FACULTEIT ECONOMIE EN BEDRIJFSWETENSCHAPPEN

Polyhedral results and branch-and-cut for the resource loading problem

Song G, Kis T, Leus R.

KBI_1908

Polyhedral results and branch-and-cut for the resource loading problem

Guopeng Song[1], Tamás Kis[2], Roel Leus[1*]

[1]ORSTAT, Faculty of Economics and Business, KU Leuven, Belgium
[2]Institute for Computer Science and Control, Hungarian Academy of Sciences, Budapest, Hungary

Abstract

We study the resource loading problem, which arises in tactical capacity planning. In this problem, one has to plan the intensity of execution of a set of orders so as to minimize a cost function that penalizes the resource usage above given capacity limits and the completion of the orders after their due dates. Our main contributions include a novel mixed-integer linear-programming (MIP) based formulation, the investigation of the polyhedra associated with the feasible intensity assignments of individual orders, and a comparison of our branch-and-cut algorithm based on the novel formulation and the related polyhedral results with other MIP formulations. The computational results demonstrate the superiority of our approach. In our formulation and in one of the proofs, we use fundamental results of Egon Balas on disjunctive programming.

Keywords: capacity planning, mixed-integer programming, facets, branch-and-cut

1 Introduction

In the resource loading problem (RLP), a portfolio J of n independent orders (jobs) is to be executed over a time horizon, which is discretized into equal-length time periods (for instance, days or weeks) indexed by t ($t = 1, \ldots, H$). A pre-specified regular workforce capacity C_t is available in each period t. Each order j ($j = 1, \ldots, n$) has a work content denoted as p_j (e.g., man-hours), a release date (period) r_j in which it can start and a due date (period) d_j, where $1 \leq r_j \leq d_j \leq H$. Each order must be executed without preemption. During the execution of order j, an upper bound UB_j and a lower bound LB_j are imposed on the intensity (fraction) of its work content to be performed in each period (with $0 < LB_j \leq UB_j \leq 1$); thus the duration of an order is not fixed,

*Corresponding author. E-mail: guopeng.song@kuleuven.be (G. Song), kis.tamas@sztaki.mta.hu (T. Kis), roel.leus@kuleuven.be (R. Leus).

but rather dependent on the intensity assigned to it in each period of execution. A feasible solution specifies an intensity assignment y_{jt} for each order j and time period t such that there is a starting period s_j and finishing period f_j with the following properties: (i) $y_{jt} = 0$ when $t < s_j$ or $t > f_j$, (ii) $LB_j \leq y_{jt} \leq UB_j$ for $t \in \{s_j, \ldots, f_j\}$, (iii) $s_j \geq r_j$, (iv) $\sum_t y_{jt} = 1$. To express the objective function, let $z_t = \max\{0, \sum_j y_{jt}p_j - C_t\}$, and $T_j = \max\{0, f_j - d_j\}$. The goal is to find a feasible solution which minimizes the cost $\sum_j w_j T_j + \sigma \sum_t z_t$, where w_j is the tardiness penalty per time period for order j, and σ the unit cost of non-regular capacity.

Example 1. Given a planning horizon with $H = 5$, with a regular capacity $C_t = 3$ in each period t, there are three orders with $p_1 = p_3 = 4$ and $p_2 = 5$; $r_1 = r_2 = 1$, $r_3 = 2$; $d_1 = d_2 = 3$ and $d_3 = 4$. As for lower and upper bounds, $LB_1 = 1/4$, $UB_1 = 1/2$; $LB_2 = 1/5$, $UB_2 = 3/5$; $LB_3 = 1/4$, $UB_3 = 3/4$. The tardiness penalties per period are $w_1 = w_2 = w_3 = 5$, and the unit cost of non-regular capacity in each period is $\sigma = 2$. An optimal solution is depicted in Figure 1 with execution intensities $(1/2,\ 1/4,\ 1/4,\ 0,\ 0)^\mathsf{T}$ for order 1, $(2/5,\ 2/5,\ 1/5,\ 0,\ 0)^\mathsf{T}$ for order 2 and $(0,\ 0,\ 1/4,\ 3/4,\ 0)^\mathsf{T}$ for order 3, where one unit of non-regular capacity is used in the first period, rather than delaying any order.

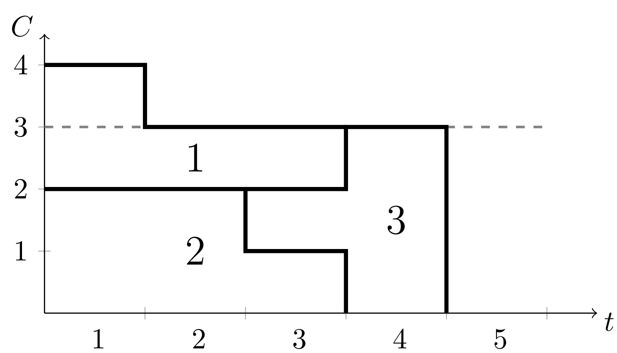

Figure 1: Graphical illustration of an optimal solution to Example 1

The RLP appears in tactical capacity planning, for instance in a multi-project environment. It has been motivated by many real-world applications, e.g., capacity planning at the Royal Netherlands Navy Dockyard [11] and workforce staffing and scheduling for aircraft maintenance [6]. Talla Nobibon et al. [27] study the RLP without allowing order tardiness, and prove the general setting with non-preemption constraint to be NP-complete in the strong sense, while many special cases allowing preemption are shown to be solvable in (pseudo-)polynomial time. Hans studies the RLP in his PhD thesis [18], where preemption is allowed and precedence constraints are considered, and Kis [19] investigates the same problem under the term "project scheduling with variable-intensity activities," and proposes an efficient branch-and-cut algorithm. Some LP-based heuristic methods are proposed in Gademann and Schutten [16] for the preemptive RLP. Wullink et al. [29] incorporate uncertainty into the preemptive RLP and apply a scenario-based approach to solve it.

Some aspects of the RLP have been investigated in other problems by different researchers. The scheduling problem of malleable tasks on parallel processors is studied by Błażewicz et al. [8], where a task may be executed by several processors simultaneously and the processing speed of a task is a nonlinear function of the number of processors allocated to it. Nattaf et al. [23] study a continuous energy-constrained scheduling problem aiming to minimize the total resource consumption under a continuous-time setting, where at any time a minimum and maximum energy requirement is considered for each task, but no job tardiness or non-regular resource usage is allowed. Project scheduling with work-content constraints has been addressed by several authors. Fündeling and Trautmann [15] consider a minimum and maximum amount of resource usage once the activity is started. Naber and Kolisch [22] investigate the resource-constrained project scheduling problem (RCPSP) with flexible resource profiles, and the authors compare several MIP formulations for solving this problem. For a survey of further related work, we refer the interested readers to Talla Nobibon et al. [27].

The remainder of the paper is organized as follows. In Section 2, we describe multiple time-indexed formulations for the RLP, and we propose valid inequalities to improve the formulations. A novel execution-interval formulation is introduced in Section 3, for which a class of valid inequalities is derived that provides a complete description of the polytope of feasible intensity assignments for individual orders, and a polynomial-time separation algorithm is presented. In Section 4, computational results are reported that show the effectiveness of the branch-and-cut algorithm based on our polyhedral results. Finally, we provide a summary and conclusions in Section 5.

2 Time-indexed formulations

In operational scheduling with fixed job processing times, time-indexed formulations have been extensively studied. Sousa [25] discusses various time-indexed formulations for single machine scheduling, and establishes the equivalence of the formulations in terms of the strength of their LP relaxation. For RCPSP, similar polyhedral results are presented by Artigues [3]. Computational results for time-indexed MIP formulations of different variants of RCPSP have also been reported (see Burgelman and Vanhoucke [9] and Naber and Kolisch [22], for example).

Decisions in the RLP, however, involve when to execute each order and how to fit the intensities of the orders into each time period. The RLP is formulated using three different sets of time-indexed variables in this section. Different from scheduling with fixed job durations, for each order, an execution interval (consecutive time periods in which the order is executed) needs to be identified.

2.1 Pulse formulation

The first model uses the binary pulse variables $s_{jt} \in \{0,1\}$ and $f_{jt} \in \{0,1\}$ for each order j, which take value 1 only if the order starts, respectively finishes, in time period t. This modeling choice has been extensively applied for scheduling problems (such as in [10, 20, 21, 24, 26]), where only the start pulse is needed for each job if the job durations are given and fixed. We also use continuous variables $y_{jt} \in [0,1]$ denoting the intensity (fraction) of order j in period t, variables T_j representing the tardiness of each order j, and z_t the non-regular capacity required for time period t. A formulation for RLP with pulse variables can be stated as follows.

$$\min \quad \sum_{j \in J} w_j T_j + \sigma \cdot \sum_{t=1}^{H} z_t \tag{1}$$

$$\text{s.t.} \quad \sum_{t=r_j}^{H} s_{jt} = \sum_{t=r_j}^{H} f_{jt} = 1 \qquad \forall j \in J \tag{2}$$

$$\sum_{k=r_j}^{t} f_{jk} \leq \sum_{k=r_j}^{t} s_{jk} \qquad \forall j \in J,\ t = r_j, \ldots, H \tag{3}$$

$$T_j \geq \sum_{t=r_j}^{H} f_{jt} \cdot t - d_j \qquad \forall j \in J \tag{4}$$

$$LB_j \cdot (\sum_{k=r_j}^{t} s_{jk} - \sum_{k=r_j}^{t-1} f_{jk}) \leq y_{jt} \leq UB_j \cdot (\sum_{k=r_j}^{t} s_{jk} - \sum_{k=r_j}^{t-1} f_{jk}) \quad \forall j \in J,\ t = r_j, \ldots, H \tag{5}$$

$$\sum_{t=r_j}^{H} y_{jt} = 1 \quad \forall j \in J \tag{6}$$

$$z_t \geq \sum_{j \in J} y_{jt} \cdot p_j - C_t \quad t = 1, \ldots, H \tag{7}$$

$$y_{jt} \geq 0 \quad \forall j \in J,\ t = r_j, \ldots, H \tag{8}$$

$$z_t \geq 0 \quad t = 1, \ldots, H \tag{9}$$

$$T_j \geq 0 \quad \forall j \in J \tag{10}$$

$$s_{jt}, f_{jt} \in \{0, 1\} \quad \forall j \in J,\ t = r_j, \ldots, H \tag{11}$$

The objective function (1) consists of two terms representing the penalty for order tardiness and the cost of hiring non-regular capacity, respectively. Constraints (2) stipulate that each order has exactly one start time and one finish time. Constraints (3) enforce that orders cannot finish before their start times. Constraints (4) determine the tardiness of each order, and (5) set the bounds on the working intensity for each order in each time period within its time window, which also guarantees that the order is executed without preemption. The term $\sum_{k=r_j}^{t} s_{jk} - \sum_{k=r_j}^{t-1} f_{jk}$ in (5), indicates the processing status of order j in period t (whether order j is executed in period t). Constraints (6) ensure that each order is finished within the time horizon. Constraints (7) determine the amount of non-regular resource capacity required in each time period.

2.2 Step formulation

In the second formulation, we use binary step variables $s^a_{jt} \in \{0, 1\}$ and $f^a_{jt} \in \{0, 1\}$, denoting whether order j has started, resp. finished, by period t. In other words, s^a_{jt} takes value 1 only if order j starts in a time period before or in t, and $f^a_{jt} = 1$ only when order j is completed before or in t. Step variables have also been used in the scheduling literature (see for example [7, 22]). With s^a_{jt} and f^a_{jt}, the processing status of order j in time period t is $s^a_{jt} - f^a_{j,t-1}$. We formulate the RLP with step variables as follows.

$$\min \quad (1)$$

s.t. (6)–(10)

$$s^a_{jt} \leq s^a_{j,t+1} \qquad \forall j \in J,\ t = r_j, \dots, H-1 \tag{12}$$

$$f^a_{jt} \leq f^a_{j,t+1} \qquad \forall j \in J,\ t = r_j, \dots, H-1 \tag{13}$$

$$s^a_{jH} = f^a_{jH} = 1 \qquad \forall j \in J \tag{14}$$

$$f^a_{jt} \leq s^a_{jt} \qquad \forall j \in J,\ t = r_j, \dots, H \tag{15}$$

$$T_j \geq (H - \sum_{t=r_j}^{H} f^a_{jt} + 1) - d_j \qquad \forall j \in J \tag{16}$$

$$LB_j \cdot (s^a_{jt} - f^a_{j,t-1}) \leq y_{jt} \leq UB_j \cdot (s^a_{jt} - f^a_{j,t-1}) \qquad \forall j \in J,\ t = r_j, \dots, H \tag{17}$$

$$s^a_{jt}, f^a_{jt} \in \{0, 1\} \qquad \forall j \in J,\ t = r_j, \dots, H \tag{18}$$

Constraints (12) and (13) impose that once an order has started or finished, the status is retained until the end of the planning horizon. Constraints (14) stipulate that the orders have to be executed by the end of the planning horizon. Equations (15) imply that orders cannot finish before they have started. Constraint sets (16) and (17) correspond with constraints (4) and (5).

2.3 Mask formulation

The third time-indexed formulation involves binary decision variables x_{jt} indicating whether order j is executed in time period t (in which case $x_{jt} = 1$); these are commonly referred to as mask (on/off) variables. Since preemption is not allowed, a feasible binary vector x_j for order j has consecutive 1s in a specific time interval and 0s for all the other entries. This modeling choice has been used for the RLP by Talla Nobibon et al. [27], and has also been widely applied for other scheduling problems (see for example [2, 19, 25]). With the mask variables, the RLP can be formulated as follows.

min (1)

s.t. (6)–(10)

$$u_{jH} \geq x_{jH} \qquad \forall j \in J \tag{19}$$

$$u_{jt} \geq x_{jt} - x_{j,t+1} \qquad \forall j \in J,\ t = r_j, \dots, H-1 \tag{20}$$

$$\sum_{t=r_j}^{H} u_{jt} = 1 \qquad \forall j \in J \tag{21}$$

$$T_j \geq \sum_{t=r_j}^{H} u_{jt} \cdot t - d_j \qquad \forall j \in J \tag{22}$$

$$LB_j \cdot x_{jt} \leq y_{jt} \leq UB_j \cdot x_{jt} \qquad \forall j \in J,\ t = r_j, \ldots, H \tag{23}$$

$$u_{jt} \geq 0 \qquad \forall j \in J,\ t = r_j, \ldots, H \tag{24}$$

$$x_{jt} \in \{0, 1\} \qquad \forall j \in J,\ t = r_j, \ldots, H \tag{25}$$

Constraints (19)–(21) use auxiliary variables u_{jt} to indicate when each order finishes, and guarantee that there is only one finish period throughout the planning horizon so that the non-preemption constraint is respected. These constraints imply that the non-zero mask variables should always be consecutive. Constraints (22) and (23) are in accordance to constraints (4) and (5) respectively.

2.4 Strength of the formulations

Proposition 1. *In terms of LP relaxation, the three time-indexed formulations are equivalent.*

Proof. Any feasible solution to the LP relaxation of the pulse formulation can be transformed into a feasible solution to the LP relaxation of the mask formulation, with $x_{jt} = \sum_{k=r_j}^{t} s_{jk} - \sum_{k=r_j}^{t-1} f_{jk}$ and $u_{jt} = f_{jt}$. The inverse transformation can be given as $s_{jt} = x_{jt} - x_{j,t-1} + u_{j,t-1}$ and $f_{jt} = u_{jt}$. Therefore, the LP relaxations of the pulse and the mask formulation lead to the same lower bound on the optimum.

Similarly, for the pulse and step formulation, we have $s^a_{jt} = \sum_{k=r_j}^{t} s_{jk}$ and $f^a_{jt} = \sum_{k=r_j}^{t} f_{jk}$, and conversely the inverse transformation is $s_{jt} = s^a_{jt} - s^a_{j,t-1}$ and $f_{jt} = f^a_{jt} - f^a_{j,t-1}$. Given any fractional solution to the LP relaxation of the step formulation, the transformation always provides a feasible solution to the LP relaxation of the pulse formulation, and vice versa. Therefore, the LP relaxations yield the same lower bound.

The mutual transformations between the mask formulation and the step formulation are given by $x_{jt} = s^a_{jt} - f^a_{j,t-1}$, $u_{jt} = f^a_{jt} - f^a_{j,t-1}$, and by $s^a_{jt} = x_{jt} + \sum_{k=r_j}^{t-1} u_{jk}$, $f^a_{jt} = \sum_{k=r_j}^{t} u_{jk}$, which completes the proof. □

Proposition 1 generalizes the previous results on the equivalence of "pulse," "mask" and "step" formulation for scheduling problems with fixed activity durations [3, 25] to scheduling variable-duration activities.

2.5 Valid inequalities

In this section, two classes of valid inequalities are proposed to strengthen the time-indexed formulations. Since the three sets of variables are mutually transformable, we present the inequalities with the pulse variables; equivalent inequalities can be applied in the other two formulations using the aforementioned affine transformations.

From the intensity of order execution, a lower and upper bound on the length of the execution interval for each order can be derived as $\underline{l_j} = \lceil 1/UB_j \rceil$ and $\overline{l_j} = \lfloor 1/LB_j \rfloor$. The following set of inequalities can strengthen the formulation.

$$\underline{l_j} \leq \sum_{t=r_j}^{H} (f_{jt} - s_{jt}) \cdot t + 1 \leq \overline{l_j} \qquad \forall j \in J \tag{26}$$

Proposition 2. *Constraints* (26) *are valid inequalities for the RLP.*

Proof. Inequalities (26) indicate that the execution-interval length of each order j cannot be shorter than $\underline{l_j}$ nor longer than $\overline{l_j}$. Equivalently, $\forall j \in J$, in the mask formulation inequality $\underline{l_j} \leq \sum_{t=r_j}^{H} x_{jt} \leq \overline{l_j}$ is valid, and $\underline{l_j} \leq \sum_{t=r_j}^{H} (s_{jt}^a - f_{jt}^a) + 1 \leq \overline{l_j}$ is valid for the step formulation. $\square$

A set of disaggregated inequalities is also introduced below to further tighten the formulation.

$$\begin{cases} s_{jt} - \displaystyle\sum_{k=t+\underline{l_j}-1}^{\min\{H,\ t+\overline{l_j}-1\}} f_{jk} \leq 0 & \forall j \in J,\ t = r_j, \ldots, H \\ \displaystyle\sum_{k=\max\{r_j,\ t-\overline{l_j}+1\}}^{t-\underline{l_j}+1} s_{jk} - f_{jt} \geq 0 & \forall j \in J,\ t = r_j, \ldots, H \end{cases} \tag{27}$$

Proposition 3. *Constraints* (27) *are valid inequalities for the RLP.*

Proof. Given any feasible solution expressed by the pulse variables and an arbitrary order j, there is one start period t^* such that $s_{jt^*} = 1$ and $s_{jt|t \neq t^*} = 0$. The finish period of this order j then lies within interval $[t^* + \underline{l_j} - 1,\ t^* + \overline{l_j} - 1]$, and therefore inequality $s_{jt} \leq \sum_{k=t+\underline{l_j}-1}^{\min\{H,\ t+\overline{l_j}-1\}} f_{jk}$ is valid for any time period in the horizon. In a similar way, we can show that $\sum_{k=\max\{r_j,\ t-\overline{l_j}+1\}}^{t-\underline{l_j}+1} s_{jk} \geq f_{jt}$ is also valid. $\square$

Constraints (26) and (27) both require that the execution-interval length be between $\underline{l_j}$ and $\overline{l_j}$, but neither set can replace the other as they cut off different fractional solutions.

Proposition 4. *Neither* (26) *nor* (27) *implies the other.*

Proof. Given an order j on a planning horizon with five periods, where $r_j = 1$ with $UB_j = 1$ ($\underline{l_j} = 1$) and $LB_j = 0.5$ ($\overline{l_j} = 2$), a fractional solution $s_j = (0.5,\ 0,\ 0.5,\ 0,\ 0)^\mathsf{T}$ and $f_j = (0,\ 0,\ 1,\ 0,\ 0)^\mathsf{T}$ is feasible for (26), but cut off by (27). Another fractional solution $s_j = (0.4,\ 0.2,\ 0,\ 0.4,\ 0)^\mathsf{T}$ and $f_j = (0,\ 0.4,\ 0,\ 0.2,\ 0.4)^\mathsf{T}$, on the other hand, respects (27) but not (26). Therefore, inequalities (26) and (27) are not interchangeable. □

3 Execution-interval formulation and polyhedral results

In this section, we propose a novel execution-interval formulation for the RLP, and we aim to provide a complete description of the convex hull of the feasible intensity assignments for each individual order by means of facet-defining inequalities.

3.1 Execution-interval formulation

A feasible execution interval of consecutive time periods $\{k, \dots, \ell\}$ for order j must satisfy $r_j \le k \le \ell \le H$, $(\ell - k + 1) \cdot LB_j \le 1$, and $(\ell - k + 1) \cdot UB_j \ge 1$. Denote the set of all feasible execution intervals of order j as $\mathcal{E}_j$, where each tuple $(k, \ell) \in \mathcal{E}_j$ refers to the execution interval $\{k, \dots, \ell\}$. In a valid plan, exactly one execution interval $(k, \ell) \in \mathcal{E}_j$ is chosen for each order j, and the intensity y_{jt} of order j satisfies $y_{jt} = 0$ for $t \notin \{k, \dots, \ell\}$, $LB_j \le y_{jt} \le UB_j$ for all $t \in \{k, \dots, \ell\}$, and $\sum_{t=1}^{H} y_{jt} = 1$. The variables $a^j_{k\ell}$ indicate this choice, i.e., $a^j_{k\ell} = 1$ if and only if the chosen execution interval for order j is $(k, \ell) \in \mathcal{E}_j$. The tardiness $T^j_{k\ell}$ associated to execution interval $(k, \ell) \in \mathcal{E}_j$ is $\max\{0, \ell - d_j\}$. With these preliminaries, the execution-interval formulation is as follows:

$$\min \quad \sum_{j \in J} \sum_{(k,\ell) \in \mathcal{E}_j} w_j \cdot T^j_{k\ell} \cdot a^j_{k\ell} + \sigma \cdot \sum_{t=1}^{H} z_t \tag{28}$$

$$\text{s.t.} \quad (6)\text{–}(9)$$

$$LB_j \cdot \sum_{(k,\ell) \in \mathcal{E}_j:\ k \le t \le \ell} a^j_{k\ell} \le y_{jt} \le UB_j \cdot \sum_{(k,\ell) \in \mathcal{E}_j:\ k \le t \le \ell} a^j_{k\ell} \qquad j \in J,\ t = r_j, \dots, H \tag{29}$$

$$\sum_{(k,\ell) \in \mathcal{E}_j} a^j_{k\ell} = 1 \qquad j \in J \tag{30}$$

$$a^j_{k\ell} \in \{0,1\} \qquad j \in J,\ (k,\ell) \in \mathcal{E}_j \tag{31}$$

Proposition 5. *The execution-interval formulation is stronger than the time-indexed formulations.*

Proof. Since the time-indexed formulations are equivalent regarding their relaxation strength, we take the pulse formulation and compare with the execution-interval formulation. Denote the set of feasible solutions of the LP relaxation of the pulse formulation augmented with the inequalities (26) and (27) by P_{pulse}, and for the execution-interval formulation by P_{EI}. Any solution in P_{EI} can be projected to P_{pulse} with $s_{jt} = \sum_{(t,\ell)\in\mathcal{E}_j} a^j_{t\ell}$ and $f_{jt} = \sum_{(k,t)\in\mathcal{E}_j} a^j_{kt}$. This transformation is not invertible in general, we can have $P_{EI} \subset P_{pulse}$. An example is established here to illustrate the proper inclusion: given an order j on a planning horizon with seven periods, where $r_j = 1$ with $UB_j = 1/2$ ($\underline{l_j} = 2$) and $LB_j = 1/3$ ($\overline{l_j} = 3$). Consider the vector $y_j = (0.36,\ 0.24,\ 0.24,\ 0,\ 0.00\bar{8},\ 0.07\bar{5},\ 0.07\bar{5})$. If we set $s_j = (0.72,\ 0,\ 0.02\bar{6},\ 0,\ 0.02\bar{6},\ 0.22\bar{6},\ 0)$ and $f_j = (0,\ 0.02\bar{6},\ 0.72,\ 0,\ 0.02\bar{6}, 0,\ 0.22\bar{6})$, then $(y_j, s_j, f_j) \in P_{pulse}$, but there exists no vector a^j such that $(y_j, a^j) \in P_{EI}$.

Similarly, we have $x_{jt} = \sum_{(k,\ell)\in\mathcal{E}_j:\ k\le t\le \ell} a^j_{k\ell}$ for the mask variables, and for the step variables $s^a_{jt} = \sum_{k=r_j}^{t} \sum_{(k,\ell)\in\mathcal{E}_j} a^j_{k\ell}$ and $f^a_{jt} = \sum_{\ell=r_j}^{t} \sum_{(k,\ell)\in\mathcal{E}_j} a^j_{k\ell}$. Therefore, the execution-interval formulation is stronger than the time-indexed formulations. □

One can interpret the execution-interval formulation as a disjunctive program (see [5]), where for each order j there is a set $\mathcal{E}_j$ of alternatives from which exactly one must be chosen.

3.2 Convex hull for individual orders and valid inequalities

Given an arbitrary order j, we consider the following subsystem, where index j is omitted for brevity:

$$LB \cdot \sum_{(k,\ell)\in\mathcal{E}:k\le t\le \ell} a_{k\ell} \le y_t \le UB \cdot \sum_{(k,\ell)\in\mathcal{E}:k\le t\le \ell} a_{k\ell} \qquad t = 1,\dots,H \tag{32}$$

$$\sum_{t=1}^{H} y_t = 1 \tag{33}$$

$$\sum_{(k,\ell)\in\mathcal{E}} a_{k\ell} = 1 \tag{34}$$

$$y_t \ge 0 \qquad t = 1,\dots,H \tag{35}$$

$$a_{k\ell} \in \{0,1\} \qquad (k,\ell) \in \mathcal{E} \tag{36}$$

Let $\mathcal{S}$ be the set of feasible solutions, i.e., the set of all $(y,a) \subseteq \mathbb{R}_+^H \times \{0,1\}^{|\mathcal{E}|}$ vectors that satisfy the constraints (32)–(34). Our goal is to determine all the facets of the polyhedron $P = \operatorname{conv}(\mathcal{S})$.

We start with stronger lower and upper bounds on each intensity variable y_t:

$$y_t \leq \sum_{(k,\ell)\in\mathcal{E}:k\leq t\leq\ell} UB_{k\ell} \cdot a_{k\ell} \tag{37}$$

$$y_t \geq \sum_{(k,\ell)\in\mathcal{E}:k\leq t\leq\ell} LB_{k\ell} \cdot a_{k\ell} \tag{38}$$

where $UB_{k\ell}$ and $LB_{k\ell}$ are defined as follows:

$$UB_{k\ell} \coloneqq \min\{UB,\ 1 - LB \cdot (\ell - k)\} \tag{39}$$

$$LB_{k\ell} \coloneqq \max\{LB,\ 1 - UB \cdot (\ell - k)\} \tag{40}$$

Proposition 6. *Inequalities* (37) *and* (38) *are valid for* P.

Proof. First we verify (37). Given any feasible solution in $\mathcal{S}$, there is exactly one execution interval $\{k^*,\ldots,\ell^*\}$ selected. Given any period t, if $k^* \leq t \leq \ell^*$, there are two easy upper bounds on y_t: UB, and the remaining fraction when the lower bound is taken for each period $\{k^*,\ldots,\ell^*\} \setminus t$, which is $1 - LB \cdot (\ell^* - k^*)$. If $t \in \{1,\ldots,H\} \setminus \{k^*,\ldots,\ell^*\}$ then $y_t = 0$. Therefore, for any period t, summing over all execution intervals including t leads to inequality (37), which is thus valid for $\mathcal{S}$, and thus for P. Analogously, inequalities (38) are also valid for P. □

More generally, for an arbitrary subset $S \subseteq \{1,\ldots,H\}$ of time periods, the intensity assigned to set S can be bounded in a similar way. If we define the complement $\bar{S} \coloneqq \{1,\ldots,H\} \setminus S$ then the bounds on the intensity assigned to S can be computed as follows:

$$\sum_{\tau\in S} y_\tau \leq \sum_{(k,\ell)\in\mathcal{E}} a_{k\ell} \cdot \min\Big\{UB_{k\ell} \cdot \big|S \cap \{k,\ldots,\ell\}\big|,\ 1 - LB_{k\ell} \cdot \big|\bar{S} \cap \{k,\ldots,\ell\}\big|\Big\} \tag{41}$$

$$\sum_{\tau\in S} y_\tau \geq \sum_{(k,\ell)\in\mathcal{E}} a_{k\ell} \cdot \max\Big\{LB_{k\ell} \cdot \big|S \cap \{k,\ldots,\ell\}\big|,\ 1 - UB_{k\ell} \cdot \big|\bar{S} \cap \{k,\ldots,\ell\}\big|\Big\} \tag{42}$$

Proposition 7. *Inequalities* (41) *and* (42) *are valid for* P.

Proof. A generalization of Proposition 6, the proof is omitted. $\square$

Proposition 8. *The inequality* (41) *for a fixed set* S *is equivalent to the inequality* (42) *for* $\bar{S}$.

Proof. Given $\sum_{\tau=1}^{H} y_\tau = \sum_{\tau \in S} y_\tau + \sum_{\tau \in \bar{S}} y_\tau = 1$, $\sum_{(k,\ell)\in\mathcal{E}} a_{k\ell} = 1$, and the lower bounding inequality (42) on an arbitrary set S:

$$\begin{aligned}\sum_{\tau\in\bar{S}} y_\tau = 1 - \sum_{\tau \in S} y_\tau \le 1 - \sum_{(k,\ell)\in\mathcal{E}} a_{k\ell} \cdot \max\Big\{LB_{k\ell}\cdot\big|S\cap\{k,\dots,\ell\}\big|,\ 1 - UB_{k\ell}\cdot\big|\bar{S}\cap\{k,\dots,\ell\}\big|\Big\}\\ = \sum_{(k,\ell)\in\mathcal{E}} a_{k\ell} - \sum_{(k,\ell)\in\mathcal{E}} a_{k\ell}\cdot\max\Big\{LB_{k\ell}\cdot\big|S\cap\{k,\dots,\ell\}\big|,\ 1 - UB_{k\ell}\cdot\big|\bar{S}\cap\{k,\dots,\ell\}\big|\Big\}\\ = \sum_{(k,\ell)\in\mathcal{E}} a_{k\ell}\cdot\min\Big\{1 - LB_{k\ell}\cdot\big|S\cap\{k,\dots,\ell\}\big|,\ UB_{k\ell}\cdot\big|\bar{S}\cap\{k,\dots,\ell\}\big|\Big\}\end{aligned}$$

This is exactly the upper bounding inequality (41) for the corresponding complement set $\bar{S}$. In the same way, inequality (41) on set S also implies inequality (42) for its complement $\bar{S}$. $\square$

3.3 A linear representation of P

Consider polyhedron $\mathcal{P} := \big\{(y,a) \in \mathbb{R}_+^H \times [0,1]^{|\mathcal{E}|} \mid (y,a)$ satisfies (33), (34) and (42) for all $S \subseteq \{1,\dots,H\}\big\}$. Our goal is to show that $\mathcal{P}$ is equivalent to P. In order to achieve this, we first establish a necessary and sufficient condition for a solution (y,a) to be in P.

Based on a given solution (y,a), we define a capacitated network $G(y,a) = (V,E,b,c)$, with the set of nodes V consisting of the source node s, the sink q, a node $v_{k\ell}$ for each $(k,\ell)\in\mathcal{E}$ and a node w_t for each $t \in \{1,\dots,H\}$. Each node $v_{k\ell}$ is connected from the source s with capacity $c(s,v_{k\ell}) = a_{k\ell}$, while each node w_t is linked to the sink q with capacity $c(w_t,q) = y_t$. From each $v_{k\ell}$, there is an arc pointed to every $t\in\{k,\dots,\ell\}$, and the flow through arc $(v_{k\ell},w_t)$ has lower bound $b(v_{k\ell},w_t) = a_{k\ell}LB_{k\ell}$ and upper bound $c(v_{k\ell},w_t) = a_{k\ell}UB_{k\ell}$, respectively. The lower bounds on all other arcs are 0. Figure 2 shows how network $G(y,a)$ is constructed.

Lemma 1. $(y,a)\in P$ *if and only if there exists a feasible flow in* $G(y,a)$ *of value 1.*

Proof. For each $(k,\ell)\in\mathcal{E}$, construct the following polyhedron $P_{(k,\ell)} = \big\{(y,a)\in\mathbb{R}_+^H\times[0,1]^{|\mathcal{E}|} \mid a_{k\ell} = 1, \sum_{(k',\ell')\in\mathcal{E}\setminus(k,\ell)} a_{k'\ell'} = 0, \sum_{t=k}^{\ell} y_t = 1, LB_{k\ell} \le y_t \le UB_{k\ell}$ if $t\in\{k,\dots,\ell\}$ and $y_t = 0$ otherwise$\big\}$.

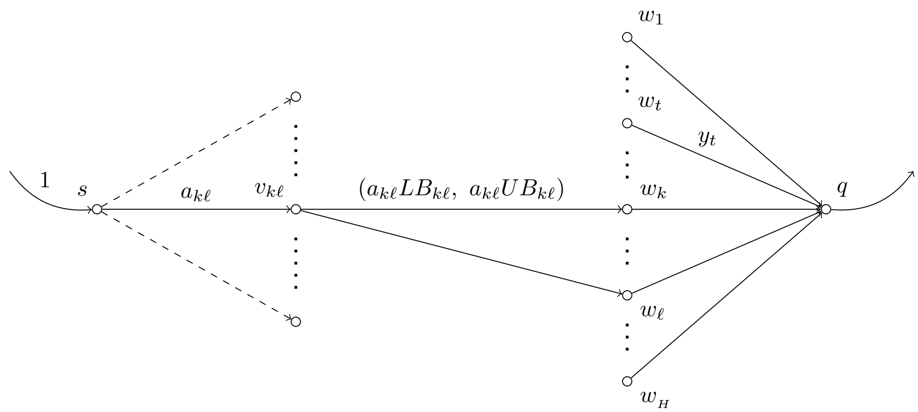

Figure 2: An illustration of network $G(y,a)$

Let $\text{conv}\left(\bigcup_{(k,\ell)\in\mathcal{E}} P_{(k,\ell)}\right)$ denote the convex hull of the union of the polytopes $P_{(k,\ell)}$. Clearly, this is the set of those vectors $(y,a) \in \mathbb{R}_+^H \times [0,1]^{|\mathcal{E}|}$, for which there exist vectors $(y^{(k,\ell)}, a^{(k,\ell)}) \in \mathbb{R}_+^H \times [0,1]^{|\mathcal{E}|}$, $(k,\ell) \in \mathcal{E}$ satisfying

$$
\begin{aligned}
(y,a) - \sum_{(k,\ell)\in\mathcal{E}} (y^{(k,\ell)}, a^{(k,\ell)}) &= 0 \\
\sum_{(k',\ell')\in\mathcal{E}\setminus(k,\ell)} a_{k'\ell'}^{(k,\ell)} &= 0 \qquad (k,\ell)\in\mathcal{E} \\
\sum_{t=k}^{\ell} y_t^{(k,\ell)} &= a_{k\ell}^{(k,\ell)} \qquad (k,\ell)\in\mathcal{E} \\
a_{k\ell}^{(k,\ell)} \cdot LB_{k\ell} \le y_t^{(k,\ell)} &\le a_{k\ell}^{(k,\ell)} \cdot UB_{k\ell} \qquad t\in\{k,\dots,\ell\}, (k,\ell)\in\mathcal{E} \\
\sum_{t\notin\{k,\dots,\ell\}} y_t^{(k,\ell)} &= 0 \qquad (k,\ell)\in\mathcal{E} \\
\sum_{(k,\ell)\in\mathcal{E}} a_{k\ell}^{(k,\ell)} &= 1
\end{aligned}
\tag{43}
$$

One can easily verify that system (43) describes exactly a feasible flow of value 1 in network $G(y,a)$. Based on the theorem on the convex hull of the union of polyhedra in disjunctive programming (see Theorem 2.1 of Balas [4]), $P = \text{conv}\left(\bigcup_{(k,\ell)\in\mathcal{E}} P_{(k,\ell)}\right)$, and (43) is an extended formulation for P. □

In order to remove the lower bounds on all the arcs $(v_{k\ell}, w_t)$ in network $G(y,a)$, one can transform $G(y,a)$ into $G'(y,a)$ in the following manner: for each $(k,\ell)\in\mathcal{E}$ and each $t\in\{k,\dots,\ell\}$, introduce an arc connecting s to w_t with capacity $c(s,w_t) = a_{k\ell}LB_{k\ell}$; change the capacity of the arc linking the corresponding nodes $v_{k\ell}$ and w_t to $c(v_{k\ell}, w_t) = a_{k\ell}(UB_{k\ell} - LB_{k\ell})$; and add a new arc from each $v_{k\ell}$ to q with capacity $c(v_{k\ell}, q) = a_{k\ell}(\ell - k + 1)LB_{k\ell}$. The transformed network $G'(y, a)$ of $G(y, a)$ is illustrated in Figure 3. This is a standard transformation (see section 6.7 of Ahuja et al. [1]), which leads to the following lemma.

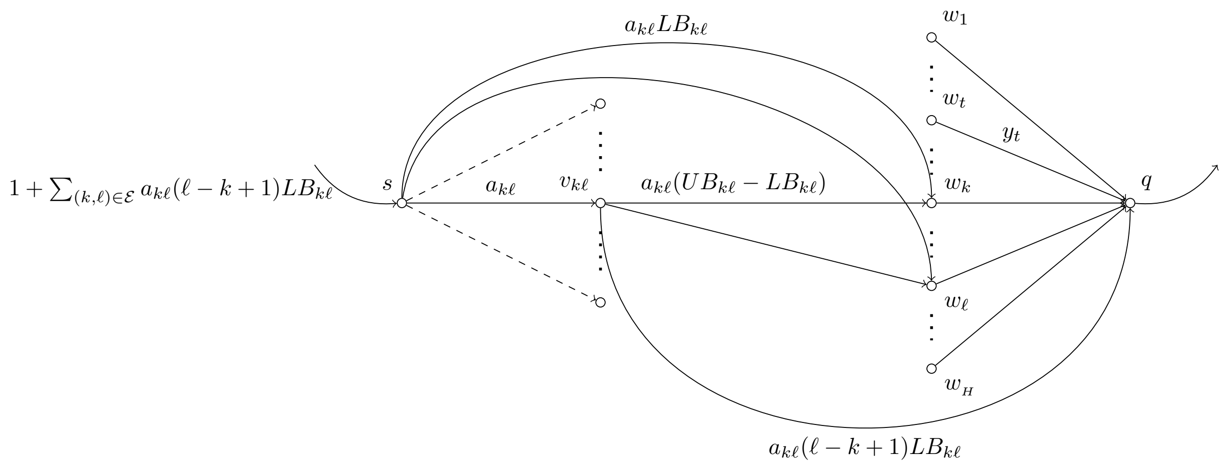

Figure 3: The transformed network $G'(y, a)$ of $G(y, a)$

Lemma 2. *There exists a feasible flow in $G(y, a)$ of value 1 if and only if one can find a feasible flow in $G'(y, a)$ of value $1 + \sum_{(k,\ell)\in\mathcal{E}} a_{k\ell}(\ell - k + 1)LB_{k\ell}$ with all arcs (s, w_t) and $(v_{k\ell}, q)$ saturated.*

Theorem 1. $\mathcal{P} = P$.

Proof. Since inequalities (33), (34), and (42) are all valid for P, one can straightforwardly get $P \subseteq \mathcal{P}$, thus we focus on proving $\mathcal{P} \subseteq P$.

Hereinafter, we use $v_{k\ell}$ and $(k, \ell) \in \mathcal{E}$, w_t and period t interchangeably. Following Lemma 1 and 2, we consider an arbitrary s-q cut $\left[\{s\}\cup S_v \cup S_w, \{q\}\cup \bar{S}_v \cup \bar{S}_w\right]$ in $G'(y, a)$, which corresponds with a partition of V, where nodes $v_{k\ell}$ are partitioned into sets S_v and $\bar{S}_v$ and the w_t are partitioned into S_w and $\bar{S}_w$. Based on the MAX-FLOW MIN-CUT Theorem of Ford and Fulkerson [14], there exists a feasible flow of $1+\sum_{(k,\ell)\in\mathcal{E}} a_{k\ell}(\ell-k+1)LB_{k\ell}$ in $G'(y, a)$ if and only if the capacity $c(S_v\cup S_w, \bar{S}_v\cup \bar{S}_w)$

of any cut $\left[\{s\} \cup S_v \cup S_w, \{q\} \cup \bar{S}_v \cup \bar{S}_w\right]$ is not smaller than $1 + \sum_{(k,\ell)\in\mathcal{E}} a_{k\ell}(\ell - k + 1)LB_{k\ell}$:

$$\begin{aligned}
c(S_v \cup S_w, \bar{S}_v \cup \bar{S}_w) = &\sum_{t\in\bar{S}_w} \sum_{(k,\ell)\in\mathcal{E}|t\in\{k,\dots,\ell\}} a_{k\ell} \cdot LB_{k\ell} + \sum_{(k,\ell)\in\bar{S}_v} a_{k\ell} + \sum_{(k,\ell)\in S_v} a_{k\ell} \cdot (\ell - k + 1) \cdot LB_{k\ell} \\
&+ \sum_{(k,\ell)\in S_v} \sum_{t\in\bar{S}_w\cap\{k,\dots,\ell\}} a_{k\ell} \cdot (UB_{k\ell} - LB_{k\ell}) + \sum_{t\in S_w} y_t \\
&\geq 1 + \sum_{(k,\ell)\in\mathcal{E}} a_{k\ell} \cdot (\ell - k + 1) \cdot LB_{k\ell}
\end{aligned} \tag{44}$$

With the terms rearranged, we obtain the following:

$$\begin{aligned}
\sum_{t\in S_w} y_t \geq &- \sum_{t\in\bar{S}_w} \sum_{(k,\ell)\in\mathcal{E}|t\in\{k,\dots,\ell\}} a_{k\ell} \cdot LB_{k\ell} - (1 - \sum_{(k,\ell)\in S_v} a_{k\ell}) - \sum_{(k,\ell)\in S_v} a_{k\ell} \cdot (\ell - k + 1) \cdot LB_{k\ell} \\
&- \sum_{(k,\ell)\in S_v} \sum_{t\in\bar{S}_w\cap\{k,\dots,\ell\}} a_{k\ell} \cdot (UB_{k\ell} - LB_{k\ell}) + 1 + \sum_{(k,\ell)\in\mathcal{E}} a_{k\ell} \cdot (\ell - k + 1) \cdot LB_{k\ell} \\
= &- \sum_{(k,\ell)\in\mathcal{E}} \sum_{t\in\bar{S}_w\cap\{k,\dots,\ell\}} a_{k\ell} \cdot LB_{k\ell} + \sum_{(k,\ell)\in S_v} a_{k\ell} + \sum_{(k,\ell)\in\bar{S}_v} a_{k\ell} \cdot (\ell - k + 1) \cdot LB_{k\ell} \\
&- \sum_{(k,\ell)\in S_v} \sum_{t\in\bar{S}_w\cap\{k,\dots,\ell\}} a_{k\ell} \cdot (UB_{k\ell} - LB_{k\ell}) \\
= &\sum_{(k,\ell)\in S_v} a_{k\ell} \cdot \left[- \sum_{t\in\bar{S}_w\cap\{k,\dots,\ell\}} LB_{k\ell} + 1 - \sum_{t\in\bar{S}_w\cap\{k,\dots,\ell\}} (UB_{k\ell} - LB_{k\ell}) \right] \\
&+ \sum_{(k,\ell)\in\bar{S}_v} a_{k\ell} \cdot \left[- \sum_{t\in\bar{S}_w\cap\{k,\dots,\ell\}} LB_{k\ell} + (\ell - k + 1) \cdot LB_{k\ell} \right] \\
= &\sum_{(k,\ell)\in S_v} a_{k\ell} \cdot \left(1 - UB_{k\ell} \cdot \left|\bar{S}_w \cap \{k,\dots,\ell\}\right|\right) + \sum_{(k,\ell)\in\bar{S}_v} a_{k\ell} \cdot \left(LB_{k\ell} \cdot \left|S_w \cap \{k,\dots,\ell\}\right|\right)
\end{aligned} \tag{45}$$

where $\sum_{t\in\bar{S}_w} \sum_{(k,\ell)\in\mathcal{E}|t\in\{k,\dots,\ell\}} a_{k\ell} \cdot LB_{k\ell} = \sum_{(k,\ell)\in\mathcal{E}} \sum_{t\in\bar{S}_w\cap\{k,\dots,\ell\}} a_{k\ell} \cdot LB_{k\ell}$, and $\sum_{(k,\ell)\in\bar{S}_v} a_{k\ell} + \sum_{(k,\ell)\in S_v} a_{k\ell} = \sum_{(k,\ell)\in\mathcal{E}} a_{k\ell} = 1$.

If we substitute set S_w with S, the cut derived in (45) is very similar to (42), except that the coefficient of $a_{k\ell}$ depends on whether $(k,\ell) \in S_v$ or not. It is $\left(1 - UB_{k\ell} \cdot \left|\bar{S}_w \cap \{k,\dots,\ell\}\right|\right)$ if $(k,\ell) \in S_v$, and $\left(LB_{k\ell} \cdot \left|S_w \cap \{k,\dots,\ell\}\right|\right)$ if $(k,\ell) \in \bar{S}_v$. Clearly, we get a stronger cut if $(k,\ell) \in S_v$ whenever $\left(1 - UB_{k\ell} \cdot \left|\bar{S}_w \cap \{k,\dots,\ell\}\right|\right) > \left(LB_{k\ell} \cdot \left|S_w \cap \{k,\dots,\ell\}\right|\right)$, and $(k,\ell) \in \bar{S}_v$ otherwise. With this choice of S_v (with respect to S_w) we get precisely (42). On the other hand, if $\left(1 - UB_{k\ell} \cdot \left|\bar{S}_w \cap \{k,\dots,\ell\}\right|\right)$ is bigger (smaller) than $\left(LB_{k\ell} \cdot \left|S_w \cap \{k,\dots,\ell\}\right|\right)$, but $(k,\ell) \in \bar{S}_v$

$((k,\ell) \in S_v)$, then the pair (S_v, S_w) induces a dominated cut, which is superfluous in the description of P. It is therefore sufficient to include the inequalities (42) in the linear description of P, which completes the proof. $\square$

3.4 The dimension of P

In this section, in order to further explore the structure of P and its facets, we determine the dimension of P. To this end, we need some more definitions. Let $\mathcal{E}_B$ the subset of those $(k,\ell) \in \mathcal{E}$ such that $(\ell-k+1)LB_{k\ell} = 1$ or $(\ell-k+1)UB_{k\ell} = 1$. Notice that in either case, when $a_{k\ell} = 1$, then the y variables are uniquely determined by $y_t = 1/(\ell - k + 1)$ for $t \in \{k, \dots, \ell\}$, and 0 otherwise. Now, let $T_B := \{t \in \{1, \dots, H\} \ : \ \nexists (k,\ell) \in \mathcal{E} \setminus \mathcal{E}_B$ such that $t \in \{k, \dots, \ell\}\}$, which includes all periods with no flexibility on the intensities. Clearly, the following equations are satisfied by all $(y,a) \in P$:

$$y_t - \sum_{(k,\ell)\in\mathcal{E}, k\le t\le\ell} a_{k\ell}/(\ell - k + 1) = 0, \quad t \in T_B. \tag{46}$$

The reason is that for $t \in T_B$, y_t takes the value $1/(\ell - k + 1)$ if $a_{k\ell} = 1$ for any $(k,\ell) \in \mathcal{E}_B$ and 0 otherwise.

Lemma 3. *If $\mathcal{E} = \mathcal{E}_B$ then $T_B = \{1, \dots, H\}$, otherwise $T_B = \emptyset$.*

Proof. If $\mathcal{E} = \mathcal{E}_B$ then the definition of set T_B implies $T_B = \{1, \dots, H\}$. Now suppose that $\mathcal{E}_B$ is a proper subset of $\mathcal{E}$. We claim that for each $t \in \{1, \dots, H\}$, there exists $(k,\ell) \in \mathcal{E} \setminus \mathcal{E}_B$ such that $t \in \{k, \dots, \ell\}$. Since $\mathcal{E} \ne \mathcal{E}_B$, there exists some $(k',\ell') \in \mathcal{E}$ such that $(\ell' - k' + 1)LB_{k\ell} < 1 < (\ell' - k' + 1)UB_{k\ell}$. Since $\mathcal{E}$ contains all the feasible execution intervals of order j, there must exist $(k,\ell) \in \mathcal{E}$ such that $\ell - k = \ell' - k'$, and $k \le t \le \ell$. $\square$

Theorem 2.

$$\dim(P) = \begin{cases} |\mathcal{E}| - 1, & \text{if } \mathcal{E}_B = \mathcal{E} \\ |\mathcal{E}| + H - 2, & \text{if } \mathcal{E}_B \subset \mathcal{E}, \mathcal{E}_B \ne \mathcal{E} \end{cases}$$

Proof. First suppose $\mathcal{E}_B = \mathcal{E}$. Then $T_B = \{1, \dots, H\}$ by Lemma 3. The valid equations for P are then the following: (33), (34), and (46). Notice that (34) and (46) are linearly independent, but (33) can be expressed as a linear combination of the former two classes of equations. To see this, it suffices to take the sum of the equations (46), leading to $\sum_{t=1}^{H} y_t - \sum_{(k,\ell)\in\mathcal{E}} a_{k\ell} = 0$. The second term equals 1 for any $(y, a) \in P$, since (34) is valid for P. This shows that $\dim(P) \leq |\mathcal{E}| - |T_B| + H - 1 = |\mathcal{E}| - 1$. To finish the proof of this case, we provide $|\mathcal{E}|$ affinely independent points in P. For each $(k, \ell) \in \mathcal{E}$, we define a point with $a_{k\ell} = 1$ and $y_t = 1/(\ell - k + 1)$ for $t \in \{k, \ldots, \ell\}$, and 0 otherwise. These points are affinely independent, since for each $(k, \ell) \in \mathcal{E}$, there is a unique point with $a_{k\ell} = 1$.

Now suppose that $\mathcal{E}_B$ is a proper subset of $\mathcal{E}$. Then $T_B = \emptyset$ by Lemma 3. We argue that any equation $\alpha y + \beta a = \delta$ that is satisfied by all points $(y, a) \in P$ must be a linear combination of (33), and (34).

First we claim that $\alpha_{t_1} = \alpha_{t_2}$ for all $t_1, t_2 \in \{1, \ldots, H\}$. To prove this, observe that in $\mathcal{E}$ there must be execution intervals of length at least 2, otherwise $\mathcal{E}_B = \mathcal{E}$ and we are in the previous case. Since $\mathcal{E}$ contains all the feasible execution intervals, there is a chain of intervals $(k_1, \ell_1), \ldots, (k_s, \ell_s) \in \mathcal{E} \setminus \mathcal{E}_B$, each of length at least 2, such that $k_1 = 1$, $\ell_s = H$ and $\ell_i \geq k_{i+1}$ for each $i = 1, \ldots, s-1$. For each of these intervals it holds that $(\ell_i - k_i + 1)LB_{k\ell} < 1 < (\ell_i - k_i + 1)UB_{k\ell}$, since $(k_i, \ell_i) \notin \mathcal{E}_B$ by definition. Hence, setting $a_{k_i\ell_i} = 1$ and $y_t = 1/(\ell_i - k_i + 1)$ for $t \in \{k, \ldots, \ell\}$ and 0 otherwise yields a point in P. Moreover, there exists $\varepsilon > 0$ such that for any $t_1 \neq t_2 \in \{k, \ldots, \ell\}$, increasing y_{t_1} by ε and decreasing y_{t_2} by the same amount also yields a point in P. Hence $\alpha_{t_1} = \alpha_{t_2}$. By the choice of the execution intervals $(k_1, \ell_1), \ldots, (k_s, \ell_s)$, our claim follows.

Now we claim that $\beta_{k\ell} = \beta_{k'\ell'}$ for all $(k, \ell), (k', \ell') \in \mathcal{E}$. To see this, notice that for each execution interval $(k, \ell) \in \mathcal{E}$, and for any point $(y, a) \in P$ with $a_{k\ell} = 1$, the expression αy is of the same value, say v, since y satisfies (33), and $\alpha_{t_1} = \alpha_{t_2}$ for any $t_1, t_2 \in \{1, \ldots, H\}$. Hence, $\beta_{k\ell} = \delta - v$ for each $(k, \ell) \in \mathcal{E}$, and the claim is proved.

Finally, notice that the above two claims imply the theorem. $\square$

3.5 Facets of P

In this section, we aim to determine when inequality (42) defines a facet of P. To this end, necessary and sufficient conditions are established. Throughout this section, we assume that $T_B = \emptyset$, and thus by Theorem 2, $\dim(P) = |\mathcal{E}| + H - 2$. Inequality (42) associated to set S induces a facet of the polyhedron P if the dimension of the face

$$F = \Big\{(y,a) \in P : \sum_{\tau \in S} y_\tau = \sum_{(k,\ell)\in\mathcal{E}} a_{k\ell} \cdot \max\big\{LB_{k\ell} \cdot \big|S \cap \{k,\ldots,\ell\}\big|,\ 1 - UB_{k\ell} \cdot \big|\bar{S} \cap \{k,\ldots,\ell\}\big|\big\}\Big\}$$

equals $\dim(P) - 1$.

We partition $\mathcal{E}$ into three disjoint subsets. Let $\mathcal{E}_1^{all}$ consist of *all* the execution intervals $(k,\ell) \in \mathcal{E}$ such that either $S \cap \{k,\ldots,\ell\} = \emptyset$, and $(\ell-k+1)\cdot LB_{k\ell} < 1 < (\ell-k+1)\cdot UB_{k\ell}$, or $S \cap \{k,\ldots,\ell\} \neq \emptyset$ and $LB_{k\ell} \cdot \big|S \cap \{k,\ldots,\ell\}\big| > 1 - UB_{k\ell} \cdot \big|\bar{S} \cap \{k,\ldots,\ell\}\big|$. Let $\mathcal{E}_2^{all}$ consist of *all* the execution intervals $(k,\ell) \in \mathcal{E}$ such that $S \cap \{k,\ldots,\ell\} \neq \emptyset$ and $LB_{k\ell} \cdot \big|S \cap \{k,\ldots,\ell\}\big| < 1 - UB_{k\ell} \cdot \big|\bar{S} \cap \{k,\ldots,\ell\}\big|$. The remaining execution intervals, if any, constitute the third subset.

Observe that if $(k,\ell) \in \mathcal{E}_1^{all}$, then for any vector $(y,a) \in F$ such that $a_{k\ell} = 1$, $y_\tau = LB_{k\ell}$ for $\tau \in S \cap \{k,\ldots,\ell\}$ and $y_\tau = 0$ for $S \setminus \{k,\ldots,\ell\}$. Analogously, if $(k,\ell) \in \mathcal{E}_2^{all}$, then for any vector $(y,a) \in F$ such that $a_{k\ell} = 1$, $y_\tau = UB_{k\ell}$ for $\tau \in \bar{S} \cap \{k,\ldots,\ell\}$ and $y_\tau = 0$ for $\bar{S} \setminus \{k,\ldots,\ell\}$. Finally, for each $(k,\ell) \in \mathcal{E} \setminus (\mathcal{E}_1^{all} \cup \mathcal{E}_2^{all})$, there exists a unique vector $(y,a) \in F$ such that $a_{k\ell} = 1$. That is, if $LB_{k\ell} \cdot (\ell - k + 1) = 1$ or $UB_{k\ell} \cdot (\ell - k + 1) = 1$, then $y_\tau = 1/(\ell - k + 1)$ for all $\tau \in \{k,\ldots,\ell\}$ and 0 otherwise; and if $LB_{k\ell} \cdot |S \cap \{k,\ldots,\ell\}| = 1 - UB_{k\ell} \cdot |\bar{S} \cap \{k,\ldots,\ell\}|$, then $y_\tau = LB_{k\ell}$ for all $\tau \in S \cap \{k,\ldots,\ell\}$, $y_\tau = UB_{k\ell}$ for all $\tau \in \bar{S} \cap \{k,\ldots,\ell\}$, and 0 otherwise.

Theorem 3. *The inequality* (42) *associated to set S defines a facet of P if and only if the following conditions hold:*

i) If $|\bar{S}| \geq 2$, then $\mathcal{E}_1^{all} \neq \emptyset$, and for any $t \in \{1,\ldots,H\}$, either $\{1,\ldots,t\} \cap \bar{S} = \emptyset$, or $\{t+1,\ldots,H\} \cap \bar{S} = \emptyset$, or there exists $(k,\ell) \in \mathcal{E}_1^{all}$ such that $\{1,\ldots,t\} \cap \bar{S} \cap \{k,\ldots,\ell\} \neq \emptyset$ and $\{t+1,\ldots,H\} \cap \bar{S} \cap \{k,\ldots,\ell\} \neq \emptyset$.

ii) If $|S| \geq 2$, then $\mathcal{E}_2^{all} \neq \emptyset$, and for any $t \in \{1,\ldots,H\}$, either $\{1,\ldots,t\} \cap S = \emptyset$, or $\{t+1,\ldots,H\} \cap S = \emptyset$, or there exists $(k,\ell) \in \mathcal{E}_2^{all}$ such that $\{1,\ldots,t\} \cap S \cap \{k,\ldots,\ell\} \neq \emptyset$ and $\{t+1,\ldots,H\} \cap S \cap \{k,\ldots,\ell\} \neq \emptyset$.

Proof. Necessity: Assume that F is a facet, but at least one of the conditions i) and ii) does not hold. Without loss of generality, assume that $|\bar{S}| \geq 2$, and there exists $t \in \{1,\ldots,H\}$ such that $\{1,\ldots,t\} \cap \bar{S} \neq \emptyset$, $\{t+1,\ldots,H\} \cap \bar{S} \neq \emptyset$, and for each $(k,\ell) \in \mathcal{E}_1^{all}$, if any, the set $\bar{S} \cap \{k,\ldots,\ell\}$ is either a subset of $\{1,\ldots,t\}$ or a subset of $\{t+1,\ldots,H\}$. Now we define two new equations satisfied by all points in F, but linearly independent from the other equations valid for F. The two equations take the following form:

$$\sum_{\tau \in \bar{S} \cap \{1,\dots,t\}} y_\tau = \sum_{(k,\ell)\in\mathcal{E}} \beta_{k\ell} a_{k\ell} \tag{47}$$

$$\sum_{\tau \in \bar{S} \cap \{t+1,\dots,H\}} y_\tau = \sum_{(k,\ell)\in\mathcal{E}} \gamma_{k\ell} a_{k\ell} \tag{48}$$

To determine the $\beta_{k\ell}$, notice that if $(k,\ell) \in \mathcal{E} \setminus \mathcal{E}_1^{all}$, the coordinates y_τ $(\tau \in \bar{S})$ of any vector $(y,a) \in F$ such that $a_{k\ell} = 1$ are fixed, so $\beta_{k\ell}$ is just the sum of these fixed values. Now suppose that $(k,\ell) \in \mathcal{E}_1^{all}$. If $\bar{S} \cap \{k,\dots,\ell\}$ is a subset of $\{1,\dots,t\}$, then $y_\tau = 0$ $(\tau \in \bar{S} \cap \{t+1,\dots,H\})$ for any $(y,a) \in F$ such that $a_{k\ell} = 1$, and thus $\sum_{\tau\in\bar{S}\cap\{1,\dots,t\}} y_\tau = 1 - |S \cap \{k,\dots,\ell\}| \cdot LB_{k\ell}$. Therefore, $\beta_{k\ell} = 1 - |S \cap \{k,\dots,\ell\}| \cdot LB_{k\ell}$ will do. If $\bar{S} \cap \{k,\dots,\ell\}$ is a subset of $\{t+1,\dots,H\}$, then $y_\tau = 0$ $(\tau \in \bar{S} \cap \{1,\dots,t\})$, and $\beta_{k\ell} = 0$. The calculation of the coefficients $\gamma_{k\ell}$ are analogous. Since $|\bar{S}| \geq 2$ by assumption, the equations (47) and (48) are linearly independent from the equations (33), (34) and (42) for S. But then, F must have a lower dimension than $\dim(P) - 1$, so it is not a facet, which leads to a contradiction.

Sufficiency: Under the conditions of the theorem, we construct $\dim(P)$ affinely independent points in F. Firstly, observe that the conditions ensure that if $\mathcal{E}_1^{all} \neq \emptyset$, then we can choose a subset $\mathcal{E}_1 \subseteq \mathcal{E}_1^{all}$ and an ordering $(k_1,\ell_1),\dots,(k_{|\mathcal{E}_1|},\ell_{|\mathcal{E}_1|})$ of the members of $\mathcal{E}_1$ such that for $t_{1,h}^{min} = \min\{t : t \in \bar{S} \cap \{k_h,\dots,\ell_h\}\}$ and $t_{1,h}^{max} = \max\{t : t \in \bar{S} \cap \{k_h,\dots,\ell_h\}\}$ we have $t_{1,h}^{min} \leq t_{1,h-1}^{max} < t_{1,h}^{max}$ for $h \in \{2,\dots,|\mathcal{E}_1|\}$, where $t_{1,0}^{max} = t_{1,1}^{min}$. Similarly, if $\mathcal{E}_2^{all} \neq \emptyset$, then we can choose a subset $\mathcal{E}_2 \subseteq \mathcal{E}_2^{all}$ and an ordering $(k'_1,\ell'_1),\dots,(k'_{|\mathcal{E}_2|},\ell'_{|\mathcal{E}_2|})$ such that for $t_{2,h}^{min} = \min\{t : t \in S \cap \{k'_h,\dots,\ell'_h\}\}$ and $t_{2,h}^{max} = \max\{t : t \in S \cap \{k'_h,\dots,\ell'_h\}\}$, we have $t_{2,h}^{min} \leq t_{2,h-1}^{max} < t_{2,h}^{max}$ for $h \in \{2,\dots,|\mathcal{E}_2|\}$, where $t_{2,0}^{max} = t_{2,1}^{min}$.

For each $(k,\ell) \in \mathcal{E} \setminus (\mathcal{E}_1 \cup \mathcal{E}_2)$, we define a point in F as follows. If $S \cap \{k,\dots,\ell\} = \emptyset$, then let $a_{k\ell} = 1$, and $y_t = 1/(\ell - k + 1)$, for $t \in \{k,\dots,\ell\}$, while all the remaining entries are set to 0. If $S \cap \{k,\dots,\ell\} \neq \emptyset$, then let $a_{k\ell} = 1$, $y_\tau = \max\left\{LB_{k\ell} \cdot \left|S \cap \{k,\dots,\ell\}\right|,\ 1 - UB_{k\ell} \cdot \left|\bar{S} \cap \{k,\dots,\ell\}\right|\right\} \Big/ \left|S \cap \{k,\dots,\ell\}\right|$ for $\tau \in S \cap \{k,\dots,\ell\}$, and $y_t = \min\left\{1 - LB_{k\ell} \cdot \left|S \cap \{k,\dots,\ell\}\right|,\ UB_{k\ell} \cdot \left|\bar{S} \cap \{k,\dots,\ell\}\right|\right\} \Big/ \left|\bar{S} \cap \{k,\dots,\ell\}\right|$ for each $t \in \bar{S} \cap \{k,\dots,\ell\}$, while all the remaining entries are set to 0. All these points are in F, since $\sum_{\tau\in S} y_\tau = \sum_{(k,\ell)\in\mathcal{E}} a_{k\ell} \cdot \max\left\{LB_{k\ell} \cdot \left|S \cap \{k,\dots,\ell\}\right|,\ 1 -\right.$

$UB_{k\ell} \cdot \left|\bar{S} \cap \{k, \ldots, \ell\}\right|\}$ holds. Since $\mathcal{E}_1 \cap \mathcal{E}_2 = \emptyset$, we obtain $|\mathcal{E}| - |\mathcal{E}_1| - |\mathcal{E}_2|$ linearly independent points in F.

Now we define additional linearly independent points in F based on the subset $\mathcal{E}_1 \subseteq \mathcal{E}$. Denote $\epsilon > 0$ a sufficiently small positive value. For any $(k_h, \ell_h) \in \mathcal{E}_1$, we construct $|\bar{S} \cap \{t^{max}_{1,h-1}, \ldots, t^{max}_{1,h}\}|$ new points in F. If $S \cap \{k_h, \ldots, \ell_h\} = \emptyset$, then by definition $(\ell_h - k_h + 1) \cdot LB_{k_h,\ell_h} < 1 < (\ell_h - k_h + 1) \cdot UB_{k_h,\ell_h}$, and for each $\tau \in \bar{S} \cap \{t^{max}_{1,h-1}, \ldots, t^{max}_{1,h}\}$, we define a point in F with $a_{k\ell} = 1$, $y_\tau = 1/(\ell_h - k_h + 1) + (\ell_h - k_h) \cdot \epsilon$ and $y_t = 1/(\ell_h - k_h + 1) - \epsilon$ for $t \in \{k_h, \ldots, \ell_h\} \setminus \{\tau\}$, and all other components are set to 0. If $S \cap \{k_h, \ldots, \ell_h\} \neq \emptyset$, then by definition $LB_{k_h,\ell_h} \cdot \left|S \cap \{k_h, \ldots, \ell_h\}\right| > 1 - UB_{k_h,\ell_h} \cdot \left|\bar{S} \cap \{k_h, \ldots, \ell_h\}\right|$, and for each $\tau \in \bar{S} \cap \{t^{max}_{1,h-1}, \ldots, t^{max}_{1,h}\}$, we define a point with $a_{k\ell} = 1$, $y_t = LB_{k_h,\ell_h}$ for $t \in S \cap \{k_h, \ldots, \ell_h\}$, $y_\tau = \left(1 - LB_{k_h,\ell_h} \cdot \left|S \cap \{k_h, \ldots, \ell_h\}\right|\right) \Big/ \left|\bar{S} \cap \{k_h, \ldots, \ell_h\}\right| + \left(\left|\bar{S} \cap \{k_h, \ldots, \ell_h\}\right| - 1\right) \cdot \epsilon$ and $y_t = \left(1 - LB_{k_h,\ell_h} \cdot \left|S \cap \{k_h, \ldots, \ell_h\}\right|\right) \Big/ \left|\bar{S} \cap \{k_h, \ldots, \ell_h\}\right| - \epsilon$, for every $t \in \bar{S} \cap \{k_h, \ldots, \ell_h\} \setminus \{\tau\}$. It is easy to verify that all these points are in F. Let M be a matrix containing all these points without the coordinates fixed to $LB_{k\ell}$ ($t \in S \cap \{k, \ldots, \ell\}$):

$$M = \left(\begin{array}{c} M_1 \\ \hline M_2 \\ \hline \vdots \end{array}\right) = \left(\begin{array}{cccccccc}
a+b & a & \ldots & a & & & 1 & \\
a & a+b & \ldots & a & & & 1 & \\
\vdots & \vdots & \ddots & \vdots & & & \vdots & \\
a & a & \ldots & a+b & & & 1 & \\
\hline
& c & \ldots & c+d & \ldots & c & & 1 \\
& \vdots & \ddots & \vdots & \ddots & \vdots & & \vdots \\
& c & \ldots & c & \ldots & c+d & & 1 \\
\hline
& & & \vdots & & & &
\end{array}\right)$$

where a, b, c and d are all positive values. One can verify that M has full row rank. Hence, we have constructed $|\bar{S} \cap \{t^{max}_{1,h-1}, \ldots, t^{max}_{1,h}\}|$ linearly independent points for each $(k_h, \ell_h) \in \mathcal{E}_1$. Thus in total, we have defined $\sum_{h=1}^{|\mathcal{E}_1|} |\bar{S} \cap \{t^{max}_{1,h-1}, \ldots, t^{max}_{1,h}\}| = |\bar{S}| + |\mathcal{E}_1| - 1$ linearly independent points in F based on $\mathcal{E}_1$.

Analogously to the previous case, additional linearly independent points can be constructed in F based on $\mathcal{E}_2 \subseteq \mathcal{E}$. For each $(k_i, \ell_i) \in \mathcal{E}_2^+$ we construct $|S \cap \{t^{max}_{2,i-1}, \ldots, t^{max}_{2,i}\}|$ points. For each $(k_i, \ell_i) \in \mathcal{E}_2$, by definition $LB_{k_i,\ell_i} \cdot \left|S \cap \{k_i, \ldots, \ell_i\}\right| < 1 - UB_{k_i,\ell_i} \cdot \left|\bar{S} \cap \{k_i, \ldots, \ell_i\}\right|$, and for each

$\tau \in S \cap \{t^{max}_{2,i-1}, \dots, t^{max}_{2,i}\}$, we define a point with $a_{k\ell} = 1$, $y_\tau = (1 - UB_{k_i,\ell_i} \cdot |\bar{S} \cap \{k_i, \dots, \ell_i\}|) \big/ |S \cap \{k_i, \dots, \ell_i\}| + (|S \cap \{k_i, \dots, \ell_i\}| - 1) \cdot \epsilon$, $y_t = (1 - UB_{k_i,\ell_i} \cdot |\bar{S} \cap \{k_i, \dots, \ell_i\}|) \big/ |S \cap \{k_i, \dots, \ell_i\}| - \epsilon$, $t \in S \cap \{k_i, \dots, \ell_i\} \setminus \{\tau\}$, and $y_t = UB_{k\ell}$ for $t \in \bar{S} \cap \{k_i, \dots, \ell_i\}$. Similarly to the previous case, all these points are linearly independent. Therefore, set $\mathcal{E}_2$ yields $\sum_{h=1}^{|\mathcal{E}_2|} |S \cap \{t^{max}_{2,h-1}, \dots, t^{max}_{2,h}\}| = |S| + |\mathcal{E}_2| - 1$ new, linearly independent points in F.

To finish the proof, we count the number of linearly independent points that have been constructed. If $|S| \geq 2$ and $|\bar{S}| \geq 2$, we got $|\mathcal{E}_1| + |\mathcal{E}_2| + H - 2$ new linearly independent points in F in addition to the $|\mathcal{E}| - |\mathcal{E}_1| - |\mathcal{E}_2|$ points constructed for $(k, \ell) \in \mathcal{E} \setminus (\mathcal{E}_1 \cup \mathcal{E}_2)$, giving a total of $|\mathcal{E}| + H - 2$ linearly independent points. If $|S| = 1$, then $\mathcal{E}_2 = \emptyset$, $|\bar{S}| + |\mathcal{E}_1| - 1 = |\mathcal{E}_1| + H - 2$, and again, we got $|\mathcal{E}| + H - 2$ linearly independent points. Finally, the case when $|\bar{S}| = 1$ is analogous. □

Example 2. Consider the polytope P of feasible intensity assignments for $LB = 0.25$ and $UB = 0.5$ for $H = 4$ time periods. Then $\mathcal{E} = \{(1,2), (1,3), (1,4), (2,3), (2,4), (3,4)\}$, which implies $P \subseteq [0,1]^{10}$, and $\dim(P) = 8$. For $S = \{1\}$, the corresponding inequality is

$$y_1 \geq 0.5a_{12} + 0.25a_{13} + 0.25a_{14}. \tag{49}$$

In fact, we can apply Theorem 3 to show that this inequality induces a facet of P. With $\mathcal{E}_1 = \{(1,3), (2,4)\}$ and $\mathcal{E}_2 = \emptyset$ we obtain eight linearly independent points in P satisfying (49):

$\mathcal{E}$	S	$\bar{S}$			$\{1, \dots, \|\mathcal{E}\|\}$					
	1	2	3	4	1	2	3	4	5	6
(1,2)	0.5	0.5	0	0	1	0	0	0	0	0
(1,3)	0.25	$0.375 + \varepsilon$	$0.375 - \varepsilon$	0	0	1	0	0	0	0
(1,3)	0.25	$0.375 - \varepsilon$	$0.375 + \varepsilon$	0	0	1	0	0	0	0
(1,4)	0.25	0.25	0.25	0.25	0	0	1	0	0	0
(2,3)	0	0.5	0.5	0	0	0	0	1	0	0
(2,4)	0	0.334	$0.333 + \varepsilon$	$0.333 - \varepsilon$	0	0	0	0	1	0
(2,4)	0	0.334	$0.333 - \varepsilon$	$0.333 + \varepsilon$	0	0	0	0	1	0
(3,4)	0	0	0.5	0.5	0	0	0	0	0	1

For $S = \{2\}$ we obtain the facet-defining inequality

$$y_2 \geq 0.5a_{12} + 0.25a_{13} + 0.25a_{14} + 0.5a_{23}, \tag{50}$$

certified by $\mathcal{E}_1 = \{(2,4)\}$, $\mathcal{E}_2 = \emptyset$ (details omitted).

Finally, for $S = \{1, 2\}$, we derive the inequality

$$y_1 + y_2 \geq a_{12} + 0.5a_{13} + 0.5a_{14} + 0.5a_{23}. \tag{51}$$

The only choice for $\mathcal{E}_1$ is $\{(2,4)\}$, while $\mathcal{E}_2$ must be empty. However, the construction of Theorem 3 yields only seven linearly independent points, and in fact, (51) is the sum of (49) and (50), so it cannot induce a facet of P.

3.6 Separation of the valid inequalities

As has been shown in Proposition (8) and Theorem (1), it suffices to considered only inequalities (42). There is one lower bounding constraint (42) for any $S \subseteq \{1, \ldots, H\}$, the total number of which is exponential. Therefore, it is not practical to generate all inequalities (42) and include them in the model from the start, and a separation procedure needs to be designed.

Theorem 4. *Inequalities* (42) *can be separated in polynomial time.*

Proof. Given the current fractional solution $(\tilde{y}, \tilde{a}) \in \mathbb{R}_+^H \times [0,1]^{|\mathcal{E}|}$, the goal is to check whether there is any violated inequality in (42). By Theorem 1, if $(\tilde{y}, \tilde{a}) \notin P$, one cannot find a feasible flow of value 1 in network $G(\tilde{y}, \tilde{a})$.

Therefore, to solve the separation problem, we determine the maximum s-q flow value in the transformed network $G'(\tilde{y}, \tilde{a})$ of $G(\tilde{y}, \tilde{a})$, and if it is smaller than $1 + \sum_{(k,\ell)\in\mathcal{E}} \tilde{a}_{k\ell}(\ell - k + 1)LB_{k\ell}$, identify the subset $S \subseteq \{1, \ldots, H\}$ of the minimum capacity s-q cut and in the node partition along with s. One can construct inequality (42) based on set S, which cuts off current solution $(\tilde{y}, \tilde{a})$. The separation problem thus requires the solution of a maximum flow problem in a network of size polynomial in H and $|\mathcal{E}|$. Since the maximum s-q flow and the corresponding minimum cut can be determined in a runtime that is polynomial in the size of the network [1], the entire procedure has polynomial time complexity in H and $|\mathcal{E}|$. $\square$

3.7 On the facets of the execution-interval formulation

In the foregoing, we have examined the polyhedron P and its facets for one individual order; we are now interested to know how strong the facets of P are for the polyhedron of the execution-interval formulation. In this section, we prove that the inequalities that induce facets of the polyhedra of feasible intensity assignments of individual orders also induce facets of the polyhedron associated with the execution-interval formulation. Let

$$Q := \operatorname{conv}\{(y, a, z) \mid (y, a, z) \text{ is a feasible solution of } (6)–(9), (29)–(31)\}.$$

Furthermore, let P_j be the polyhedron P for order j. We have the following:

Theorem 5. *Let $F_j := \{(y^j, a^j) \in P_j \mid \alpha y^j + \gamma a^j = \beta\}$ be a facet of P_j induced by the inequality $\alpha y^j + \gamma a^j \leq \beta$ valid for P_j. Then $F := \{(y, a, z) \in Q \mid \alpha y^j + \gamma a^j = \beta\}$ is a facet of Q.*

Proof. (sketch) First notice that the dimension of Q is at most $H + \sum_{j=1}^{n} \dim(P_j)$. Hence, the dimension of any of its facets is no more than $H - 1 + \sum_{j=1}^{n} \dim(P_j)$. Without loss of generality, assume that $j = 1$, and F_1 is a facet of P_1. We construct $H + \sum_{j=1}^{n} \dim(P_j)$ affinely independent points in F, which proves the theorem, since $F \neq Q$ by the choice of $\alpha y^j + \gamma a^j \leq \beta$.

Let $z^0 = (M, \ldots, M) \in \mathbb{R}^H$, where $M = \sum_{j=1}^{n} p_j$, and e_t the unit vector in $\mathbb{R}^H$ with a 1 in position t and 0 in all other positions. Let $v_0^1, \ldots, v_{\dim(F_1)}^1$ be $\dim(F_1) + 1 = \dim(P_1)$ affinely independent points in F_1, and $v_0^j, \ldots, v_{\dim(P_j)}^j$ be $\dim(P_j) + 1$ affinely independent points in P_j for $j = 2, \ldots, n$ (such points must exist by the basic properties of dimension of polyhedra). Consider the following points in Q:

- $(v_0^1, v_0^2, \ldots, v_0^n, z^0)$,
- $(v_0^1, v_0^2, \ldots, v_0^n, z^0 + e_t)$, $t = 1, \ldots, H$,
- $(v_i^1, v_0^2, \ldots, v_0^n, z^0)$, $i = 1, \ldots, \dim(F_1)$,
- $(v_0^1, v_0^2, \ldots, v_i^j, \ldots, v_0^n, z^0)$, for $j = 2, \ldots, n$, and $i = 1, \ldots, \dim(P_j)$.

This is a set of $H + 1 + \dim(F_1) + \sum_{j=2}^{n} \dim(P_j) = H + \sum_{j=1}^{n} \dim(P_j)$ affinely independent points in Q, all of which satisfy $\alpha a^1 + \gamma y^1 \leq \beta$ at equality, and the theorem follows. $\square$

From this theorem, it follows that those inequalities in (42) that induce facets of P_j, induce facets of Q as well.

4 Computational experiments

4.1 Experimental setup

The computational experiments are performed on a PC equipped with Intel Core i7-4790 CPU at 3.6 GHz with 16 GB of RAM on a Windows 10 64-bit OS. All formulations are solved with CPLEX 12.6.3 implemented with C++ using Concert Technology. The CPLEX solver is set to use only one thread with all other parameters set to their default values.

To the best of our knowledge, there are no benchmark datasets for the RLP, so we have generated our own instance sets. The instances are generated for $|J| = n = 20,\ 40,\ 60,\ 80$, and 100, on a planning horizon with $H = 50$ periods. The work content p_j for each order j is drawn as an integer from a uniform distribution $U(100, 200)$. Instead of parameterizing the bounds on the intensity in each period for the orders, we set the bounds on the execution-interval length for each order. Each $\underline{l_j}$ is generated from the uniform distribution $U(2, 10)$. As the difference between $\overline{l_j}$ and $\underline{l_j}$ determines most of the flexibility for the timing of the orders, it has direct impact on the size of the solution space. Two different levels are given for generating $\overline{l_j} - \underline{l_j}$, with uniform distributions $U(2, 6)$ and $U(8, 12)$, and the bounds on the execution intensity UB_j and LB_j take the reciprocals of $\underline{l_j}$ and $\overline{l_j}$ correspondingly. The release period r_j of each order j is drawn uniformly from $[1,\ H - \underline{l_j} - 1]$. The slack periods of an order are defined as $slack_j = d_j - r_j - \underline{l_j} + 1$, which is a factor shown to have an impact on the difficulty of the instances in Hans [18] and Kis [19]. The parameter $slack_j$ is generated uniformly from three different intervals $[1,\ 5]$, $[6,\ 10]$ and $[11,\ 15]$, and the due date of each order j is then obtained as $d_j = \min\{H, r_j + \underline{l_j} + slack_j - 1\}$. The unit cost for non-regular capacity usage σ takes the value 0.5, and the tardiness cost per period w_j for each order j is either set as a fixed value 5 or drawn from uniform distribution $U(1, 10)$. The available capacity C_t in each period t is generated uniformly from interval $[\sum_{j \in J} p_j/H \cdot 0.9,\ \sum_{j \in J} p_j/H \cdot 1.1]$. For each combination of the above factors, 20 instances are generated, which gives a total number of instances equal to $5 \times 2 \times 3 \times 2 \times 20 = 1200$. The time-indexed mask, pulse and step formulations are tested and compared, together with the execution-interval reformulation with and without cuts (42) separated. The runtime limit for each formulation on each instance is set at 1800 seconds (30 minutes).

Table 1: Results of five methods on instances with different values of n

n	*Mask*			*Pulse*			*Step*		
	opt	*time*	*gap*(%)	*opt*	*time*	*gap*(%)	*opt*	*time*	*gap*(%)
20	227	161.74	26.09	236	79.05	12.95	240	6.10	0.00
40	207	302.52	28.64	225	170.35	12.32	236	69.31	3.84
60	189	414.44	31.21	212	264.57	13.83	225	169.41	6.24
80	191	397.14	23.39	220	198.11	12.33	225	157.04	8.18
100	194	370.85	26.89	212	240.22	12.82	217	209.26	8.30
Overall	1008	329.34	27.39	1105	190.46	12.94	1143	122.22	7.41

n	*EI*			*EI_cut*		
	opt	*time*	*gap*(%)	*opt*	*time*	*gap*(%)
20	237	51.17	8.73	240	5.45	0.00
40	225	139.51	3.39	240	19.33	0.00
60	219	196.46	2.40	240	45.07	0.00
80	226	131.73	1.61	238	43.49	1.31
100	218	191.82	1.63	233	104.87	0.50
Overall	1125	142.14	2.48	1191	43.64	0.68

4.2 Computational results

In Table 1, computational results for five different methods are aggregated and reported for each n value. We use *Mask*, *Pulse* and *Step* to denote the time-indexed mask, pulse and step formulation respectively, and the valid inequalities in Section 2.5 and their equivalent forms are incorporated in all of the three time-indexed formulations. *EI* stands for the plain execution-interval formulation, while *EI_cut* denotes the branch-and-cut method based on the execution-interval formulation with cuts (42) generated. The maximum flow problem in the separation routine is solved by the preflow algorithm by Goldberg and Tarjan [17], using the LEMON library [13]. Cuts (42) are separated in the search tree of *EI_cut* in the nodes on the first five levels and every fifth level, and we also incorporate the generalized upper bound (GUB) branching scheme [28] to further enhance the algorithm. Columns labeled by *opt* show the number of instances solved to optimality within the time limit, out of 240 instances per n value, and columns labeled *time* contain the average CPU time over the 240 instances in seconds. Entries *gap* show, for the unsolved instances, the average gap between the upper and lower bounds reported by the solver when the runtime limit is reached.

Table 1 shows that *EI_cut* is the clear winner out of the five methods. It solves all instances with up to 60 orders, and is only unable to solve two and seven instances respectively for $n = 80$ and $n = 100$. The average runtime of *EI_cut* is mostly a fraction of the runner-up method, and it

Table 2: Results of *EI_cut* on instances with $n = 60$ across different settings

$\bar{l} - \underline{l}$	*slack*	$w = 5$			$w \sim U(1, 10)$		
		opt	*time*	*gap*(%)	*opt*	*time*	*gap*(%)
[2, 6]	[1, 5]	20	136.95	0.00	20	77.84	0.00
	[6, 10]	20	7.03	0.00	20	3.24	0.00
	[11, 15]	20	2.83	0.00	20	2.97	0.00
[8, 12]	[1, 5]	20	228.41	0.00	20	72.30	0.00
	[6, 10]	20	2.45	0.00	20	2.91	0.00
	[11, 15]	20	1.82	0.00	20	2.17	0.00

provides the smallest gap for those unsolved instances. Among the three time-indexed formulations, *Step* outperforms the other two, despite the fact that the three time-indexed formulations are equally strong in terms of LP relaxation. One advantage of *Step* over *Pulse* is that branching on a step variable is equivalent to branching on the sum of a group of pulse variables due to their mutual transformation, which coincides with the idea of GUB branching. A similar pattern of the computational behavior of these time-indexed formulations has been observed by other researchers; see, e.g., Burgelman and Vanhoucke [9], Naber and Kolisch [22]. We should also point out that the best solutions found by the time-indexed formulations are not as far from the optimal solutions as indicated by the values in *gap*, e.g., the average gap of *Mask* for $n = 60$ can be brought down to 2.90 from 31.21, when compared with the optimal solutions found by *EI_cut*. This also proves that one main disadvantage of the time-indexed formulations is the inferior lower bound. The plain execution-interval formulation *EI* does not always outperform the time-indexed formulations, but it does on average provide a better optimality gap thanks to its stronger LP bound.

In order to show how the difficulty of the instances scales with different parameter settings, Table 2 details the results of *EI_cut* on instances with $n = 60$ for different levels of $\bar{l} - \underline{l}$ and *slack*, and different choices for the generation of w. Clearly, instances with low slackness are hard to solve, and this is true for all formulations and for each value of n. The reason is that in case of low slackness, the choice between delaying an order (incurring a tardiness penalty) and invoking non-regular capacity is crucial, and the trade-off becomes difficult. Comparing the left and right half of the table, we see that instances with identical unit tardiness cost w tend to be more difficult than those with heterogeneous w values. This is possibly because with w homogeneous among different orders, there are many similar solutions and thus a larger feasible space needs to be examined.

Table 3: Results of five methods on selected difficult instances

n	*Mask*			*Pulse*			*Step*		
	opt	*time*	*node*	*opt*	*time*	*node*	*opt*	*time*	*node*
20 (14)	1	1703.56	133431.93	10	726.92	145272.36	14	58.26	5889.00
40 (34)	1	1759.15	103747.41	19	1030.18	135218.74	30	466.51	23552.38
60 (51)	0	1800.00	67016.61	23	1169.77	114530.65	36	783.60	27103.71
80 (49)	0	1800.00	55277.88	29	906.17	62623.43	34	742.25	16174.90
100 (46)	0	1800.00	49502.76	18	1182.98	65889.59	23	1057.87	20420.37

n	*EI*			*EI_cut*			
	opt	*time*	*node*	*opt*	*time*	*node*	*cut*
20 (14)	11	584.76	547724.14	14	25.26	1245.93	908.14
40 (34)	19	962.12	475328.12	34	120.17	4281.71	1107.09
60 (51)	30	915.79	352199.51	51	201.42	7780.43	1142.14
80 (49)	35	636.59	231829.41	47	199.46	4743.94	838.84
100 (46)	24	990.21	285042.22	39	529.25	11614.96	1348.65

Table 3 reports the results for selected difficult instances, which at least one method fails to solve. The number of the selected instances for each value of n is indicated between parentheses. Columns labeled by *node* contain the average number of nodes examined in the branch-and-bound tree, and column *cut* of *EI_cut* presents the average number of valid inequalities (42) separated for solving these instances. We clearly see that for these difficult instances, *EI_cut* dominates all other methods, since on average it consumes the least solving time and explores the lowest number of nodes. Without the enhancements, the average number of nodes examined for *EI* is in fact the largest among all five methods. Between the three time-indexed formulations, *Mask* and *Pulse* are outperformed by *Step*, and we should point out that *Mask* and *Pulse* cannot solve a single instance that is not solved by *Step*.

4.3 Results on instances with precedence constraints

In order to examine the flexibility of the different models for incorporating various practical scheduling extensions, we compare the model *Step* and our branch-and-cut *EI_cut*, which are the two best performers, on instances with general precedence constraints. The aim is to show the difference in effectiveness of the models in handling the more generalized problem setting. The instances are generated for $n = 60$, and the precedence graphs are generated by the RanGen software by Demeulemeester et al. [12], with three levels of order strength $OS = 0.2,\ 0.4$ and 0.6 (order strength is a measure for the density of the graph). For each order j, $\underline{l_j}$ is generated from uniform distribution

$U(2,10)$, and $\overline{l_j} - \underline{l_j}$ is drawn from uniform distribution $U(2,6)$. We take the planning horizon as the length of the critical path calculated by $\underline{l_j}$, multiplied by a factor CPL with three choices $CPL = 1.2,\ 1.4$ and 1.6, and rounded down to the nearest integer. The tardiness cost w_j per period of each order j is set as a fixed value 5, the release period r_j is set as the earliest start time ES_j and the due date d_j is set as the first quartile between the earliest finish time EF_j and the latest finish time LF_j, rounded down to the nearest integer. The other parameters are generated in the same way as described before. For each combination of factor OS and CPL, 20 instances are generated, leading to $3 \times 3 \times 20 = 180$ instances in total.

We incorporate the precedence constraints in the *Step* formulation as follows: the set of all immediate precedence relationships is denoted as set E, such that order pair $(i,j) \in E$ iff order i is a direct predecessor of order j, meaning that i must be completed before the start of j. *Step* can be extended with precedence constraints as follows:

$$s^a_{j,t+1} \le f^a_{it} \qquad t = 1, \dots, H-1, \forall (i,j) \in E \tag{52}$$

$$s^a_{jt} = 0 \qquad t = ES_j - 1, \forall j \in J \tag{53}$$

$$s^a_{jt} = 1 \qquad t = LS_j, \forall j \in J \tag{54}$$

$$f^a_{jt} = 0 \qquad t = EF_j - 1, \forall j \in J \tag{55}$$

$$f^a_{jt} = 1 \qquad t = LF_j, \forall j \in J \tag{56}$$

The same set of constraints are also applied in *EI_cut*, using the affine transformation described in Proposition 5.

Table 4 presents the results of *Step* and *EI_cut* on the instances with precedence constraints, with columns labeled by 1% containing the number of instances with optimality gap less than one percent by the runtime limit. *EI_cut* solves half of the instances with $OS = 0.2$, and fails for only two with $OS = 0.6$. It also closes the optimality gap to less than 1% for all but one of the instances, within the runtime limit. On the other hand, *Step* fails nearly completely and is clearly dominated by *EI_cut* in all aspects. Considering how the difficulty scales with the parameters, instances with lower OS and higher CPL values tend to be harder to solve.

Table 4: Results of *Step* and *EI_cut* on instances of $n = 60$ with precedence constraints

OS	*CPL*	*Step*				*EI_cut*			
		opt	*time*	*gap*(%)	1%	*opt*	*time*	*gap*(%)	1%
0.2	1.2	1	1782.87	6.53	4	10	1246.51	0.49	19
	1.4	0	1800.00	12.18	1	13	1048.14	0.45	20
	1.6	0	1800.00	23.75	0	7	1510.02	0.25	20
0.4	1.2	1	1731.58	5.43	3	16	748.15	0.47	20
	1.4	0	1800.00	18.85	0	15	902.91	0.54	20
	1.6	0	1800.00	25.61	0	17	887.57	0.27	20
0.6	1.2	1	1757.18	6.58	3	20	292.06	0.00	20
	1.4	0	1800.00	12.73	0	19	448.09	0.46	20
	1.6	0	1800.00	16.99	0	19	572.63	0.27	20
Overall		3	1785.74	14.43	11	136	850.67	0.40	179

5 Conclusions

In this paper, we model and solve the RLP with different MIP formulations: the time-indexed mask, pulse and step formulation, and a novel execution-interval formulation. The strength of these formulations is compared based on polyhedral analysis, and valid inequalities are proposed to further enhance the formulations. Based on the execution-interval formulation, a complete description of the polytope of feasible intensity assignments for individual orders is established, along with a polynomial-time separation algorithm. The computational results show that the proposed branch-and-cut algorithm provides superior performance. Topics for future research are manifold, such as establishing polyhedral results for the incorporation of precedence constraints among the orders, and the consideration of uncertainty on the work content of orders.

Acknowledgments

This work was supported by the CELSA joint research funding program (project CELSA/17/007). The research of Tamás Kis has been supported by the National Research, Development and Innovation Office of Hungary–NKFIH, grant no. SNN 129178, and ED_18-2-2018-0006.

References

[1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, Upper Saddle River, New Jersey, 1993.

[2] A. Alfieri, T. Tolio, and M. Urgo. A project scheduling approach to production planning with feeding precedence relations. *International Journal of Production Research*, 49(4):995–1020, 2011.

[3] C. Artigues. On the strength of time-indexed formulations for the resource-constrained project scheduling problem. *Operations Research Letters*, 45(2):154–159, 2017.

[4] E. Balas. Disjunctive programming: Properties of the convex hull of feasible points. *Discrete Applied Mathematics*, 89(1-3):3–44, 1998.

[5] E. Balas. *Disjunctive Programming*. Springer, 2018.

[6] J. Beliën, B. Cardoen, and E. Demeulemeester. Improving workforce scheduling of aircraft line maintenance at Sabena Technics. *Interfaces*, 42(4):352–364, 2012.

[7] L. Bianco and M. Caramia. A new formulation for the project scheduling problem under limited resources. *Flexible Services and Manufacturing Journal*, 25(1-2):6–24, 2013.

[8] J. Błażewicz, M. Machowiak, J. Weglarz, M. Y. Kovalyov, and D. Trystram. Scheduling malleable tasks on parallel processors to minimize the makespan. *Annals of Operations Research*, 129(1-4):65–80, 2004.

[9] J. Burgelman and M. Vanhoucke. Maximising the weighted number of activity execution modes in project planning. *European Journal of Operational Research*, 270(3):999–1013, 2018.

[10] C. C. Cavalcante, C. C. de Souza, M. W. Savelsbergh, Y. Wang, and L. A. Wolsey. Scheduling projects with labor constraints. *Discrete Applied Mathematics*, 112(1):27–52, 2001.

[11] R. De Boer. *Resource-constrained multi-project management*. PhD thesis, University of Twente, The Netherlands, 1998.

[12] E. Demeulemeester, M. Vanhoucke, and W. Herroelen. Rangen: A random network generator for activity-on-the-node networks. *Journal of Scheduling*, 6(1):17–38, 2003.

[13] B. Dezső, A. Jüttner, and P. Kovács. LEMON–an open source c++ graph template library. *Electronic Notes in Theoretical Computer Science*, 264(5):23–45, 2011.

[14] L. R. Ford and D. R. Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 8(3):399–404, 1956.

[15] C. Fündeling and N. Trautmann. A priority-rule method for project scheduling with work-content constraints. *European Journal of Operational Research*, 203:568–574, 2010.

[16] N. Gademann and M. Schutten. Linear-programming-based heuristics for project capacity planning. *IIE Transactions*, 37:153–165, 2005.

[17] A. V. Goldberg and R. E. Tarjan. A new approach to the maximum-flow problem. *Journal of the ACM*, 35(4):921–940, 1988.

[18] E. Hans. *Resource loading by branch-and-price techniques.* PhD thesis, University of Twente, The Netherlands, 2001.

[19] T. Kis. A branch-and-cut algorithm for scheduling of projects with variable-intensity activities. *Mathematical Programming*, 103(3):515–539, 2005.

[20] 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.

[21] R. H. Möhring, A. S. Schulz, F. Stork, and M. Uetz. Solving project scheduling problems by minimum cut computations. *Management Science*, 49(3):330–350, 2003.

[22] A. Naber and R. Kolisch. MIP models for resource-constrained project scheduling with flexible resource profiles. *European Journal of Operational Research*, 239:335–348, 2014.

[23] M. Nattaf, M. Horváth, T. Kis, C. Artigues, and P. Lopez. Polyhedral results and valid inequalities for the continuous energy-constrained scheduling problem. *Discrete Applied Mathematics*, 258:188–203, 2019.

[24] A. A. B. Pritsker, L. J. Watters, and P. M. Wolfe. Multiproject scheduling with limited resources: A zero-one programming approach. *Management Science*, 16(1):93–108, 1969.

[25] J. P. Sousa. *Time indexed formulations of non-preemptive single-machine scheduling problems*. PhD thesis, Université Catholique de Louvain, Belgium, 1989.

[26] J. P. Sousa and L. A. Wolsey. A time indexed formulation of non-preemptive single machine scheduling problems. *Mathematical Programming*, 54(1-3):353–367, 1992.

[27] F. Talla Nobibon, R. Leus, K. Nip, and Z. Wang. Resource loading with time windows. *European Journal of Operational Research*, 244(2):404–416, 2015.

[28] L. A. Wolsey. *Integer Programming*. Wiley-Interscience Series in Discrete Mathematics and Optimization, John Wiley & Sons, New York, 1998.

[29] G. Wullink, A. Gademann, E. W. Hans, and A. van Harten. Scenario-based approach for flexible resource loading under uncertainty. *International Journal of Production Research*, 42 (24):5079–5098, 2004.

FACULTY OF ECONOMICS AND BUSINESS
Naamsestraat 69 bus 3500
3000 LEUVEN, BELGIË
tel. + 32 16 32 66 12
fax + 32 16 32 67 91
info@econ.kuleuven.be
www.econ.kuleuven.be