s$  denote the (non-linear) efficiency function for source  $s$ , i.e. a cost or energy consumption of  $x$  produces an amount of usable energy of  $\rho^s(x)$ . In other words,  $(\rho^s)^{-1}(x)$  is required from  $s$  to satisfy a demand of  $x$ . Since all  $s \in \mathcal{S}$  are non-reversible, then  $\rho^s(x) = 0, \forall x < 0, \forall s \in \mathcal{S}$ . The objective is to schedule the tasks so as to minimize the total energy consumption from all energy sources. Let (P) denote the resulting problem. It can be expressed using binary decision variables  $x_{it}$ , equal to 1 iff activity  $i$  is being executed at time period  $t$ ; and continuous positive decision variables  $w_{st}$ , equal to the amount of energy demand covered by energy source  $s$  at instant  $t$ . A MILP formulation of (P) is:

$$\text{(P)} \quad \min \quad \sum_{s \in \mathcal{S}} \sum_{t \in \mathcal{T}} (\rho^s)^{-1}(w_{st}) \quad (4)$$

s.t.

$$\sum_{t \in \mathcal{T}} a_{it} x_{it} \geq p_i, \quad \forall i \in \mathcal{A} \quad (5)$$

$$\sum_{s \in \mathcal{S}} w_{st} = \sum_{i \in \mathcal{A}} b_i x_{it}, \quad \forall t \in \mathcal{T} \quad (6)$$

$$x_{it} \in \{0, 1\}, \quad \forall i \in \mathcal{A}, t \in \mathcal{T} \quad (7)$$

$$w_{st} \geq 0, \quad \forall s \in \mathcal{S}, t \in \mathcal{T} \quad (8)$$

The objective function (4) aims at minimizing the total energy cost. Constraints (5) set each task  $i$  in process during at least  $p_i$  time periods. Constraints (6) ask to cover the energy

	T1	T2	T3
release date	0	2	1
due date	7	10	6
duration	6	4	3
energy demand	2	3	2

Table 1: Scheduling instance.

demand of each task by the available sources. Constraints (7)–(8) give the domains of the variables. Because of the piecewise-linear bounding-based solution methodology, all efficiency functions  $\rho^s$  (and thus  $(\rho^s)^{-1}$ ) can be assumed to be piecewise linear. Such functions can be modeled either with dedicated piecewise-linear functions in black-box solvers or by adding additional binary variables aiming at identifying which sector of the piecewise-linear function has to be activated at each time period. These additional variables may weaken the linear relaxation of the resulting model. Still, the mathematical model has a pseudo-polynomial (depending on the size of the time windows) number of variables and constraints and therefore can be loaded into any MILP solver for a reasonable size of the time horizon.

A scheduling instance and efficiency function are given in Table 1 and Fig. 2, respectively. A feasible solution and the corresponding energy cost profile are shown in Fig. 3(a). At time 5 for example, the tasks being executed are T1 and T2. As a consequence the total instantaneous energy demand is  $b_1 + b_2 = 2 + 3 = 5$ , which would lead to an energy cost of 4 according to the efficiency function curve of the energy source. The optimal solution is provided in Fig. 3(b). It possesses a higher makespan than the previous solution. However, any solution with a higher makespan also has a higher energy cost, which shows that minimizing the energy cost does not necessarily translate into minimizing or maximizing the makespan.

The structural analysis of (P) led to findings expressed with Theorems 3.1, 3.2, 3.3, and 3.4. The first result shows that, due to the discrete nature of the problem, replacing general non-linear functions by piecewise-linear approximations can be made without loss of generality.

**Theorem 3.1** *For any problem (P) with non-linear functions  $(\rho^s)^{-1}$ , there exist piecewise-linear approximations  $(\bar{\rho}^s)^{-1}$  of functions  $(\rho^s)^{-1}$  such that problem  $(\bar{P})$  obtained by replacing  $(\rho^s)^{-1}$  by  $(\bar{\rho}^s)^{-1}$  for all  $s$  has the same optimal solution as (P).*

**Proof** As there is a finite number  $n$  of tasks, there are at most  $2^n$  different values for the possible energy demand. In addition, if all  $b_i$  are integer, there are at most  $\max(2^n, \sum_{i \in \mathcal{A}} b_i)$  different values of the demand. Let  $x_q$  denote the  $q^{\text{th}}$  different value of the demand in increasing order. For each source  $s$  a continuous piecewise linear function can be built by connecting consecutive points of the series  $(x_q, y_q)$  where  $y_q = (\rho^s)^{-1}(x_q)$ . Note that this yields a purely combinatorial model where a binary variable is necessary for each possible demand value. ■

Theorems 3.2 states that there is in fact no need to explicitly consider multiple non-reversible sources.



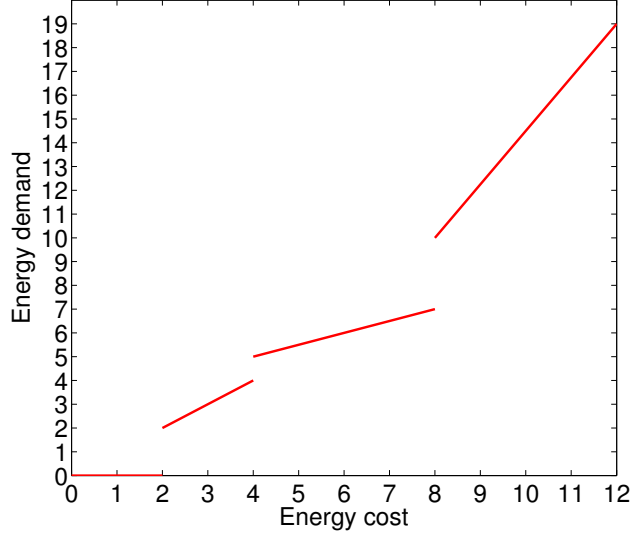


Figure 2: Efficiency function.

**Theorem 3.2** *For any problem (P) with multiple non-reversible energy sources, there is an equivalent single non-reversible energy source problem.*

**Proof** The proof is based on the fact that (P) can be reformulated as:

$$(P') \quad \min \sum_{t \in \mathcal{T}} (\rho')^{-1} \left( \sum_{i \in \mathcal{A}} b_i x_{it} \right) \quad (9)$$

s.t.

$$\sum_{t \in \mathcal{T}} a_{it} x_{it} \geq p_i, \quad \forall i \in \mathcal{A} \quad (10)$$

$$x_{it} \in \{0, 1\}, \quad \forall i \in \mathcal{A}, t \in \mathcal{T} \quad (11)$$

where  $\forall x \in \mathbb{R}$ ,  $\rho'(x)^{-1}$  can be defined as the solution cost of the problem:

$$(\text{Cost}_x) \quad \min \left( \sum_{s \in \mathcal{S}} (\rho^s)^{-1}(w_s) \right) \quad (12)$$

s.t.

$$\sum_{s \in \mathcal{S}} w_s = x \quad (13)$$

$$w_s \in \mathbb{R}^+, \quad \forall s \in \mathcal{S} \quad (14)$$

Therefore, the efficiency function of an equivalent single non-reversible energy source can be obtained from an optimal pre-aggregation of the  $|\mathcal{S}|$  efficiency functions into a single one  $\rho'(x)$ . ■

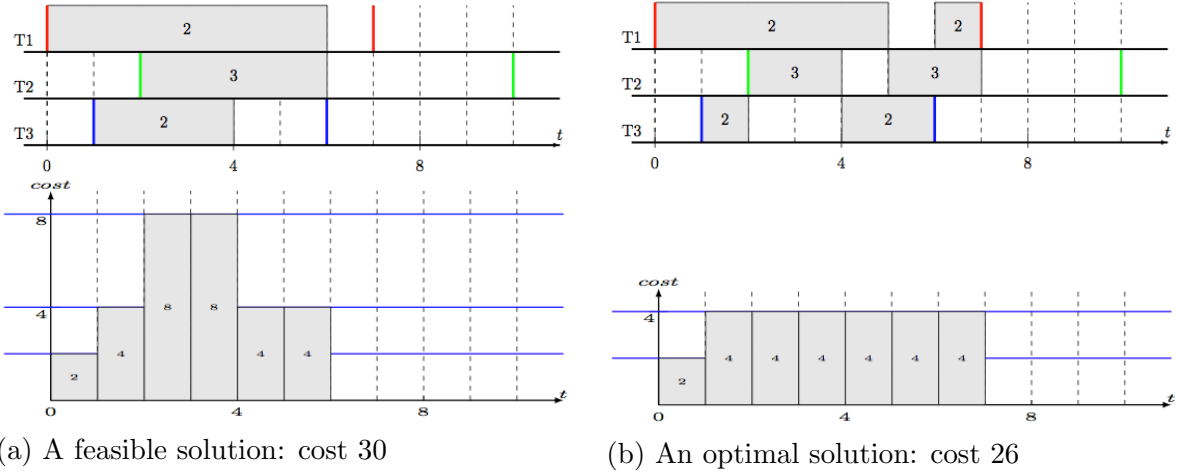


Figure 3: Illustrative example.

Now, considering the single source problem, the complexity of a special case and of the general problem can be established.

**Theorem 3.3** *The single non-reversible energy source problem with an efficiency function that is linear except in 0 can be solved exactly by a greedy algorithm in  $O(n^2)$  time.*

**Proof** Consider that  $\rho^{-1}(x) = ax + b$  if  $x > 0$  and  $\rho^{-1}(0) = 0$ . Let  $\delta_t = 1$  if there is an activity scheduled at  $t$ , assuming that all demands  $b_i$  are strictly positive. Given a solution  $(x_{it})$  of the problem we have a total cost of  $\sum_{t \in \mathcal{T}, \delta_t=1} (a(\sum_{i \in \mathcal{A}} b_i x_{it}) + b) = a(\sum_{i \in \mathcal{A}} b_i)(\sum_{t \in \mathcal{T}} x_{it}) + b \sum_{t \in \mathcal{T}} \delta_t = a \sum_{i \in \mathcal{A}} b_i + b \sum_{t \in \mathcal{T}} \delta_t$ . This amounts to minimizing the number of idle time periods for a preemptive scheduling problem with release dates, deadlines, and jobs of arbitrary durations. This problem is known as the preemptive busy time model, for which an  $O(n^2)$  exact greedy algorithm has been proposed in [10]. ■

**Theorem 3.4** *The single non-reversible energy source problem is NP-hard even for a concave linear efficiency function with two pieces that is part of the input.*

**Proof** The proof is based on the fact that any decision instance of the discrete bin-packing problem can be transformed into a particular decision instance of (P'). The discrete bin-packing problem can be stated as follows. Given a list of  $n$  items of size  $b_i, \forall i \in 1..n$ , is it possible to assign each item to a bin such that the sum of the item sizes in a bin does not exceed  $C$  and the number of bins used does not exceed  $B$ ? This is equivalent to stating the following ( $\tilde{P}'$ ) problem: Given a list of  $n$  unit-time activities, each activity  $i$  having an energy demand  $b_i$ , a time window  $[0, B]$  and given an energy source of inverse efficiency function (15), does it exist a solution of ( $\tilde{P}'$ ) that has a cost not exceeding  $B$ ?

$$(\tilde{\rho}')^{-1}(x) = \begin{cases} x/C & \text{if } 0 \leq x \leq C \\ (B-1)x + 1 - (B-1)C & \text{if } x \geq C \end{cases} \quad (15)$$

Indeed, since the total cost can be expressed as  $\sum_{t=0}^B (\tilde{\rho}')^{-1}(\sum_{i=1}^n b_i * x_{it})$  where  $x_{it}$  is a binary variable equal to 1 if and only if activity  $i$  is scheduled at time  $t$ , the equivalence between the decision variant of the discrete bin-packing problem and  $(\tilde{P}')$  can be easily shown. If the demand stays in time window  $[0, B]$  below threshold  $C$  then the cost is below  $B$  as  $x/C \leq 1$ ,  $\forall x \leq C$ . Otherwise, as soon as the demand is larger than or equal to  $C + 1$  the cost is larger than  $B$ . Therefore the decision variant of  $(P')$  generalizes the decision variant of the discrete bin-packing. As a consequence, the decision variant of  $(P')$  is NP-complete, and therefore  $(P')$  is NP-hard. We have proved NP-hardness for a function  $(\tilde{\rho}')^{-1}$ , which is strictly increasing, convex and inversible. Hence  $(\tilde{\rho}')$  is concave.  $\blacksquare$

## 4 Extended formulation and Dantzig-Wolfe Decomposition

Because of Theorem 3.2, the remainder of the paper focusses on solving efficiently problem  $(P')$ , which considers a single non-reversible energy source having a piecewise-linear efficiency function. Consequently, variables  $w_{st}$  can be disregarded from Model (4)–(8) and the efficiency function of the unique energy source will be simply denoted  $\rho$  and its reverse  $\rho^{-1}$ .

### 4.1 Extended formulation

The presence of piecewise-linear function and the resulting additional variables in the mathematical model can result into a weaker linear relaxation. To counter that, we propose a set partitioning-based reformulation with a purely linear objective-function. This formulation is based on the identification of sets of feasible subsets. A *feasible subset* [18] is a set of activities that can be in progress simultaneously without exceeding any resource availability, and that are not pairwise linked by a precedence constraint. Let  $l \in \mathcal{L}$  be a set of activities that can be processed simultaneously according to the constraints. The set of activities belonging to  $l$  is denoted  $\mathcal{A}_l (\subseteq \mathcal{A})$ . Each  $l \in \mathcal{L}$  is assigned an energy demand  $b_l = \sum_{i \in \mathcal{A}_l} b_i$ , an energy cost  $c_l = \rho^{-1}(b_l)$ , a release date  $r_l = \max_{i \in \mathcal{A}_l} r_i$ , and a due date  $d_l = \min_{i \in \mathcal{A}_l} d_i$ . Finally, the set of activity sets executable at instant  $t$  is denoted  $\mathcal{L}_t$ , and is composed of sets  $l$  that verify  $r_l \leq t < d_l$ . Binary variables  $y_{lt}$  are defined, equal to 1 if activity set  $l$  is chosen at time  $t$  and 0 otherwise.

The resulting formulation (EF) of problem  $(P)$  is expressed by equations (16)–(21) and requires the introduction of constant terms  $a_{il}$  equal to 1 if  $i \in \mathcal{A}_l$  and 0 otherwise. Its validity is stated by Theorem 4.1.

$$(EF) \quad \min \sum_{t \in \mathcal{T}} \sum_{l \in \mathcal{L}_t} c_l y_{lt} \quad (16)$$

s.t.

$$x_{it} - \sum_{l \in L_t} a_{il} y_{lt} = 0, \quad \forall i \in A, t \in \mathcal{T} \quad (17)$$

$$\sum_{l \in L} \sum_{t=R_l}^{D_l-1} a_{il} y_{lt} \geq p_i, \quad \forall i \in A \quad (18)$$

$$\sum_{l \in L_t \cup \emptyset} y_{lt} = 1, \quad \forall t \in \mathcal{T} \quad (19)$$

$$x_{it} \in \{0, 1\}, \quad \forall i \in A, t \in \mathcal{T} \quad (20)$$

$$y_{lt} \geq 0, \quad \forall t \in \mathcal{T}, l \in L_t \cup \emptyset \quad (21)$$

The objective function (16) to be minimized is the sum over time of the energy cost of each activity set being used at each instant. Constraints (17) ensure the coherence between variables  $x$  and  $y$ . Constraints (18) ensure that the duration of each activity is satisfied. Constraints (19) state that one activity set is active at each instant, empty set included. Constraints (20) and (21) specify the domain of each variable.

**Theorem 4.1** *An optimal solution  $(x, y)$  of MILP (EF) provides an optimal solution for problem (P).*

**Proof** Proving that an optimal solution  $(x, y)$  of MILP (EF) verifying  $y \in [0, 1]^{|L||\mathcal{T}|}$  is also an optimal solution for problem (P) is straightforward. It remains to be shown, however, that relaxing constraints  $y \in [0, 1]^{|L||\mathcal{T}|}$  into constraints (21) still yields an optimal solution for the original problem. The proof of validity of Theorem 4.1 consists in proving that although decision variables  $y_{lt}$  are continuous, the model ensures that their values are always binary in feasible solutions, equal to 1 if activity set  $l$  is being executed at time  $t$ , and 0 otherwise.

Let  $\tilde{S}$  be a feasible solution of MILP (EF):  $\tilde{x}_{it} \in [0, 1]^{|A||\mathcal{T}|}$  and  $\tilde{y}_{lt} \in \mathbb{R}^{|L||\mathcal{T}|}$ . Given an instant  $t^* \in \mathcal{T}$ , let us denote:

- $L^{>0}$  the subset of activity sets used at instant  $t^*$ . In other words  $L^{>0} = \{l : \tilde{y}_{lt^*} > 0, \forall l \in L_{t^*}\}$ .
- $A^{L^{>0}}$  the subset of activities that appear in at least one set  $l$  of  $L^{>0}$ . In other words  $A^{L^{>0}} = \{i : \exists l \in L^{>0} \text{ that verifies } a_{il} = 1\}$ .
- $L^{>0}(i)$  the subset of activity sets from  $L^{>0}$  that contain activity  $i$ .

Since  $\tilde{y}_{lt^*} > 0, \forall l \in L^{>0}$  (by definition) and  $\sum_{l \in L^{>0}} \tilde{y}_{lt^*} = 1$  (from constraints (19)), we deduce proposition 4.2 which is necessary to continue the proof.

**Proposition 4.2**  $\forall \tilde{L} \subseteq L^{>0}$ , if  $\sum_{l \in \tilde{L}} \tilde{y}_{lt^*} = 1$  then  $\tilde{L} = L^{>0}$ .

Since by definition  $\forall i \in A^{L^{>0}}, \tilde{x}_{it^*} = 1$ , we can deduce from constraints (17) that:

$$\sum_{l \in L^{>0}(i)} \tilde{y}_{lt^*} = 1 \quad (22)$$

where  $L^{>0}(i) \subseteq L^{>0}$ .

Finally, combining Proposition 4.2 and constraint (22) we can deduce:

$$L^{>0}(i) = L^{>0}, \forall i \in A^{L^{>0}}. \quad (23)$$

Constraints (23) imply that:

- either  $|L^{>0}| = 1$  and therefore all  $\tilde{y}_{lt^*}$  are integer
- or  $|L^{>0}| > 1$  but all activity sets from  $L^{>0}$  are identical.

The latter assertion is impossible because the model does not authorize multiple identical columns. Therefore  $\tilde{y}_{lt^*}$  are integer. Combined with constraints (19), this proves that  $\tilde{y} \in [0, 1]^{|L||T|}$ . Thus, solving (EF) produces optimal solutions for the original problem (P).  $\blacksquare$

Formulation (EF) has a polynomial number of constraints, a polynomial number of variables  $x_{it}$ , but an exponential number of variables  $y_{lt}$  and therefore its linear relaxation can be solved with column generation. The linear relaxation of (EF), which serves as the master problem (MP) of the Dantzig-Wolfe decomposition is given by equations (24)–(29). Note that the empty set does not belong to  $\mathcal{L}$ , therefore if no task is being executed then the left-hand-side of equations (27) is equal to 0.

$$(MP) \quad \min \sum_{t \in T} \sum_{l \in \bar{L}_t} c_l y_{lt} \quad (24)$$

s.t.

$$x_{it} - \sum_{l \in \bar{L}_t} a_{il} y_{lt} = 0, \quad \forall i \in A, t \in T \quad (25)$$

$$\sum_{l \in \mathcal{L}} \sum_{t=R_l}^{D_l-1} a_{il} y_{lt} \geq p_i, \quad \forall i \in A \quad (26)$$

$$\sum_{l \in \mathcal{L}_t} y_{lt} \leq 1, \quad \forall t \in T \quad (27)$$

$$0 \leq x_{it} \leq 1 \quad \forall i \in A, t \in T \quad (28)$$

$$y_{lt} \geq 0 \quad \forall t \in T, l \in \mathcal{L}_t \quad (29)$$

At each iteration  $\alpha$  of a column generation we solve a restricted master problem (RMP $^\alpha$ ) obtained by restricting  $\mathcal{L}$  to a subset of activity sets  $\mathcal{L}^\alpha \subseteq \mathcal{L}$ , then try to generate one or

several activity sets of negative reduced cost using the dual variables values of the solution found. If no such activity set can be generated, then the current solution is optimal for the current linear relaxation (MP). Otherwise, the best sets with negative reduced cost are added to  $\mathcal{L}^\alpha$  to obtain  $\mathcal{L}^{\alpha+1}$  and the new master problem (RMP $^{\alpha+1}$ ) is solved.

Let  $w_{it}$ ,  $u_i$ ,  $v_t$  and  $z_{it}$  be the dual variables associated with constraints (25), (26), (27) and (28), respectively. The resulting dual of (MP) is:

$$\text{(DMP)} \quad \max \sum_{i \in A} p_i u_i - \sum_{t \in T} v_t - \sum_{i \in A} \sum_{t \in T} z_{it} \quad (30)$$

s.t.

$$\sum_{i \in A} a_{il}(u_i - w_{it}) - v_t \leq c_l, \quad \forall t \in T, l \in \tilde{L}_t \quad (31)$$

$$w_{it} - z_{it} \leq 0, \quad \forall i \in A, t \in T \quad (32)$$

$$w_{it} \in \mathbb{R}, \quad \forall i \in A, t \in T \quad (33)$$

$$u_i \geq 0, \quad \forall i \in A \quad (34)$$

$$v_t \geq 0, \quad \forall t \in T \quad (35)$$

$$z_{it} \geq 0, \quad \forall i \in A, t \in T \quad (36)$$

Any column  $y_{lt}$  missing from (RMP $^\alpha$ ) corresponds to a constraint (31) missing from its dual. Therefore, identifying columns of negative reduced cost is equivalent to identifying missing violated inequalities (31). As a consequence, the subproblem resolution (also called pricing procedure) consists in building an activity set  $\tilde{l}$  and identifying an instant-time  $\tilde{t}$  that maximizes the difference between the left-hand-side and the right-hand-side of inequality (31). If the difference is strictly positive, then the corresponding column  $y_{\tilde{l}, \tilde{t}}$  is introduced into the model. Otherwise it can be disregarded and the current optimal solution of (RMP $^\alpha$ ) is the optimal solution of the current (RMP).

## 4.2 Subproblem SP1: With variable $t$

Generating the columns of negative reduced cost consists in building the pair  $(\tilde{l}, \tilde{t})$  that maximize the violation of constraints (31). The corresponding subproblem can be modeled with binary decision variables  $\alpha_i$  equal to 1 iff activity  $i$  is selected to compose the set  $\tilde{l}$ ,  $\beta_t$  equal to 1 iff time  $t$  is selected ( $t = \tilde{t}$ ), and finally variables  $\gamma_{it}$  equal to 1 iff both activity  $i$  and time  $t$  have been selected. The resulting subproblem denoted (SP1) takes as input data the current dual variables values  $u_i$ ,  $v_t$  and  $w_{it}$ .

$$\text{(SP1)} \quad \max \sum_{i \in A} u_i \alpha_i - \sum_{i \in A} \sum_{t \in T} w_{it} \gamma_{it} - \sum_{t \in T} v_t \beta_t - \rho^{-1} \left( \sum_{i \in A} b_i \alpha_i \right) \quad (37)$$

s.t.

$$\alpha_i + \beta_t \leq 1, \quad \forall i \in A, t \leq r_i - 1 \text{ or } t \geq d_i \quad (38)$$

$$\sum_{t \in T} \beta_t = 1 \quad (39)$$

$$\gamma_{it} - 0.5\alpha_i - 0.5\beta_t \leq 0, \quad \forall i \in A, t \in T \quad (40)$$

$$\gamma_{it} - \alpha_i - \beta_t \geq -1, \quad \forall i \in A, t \in T \quad (41)$$

$$\alpha_i \in [0, 1], \quad \forall i \in A \quad (42)$$

$$\beta_t \in [0, 1], \quad \forall t \in T \quad (43)$$

$$\gamma_{it} \in [0, 1], \quad \forall i \in A, \forall t \in T \quad (44)$$

The objective function (37) is piecewise-linear and maximizes the violation of constraints (31). Constraints (38) forbid the choice of an instant  $t$  outside of the time window of the activities selected. Constraint (39) enforces the choice of a single instant. Constraints (40) and (41) ensure the coherence between the values of  $\gamma_{it}$ ,  $\alpha_i$  and  $\beta_t$ . Finally, constraints (42)–(44) specify the domain of the decision variables.

The subproblem (SP1) has a piecewise-linear objective function, a polynomial number of constraints and variables. Thus it can be solved with any MILP black box solver. A column generation algorithm can therefore be obtained where the pricing procedure consists in solving (SP1) with a MILP black box solver and then adding to the master problem the column  $y_{\tilde{l}, \tilde{t}}$  corresponding to the optimal pair  $(\tilde{l}, \tilde{t})$  if the optimal cost of (SP1) was strictly positive.

Preliminary results showed that the resulting algorithm required high computing times spent mostly in the resolution of (SP1). To counter that, two additional policies for the generation and insertion of columns into the master problem were designed and implemented as explained in Section 5. We also proposed three other subproblem formulations (SP2 $_{\tilde{t}}$ ), (SP3 $_{\tilde{s}}$ ) and (SP4 $_{\tilde{t}, \tilde{s}}$ ) aiming at reducing the time necessary to generate new columns.

### 4.3 Subproblem SP2 $_{\tilde{t}}$ : With fixed $t$

Based on the idea that predefining  $t$  leads to a subproblem easier to solve, SP2 inputs the current dual variables values, but also a predefined time  $\tilde{t}$ , and then outputs the best activity set  $\tilde{l}$  obtained by solving model (45)–(47).

$$(SP2_{\tilde{t}}) \quad \max \quad \sum_{i \in A} \alpha_i (u_i - w_{i\tilde{t}}) - v_{\tilde{t}} - \rho^{-1} \left( \sum_{i \in A} \alpha_i b_i \right) \quad (45)$$

s.t.

$$\alpha_i \leq a_{i\tilde{t}}, \quad \forall i \in A \quad (46)$$

$$\alpha_i \in [0, 1], \quad \forall i \in A \quad (47)$$

The objective function (45) is piecewise-linear and maximizes the violation of constraint (31). Note that  $v_{\tilde{t}}$  is a constant term. Constraints (46) ensure that an activity can only be

selected if it is executable during the time  $\tilde{t}$ . Finally, constraints (47) enforce the binarity of variables  $\alpha_i$ . This formulation contains a polynomial number of variables and constraints and is therefore suitable for MILP solvers.

Using SP2 in the column generation means that instead of solving SP1 once during the pricing procedure to obtain both  $\tilde{t}$  and  $\tilde{l}$ , multiple subproblems (SP2 $_{\tilde{t}}$ ) are solved with different predefined values of  $\tilde{t}$  and for each subproblem (SP2 $_{\tilde{t}}$ ) the activity set that maximizes the violation is generated. However each (SP2 $_{\tilde{t}}$ ) contains much less binary variables and constraints than (SP1) and therefore requires a smaller computing time. The different variants and parameter settings implemented are presented in Section 5.

#### 4.4 Subproblem SP3 $_{\tilde{s}}$ : Efficiency-function-based

This subproblem results from the observation that each sector of the efficiency function is linear, therefore, subproblems easier to solve might be obtained by imposing the specific sector in use. In addition to the current dual variables values, (SP3 $_{\tilde{s}}$ ) inputs a predefined sector  $\tilde{s} \in S$ . The lower limit is  $x_{\tilde{s}}^{\min}$  and the higher limit is  $x_{\tilde{s}}^{\max}$ . Within these limits the cost function can be expressed with the expression  $\tilde{a}_{\tilde{s}}x + \tilde{b}_{\tilde{s}}$  where  $\tilde{a}_{\tilde{s}}$  is the slope of sector  $\tilde{s}$  and  $\tilde{b}_{\tilde{s}}$  is the y-intercept of sector  $\tilde{s}$ .

The best activity set  $\tilde{l}$  and associated time  $\tilde{t}$  are obtained by solving model (SP3 $_{\tilde{s}}$ ).

$$(SP3_{\tilde{s}}) \quad \max \sum_{i \in A} \alpha_i (u_i - \tilde{a}_{\tilde{s}} b_i) - \sum_{i \in A} \sum_{t \in T} \gamma_{it} w_{it} - \sum_{t \in T} \beta_t v_t - \tilde{b}_{\tilde{s}} \quad (48)$$

s.t.

$$x_s^{\min} \leq \left( \sum_{i \in S} b_i \alpha_i \right) \leq x_s^{\max} \quad (49)$$

$$\alpha_i + \beta_t \leq 1, \quad \forall i \in A, t \leq r_i - 1 \text{ or } t \geq d_i \quad (50)$$

$$\sum_{t \in T} \beta_t = 1 \quad (51)$$

$$\gamma_{it} - 0.5\alpha_i - 0.5\beta_t \leq 0, \quad \forall i \in A, t \in T \quad (52)$$

$$\gamma_{it} - \alpha_i - \beta_t \geq -1, \quad \forall i \in A, t \in T \quad (53)$$

$$\alpha_i \in \{0, 1\}, \quad \forall i \in A \quad (54)$$

$$\beta_t \in \{0, 1\}, \quad \forall t \in T \quad (55)$$

$$\gamma_{it} \in \{0, 1\}, \quad \forall i \in A, t \in T \quad (56)$$

The objective function (48) maximizes the violation of constraint (31). Constraint (49) ensures that the total energy demand of the activity set is within the limits of the corresponding sector  $\tilde{s}$ . Constraints (50) forbid the choice of an instant  $t$  outside of the time window of the activity selected. Constraint (51) enforces the choice of a single instant. Constraints (52) and (53) ensure the coherence between the values of  $\gamma_{it}$ ,  $\alpha_i$  and  $\beta_t$ . Finally, constraints (54)–(56) specify the domain of the decision variables. This formulation has a fully linear



objective function, a polynomial number of variables and constraints. It is therefore suitable for a black box MILP solver.

#### 4.5 Subproblem $\text{SP4}_{\tilde{t}, \tilde{s}}$ : With fixed $t$ and $s$

This subproblem results from the combination of  $(\text{SP2}_{\tilde{t}})$  with  $(\text{SP3}_{\tilde{s}})$ . In addition to the current dual variables values,  $(\text{SP4}_{\tilde{t}, \tilde{s}})$  inputs a predefined sector  $\tilde{s} \in S$  and a predefined time  $\tilde{t}$ . The slope is  $\tilde{a}_{\tilde{s}}$  and the y-intercept is  $\tilde{b}_{\tilde{s}}$ . Within the limits  $[x_{\tilde{s}}^{\min}, x_{\tilde{s}}^{\max}]$  the cost function can be expressed with the expression  $\tilde{a}_{\tilde{s}}x + \tilde{b}_{\tilde{s}}$ .

The best activity set  $\tilde{l}$  is obtained by solving model (57)–(60).

$$(\text{SP4}_{\tilde{t}, \tilde{s}}) \quad \max\left(\sum_{i \in A} \alpha_i (u_i - w_{i\tilde{t}} - \tilde{a}_{\tilde{s}} b_i)\right) - v_{\tilde{t}} - \tilde{b}_{\tilde{s}} \quad (57)$$

s.t.

$$x_{\tilde{s}}^{\min} \leq \left(\sum_{i \in S} b_i \alpha_i\right) \leq x_{\tilde{s}}^{\max} \quad (58)$$

$$\alpha_i \leq a_{i\tilde{t}}, \quad \forall i \in A \quad (59)$$

$$\alpha_i \in \{0, 1\}, \quad \forall i \in A \quad (60)$$

The objective function (57) maximizes the violation of constraint (31). Constraint (58) ensures that the total energy demand of the activity set is within the limits of the corresponding sector  $\tilde{s}$ . Constraints (59) forbid the selection of activities with a time window outside of time  $\tilde{t}$ . Finally constraints (60) impose the domain of decision variables  $\alpha_i$ . This formulation has a fully linear objective function, a polynomial number of variables and constraints. It is therefore suitable for a black box MILP solver.

Note that the terms  $v_{\tilde{t}}$  and  $\tilde{b}_{\tilde{s}}$  are constant for each subproblem. Therefore the resulting problem is equivalent to a classical binary knapsack problem with an additional constraint imposing a minimum total weight  $x_{\tilde{s}}^{\min}$ . Each object has a profit  $u_i - w_{i\tilde{t}} - \tilde{a}_{\tilde{s}} b_i$  and a weight  $b_i$ . The capacity of the knapsack is  $x_{\tilde{s}}^{\max}$ .

## 5 Branch & Price

Solving (MP) requires a column generation, therefore solving (EF) requires a branch & price to ensure the integrity of binary variables  $x_{it}$ . Note that it is possible to eliminate variables  $x_{it}$  and just impose variables  $y_{it}$  to be binary, but doing so would force the resulting Branch & Price to branch on the same variables  $y_{it}$  that are being generated, which means that the resulting pricing procedures (or subproblems) would vary in function of the node of the Branch & Price. Instead, introducing binary variables  $x_{it}$ , despite some redundancy, allows variables  $y_{it}$  to be continuous and therefore ensures that the optimal solution can be obtained (see Theorem 4.1) without ever branching on variables  $y_{it}$ . As a consequence, the

same subproblem must be solved at each node of the Branch & Price, which means that only one pricing procedure needs to be designed and implemented, to be chosen between (SP1), (SP2 $_{\tilde{t}}$ ), (SP3 $_{\tilde{s}}$ ), and (SP4 $_{\tilde{t},\tilde{s}}$ ).

Different variants of column generation or Branch & Price algorithms can be obtained depending on the values of parameters a, b, c and d set up as follows:

- a) type of subproblems solved in the pricing procedure
  - 1 (SP1)
  - 2 (SP2 $_{\tilde{t}}$ )
  - 3 (SP3 $_{\tilde{s}}$ )
  - 4 (SP4 $_{\tilde{t},\tilde{s}}$ )
- b) column adding policy
  - 1 add the activity set  $\tilde{l}$  at instant  $\tilde{t}$  only
  - 2 add the activity set  $\tilde{l}$  at all feasible instants  $t \in [r_{\tilde{t}}, d_{\tilde{t}} - 1]$
  - 3 add the activity set  $\tilde{l}$  only at feasible instants where it has negative reduced cost
- c) multiple sets? (only available for a=1 or a=4)
  - 0 stop the pricing procedure as soon as one column is added in the master problem
  - 1 try to generate columns for each instant  $t \in \mathcal{T}$  before exiting the pricing procedure
- d) time increment at the start of the pricing procedure (only available for a=1 or a=4)
  - 0 always restart the pricing procedure from  $\tilde{t} = 0$
  - 1 only restart from the instant where the last pricing procedure could not find columns of negative reduced cost

Each variant of the Branch & Price implemented for solving (P) is therefore denoted a-b-c-d. For instance, 1-1-0-0 is the basic version with subproblem (SP1), where each pricing procedure consists in solving (SP1) and adding the activity set found at the instant found. Likewise, 2-1-0-0 is the basic version with (SP2 $_{\tilde{t}}$ ). In total 30 variants were obtained (3 with (SP1) and (SP3 $_{\tilde{s}}$ ), which are only concerned by parameter b, and 12 with (SP2 $_{\tilde{t}}$ ) and (SP4 $_{\tilde{t},\tilde{s}}$ )).

The restricted master problem is initialized with columns obtained by starting all activities at their release date, as in Fig. 3(b). An initial upper bound is obtained by computing the cost of the resulting solution.

## 6 Computational evaluation

### 6.1 Instances, Implementation and Parameter settings for best variant identification

Instances\* were obtained by combining adapted scheduling instances with the test efficiency function  $\rho_0$  illustrated in Fig. 3(a) for the purpose of identifying the best variant of the branch-and-price and to compare with the pseudo-polynomial time-indexed formulation. The scheduling instances used were initially proposed for the energy-constrained scheduling problem [2, 3, 19, 20] but were adapted to our setting. The set contains 288 instances: 144 with 30 activities and 144 with 60 activities. Each instance is characterized by one value of each of the following parameters:

- scale  $\in \{1, 10\}$ : scaling of the time horizon (1: small, 10: large);
- k  $\in \{1.25, 2.5, 5\}$ : magnitude of required energy ( $p_i \times b_i$ );
- df  $\in \{0.1, 0.15, 0.2, 0.25\}$ : release date dispersion ( $r_i$ );
- mf  $\in \{0.1, 0.2, 0.4, 0.8, 1.6, 3.2\}$ : time window size ( $d_i - r_i$ ).

Algorithms were coded mainly in C++ and with the framework SCIP 3.1.1 on an Intel Core i7-4770 CPU and 8 GB of RAM. Each master problem was solved with SOPLEX 2.0.1 whereas each subproblem was solved with IBM CPLEX 12.6 which offers the possibility to use the function *IloPiecewiseLinear* to model the piecewise linear functions.

The computational evaluation was done in two phases: first, the best variant of the branch-and-price method was identified by applying different variants on the 288 instances with a time limit of 600 s, then the best branch-and-price was applied to solve all 288 instances with a time limit of 7200 s and compared to IBM CPLEX applied on the pseudo-polynomial size piecewise-linear formulation (4)–(8) with the same time limit.

Table 2 summarizes, for an illustrative subset of settings, the number of instances for which upper and lower bounds were available at the end of the time limit of 600 s. Table 3 focusses on the two best variants identified. Finally Table 4 illustrates the results of the Branch-and-Price compared to the ones from a black-box MILP solver (CPLEX 12.6).

### 6.2 Identification of the best variant

Preliminary tests were performed on the 288 instances with a time limit of 600 s to identify the best parameter settings for the Branch & Price. Table 2 records, for each setting, the number of instances for which a lower bound was obtained (column NbObs). In a column generation or a Branch & Price, no lower bound is available until at least the root node has been solved. Upper bounds were provided at the beginning of each algorithm, equal to the

---

\*Instances available at <http://homepages.laas.fr/sungueve/EnergySchedulingInstances.html>

total energy cost when all activities are started at their release date. Table 2 shows that for half of the displayed variants, the column generation at the root node was not completed yet for a significant number of instances (from 15% to 50%). Variant 2-1-0-0 for example, was not able to produce lower bounds within the 600 s time limit for half of the instances.

Setting a-b-c-d	NbObs
2-1-1-0	278
2-1-1-1	278
2-3-1-1	267
2-3-1-0	266
2-3-0-1	265
2-2-0-1	265
2-2-1-1	262
2-2-1-0	261
2-1-0-1	249
4-1-1-1	214
2-2-0-0	197
1-3-0-0	193
1-2-0-0	192
2-3-0-0	192
1-1-0-0	168
2-1-0-0	146

Table 2: Preliminary results: Number of instances with a non-infinite lower bound after 600 s.

All alternatives produced improvements over their respective basic settings (i.e. all 1-X-X-X are better than 1-1-0-0 and all 2-X-X-X are better than 2-1-0-0). Although basic setting 2-1-0-0 of (SP $2_{\tilde{t}}$ ) performs poorly in comparison to basic setting 1-1-0-0 of (SP1), the two settings that perform best are based on (SP $2_{\tilde{t}}$ ). Note that the results of variants involving (SP $3_{\tilde{s}}$ ) and (SP $4_{\tilde{t},\tilde{s}}$ ) were disappointing, meaning that, at least for the test efficiency function, incorporating the piecewise-linear function in the subproblem solving is better than solving a series of linear subproblems. We only display here variant 4-1-1-1. Table 3 focusses on the two best parameter settings.

Although setting 2-1-1-0 has a better average number of nodes, 2-1-1-1 generates less columns and less pricing attempts. We consequently selected setting 2-1-1-1 for further experiments.

### 6.3 Comparison with the pseudo-polynomial formulation and impact of instance parameters

In the second phase of the computational analysis, all 288 instances were solved with a Branch & Price with the best parameter setting 2-1-1-1. The results were compared to the ones of a black box MILP solver (CPLEX 12.6) applied on the piecewise-linear formulation (4)–(8). The time limit was set to 7200 s. To avoid out-of-memory errors that occurred on the black

		Settings	
		2-1-1-0	2-1-1-1
Number of columns generated	min	95	95
	max	118954	31424
	avg	4483.7	<b>4126.6</b>
Number of pricing attempts	min	2	2
	max	2058	1674
	avg	65.5	<b>62.0</b>
Number of nodes	min	1	1
	max	1317	1558
	avg	<b>34.6</b>	36.7

Table 3: Preliminary results: focus on the two best settings.

box solver, a memory usage limit (1 GB) was also imposed on both algorithms. The results obtained are summarized in Table 4. It shows that the Branch & Price described in this paper is efficient for solving the energy optimization problem with non-reversible energy sources: it was able to solve more than 90% of the instances to optimality and reached an average ratio of 99.98% between the best-known lower and upper bound, which suggests that all instances may be solved to optimality with only a small increase of the time limit. The black box MILP solver applied on the pseudo-polynomial-size formulation solves less than 2% of the instances to optimality and has an average ratio of 88.01%. Note that all executions of the MILP solver but 5 were stopped because the memory limit was reached as shown by the maximum CPU time of 1674.3 s. We also recall the results of the branch-and-price for 600 s of CPU time. This illustrates the trade-off between quality and CPU time that the column generation approach brings as the average CPU time falls significantly under the average CPU time needed by the MILP solver while finding in average way better solutions (although no LB could be found for 10 instances). We also ran the branch-and-price method with a single node and no time limit (so all LB were found) and the results are still good with an average ratio of 98.1% between the best-obtained lower and upper bound and 119 optimal solutions found (as no branching was allowed, this is the gap between the relaxation of the extended formulation and the heuristic ran by SCIP at the root node). This illustrates the high quality of the LP relaxation of the extended formulation. Finally computing the pure LP relaxation of the extended formulation by column generation (without any preprocessing nor heuristic search) takes 48.7 s in average with a maximum of 1544 s and yields an average lower bound close at 99.97% in average from the best known feasible solution with a minimum of 98.12 %. This illustrates the high quality of the extended formulation relaxation.

Finally we analyze the instance difficulty w.r.t. their generation parameters scale,  $k$ ,  $df$  and  $mf$ . In Table 5 we give the number of optimal solutions found for each parameter value.

The results show that, as it could be expected, the time horizon scale has a large influence on the instance hardness for time-indexed formulations. To a lesser extent, the study of the influence of parameter  $k$  suggests that large energy requirements yield instances that are

		Extended Model			Pseudo-polynomial-size Model
		B&P 1 node	B&P / 600 s	B&P / 7200s	Black Box Solver
# lower bounds		288	278	<b>288</b>	<b>288</b>
# opt		119	220	<b>268</b>	5
Ratio LB/UB	min	88.0%	94.79%	9.1%	58.1%
	max	100%	100%	100%	100%
	avg	99.12%	99.81%	99.98%	88.01%
Time	min	1 s	1 s	1 s	7 s
	max	2625.1	600 s	7200 s	1674.3 s
	avg	80.8 s	189.99 s	879.9 s	544.2 s
# nodes	min	1	1	1	20577
	max	1	1558	7100	12478166
	avg	1	46.75	165.7	926119

Table 4: Efficiency of the Branch & Price compared to a black-box solver.

scale	1	10				
# opt	141	127				
k	1.25	2.5	5			
# opt	92	89	87			
df	0.1	0.15	0.2	0.25		
# opt	64	66	70	68		
mf	0.1	0.2	0.4	0.8	1.6	3.2
# opt	48	48	46	46	45	35

Table 5: Impact of the instance parameters on the number of optimally solved instances.

harder to solve than instances where tasks have lower energy requirement. No conclusion can be drawn on the influence of the release date dispersion parameter (df). However the category of instances with the larger time window (large mf) seems harder.

#### 6.4 Application using a realistic non-linear efficiency function

A realistic efficiency function derived from the conversion function of a real-world fuel cell was considered (see Fig. 1(b)). The function provided by researchers in Electrical Engineering expresses the energy demand (or energy usable) in function of the energy cost (or energy consumed) and therefore corresponds to  $\rho^{-1}$  and could not be easily inverted. Following the methodology of [21], the non-linear function was bounded from above and below with two piecewise linear functions  $\overline{\rho^{-1}^\epsilon}$  and  $\underline{\rho^{-1}^\epsilon}$  that verify equations (1)–(3) for a given value of  $\epsilon$ . To that end, the algorithms proposed in [21] were applied on convex or concave sections of  $\rho^{-1}$ . The resulting piecewise linear functions could then be easily inverted without any loss of precision before their introduction into the mathematical model of the preemptive scheduling

problem.

We considered four levels of precision:  $\epsilon = 5\%, 1\%, 0.5\%$  and  $0.1\%$ . Table 6 summarizes the number of sectors obtained for the different piecewise linear functions. Fig. 4 illustrates the bounding results for a precision  $\epsilon = 10\%$ . It can be observed that the abscissas of the extremities of the sectors of  $\overline{\rho^{-1}^\epsilon}$  differ from the ones of the sectors of  $\underline{\rho^{-1}^\epsilon}$ . This contributes to the reduction of the number of sectors for each of the piecewise functions, contrary to the cases where the same limits are used for both the over- and the under-estimation functions, as is done by Interval Branch-and-Bound methods or in box-bounded global optimization.

	$\epsilon=5\%$		$\epsilon=1\%$		$\epsilon=0.5\%$		$\epsilon=0.1\%$	
	$\overline{\rho^{-1}^\epsilon}$	$\underline{\rho^{-1}^\epsilon}$	$\overline{\rho^{-1}^\epsilon}$	$\underline{\rho^{-1}^\epsilon}$	$\overline{\rho^{-1}^\epsilon}$	$\underline{\rho^{-1}^\epsilon}$	$\overline{\rho^{-1}^\epsilon}$	$\underline{\rho^{-1}^\epsilon}$
# sectors	9	9	18	24	42	28	60	58

Table 6: Number of sectors of the piecewise linear functions  $\overline{\rho^{-1}^\epsilon}$  and  $\underline{\rho^{-1}^\epsilon}$ .

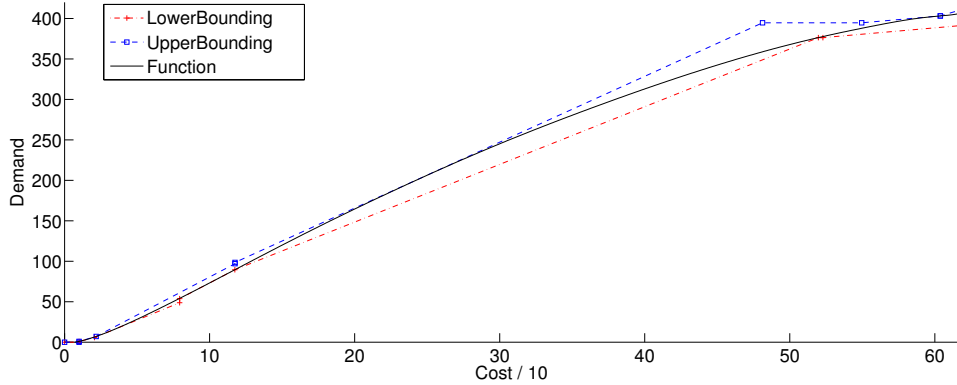


Figure 4: A piecewise linear bounding result for  $\epsilon = 10\%$ .

For each value of  $\epsilon$ , two MILPs, denoted by  $\overline{\text{MILP}}$  and  $\underline{\text{MILP}}$  respectively, are obtained by substituting  $\rho$  with  $(\underline{\rho^{-1}^\epsilon})^{-1}$  and  $(\overline{\rho^{-1}^\epsilon})^{-1}$ , respectively. As the non-linear function appears only in the objective function of the problem studied in this paper, the resulting  $\overline{\text{MILP}}$  and  $\underline{\text{MILP}}$  both generate solutions feasible for the non-linear problem. The optimal solution cost of  $\overline{\text{MILP}}$  is an upper bound for the non-linear problem, less than  $\epsilon\%$  higher than its optimal solution cost. The optimal solution cost of  $\underline{\text{MILP}}$  is a lower bound for the non-linear problem, less than  $\epsilon\%$  lower than its optimal solution cost. Solving both  $\overline{\text{MILP}}$  and  $\underline{\text{MILP}}$  and then comparing their solution costs aims at lowering the gap between the best known upper and lower bounds for the scheduling problem with non-linear efficiency function, which in the worst case would be equal to  $\epsilon\%$ .

Table 7 shows the performance of the branch-and-price with the best parameter setting 2-1-1-1 previously identified, considering a time limit of 600 s on all 288 instances and given different values of  $\epsilon$ . The results confirm the efficiency of the algorithm: for 2105 of the 2304

cases the optimal solution of the MILP was found. Only in 151 cases was the algorithm not able to produce lower bounds before the time limit. Among the 2153 remaining cases, the average ratio between the lower and upper bound exceeds 99.85% for the four  $\epsilon$  values tested.

		$\epsilon=5\%$		$\epsilon=1\%$		$\epsilon=0.5\%$		$\epsilon=0.1\%$	
		<u>MILP</u>	<u>MILP</u>	<u>MILP</u>	<u>MILP</u>	<u>MILP</u>	<u>MILP</u>	<u>MILP</u>	<u>MILP</u>
# lower bounds		274	260	271	270	267	267	271	273
# opt		254	239	268	268	266	266	271	273
Ratio LB/UB	min	77.51%	93.39%	99.93%	99.94%	99.84%	99.99%	100%	100%
	max	100%	100%	100%	100%	100%	100%	100%	100%
	avg	99.85%	99.94%	99.99%	99.99%	99.99%	99.99%	100%	100%
Time	min	1 s	2 s	1 s	1 s	1 s	1 s	1 s	1 s
	max	600 s	600 s	600 s	600 s	600 s	600 s	600 s	600 s
	avg	127.99 s	151.64 s	101.72 s	111.00 s	98.60 s	96.77 s	89.37 s	88.97 s
# columns	min	11 4	119	116	114	112	114	115	116
	max	56360	60835	56295	58424	43296	43526	50756	52719
	avg	4666	4735	5139	5263	5142	5161	5427	5425
# pricing attempts	min	4	4	5	4	4	5	5	4
	max	633	2711	185	1086	1379	1066	492	442
	avg	32.06	74.90	20.95	25.65	25.69	24.27	21.45	21.42
# nodes	min	1	1	1	1	1	1	1	1
	max	305	307	54	86	23	28	19	13
	avg	10.60	9.58	2.16	2.81	1.47	1.57	1.14	1.10

Table 7: Branch-and-Price output.

Finally, we focussed on the instances for which both MILP and MILP could be solved to optimality. This concerned respectively 233, 265, 264 and 270 instances for  $\epsilon = 5\%$ ,  $1\%$ ,  $0.5\%$  and  $0.1\%$ . The best upper and lower bounds were obtained with equations (61)–(62).

$$\text{Best UB} = \min\{\overline{\text{MILP}}, \underline{\text{MILP}} * (1 + \epsilon)\%\} \quad (61)$$

$$\text{Best LB} = \max\{\underline{\text{MILP}}, \overline{\text{MILP}} * (1 - \epsilon)\%\} \quad (62)$$

Table 8 shows that except for one instance, the best upper bounds were derived from MILP whereas the best lower bounds were derived from MILP. The final gaps between the best upper and lower bounds found, also available in Table 8, are on average smaller (and therefore better) than  $\epsilon\%$ , in the best cases even smaller than half of  $\epsilon\%$  and in the worst cases equal to  $\epsilon\%$ . This illustrates the benefits of the piecewise bounding approach: (i) the guarantee of feasibility for the solution obtained with respect to the non-linear problem and; (ii) the tighter gap between the best-known upper and lower bounds.

## 7 Conclusions

This paper contributes to the definition of a novel and efficient methodology for the integration of realistic energy characteristics in combinatorial optimization problems via piecewise-linear lower and upper bounding of the non-linear energy conversion functions. Focussing on a



		$\epsilon=5\%$	$\epsilon=1\%$	$\epsilon=0.5\%$	$\epsilon=0.1\%$
# instances		233	265	264	270
Best UB	$\overline{\text{MILP}}$	232	265	264	270
	$\overline{\text{MILP}} * (1 + \epsilon)\%$	1	0	0	0
Best LB	$\overline{\text{MILP}}$	1	0	0	0
	$\overline{\text{MILP}} * (1 - \epsilon)\%$	233	265	264	270
Gap (= $\frac{\text{Best UB} - \text{Best LB}}{\text{Best LB}}$ )	min	2.64%	0.53%	0.27%	0.04%
	max	5.00%	0.83%	0.40%	0.08%
	avg	3.54%	0.63%	0.31%	0.06%

Table 8: Final results for the scheduling problem with a non-linear cost function.

scheduling problem where the objective is to minimize the total energy cost of a set of preemptive tasks subject to time windows, we show that the preemptive scheduling problem with piecewise-linear energy cost from multiple non-reversible sources, can be transformed into an equivalent single-source problem. Proof of the NP-hardness of the problem in the general case and some polynomial particular cases are provided. We present two mixed-integer linear programming (MILP) formulations, among which an extended formulation. A branch-and-price procedure is proposed to solve the extended formulation. Computational experiments on a set of scheduling instances with a realistic efficiency function from a fuel cell energy conversion function show the efficiency of the approach, with a final gap between best known upper and lower bounds between 0.04% and 0.08%.

Future work includes the introduction of reversible energy sources, able not only to produce energy but also to retrieve it. An additional level of complexity would be added since the non-reversible sources could produce more energy than the demand at certain times and store the excess in the reversible sources to satisfy the demand of a later time. This also invalidates the proofs of theorems of the problem without reversible energy sources and the extended formulation presented would no longer apply.

## Acknowledgements

This work is funded in part by the Gaspard Monge Program for Optimization and operations research (PGMO), launched by EDF and the Jacques Hadamard Mathematical Foundation (FMJH).

## References

- [1] S. Albers. Energy-efficient algorithms. *Communications of the ACM*, 53(5):86–96, 2010.
- [2] C. Artigues and P. Lopez. Energetic reasoning for energy-constrained scheduling with a continuous resource. *Journal of Scheduling*, 18(3):225–241, 2015.

- [3] C. Artigues, P. Lopez, and A. Haït. The energy scheduling problem: Industrial case-study and constraint propagation techniques. *International Journal of Production Economics*, 143(1):13–23, 2013.
- [4] J. Błażewicz, K. H. Ecker, E. Pesch, G. Schmidt, and J. Węglarz. *Scheduling computer and manufacturing processes*. Springer-Verlag, Berlin/Heidelberg, 2001.
- [5] J. Błażewicz, K. H. Ecker, E. Pesch, G. Schmidt, and J. Węglarz. *Handbook on Scheduling: From Theory to Applications*. Springer, 2007.
- [6] J. Błażewicz, M. Machowiak, J. Węglarz, M. Kovalyov, and D. Trystram. Scheduling malleable tasks on parallel processors to minimize the makespan. *Annals of Operations Research*, 129(1-4):65–80, 2004.
- [7] A. Borghetti, C. D’Ambrosio, A. Lodi, and S. Martello. An MILP approach for short-term hydro scheduling and unit commitment with head-dependent reservoir. *IEEE Transactions on Power Systems*, 23(3):1115–1124, 2008.
- [8] E. Camponogara, M. P. de Castro, and A. Plucenio. Compressor scheduling in oil fields: A piecewise-linear formulation. In *IEEE International Conference on Automation Science and Engineering (CASE)*, pages 436–441, September 2007.
- [9] R. Carli and M. Dotoli. Energy scheduling of a smart home under nonlinear pricing. In *IEEE 53rd Annual Conference on Decision and Control (CDC)*, pages 5648–5653, 2014.
- [10] J. Chang, S. Khuller, and K. Mukherjee. LP rounding and combinatorial algorithms for minimizing active and busy time. In G. E. Blelloch and P. Sanders, editors, *26th ACM Symposium on Parallelism in Algorithms and Architectures, SPAA*, pages 118–127, 2014.
- [11] Y. Gaoua, S. Caux, and P. Lopez. A combinatorial optimization approach for the electrical energy management in a multi-source system. In *International Conference on Operations Research and Enterprise Systems (ICORES)*, pages 55–59, February 2013.
- [12] M. Guemri, S. Caux, and S. U. Ngueveu. Using quasi-Newton method for energy management in electrical multi source systems. In *IEEE International Conference on Environment and Electrical Engineering (EEEIC)*, pages 194–199, May 2012.
- [13] M. Guemri, S. Caux, S. U. Ngueveu, and F. Messine. Heuristics and lower bound for energy management in hybrid-electric vehicles. In *9th International Conference on Modeling, Optimization & SIMulation (MOSIM)*, Bordeaux, June 2012.
- [14] A. Haït and C. Artigues. A hybrid CP/MILP method for scheduling with energy costs. *European Journal of Industrial Engineering*, 5(4):471–489, 2011.
- [15] J. Józefowska, M. Mika, R. Różycki, G. Waligóra, and J. Węglarz. Project scheduling under discrete and continuous resources. In J. Węglarz, editor, *Project Scheduling: Recent Models, Algorithms, and Applications*, pages 289–308. Kluwer Academic Publishers, Boston, 1999.

- [16] T. Kis. A branch-and-cut algorithm for scheduling of project with variable-intensity activities. *Mathematical Programming: Series A*, 103:515–539, 2005.
- [17] O. L. Mangasarian, J. B. Rosen, and M. E. Thompson. Global minimization via piecewise-linear underestimation. *Journal of Global Optimization*, 32(1):1–9, 2005.
- [18] A. Mingozzi, V. Maniezzo, S. Ricciardelli, and L. Bianco. An exact algorithm for the resource-constrained project scheduling problem based on a new mathematical formulation. *Management Science*, 44(5):714–729, 1998.
- [19] M. Nattaf, C. Artigues, and P. Lopez. A hybrid exact method for a scheduling problem with a continuous resource and energy constraints. *Constraints*, 20(3):303–324, 2015.
- [20] 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. Working paper.
- [21] S. U. Ngueveu, B. Sareni, and X. Roboam. Piece-wise bounding and integer linear programming for the optimal management of a water pumping and desalination process. Technical report, LAAS-CNRS, April 2014.
- [22] F. Shrouf, J. Ordieres-Meré, A. García-Sánchez, and M. Ortega-Mier. Optimizing the production scheduling of a single machine to minimize total energy consumption costs. *Journal of Cleaner Production*, 67:197–207, 2014.
- [23] J. Węglarz, J. Józefowska, M. Mika, and G. Waligóra. Project scheduling with finite or infinite number of activity processing modes – A survey. *European Journal of Operational Research*, 208:177–205, 2009.
- [24] C-K. Woo, P. Sreedharan, J. Hargreaves, F. Kahrl, J. Wang, and I. Horowitz. A review of electricity product differentiation. *Applied Energy*, 114:262–272, 2014.