



HAL
open science

Polyhedral results and valid inequalities for the Continuous Energy-Constrained Scheduling Problem

Margaux Nattaf, Tamás Kis, Christian Artigues, Pierre Lopez

► **To cite this version:**

Margaux Nattaf, Tamás Kis, Christian Artigues, Pierre Lopez. Polyhedral results and valid inequalities for the Continuous Energy-Constrained Scheduling Problem. *Discrete Applied Mathematics*, 2019, 258, pp.188-203. 10.1016/J.DAM.2018.11.008 . hal-01391403

HAL Id: hal-01391403

<https://hal.science/hal-01391403>

Submitted on 3 Nov 2016

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.

Polyhedral results and valid inequalities for the Continuous Energy-Constrained Scheduling Problem

Margaux Nattaf¹, Tamás Kis², Christian Artigues¹, and Pierre Lopez¹

¹LAAS-CNRS, Université de Toulouse, CNRS, Toulouse, France
{nattaf,artigues,lopez}@laas.fr

²Computer and Automation Research Institute, Hungarian Academy of sciences, 1111 Budapest, Kende utca
13-17, Hungary, kis.tamas@sztaki.mta.hu

Abstract

This paper addresses a scheduling problem with a cumulative, continuously-divisible and renewable resource with limited capacity. During its processing, each task consumes a part of this resource, which lies between a minimum and a maximum requirement. A task is finished when a certain amount of energy is received by it within its time window. This energy is received via the resource and an amount of resource is converted into an amount of energy with a non-decreasing, continuous and linear efficiency function. The goal is to minimize the resource consumption. The paper focuses on an event based mixed integer linear program, providing several valid inequalities which are used to improve the performance of the model. Furthermore, we give a minimal description of the polytope of all feasible assignments to the on/off binary variable for a single activity along with a dedicated separation algorithm. Computational experiments are reported in order to show the effectiveness of the results.

keywords continuous scheduling, continuous resources, linear efficiency functions, mixed-integer programming, valid inequalities, polyhedral combinatorics

1 Introduction

Most of the scheduling problems dealing with resource constraints assume a fixed duration and does not allow the resource usage to vary over time. However, several extensions of existing problem, such as the resource-constrained project scheduling problem or the cumulative scheduling problem, has been developed to tackle this issue. Among them there are the multi-mode resource-constrained project scheduling problem [6], project scheduling with variable-intensity activities [9] or with work-content resources [8] or the malleable tasks scheduling problem [7]. In this paper, we study a problem called the continuous energy-constrained scheduling problem (CECSP), a generalization of the cumulative scheduling problem which no longer assumes fixed duration and resource requirement.

In the CECSP, we are given as input a set of non preemptive tasks $\mathcal{A} = \{1, \dots, n\}$ and a continuously-divisible resource, renewable and cumulative resource of capacity B . For each task, a release date r_i and a deadline d_i define an interval in which the task must be executed. At each time t during its execution, each task consumes a quantity of resource $b_i(t)$ that has to be determined. This resource usage has to lie between a minimum requirement, b_i^{min} , and a maximum requirement, b_i^{max} .

The particularity of the CECSP is that a task no longer have a fixed duration but instead an energy requirement W_i needs to be fulfilled before the task deadline. This energy is computed from the task resource usage, using an efficiency function f_i^1 . We assume these functions to be continuous, non-decreasing and linear. An efficiency functions f_i can be defined as follows:

$$f_i(b) = \begin{cases} 0 & \text{if } b = 0 \\ a_i * b + c_i & \text{if } b_i^{min} = 0 \text{ and } b \in]b_i^{min}, b_i^{max}] \\ a_i * b + c_i & \text{if } b_i^{min} \neq 0 \text{ and } b \in [b_i^{min}, b_i^{max}] \end{cases}$$

¹Some authors call this function the power processing rate function [3, 4, 6].

with $a_i > 0$ and $-a_i * b_i^{min} \geq c_i$ to ensure that $f_i(b) \leq 0, \forall b \in [b_i^{min}, b_i^{max}]$.

Therefore, to solve the CECSP, we have to find, for each task $i \in \mathcal{A}$, its start time st_i , its end time et_i and its resource allocation function $b_i(t), \forall t \in \mathcal{T} = [\min_{i \in \mathcal{A}} r_i, \max_{i \in \mathcal{A}} d_i]$. These quantities have to satisfy the following constraints:

$$r_i \leq st_i < et_i \leq d_i \quad \forall i \in \mathcal{A} \quad (1)$$

$$b_i^{min} \leq b_i(t) \leq b_i^{max} \quad \forall i \in \mathcal{A}, \forall t \in [st_i, et_i] \quad (2)$$

$$b_i(t) = 0 \quad \forall i \in \mathcal{A}, \forall t \notin [st_i, et_i] \quad (3)$$

$$\int_{st_i}^{et_i} f_i(b_i(t)) dt = W_i \quad \forall i \in \mathcal{A} \quad (4)$$

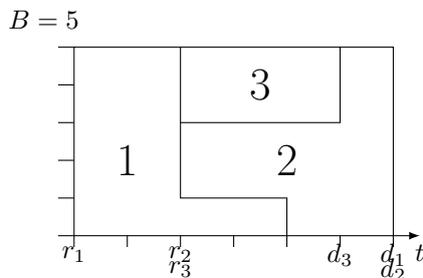
$$\sum_{i \in \mathcal{A}} b_i(t) \leq B \quad \forall t \in \mathcal{T} \quad (5)$$

The objective we are interested in is the minimization of the total resource consumption. In [5], the authors shows that the problem of finding a feasible solution is NP-complete.

Example 1 Consider an instance with $n = 3$ and $B = 5$. The other data are displayed in Table 1a, and a feasible solution is depicted in Fig. 1b.

i	r_i	d_i	W_i	b_i^{min}	b_i^{max}	$f_i(b)$
1	0	6	28	1	5	$2b + 1$
2	2	6	32	2	5	$b + 5$
3	2	5	6	2	2	b

(a) an instance of CECSP



(b) the corresponding solution

Figure 1: An example of an instance and the corresponding solution for the CECSP

*This solution is feasible since each task lies in its time window, all the constraints of maximum and minimum requirements are satisfied and the total resource usage at each time does not exceed the availability of the resource. Furthermore, the required energy is received by each task. For example, the energy received by task 1 is $(2 * 5 + 1) + (2 * 5 + 1) + (2 * 1 + 1) + (2 * 1 + 1) = 11 + 11 + 3 + 3 = 28$, while its total resource consumption is equal to 12.*

Different elements of the problem addressed in this paper have been studied by several authors. Unfortunately, the authors consider generally only a part of the problem and/or they assume a discrete time setting. Our problem belongs to the category of problems where the resource usage may vary continuously and such that the amount of resource required by a task may vary over time. Węglarz *et al.* [6] call this model the processing rate vs resource amount model. Providing a general framework for solving mixed discrete/continuous problems with concave processing rate functions, Józefowska *et al.* [4]

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.

The CECSP comes from an industrial problem occurring in the context of energy-consuming production scheduling problem. In [1], a foundry application is presented and a solution method is proposed using constraint programming and mixed integer linear programming. Due to the complexity of the problem the authors consider a time discretization and do not take efficiency functions into account. Still without considering efficiency functions but in a continuous time setting, Artigues and Lopez [2] propose a constraint satisfiability test based on the energetic reasoning. Recently, Nattaf *et al.* [5] propose mixed integer linear programs, constraint propagation algorithm and a hybrid branch and bound methods to solve the problem with linear efficiency functions and considering continuous time.

This paper focuses on the on/off model from [5], providing several valid inequalities which are used to improve the performance of the model in Section 2. Furthermore, a special set of inequality is described in Section 3. These inequalities are used to give a minimal description of the polytope Z of all feasible assignments to the on/off variable z_{ie} for a single activity. This section also describe a special separation algorithm for these inequalities. Finally, computational results are presented in Section 4.

2 On/Off MILP formulation and valid inequalities

In the on/off formulation of [5], an event corresponds either to a task start or a task end time. These events are represented by a set of continuous variables t_e and $\mathcal{E} = \{1, \dots, 2n\}$ represent the index set of these events. The authors uses a binary variable z_{ie} to assign the different event dates to the start and end time of the tasks. Indeed, z_{ie} is equal to 1 if and only if task i is in process during interval $[t_e, t_{e+1}]$. Finally, two continuous variables b_{ie} and w_{ie} are also defined. These variables stand for the quantity of resource used by task i and for the energy received by i between events t_e and t_{e+1} .

The model is described below.

$$\min \sum_{i \in \mathcal{A}} \sum_{e \in \mathcal{E} \setminus \{2n\}} b_{ie} \quad (6)$$

$$t_e \leq t_{e+1} \quad \forall e \in \mathcal{E} \setminus \{2n\} \quad (7)$$

$$r_i z_{ie} \leq t_e \quad \forall (i, e) \in \mathcal{A} \times \mathcal{E} \quad (8)$$

$$t_e \leq s_i^{max}(z_{ie} - z_{i,e-1}) + (1 - (z_{ie} - z_{i,e-1}))|\mathcal{T}| \quad \forall (i, e) \in \mathcal{A} \times \mathcal{E} \setminus \{1\} \quad (9)$$

$$e_i^{min}(z_{i,e-1} - z_{ie}) \leq t_e \quad \forall (i, e) \in \mathcal{A} \times \mathcal{E} \setminus \{1\} \quad (10)$$

$$t_e \leq d_i(z_{i,e-1} - z_{ie}) + (1 - (z_{i,e-1} - z_{ie}))|\mathcal{T}| \quad \forall (i, e) \in \mathcal{A} \times \mathcal{E} \setminus \{1\} \quad (11)$$

$$t_f \geq t_e + \quad \forall e, f \in \mathcal{E} \setminus \{1\},$$

$$((z_{ie} - z_{i,e-1}) - (z_{im} - z_{i,m-1}) - 1) \frac{W_i}{f_i(b_i^{max})} \quad f > e, \forall i \in \mathcal{A} \quad (12)$$

$$\sum_{e \in \mathcal{E}} z_{ie} \geq 1 \quad \forall i \in \mathcal{A} \quad (13)$$

$$\sum_{e'=1}^e z_{ie'} \leq e(1 - (z_{ie} - z_{i,e-1})) \quad \forall (i, e) \in \mathcal{A} \times \mathcal{E} \setminus \{1\} \quad (14)$$

$$\sum_{e'=e}^{2n} z_{ie'} \leq (2n - e)(1 + (z_{ie} - z_{i,e-1})) \quad \forall (i, e) \in \mathcal{A} \times \mathcal{E} \setminus \{1\} \quad (15)$$

$$\sum_{i \in \mathcal{A}} b_{ie} \leq B(t_{e+1} - t_e) \quad \forall e \in \mathcal{E} \setminus \{2n\} \quad (16)$$

$$b_{ie} \geq b_i^{min}(t_{e+1} - t_e) - (b_i^{min}(d_i - r_i)(1 - z_{ie})) \quad \forall (i, e) \in \mathcal{A} \times \mathcal{E} \setminus \{2n\} \quad (17)$$

$$b_{ie} \leq b_i^{max}(t_{e+1} - t_e) \quad \forall (i, e) \in \mathcal{A} \times \mathcal{E} \setminus \{2n\} \quad (18)$$

$$z_{ie}(b_i^{max}(d_i - r_i)) \geq b_{ie} \quad \forall (i, e) \in \mathcal{A} \times \mathcal{E} \setminus \{2n\} \quad (19)$$

$$\sum_{e \in \mathcal{E} \setminus \{2n\}} b_{ie} = W_i \quad \forall i \in \mathcal{A} \quad (20)$$

$$w_{ie} \leq a_i b_{ie} + c_i(t_{e+1} - t_e) \quad \forall (i, e) \in \mathcal{A} \times \mathcal{E} \setminus \{2n\} \quad (21)$$

$$w_{ie} \leq W_i z_{ie} \quad \forall (i, e) \in \mathcal{A} \times \mathcal{E} \setminus \{2n\} \quad (22)$$

$$z_{i,2n} = 0 \quad \forall i \in \mathcal{A} \quad (23)$$

$$t_e \geq 0 \quad \forall e \in \mathcal{E} \quad (24)$$

$$b_{ie} \geq 0 \quad \forall (i, e) \in \mathcal{A} \times \mathcal{E} \setminus \{2n\} \quad (25)$$

$$w_{ie} \geq 0 \quad \forall (i, e) \in \mathcal{A} \times \mathcal{E} \setminus \{2n\} \quad (26)$$

$$z_{ie} \in \{0, 1\} \quad \forall (i, e) \in \mathcal{A} \times \mathcal{E} \quad (27)$$

total resource consumption is described by equation (6). Constraint (7) arbitrarily orders the events. Inequalities (8)–(11) model the time window constraints. Constraint (12) ensures a minimum separation between two events corresponding to the start and end time of a same task i . Constraint (13) stipulates that each task has to be scheduled once while constraints (14) and (15) make sure a task is not preempted during its execution. Finally, inequalities (16)–(19) model the resource constraints while inequalities (20)–(22) represent resource conversion and energy requirement constraints.

2.1 Maximum interval between two events

Here, we describe the first set of inequalities we derive for the CECSP. In the following, we suppose an event correspond to one and only one task start/end time. Such solution always exists since there is $2 \times \#tasks$ events.

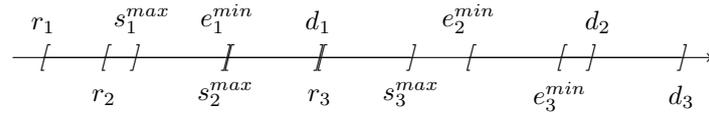
The inequalities define in this section aims to give an upper bound on the value of $t_{e+1} - t_e$, $\forall e \in \mathcal{E}$. To do so, we study the time window of each task start and end time. The main idea relies on the fact that an event must occur in each of these time windows. Therefore, we know there are at least two consecutive events in the union of two consecutive time windows.

Let \mathcal{S} be the set of all task time windows. We start by sorting each interval in \mathcal{S} according to the following rule: $[\mathcal{S}_x^1, \mathcal{S}_y^1] \leq [\mathcal{S}_x^2, \mathcal{S}_y^2] \Leftrightarrow \mathcal{S}_x^1 < \mathcal{S}_x^2 \vee (\mathcal{S}_x^1 = \mathcal{S}_x^2 \wedge \mathcal{S}_y^1 \leq \mathcal{S}_y^2)$

And, then we have:

$$t_{e+1} - t_e \leq |\mathcal{S}_e \cup \mathcal{S}_{e+1}| \quad \forall e \in \mathcal{E} \setminus \{2n\} \quad (28)$$

Example 2 Consider the following intervals:



An ordering of these intervals is: $[r_1, s_1^{max}] \leq [r_2, s_2^{max}] \leq [e_1^{min}, d_1] \leq [r_3, s_3^{max}] \leq [e_2^{min}, d_2] \leq [e_3^{min}, d_3]$.

And then, we have the following constraints:

- $t_2 - t_1 \leq s_2^{max} - r_1$
- $t_3 - t_2 \leq d_1 - r_2$
- $t_4 - t_3 \leq s_3^{max} - e_1^{min}$
- $t_5 - t_4 \leq d_2 - r_3$
- $t_6 - t_5 \leq d_3 - e_2^{min}$

We can then add these constraints to the model and/or use these upper bounds in constraint (17), replacing $b_i^{min}(d_i - r_i)$ by $b_i^{min}|\mathcal{S}_e \cup \mathcal{S}_{e+1}|$, $\forall (i, e) \in \mathcal{A} \times \mathcal{E}$.

2.2 Maximum time of an event

A similar idea than the previous one is to use the time windows to order the event times and to compute upper bounds on these dates. To do so, we sort the time window upper bounds, i.e. s_i^{max} and d_i , $\forall i \in \mathcal{A}$, in increasing order. Then, since an event must occurred in each time window, i.e. before its upper bound, we have an upper bound on each event.

Indeed, let \mathcal{UP} be the sorted set of all time window upper bounds, then, we have the following:

$$t_e \leq \mathcal{UP}_e \quad \forall e \in \mathcal{E} \quad (29)$$

Example 3 Consider the intervals given in example 2. Then, we have the following constraints:

- $t_1 \leq s_1^{max}$
- $t_2 \leq s_2^{max}$
- $t_3 \leq d_1$
- $t_4 \leq s_3^{max}$
- $t_5 \leq d_2$
- $t_6 \leq d_3$

Like the previous inequalities, we can use these bounds either as additional constraints of the model, or in constraints (9) and (11) as an alternative to T - an upper bound on the project duration - or both.

2.3 Valid inequalities from knapsack problems

Since the minimum intensity of the activities can be positive, we can consider the following knapsack type constraint for each $e \in \mathcal{E} \setminus \{2n\}$ from which one can easily derive valid inequalities:

$$\sum_{i \in \mathcal{A}^+} b_i^{\min} z_{ie} \leq B, \quad (30)$$

where \mathcal{A}^+ is the subset of activities with positive b_i^{\min} values. One may add this set of constraints to the initial formulation, and then let the solver to strengthen the LP relaxation by cutting planes for these knapsack sets.

3 Polyhedral results and non-preemptive inequalities

In this section we describe some inequalities satisfied by all feasible solutions of the MIP formulation.

3.1 Non-preemptive inequalities

Since each activity must be processed without preemption, in any feasible schedule for each activity i , the z_{ie} satisfy

$$\sum_{e_k \in S} (-1)^k z_{i,e_k} \leq 1 \quad (31)$$

where $S = \{e_0, e_1, \dots, e_{2\ell}\}$ is a subset of $\mathcal{E}^* := \mathcal{E} \setminus \{2n\}$ of odd cardinality.

Consider the polyhedron $ZP_i := \{z_i \in [0, 1]^{\mathcal{E}^*} \mid z_i \text{ satisfies (13) and (31)}\}$.

On the other hand, let $ZQ_i := \text{conv}\{z_i \in \{0, 1\}^{\mathcal{E}^*} \mid z_i \text{ satisfies (13) - (15)}\}$.

Theorem 1 $ZP_i = ZQ_i$

Proof We will use the Farkas lemma in order to derive a description of ZQ_i in terms of linear inequalities. The vertices of ZQ_i are precisely the $|\mathcal{E}^*|$ -dimensional vectors

$$z_{i,e}^{kl} = \begin{cases} 1 & k \leq e \leq \ell \\ 0 & \text{otherwise} \end{cases} \quad \forall k, \ell \in \mathcal{E}^*, k \leq \ell.$$

Consider the linear system

$$\sum_{k \leq \ell} z_{i,e}^{kl} \lambda_{kl} = \bar{z}_{i,e}, \quad e \in \mathcal{E}^* \quad (32)$$

$$\sum_{k \leq \ell} \lambda_{kl} = 1 \quad (33)$$

$$\lambda \geq 0 \quad (34)$$

Clearly, $\bar{z}_i \in ZQ_i$ if and only if this system admits a feasible solution. By the Farkas lemma, the system (32)-(34) admits a feasible solution if and only if for all μ satisfying the dual system

$$\sum_{e=k}^{\ell} \mu_e + \mu_0 \leq 0, \quad k \leq \ell \quad (35)$$

μ also satisfies the condition

$$\sum_{e \in \mathcal{E}^*} \mu_e \bar{z}_{i,e} + \mu_0 \leq 0. \quad (36)$$

In order to prove our theorem, it suffices to find all the extreme rays of the cone (35), since they define all the linear inequalities needed to describe ZQ_i . We will show that there is a one-to-one correspondence between the extreme rays of cone (35), and the inequalities of ZP_i . In order to find all the extreme rays of the cone (35), it suffices to distinguish between 3 cases:

$\mu_0 = 1$. Then for each $e \in \mathcal{E}^*$, $\mu_e \leq -1$ follows from (35) by considering the inequalities for $k = \ell = e$. Then (36) yields

$$\sum_{e \in \mathcal{E}^*} -z_{i,e} \leq -1,$$

which, by the Farkas lemma, is a valid inequality for ZQ_i . Notice that it is equivalent to (13).

$\mu_0 = 0$. Then we still have a cone, whose extreme rays are the negative unit vectors in $\mathbb{R}^{\mathcal{E}^*}$. These extreme rays give the inequalities $-z_{i,e} \leq 0$, which are the non-negativity constraints valid for ZQ_i .

$\mu_0 = -1$. We argue that there is a one-to-one correspondence between the extreme points of the polyhedron $M \subseteq \mathbb{R}^{|\mathcal{E}^*|}$ defined by

$$\sum_{e=k}^{\ell} \mu_e \leq 1, \quad k \leq \ell \quad (37)$$

and the inequalities (31).

First we claim that the coefficient vector of the left-hand-side of each inequality in (31) is an extreme point solution of (37). Let $S = \{e_0, e_1, \dots, e_{2\ell}\}$ be a set of events with $e_i < e_{i+1}$ for $i = 0, \dots, 2\ell - 1$. The corresponding vector $\bar{\mu}$ is defined as

$$\bar{\mu}_e = \begin{cases} (-1)^k, & \text{if } e_k \in S \\ 0, & \text{if } e \in \mathcal{E}^* \setminus S. \end{cases}$$

We claim that $\bar{\mu}$ is an extreme point solution of (37). To prove our claim, we exhibit a subsystem L of (37) consisting of $|\mathcal{E}^*|$ linearly independent inequalities such that each inequality in L holds at equality in $\bar{\mu}$. The subsystem contains the inequalities

$$\sum_{e=k}^{e_0} \mu_e \leq 1, \quad k = 1, \dots, e_0 - 1.$$

and

$$\sum_{e=e_{2\ell}}^k \mu_e \leq 1, \quad k = e_{2\ell}, \dots, |\mathcal{E}^*|.$$

Further on, for each 3 consecutive events $e_{2k}, e_{2k+1}, e_{2k+2} \in S$, the set of inequalities

$$\begin{aligned} \sum_{e=e_{2k}}^{e_{2k+2}} \mu_e &\leq 1, \\ \sum_{e=e_{2k}}^t \mu_e &\leq 1, \quad t = e_{2k}, \dots, e_{2k+1} - 1 \\ \sum_{e=t}^{e_{2k+2}} \mu_e &\leq 1, \quad t = e_{2k+1} + 1, \dots, e_{2k+2} - 1 \end{aligned}$$

It is easy to verify that the above system consists of $|\mathcal{E}^*|$ linearly independent inequalities, and $\bar{\mu}$ satisfies each of them at equality, which proves our claim.

Now we claim that any extreme point solution $\bar{\mu}$ of (37) is equivalent to an inequality in (31). First, notice that the constraint matrix of (37) is totally unimodular, thus any vertex of this polyhedron is an integral vector. Also observe that $\bar{\mu}_e \leq 1$ for all $e \in \mathcal{E}$, since $\mu_e \leq 1$ is an inequality of (37) for each $e \in \mathcal{E}^*$. Let k_1 be the first index such that $\bar{\mu}_e \neq 0$. We claim that $\bar{\mu}_{k_1} = 1$. Suppose not, i.e., $\bar{\mu}_{k_1} \leq -1$ (recall that the coordinates of $\bar{\mu}$ are integers). Since $\bar{\mu}$ is an extreme point of M , there must exist a subset L of $|\mathcal{E}^*|$ linearly independent inequalities from (37) that are satisfied at equality in $\bar{\mu}$. Observe that L must contain an inequality involving the variable μ_{k_1} , otherwise this variable may be made arbitrarily negative while still satisfying all inequalities in L , and thus $\bar{\mu}$ would not be an extreme point of M , a contradiction. Since $\bar{\mu}_e = 0$ for $e < k_1$, such an inequality must be of the form $\sum_{e=k_1}^{\ell_1} \mu_e \leq 1$. Since it must hold at equality in $\bar{\mu}$, and $\bar{\mu}_e \leq 1$ as we have already noticed, it follows that $\bar{\mu}$ must admit at least two coordinates q_1 and q_2 such that $k_1 < q_1 < q_2 \leq \ell_1$ with $\bar{\mu}_{q_1} = \bar{\mu}_{q_2} = 1$, and $\bar{\mu}_e = 0$ for $q_1 < e < q_2$. But then, $\bar{\mu}$ would violate the inequality $\sum_{e=q_1}^{q_2} \mu_e \leq 1$, a contradiction.

So, the first non-zero coordinate of $\bar{\mu}$ has value 1. The next nonzero coordinate, say k_2 , cannot have value 1 by the previous argument. So, it must be a negative integer. If it were smaller than -1 , then the sum of coordinates of $\bar{\mu}$ up to and including k_2 would be a negative integer. But then we could argue as above to show that k_2 must be involved in an inequality $\alpha \in L$, and then to reach a similar contradiction as above. Therefore, the second nonzero coordinate of $\bar{\mu}$ must be -1 . Moreover, α must involve a variable μ_{k_3} of value 1 in $\bar{\mu}$, otherwise it cannot hold at equality in $\bar{\mu}$. Continuing this argument if μ still has nonzero coordinates after k_3 , we recognize that μ has the pattern of $1/-1$ as in the inequalities (31). \square

3.2 Separation algorithm for the non-preemptive inequalities

In this section we describe a polynomial time separation procedure for the inequalities (31).

Separation procedure. The main idea is that we find a longest path in a properly defined acyclic digraph. There is a unique source and a unique sink node (indexed by 0 and $2n$, respectively), and there is a node for each event in \mathcal{E}^* . The arcs fall in three categories: (i) there are "starting arcs" from node 0 to every node with index $h \leq 2n - 3$, (ii) "intermediate arcs" starting from some node h with $1 \leq h \leq 2n - 3$, and ending at some node k with $h + 2 \leq k \leq 2n - 1$, and (iii) "terminal arcs" starting at some node h with $3 \leq h \leq 2n - 1$ and ending in the sink node $2n$. The cost of each starting arc is 0. The cost of each intermediate arc (h, k) is $cost(h, k) = \bar{z}_{i, e_h} - \min\{\bar{z}_{i, e_\ell} : \ell = h + 1, \dots, k - 1\}$. Finally, the cost of each terminal arc $(h, 2n)$ is \bar{z}_{i, e_h} .

To separate a vector $\bar{z}_i \in \mathbb{R}^{\mathcal{E}^*}$ we compute the values

$$F(e_k) = \max\{F(e_h) + cost(h, k) : h = 1, \dots, k - 2\}, \quad (38)$$

where $F(1) = F(2) = 0$.

Then, for each $F(e_k)$ we compute $F(e_k) + \bar{z}_{i, e_k}$ and compare it to the length of the longest path with alternating sign pattern found so far. If it is greater, then we store it. In the end, we obtain the largest value of a path.

The computation time for computing the cost of all the intermediate arcs from some fixed node h is $O(n)$. Since the number of arcs is $O(n^2)$, the time complexity of the entire separation procedure is $O(n^2)$.

4 Experiments

This section describes the computational results we obtain with the inequalities described in this paper.

The experiments are conducted on an Intel Core i7-4770 processor with 4 cores and 8 gigabytes of RAM under the 64-bit Ubuntu 12.04 operating system. We use IBM CPLEX 12.6 with one thread and a time limit of 1000 seconds for solving the MILP models.

We use the instances of [5]. These instances were generated according to the following framework. First, the instances have been randomly generated with identity power processing rate functions, i.e. $f_i(b) = b$, $\forall i \in A$. We generated 5 instances of 10 tasks and 10 instances of 20, 25 and 30 tasks according to the following framework. The resource availability B is set to 10 and all other data are randomly generated in their corresponding interval: $W_i \in [1, 1.25 * B]$, $b_i^{min} \in [0, 0.25 * W_i]$, $b_i^{max} \in [b_i^{min}, 2 * b_i^{min}]$, $r_i \in [0, 0.5 * n]$ and $d_i \in [e_i^{min}, e_i^{min} + n]$. Then, we transform them in order to obtain two families of instances with power processing rate functions in the following way. For the first family, we randomly generated the parameters of the function a_i and c_i , $\forall i \in A$, within interval $[1, 10]$ and we set W_i to a random number within $[0, f_i(W_i)]$. For the second family, we randomly generate the parameters of the function a_i , within interval $[1, 10]$, and c_i is set to 0. The required energy W_i is set to $f_i(W_i)$. Experiments are conducted on instances of Family 1 and Family 2.

For the first family, table 1 shows the percentage of instances solved optimally as well as the time needed to solve them for some combinations of inequalities. The time is computed as follow: if the instance is not solved after the time limit, then we set this value to 1000 seconds. In the table, we denote by *Int.* or *I.* the maximum interval inequalities, by *Time* or *T.* the maximum time inequalities, by *Knapsack* or *K.* the inequalities from knapsack problems and by *Preemptive* or *P.* the non-preemptive inequalities.

In this table, we can see, first, that the number of 10-task and 20-task instances solved is similar for all tested combination of inequalities. For the 25-task instances, the results are more heterogeneous but the best results are obtained using inequalities presented in this paper. Finally, the 30-task instances are

ineq. \ #tasks	10		20		25		30	
	time(s)	%opt	time(s)	%opt	time(s)	%opt	time(s)	%opt
<i>Default</i>	0,3	100	164,14	90,9	635,4	55,6	968	10
<i>Int.</i>	0,6	100	182,2	90,9	727,3	55,6	851	20
<i>Time</i>	0,5	100	167,3	90,9	629,5	88,9	961,4	20
<i>Knap.</i>	0,5	100	164,7	90,9	555,1	66,7	845,3	20
<i>Preem.</i>	0,3	100	330,9	72,7	822,4	44,4	914,8	10
<i>I. & T. & K.</i>	0,6	100	154	90,9	389,9	77,8	839,9	20
<i>I. & T. & \bar{P}.</i>	0,6	100	179,8	90,9	454,4	77,8	813,8	60
<i>I. & K. & \bar{P}.</i>	0,5	100	182,2	90,9	759,6	33,3	924,9	20
<i>T. & K. & \bar{P}.</i>	0,5	100	170	90,9	705	66,7	816	50
<i>I. & T. & K. & \bar{P}.</i>	0,9	100	278,9	81,8	510	88,9	802,8	50

Table 1: Instances of Family 1 optimally solved and time needed

ineq. \ #tasks	10		20		25		30	
	%feas.	gap	%feas.	gap	%feas.	gap	%feas.	gap
<i>Default</i>	100	0,23	81,8	0,74	22,2	0,9	0	0
<i>Int.</i>	100	<0,01	100	0,15	44,4	0,69	0	0
<i>Preem.</i>	100	0,6	81,8	0,73	0	0	0	0
<i>I. & K. & \bar{P}.</i>	100	0,09	100	0,68	44,4	0,88	10	0,92

Table 2: Instances of Family 2 solved and relative gap

solved better when the MILP is combined with these inequalities. The best result is obtained with the maximum interval inequalities, the maximum time inequalities and the non-preemptive inequalities.

For the second family, only a few instances are solved optimally. Therefore, we present only the percentage of instances for which a solution is found and the relative gap of this solution when the time limit is reached. These results are presented in table 2. The notation are as in table 1.

5 Conclusion and further researches

In this paper, we have presented 4 set of inequalities improving the existing event-based MILP of [5]. In particular, one of these set is used to give a minimal description of the polyhedra of all feasible assignments to the on/off binary variable for a single activity. Furthermore, for these inequalities, a polynomial separation algorithm is provided. Finally, computational experiments show the effectiveness of the proposed inequalities.

Further researches on this subject are numerous. In particular, find symmetry breaking inequalities along with a minimal description of the polyhedra of all asymmetric feasible assignment of the on/off variable not for a single activity but for all of them. The adaptation of this work to the Resource-Constrained Project Scheduling Problem is also an interesting direction for additional results.

References

- [1] C. Artigues, P. Lopez, and A. Haït (2013). The energy scheduling problem: Industrial case study and constraint propagation techniques. *International Journal of Production Economics*, 143(1): 13-23.
- [2] C. Artigues and P. Lopez (2014). Energetic reasoning for energy-constrained scheduling with a continuous resource. *Journal of Scheduling*, 18(3): 225-241.
- [3] J. Błażewicz, K. H. Ecker, E. Pesch, G. Schmidt and J. Węglarz (2001). *Scheduling computer and manufacturing processes*, Springer-Verlag, Berlin/Heidelberg.
- [4] J. Józefowska, M. Mika, R. Różycki, G. Waligóra and J. Węglarz (1999). Project scheduling under discrete and continuous resources. In: J. Węglarz (Ed.), *Project Scheduling: Recent Models, Algorithms, and Applications*. Kluwer Academic Publishers, Boston, pp. 289-308.

- [5] M. Nattaf, C. Artigues, P. Lopez and D. Rivreau (2015). Energetic reasoning and mixed-integer linear programming for scheduling with a continuous resource and linear efficiency functions. *OR Spectrum*, pp 1–34.
- [6] J. Węglarz, J. Józefowska, M. Mika and G. Waligóra (2009). Project scheduling with finite or infinite number of activity processing modes - A survey. *European Journal of Operational Research*, 208: 177–205.
- [7] J. Błażewicz, M. Machowiak, J. Węglarz, M. Kovalyov and D. Trystram (2004). Scheduling malleable tasks on parallel processors to minimize the makespan. *Annals of Operations Research*, 129(1-4): 65–80.
- [8] C.-U. Fündeling and N. Trautmann (2010). A priority-rule method for project scheduling with work-content constraints. *European Journal of Operational Research*, 203(3): 568–574.
- [9] T. Kis (2005). A branch-and-cut algorithm for scheduling of project with variable-intensity activities. *Mathematical Programming, Series A*, 103: 515–539.
- [10] O. Koné, C. Artigues, P. Lopez and M. Mongeau (2011). Event-based MILP models for resource-constrained project scheduling problems. *Computers & Operations Research*, 38(1): 3–13.
- [11] A. Naber and R. Kolisch (2014). MIP models for resource-constrained project scheduling with flexible resource profiles. *European Journal of Operational Research*, 239: 335–348.
- [12] A. Naber and R. Kolisch (2014). A continuous time model for the resource-constrained project scheduling with flexible resource profiles. *Proceedings of the 14th International Conference on Project Management and Scheduling*, pp. 166–168, Munich, Germany.
- [13] C. Schwindt (2005). *Resource allocation in project management*. Springer, Berlin.