Mathematical Programming 72 (1996) 207–227

Scheduling jobs of equal length: complexity, facets and computational results [1]

Yves Crama [a], Frits C.R. Spieksma [b,*]

[a] *Ecole d'Administration des Affaires, Université de Liège, 4000 Liège, Belgium*
[b] *Department of Mathematics, University of Limburg, P.O. Box 616, 6200 MD Maastricht, The Netherlands*

Received 15 November 1991; revised manuscript received 15 September 1995

Abstract

The following problem was originally motivated by a question arising in the automated assembly of printed circuit boards. Given are n jobs, which have to be performed on a single machine within a fixed timespan $[0, T]$, subdivided into T unit-length subperiods. The processing time (or length) of each job equals p, $p \in \mathbb{N}$. The processing cost of each job is an arbitrary function of its start-time. The problem is to schedule all jobs so as to minimize the sum of the processing costs.

This problem is proved to be NP-hard, already for $p = 2$ and 0–1 processing costs. On the other hand, when $T = np + c$, with c constant, the problem can be solved in polynomial time. A partial polyhedral description of the set of feasible solutions is presented. In particular, two classes of facet-defining inequalities are described, for which the separation problem is polynomially solvable. Also, we exhibit a class of objective functions for which the inequalities in the LP-relaxation guarantee integral solutions.

Finally, we present a simple cutting plane algorithm and report on its performance on randomly generated problem instances.

Keywords: Scheduling; Computational complexity; Polyhedral description

1. Introduction

The following problem is studied in this paper. Given are n jobs, which have to be processed on a single machine within the timespan $[0, T]$. In our formulation, we assume T to be an integer, and the timespan is discretized into T time periods (or

* Corresponding author. e-mail: spieksma@math.rulimburg.nl.
[1] An extended abstract of this paper appeared in the Proceedings of the fourth IPCO Conference.

SSDI 0025-5610(95)00046-1

periods) of length one, viz. [0, 1], [1, 2], ..., $[T-1, T]$. Thus, period t refers to the time slot $[t-1, t]$, $t=1,\dots,T$. The machine can handle at most one job at a time. The *processing time*, or *length*, of each job equals p, $p \in \mathbb{N}$. The *processing cost* of each job is an arbitrary function of its start-time: we denote by c_{jt} the cost of starting job j in period t. The problem is to schedule all jobs so as to minimize the sum of the processing costs. We refer to this problem as problem SEL (*Scheduling jobs of Equal Length*).

Mathematically, SEL can be formulated as follows:

$$\min \quad \sum_{j=1}^{n} \sum_{t=1}^{T-p+1} c_{jt} x_{jt}$$

$$\text{subject to} \quad \sum_{t=1}^{T-p+1} x_{jt} = 1, \quad \text{for } j=1,\dots,n, \tag{1.1}$$

$$\sum_{j=1}^{n} \sum_{t=s}^{s+p-1} x_{jt} \leqslant 1, \quad \text{for } s=1,\dots,T-2p+2, \tag{1.2}$$

$$x_{jt} \in \{0, 1\}, \quad \text{for } j=1,\dots,n,\ t=1,\dots,T-p+1, \tag{1.3}$$

where $x_{jt}=1$ if job j starts in period t, and $x_{jt}=0$ otherwise.

Constraints (1.1) ensure that each job must start in some period, and constraints (1.2) imply that no more than one job can be scheduled in p consecutive periods. Obviously, the requirement that each job must be finished before T implies that the latest possible period for any job to start (its starting period) is period $T-p+1$. Constraints (1.3) are the integrality constraints.

In Section 2, this problem is shown to be strongly NP-hard, even when all jobs have length $p=2$. In Section 3, we show that the inequalities in the LP-relaxation of (1.1)–(1.3) define facets and we focus on objective functions for which these inequalities are in some sense sufficient. Section 4 presents more facet-defining and valid inequalities for the solution set of (1.1)–(1.3). Finally, we report in Section 5 on computational experiments with a simple cutting-plane algorithm.

Notice that the input of SEL consists of the numbers n, T, p and c_{jt}, for $j=1,\dots,n$, $t=1,\dots,T-p+1$. Thus, since we can assume that $p \leqslant T$, the size of the input is $O(nT \log(\max_{j,t} c_{jt}))$. It follows that an algorithm polynomial in n, T, p is a polynomial algorithm for SEL. This observation will allow us to conclude that two separation algorithms presented in Section 4 are polynomial-time algorithms.

Notice further that SEL is a special case of a scheduling problem (say, problem S) considered by Sousa and Wolsey (1992). In problem S, the jobs may have general processing times. Sousa and Wolsey propose several classes of facets and valid inequalities for S. It is an easy observation that, if $\{1,\dots,n\}$ is any subset of the jobs occurring in S, and all the jobs in $\{1,\dots,n\}$ have the same length p, then any valid inequality for (1.1)–(1.3) is also valid for S. This suggests that the polyhedral description presented in Sections 3 and 4 may prove useful, not only when all jobs strictly have equal length, but also in situations where the number of distinct lengths is small, or where most of the jobs have the same length. We now proceed to describe an interesting application in which the latter assumptions are fulfilled, and which originally motivated our study.

The electronics industry relies on numerically controlled machines for the automated assembly of printed circuit boards (PCBs). Prior to the start of operations, a number n of feeders, containing the electronic components to be mounted on the PCBs, are positioned alongside the machine, in some available slots $1, 2, \ldots, T$. A slot can accomodate at most one feeder. Each feeder j requires a certain number of slots, say p_j, depending on the feeder type; usually, p_j only takes a small number of values, say $p_j \in \{1, 2, 3\}$. In order to minimize the production makespan, it is desirable to position the feeders "close" to the locations where the corresponding components must be inserted. More precisely, for each combination of feeder j and slot t, a coefficient c_{jt} can be computed which captures the cost of assigning feeder j to slots $t, t+1, \ldots, t+p_j-1$. It should now be clear that finding a minimum-cost assignment of feeders to slots is equivalent to solving a scheduling problem with "small number of distinct processing times" (see, e.g., Ball and Magazine (1988) for a description of this model with $p_j = 1$ for all j, and Ahmadi et al. (1995), Crama et al. (1990), and Van Laarhoven and Zijm (1993) for a more general discussion).

Let us finally mention that SEL may be regarded as a discrete analogue of scheduling problems with unit-length tasks and arbitrary rational start-times (see, e.g., Garey et al. (1981) where minimizing the makespan is the objective considered). SEL is also superficially related to an assignment problem with side constraints investigated by Aboudi and Nemhauser (1990, 1991).

2. Complexity of SEL

It is obvious that, when each job has length 1 (the case $p = 1$), SEL reduces to an assignment problem, and hence is solvable in polynomial time. The following theorem shows that SEL is already strongly NP-hard for $p = 2$:

Theorem 2.1. *SEL is NP-hard, even for $p = 2$ and $c_{jt} \in \{0, 1\}$ for all j, t.*

Proof. An instance of SEL, with $p = 2$ and processing costs equal to 0 or 1, can be described by a bipartite graph $G = (V_1 \cup V_2, E)$. Each job is represented by a vertex in V_1, each period is represented by a vertex in V_2, and there is an edge $(j, t) \in E$, with $j \in V_1$ and $t \in V_2$, if and only if starting job j at period t has processing cost $c_{jt} = 0$. The instance of SEL admits a schedule with zero cost if and only if there exists a set of edges $A \subseteq E$ such that

(i) each vertex in V_1 is incident to precisely one edge in A,
(ii) each vertex in V_2 is incident to at most one edge in A, and
(iii) if vertex $t \in V_2$ is incident to an edge in A, then vertex $t+1$ is not incident to any edge in A, for all $t = 1, \ldots, |V_2| - 1$.

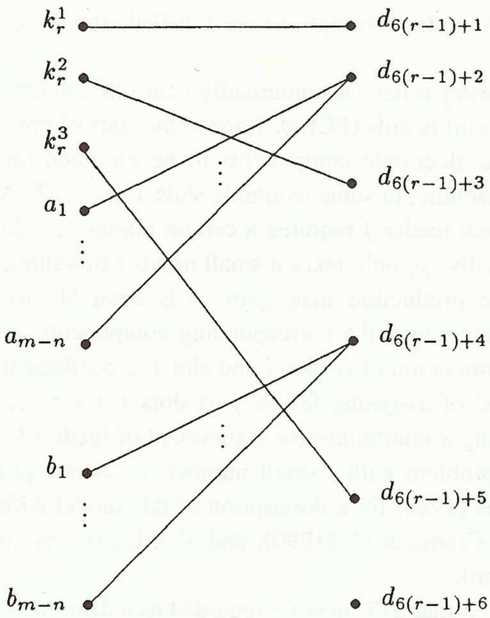

Fig. 1.

We use a reduction from the NP-hard three-dimensional matching problem (see Garey and Johnson (1979)). An instance I of three-dimensional matching is specified by three mutually disjoint sets K_1, K_2 and K_3 with $|K_i| = n$, for $i = 1, 2, 3$, and a set $Q \subseteq K_1 \times K_2 \times K_3$, with $|Q| = m$. The instance is feasible if there exists a set $Q' \subseteq Q$ such that every element of $K_1 \cup K_2 \cup K_3$ occurs in exactly one element of Q'. With I, we associate an instance of SEL as follows. Let

$$V_1 = K_1 \cup K_2 \cup K_3 \cup \{a_1, \ldots, a_{m-n}\} \cup \{b_1, \ldots, b_{m-n}\},$$
$$V_2 = \{d_1, \ldots, d_{6m}\}.$$

In order to define the edge-set E, denote by $Q_r = \{k_r^1, k_r^2, k_r^3\}$ the rth triple in Q, where

$$k_r^1 \in K_1, \quad k_r^2 \in K_2 \quad \text{and} \quad k_r^3 \in K_3 \quad (r = 1, \ldots, m).$$

Now, let E consist of the following edges:

$$\left(k_r^1, d_{6(r-1)+1}\right), \qquad \left(k_r^2, d_{6(r-1)+3}\right) \quad \text{and} \quad \left(k_r^3, d_{6(r-1)+5}\right)$$

for $r = 1, \ldots, m$ and

$$\left(a_s, d_{6(r-1)+2}\right) \quad \text{and} \quad \left(b_s, d_{6(r-1)+4}\right)$$

for $s = 1, \ldots, m-n$ and $r = 1, \ldots, m$.

A typical piece of the graph is shown in Fig. 1.

When the instance I of three-dimensional matching has a feasible solution, it is straightforward to find a set of edges $A \subseteq E$ which defines a zero-cost schedule. Conversely, assume that SEL has a feasible solution specified by an edge set A. Define

$D_r = \{d_{6(r-1)+1}, \ldots, d_{6r}\}$ for $r = 1, \ldots, m$. Notice that, for each $r = 1, \ldots, m$, at most three vertices of D_r are incident to some edge of A. Moreover, when there are exactly three such vertices, then these vertices are matched to Q_r by A. Let now

$$R_3 = \{r\text{: exactly three vertices of } D_r \text{ are incident to some edge of } A\},$$

$$R_2 = \{r\text{: at most two vertices of } D_r \text{ are incident to some edge of } A\}.$$

Since $|A| = |V_1| = 2m + n$, we get

$$2m + n \leqslant 3|R_3| + 2|R_2| = 3|R_3| + 2(m - |R_3|) = 2m + |R_3|.$$

From this, it directly follows that $Q' = \{Q_r\text{: } r \in R_3\}$ contains exactly n triples, and thus Q' defines a feasible solution of the three-dimensional matching problem. □

In fact, it can be proven that SEL remains NP-hard when each job can be processed at zero cost during three periods only (Spieksma and Crama, 1992).

Notice that the proof of Theorem 2.1 is easily adapted to show that a related problem, in which n_1 jobs have length 1, n_2 jobs have length 2, and $T = n_1 + 2 \cdot n_2$ (the minimal value of T allowing a feasible solution), is NP-hard too. This is to be contrasted with the following statement.

Theorem 2.2. *If $T = n \cdot p + c$, where $c \in \mathbb{N}$ denotes a given constant not part of the input, then SEL is polynomially solvable .*

Proof. Simply notice that, in this case, it is sufficient to solve $\binom{n+c}{c} = O(n^c)$ assignment problems, where each assignment problem corresponds to a set of starting periods allowing a feasible solution to SEL. Indeed, the feasible sets of starting periods are in 1–1 correspondence with 0–1 sequences of length $n + c$ containing exactly n ones and c zeros (with the zeros denoting idle periods between successive jobs). □

3. The LP-relaxation of SEL

Let us first recall some fundamental definitions from polyhedral theory (for a thorough introduction, the reader is referred to Nemhauser and Wolsey (1988)). Consider a polyhedron $P = \{x \in \mathbb{R}^k\colon Ax \leqslant b\}$. The *equality set* of (A, b) is the set of rows of (A, b), say $(A^=, b^=)$, such that: $A^= x = b^=$ for all x in P. The *dimension* of P is given by: $\dim(P) = k - \operatorname{rank}(A^=, b^=)$. The inequality $\alpha x \leqslant \alpha_0$ is *valid* for P if it is satisfied by all points in P. For a valid inequality $\alpha x \leqslant \alpha_0$, the set $F = \{x \in P\colon \alpha x = \alpha_0\}$ is called a *facet* of P if dim $(F) = \dim(P) - 1$. Equivalently, when $\emptyset \neq F \neq P$, F is a facet if and only if the following condition holds: if all points in F satisfy $\pi x = \pi_0$, for some $(\pi, \pi_0) \in \mathbb{R}^{k+1}$, then (π, π_0) is a linear combination of $(A^=, b^=)$ and (α, α_0) (see Nemhauser and Wolsey, 1988, p. 91).

Consider now the formulation in Section 1, and let P denote the convex hull of the feasible solutions to constraints (1.1)–(1.3). Furthermore, assume from now on that

$T \geqslant p \cdot (n+1)$. (Notice that $\dim(P) \leqslant n \cdot (T-p+1) - n = n \cdot (T-p)$. If $T < p \cdot (n+1)$, then it is easy to see that $\dim(P) < n \cdot (T-p)$; for instance $\sum_{j=1}^{n} \sum_{t=1}^{p} x_{jt} = 1$ is implied by (1.1) and (1.2)). To avoid trivialities, assume also $n > 2$, $p \geqslant 2$.

Sousa and Wolsey (1992) established the dimension of P. For the sake of completeness, we also include a proof of this result.

Theorem 3.1. $\dim(P) = n \cdot (T-p)$.

Proof. We just noticed that $\dim(P) \leqslant n \cdot (T-p)$. Suppose $\sum_{j=1}^{n} \sum_{t=1}^{T-p+1} \pi_{jt} x_{jt} = \pi_0$ for all $x \in P$; we want to show that this equality is implied by constraints (1.1).

To see this, fix j and t, $t \leqslant T-p$, and consider a solution with job j starting at period t, while the other jobs start arbitrarily at periods in $[1, t-p] \cup [t+p+1, T-p+1]$. Note that this is always possible; e.g., let $t = k \cdot p + q$, with $1 \leqslant q \leqslant p$; then, a feasible schedule can be found using only starting periods in

$$S_t = \{ j \cdot p + q : j = 0, \dots k \} \cup \{ j \cdot p + q + 1 : j = k+1, \dots, m \},$$

where m is the largest index such that $m \cdot p + q + 1 \leqslant T - p + 1$. Indeed, since $T \geqslant p \cdot (n+1)$, S_t contains at least n periods.

Consider now a second schedule, obtained by starting job j at period $t+1$, while all other jobs remain untouched. Comparing the two schedules, it follows easily that $\pi_{jt} = \pi_{j,t+1}$ for all $j = 1, \dots, n$, $t = 1, \dots, T-p$. (This construction will be used in subsequent proofs.) Thus, with $\pi_{jt} = \pi_j$ for all $j = 1, \dots, n$, $t = 1, \dots, T-p+1$, we get $\sum_{j=1}^{n} \sum_{t=1}^{T-p+1} \pi_{jt} x_{jt} = \sum_{j=1}^{n} \pi_j \sum_{t=1}^{T-p+1} x_{jt} = \pi_0$, which is a linear combination of the equalities (1.1). □

With the dimension of P established, we now can proceed to show that some inequalities define facets of P. First, we prove that the inequalities in the LP-relaxation of (1.1)–(1.3) are facet-defining.

Theorem 3.2. *The inequalities* $x_{jt} \geqslant 0$ *define facets of* P, *for all* $j = 1, \dots, n$, $t = 1, \dots, T-p+1$.

Proof. Let $F = \{x \in P\colon\ x_{is} = 0\}$ for any i, s with $1 \leqslant i \leqslant n$, $1 \leqslant s \leqslant T-p+1$ and suppose $\sum_{j=1}^{n} \sum_{t=1}^{T-p+1} \pi_{jt} x_{jt} = \pi_0$ for all $x \in F$.

To prove $\pi_j = \pi_{jt}$ for all $j = 1, \dots, n$, $j \neq i$, $t = 1, \dots, T-p+1$, we refer to the construction used in the proof of Theorem 3.1 (it is obvious that it is always possible to ensure that job i is not placed at s, for any s). Moreover, we can use this construction for job i and starting period t for all $t \leqslant s-2$ and $t \geqslant s+1$, proving that $\pi_{i1} = \pi_{i2} = \dots = \pi_{i,s-1}$ and $\pi_{i,s+1} = \pi_{i,s+2} = \dots = \pi_{i,T-p+1}$. Thus, for $s = 1$ or $s = T-p+1$, it follows that $\pi_i = \pi_{it}$ for all $t \neq s$.

If $s \neq 1$ and $s \neq T-p+1$, consider a solution with job i at period 1 and the other jobs at periods $1+p, 1+2p, \dots, 1+(n-1) \cdot p$, and a solution with job i at $T-p+1$, and all other jobs at the same periods as before (again, note that this is always possible, since we assumed $T \geqslant p \cdot (n+1)$). Comparing these solutions, it follows that $\pi_{i1} = \pi_{i,T-p+1}$ and thus $\pi_i = \pi_{it}$ for all $t \neq s$. So,

$$\sum_{j=1}^{n} \sum_{t=1}^{T-p+1} \pi_{jt} x_{jt} = \sum_{j=1}^{n} \pi_j \sum_{t=1}^{T-p+1} x_{jt} + \rho x_{is},$$

which shows that the equality $\sum_{j=1}^{n} \sum_{t=1}^{T-p+1} \pi_{jt} x_{jt} = \pi_0$ is a linear combination of (1.1) and of $x_{is} = 0$. □

Theorem 3.3. *The inequalities* (1.2) *define facets of* P.

Proof. Let $F = \{x \in P: \sum_{j=1}^{n} \sum_{t=s}^{s+p-1} x_{jt} = 1\}$, for any $1 \leqslant s \leqslant T-2p+2$, and suppose $\sum_{j=1}^{n} \sum_{t=1}^{T-p+1} \pi_{jt} x_{jt} = \pi_0$ for all $x \in F$. For any j and any t, consider a schedule using only starting periods in S_t (as in Theorem 3.1) and with $x_{jt} = 1$. There is always such a schedule corresponding to a point in F, unless $t = s-1$. Also, the schedule obtained by delaying the starting period of job j until $t+1$ is in F, unless $t = s+p-1$. From this, one easily concludes that, for all $j = 1, \ldots, n$,

$$\begin{aligned} &\pi_{j1} = \pi_{j2} = \cdots = \pi_{j,s-1} = \alpha_j, \\ &\pi_{js} = \pi_{j,s+1} = \cdots = \pi_{j,s+p-1} = \beta_j, \\ &\pi_{j,s+p} = \pi_{j,s+p+1} = \cdots = \pi_{j,T-p+1} = \gamma_j. \end{aligned} \tag{3.1}$$

If $2 \leqslant s \leqslant T-p$, then one can also show as in Theorem 3.2 that $\pi_{j1} = \pi_{j,T-p+1}$ for all $j = 1, \ldots, n$, or, more generally:

$$\gamma_j = \alpha_j \text{ for all } j = 1, \ldots, n. \tag{3.2}$$

Furthermore, simple interchange arguments yield:

$$\beta_j + \alpha_i = \alpha_j + \beta_i, \quad \text{for all } i, j \in \{1, 2, \ldots, n\}, \tag{3.3}$$

or equivalently $\delta = \beta_j - \alpha_j$ for all $j = 1, \ldots, n$. So, (3.1)–(3.3) together imply

$$\begin{aligned} \sum_{j=1}^{n} \sum_{t=1}^{T-p+1} \pi_{jt} x_{jt} &= \sum_{j=1}^{n} \alpha_j \left(\sum_{t=1}^{s-1} x_{jt} + \sum_{t=s+p}^{T-p+1} x_{jt} \right) + \sum_{j=1}^{n} \beta_j \sum_{t=s}^{s+p-1} x_{jt} \\ &= \sum_{j=1}^{n} \alpha_j \sum_{t=1}^{T-p+1} x_{jt} + \delta \sum_{j=1}^{n} \sum_{t=s}^{s+p-1} x_{jt}, \end{aligned}$$

which proves the theorem. □

Theorems 3.2 and 3.3 state that the inequalities in the LP-relaxation of (1.1)–(1.3) define facets of (1.1)–(1.3). In view of the NP-hardness of SEL, we obviously cannot hope that these inequalities alone suffice to describe P (as a matter of fact, they do not). However, it is conceivable that, by restricting ourselves to a certain class of objective functions, the inequalities in the LP-relaxation are in some sense sufficient. In the following we will explore this issue. Define

$$Q = \{x \in \mathbb{R}^+ : x \text{ satisfies } (1.1) \text{ and } (1.2)\}.$$

Notice that Q is the polytope defined by the inequalities in the LP-relaxation. Let us now address the following question: which restrictions on the objective function guarantee either that (i) all optimal vertices of Q are integral?, or – less demanding – that (ii) there exists an optimal vertex of Q that is integral?

If, for some c, condition (i) holds, we will say that *Q is integral with respect to c*. If, for some c, condition (ii) holds, we will say that *Q is weakly integral with respect to c*.

Notice that if Q is integral with respect to c, then any simplex-based LP-solver, when optimizing c over Q, will find an optimal integral solution to SEL. If Q is weakly integral with respect to c, the value found by the LP-solver will be equal to the cost of an optimal solution to SEL.

Consider the following restriction on the objective function c.

Restriction 1. For all $j = 1, \dots, n$, there exists a t_j with $1 \leqslant t_j \leqslant T - p + 1$ such that

$$c_{jt} \begin{cases} < c_{j,t+1} & \text{for } t = 1, \dots, t_j - 1, \\ > c_{j,t+1} & \text{for } t = t_j, \dots, T - p. \end{cases}$$

Theorem 3.4. *If c satisfies Restriction* 1, *then Q is integral with respect to c.*

Proof. Let us call each element of Q a feasible LP-solution, and let us call each $x \in Q$ such that $cx \leqslant cy$ for all $y \in Q$, an optimal LP-solution. It will sometimes be useful to think of $x \in Q$ as of a matrix with elements x_{jt}.

Consider an optimal LP-solution x^*. Let us refer to $\sum_{j=1}^{n} x^*_{jt}$ as the *weight of column* t, $t = 1, \dots, T - p + 1$. We claim that any optimal LP-solution satisfies the following property: there exists $r \in \{0, \dots, n\}$ such that
- columns $1, 1 + p, 1 + 2p, \dots, 1 + (r - 1)p$ have weight 1;
- column $1 + rp$ has weight $1 - \epsilon \in [0, 1]$;
- column $T - (n - r)p + 1$ has weight ϵ;
- columns $T - (n - r - 1)p + 1, T - (n - r - 2)p + 1, \dots, T - p + 1$ have weight 1.

Intuitively, one can explain this as follows. Consider a feasible LP-solution y. If some fraction $y_{jt} > 0$ with $t < t_j$ $(t > t_j)$ can be "shifted" to a smaller (greater) period, a solution with lower cost arises due to Restriction 1. Thus, such a shift cannot be possible in an optimal LP-solution, and this results in the property described.

Let us now establish the validity of the property in a more formal way. First, observe that there cannot be two jobs j_1, j_2 and two time periods s, t such that $t_{j_2} \leqslant s < t < t_{j_1}$, $x^*_{j_1,t} > 0$ and $x^*_{j_2,s} > 0$. Otherwise, indeed, we could construct a feasible LP-solution y with lower cost than x^* by setting $y_{jt} = x^*_{jt}$ for all j, t except:

$$y_{j_1,t} := x^*_{j_1,t} - \beta, \qquad y_{j_2,s} := x^*_{j_2,s} - \beta,$$

$$y_{j_1,s} := x^*_{j_1,s} + \beta, \qquad y_{j_2,t} := x^*_{j_2,t} + \beta, \quad \text{where } \beta = \min\left(x^*_{j_1,t},\, x^*_{j_2,s}\right).$$

Next, consider the first index $r \in \{0, \dots, n\}$ such that column $1 + rp$ has weight $1 - \epsilon$ with $\epsilon > 0$. Suppose that there exists a time period t such that, for some job j_1,

$1 + rp < t < t_{j_1}$ and $x^*_{j_1,t} > 0$. Choose t as small as possible with these properties. Then again, we could define a solution y with smaller cost than x^* by setting $y_{jt} = x^*_{jt}$ for all j, t, except:

$$y_{j_1,t} := x^*_{j_1,t} - \min\left(\epsilon,\ x^*_{j_1,t}\right), \qquad y_{j_1,1+rp} := x^*_{j_1,1+rp} + \min\left(\epsilon,\ x^*_{j_1,t}\right).$$

The solution y clearly satisfies constraints (1.1). It is also straightforward that y satisfies (1.2) if $t \leqslant (r+1)p$. Moreover, if $t > (r+1)p$, then the choice of t implies that $x^*_{j_2,s} = y_{j_2,s} = 0$ for all $1 + rp < s \leqslant (r+1)p$ and for all $j_2 \in \{1, \ldots, n\}$ (else we would have $x^*_{j_1,t} > 0$, $x^*_{j_2,s} > 0$ and $t_{j_2} \leqslant s < t < t_{j_1}$, which contradicts our first observation); hence (1.2) is satisfied in this case, too.

From the previous discussion we conclude that $x^*_{jt} > 0$ implies $t \geqslant t_j$ for all $t > 1 + rp$ and for all jobs j. In view of Restriction 1, it is now easy to argue that the weight of columns $T - p + 1, T - 2p + 1, \ldots, T - (n - r - 1)p + 1$ must be exactly 1, and that the weight of column $T - (n - r)p + 1$ is ϵ.

Now we will demonstrate that if x^* is fractional, it can be written as a convex combination of integral solutions, and therefore cannot be an extreme vertex of Q. This implies that Q is integral with respect to c.

Let us construct from the solution x^* a matrix M with n rows and n columns as follows. First, ''merge'' columns $1 + rp$ and $T - (n - r)p + 1$ (with weights ϵ and $1 - \epsilon$ respectively) into one column by summing the corresponding entries. Let M now consist of all columns in the solution x^* which have weight 1 (including the ''merged'' column). Obviously, M has n columns and n rows each with weight 1. Thus, we can apply Birkhoff's result (Birkhoff, 1946) on doubly stochastic matrices to show that M is a convex combination of some {0, 1}-matrices, which have the property that each column and each row contain precisely one 1. The solution to SEL corresponding to such a {0, 1}-matrix can be found straightforwardly: if an entry (i, j) is 1, then job i is scheduled at the period corresponding to the j-th column. Notice that if in x^* a job has positive fractions in both merged columns, this can be handled by ''splitting'' the corresponding {0, 1}-solution with multipliers according to those fractions. □

The reader will have no difficulty in verifying that, if we relax in Restriction 1 the ''$<$'' and ''$>$'' sign to ''$\leqslant$'' and ''$\geqslant$'' (let us call this relaxed Restriction 1), we can deduce the following corollary.

Corollary. *If c satisfies relaxed Restriction* 1, *then Q is weakly integral with respect to c.*

Notice also that, under relaxed Restriction 1, Q is not integral with respect to c, since even for a constant objective function (which certainly satisfies relaxed Restriction 1) all vertices of Q, including the nonintegral ones, are optimal.

The fact that we consider here a problem where all jobs have equal length is crucial for Theorem 3.4, as witnessed by the following example.

Example 3.1. Let $n = 2$, $p_1 = 1$, $p_2 = 2$ (where p_j, $j = 1, 2$ denotes the processing time of job j), and let

$$c_{jt} = \begin{pmatrix} 0 & 2 & 4 \\ 0 & 4 & 5 \end{pmatrix}$$

In order to accommodate jobs of different length in our formulation, we reformulate constraints (1.2) as follows:

$$\sum_{j=1}^{n} \sum_{s=\max(1,t-p_j+1)}^{t} x_{js} \leqslant 1, \quad \text{for } t = 1, 2, 3. \tag{3.4}$$

Now, a feasible solution to the model defined by constraints (1.1), (3.4) and the nonnegativity constraints is: $x_{11} = x_{12} = x_{21} = x_{23} = \frac{1}{2}$, $x_{13} = x_{22} = 0$. This solution has cost $3\frac{1}{2}$, whereas any optimal integral solution has cost 4.

Obviously, Restriction 1 subsumes the case where the cost-coefficient of each job is simply its starting period. Adding job-dependent release dates to this case translates into the following restriction on the objective function.

Restriction 2. For all j, there exist r_j, with $1 \leqslant r_j \leqslant T - np + 1$ such that, for all t:

$$c_{jt} = \begin{cases} t - r_j, & \text{if } t \geqslant r_j, \\ M, & \text{if } t < r_j, \end{cases}$$

where M is a sufficiently large number.

Theorem 3.5. *If c satisfies Restriction 2, then Q is integral with respect to c.*

Proof. Observe that in an optimal LP-solution x^*, $x^*_{jt} = 0$ if $c_{jt} = M$ for all j, t. This is due to the fact that we assumed that $r_j \leqslant T - np + 1$ for all j, which allows enough room to accommodate all weight on cost-coefficients whose value is not M.

Assume, without loss of generality, $r_1 \leqslant r_2 \leqslant \cdots \leqslant r_n$. Further, define s_i as follows, for $i = 1, \ldots, n$:

$$s_1 = r_1, \qquad s_i = \max(s_{i-1} + p, r_i).$$

We claim that in an optimal LP-solution x^*, columns s_i have weight 1 for $i = 1, \ldots, n$. The proof of this claim is by contradiction. Consider the minimal $i \in 1, \ldots, n$ for which column s_i has weight < 1. Let us refer to this column as column s_{i_1}. Obviously, there must exist a job, say job j_1, which is fractionally scheduled on a period $t_1 \leqslant s_{i_1}$ and has positive weight on a period $t_2 > s_{i_1}$, that is $x^*_{j_1,t_1} > 0$ and $x^*_{j_1,t_2} > 0$. We will now construct a feasible LP-solution y with lower cost than x^*, thereby contradicting the optimality of x^*.

The solution y can be constructed in the following way. Let us "transfer" a quantity $\epsilon = \min_{\{j,t: x^*_{jt} > 0\}} x^*_{jt}$ from $x^*_{j_1,t_2}$ to $x^*_{j_1,s_{i_1}}$. To be precise, set $y_{jt} = x^*_{jt}$, for all j, t, except:

$$y_{j_1,t_2} = x^*_{j_1,t_2} - \epsilon, \qquad y_{j_1,s_{i_1}} = x^*_{j_1,s_{i_1}} + \epsilon,$$

This solution y has gained $(t_2 - s_{i_1})\epsilon$ in cost, however, it may not satisfy constraints (1.2): constraint $\sum_{j=1}^{n} \sum_{t=s_{i_1}}^{s_{i_1}+p-1} y_{jt} \leqslant 1$ may be violated, since we added ϵ to the left-hand side. This can be repaired in the following way. Pick the first column t, with $s_{i_1} < t \leqslant s_{i_1} + p - 1$, such that $y_{jt} > 0$ for some job j, and set:

$$y_{jt} := y_{jt} - \epsilon, \qquad y_{j,t_2} := y_{j,t_2} + \epsilon.$$

Notice first that this yields a feasible LP-solution, and secondly, this deteriorates the cost of the previous solution y by at most $(t_2 - s_{i_1} - 1)\epsilon$. Thus, the solution y constructed here achieves a net gain of at least ϵ. This contradicts the optimality of x^* and therefore columns s_i have weight 1 for $i = 1, \dots, n$.

It follows that the optimal LP-solution contains n columns and n rows each with weight 1. Thus, we can use Birkhoff's result (1946) as we did in Theorem 3.4, to show that, if x^* is fractional, it can be written as convex combination of $\{0, 1\}$-solutions. □

Finally, consider the following restriction, which models a common release date and job-dependent due dates.

Restriction 3. For all j, t, $c_{jt} \in \{0, 1\}$; also, there exists $r \in \{1, \dots, T - p + 1\}$ and for all j, there exist $d_j \in \{r, \dots, T - p + 1\}$ such that

$$c_{jt} = \begin{cases} 0, & \text{for } t = r, \dots, d_j, \\ 1, & \text{for } t = 1, \dots, r-1 \text{ and for } t = d_j + 1, \dots T - p + 1. \end{cases}$$

Theorem 3.6. *If c satisfies Restriction 3, then Q is weakly integral with respect to c.*

Proof. Consider some optimal LP-solution x^*, and assume it is fractional. We prove the theorem by manipulating this solution so that an integral solution arises whose cost does not exceed the cost of the optimal LP-solution. First, we apply the following procedure. If the weight of column r is smaller than 1, find the earliest positive fraction after r (breaking ties arbitrarily) and shift it (or part of it) to period r. More formally, let the weight of column r equal $1 - \epsilon$, for some $\epsilon > 0$, and let t denote the smallest $t > r$ such that the weight of column t is positive. For some job j with $x_{jt}^* > 0$, we now set

$$x_{jt}^* := x_{jt}^* - \min(\epsilon, x_{jt}^*), \qquad x_{jr}^* := x_{jr}^* + \min(\epsilon, x_{jt}^*).$$

Repeat this step until column r has weight 1. Next, repeat this procedure for each of the columns $r + \alpha p$, $\alpha = 1, \dots, n-1$. If, for some $\alpha \in \{1, \dots, n-1\}$, $r + \alpha p > T - p + 1$, the procedure is continued for columns $1, 1 + p, \dots,$ until we obtain a solution in which n columns have weight 1. Due to the fact that "T is large enough" (we assumed $T \geqslant p \cdot (n+1)$) the solution constructed by repetition of the described procedure yields a feasible LP-solution. Also, it is easy to see that the cost of the solution constructed has not increased. Now, assume, without loss of generality that $d_1 \leqslant d_2 \leqslant \cdots \leqslant d_n$. Suppose $x_{1r} \neq 1$. Then there exists a job, say job j, such that $x_{jr}^* > 0$, $j \neq 1$, and there exists a column, say column t ($\neq r$), such that $x_{1t}^* > 0$. Let $\gamma = \min(x_{1t}^*, x_{jr}^*)$. Set

$$x_{1r}^* := x_{1r}^* + \gamma, \qquad x_{jr}^* := x_{jr}^* - \gamma,$$
$$x_{1t}^* := x_{1t}^* - \gamma, \qquad x_{jt}^* := x_{jt}^* + \gamma.$$

Notice that this solution is still a feasible LP-solution whose value is not worse than the original solution (since if $c_{1t} = 0$ then $c_{jt} = 0$ by the ordering we assumed). By repeating this step until $x^*_{1r} = 1$, and next by deleting columns $r, r+1, \ldots, r+p-1$ and jobs i with $d_i \leqslant r+p-1$, and repeating this procedure again, we finally find a $\{0, 1\}$-solution with the same cost as the cost of the LP-relaxation. □

Notice that under Restriction 3, even with $d_j = d$ for all j, Q is not integral with respect to c. This can be derived from the fact that, with $r = 1$ and $d_j = T - p + 1$ for all j, a constant objective function appears for which, as mentioned earlier, all vertices of Q are optimal.

In case we relax Restriction 3 to allow for job-dependent release times r_j, we lose the weak integrality of Q as witnessed by the following example.

Example 3.2. Let $n = 2$, $p = 2$ and let

$$c_{jt} = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

The solution $x_{12} = x_{14} = x_{21} = x_{23} = 1/2$, and all other $x_{jt} = 0$, is a feasible LP-solution with cost $1/2$. However, the optimal integral solution has cost 1.

4. More facet-defining and valid inequalities for SEL

In this section we will exhibit more facet-defining and valid inequalities for SEL. To start with, let us consider the following inequalities:

$$\sum_{t=s}^{s+p+l-1} x_{it} + \sum_{j=1}^{n} \sum_{t=s+l}^{s+p-1} x_{jt} \leqslant 1$$

$$\text{for } 1 \leqslant i \leqslant n,\ 1 \leqslant l \leqslant p-1 \text{ and } 1 \leqslant s \leqslant T-2p-l+2. \tag{4.1}$$

These inequalities are introduced in Sousa and Wolsey (1992). Notice that the inequalities (1.2) are the special case of (4.1) for $l = 0$. However, for reasons of convenience, we maintain the distinction between these two classes. It is not difficult to see that the inequalities (4.1) are valid, but they are also facet-defining, as witnessed by the next theorem (due to Sousa and Wolsey, 1992).

Theorem 4.1. *The inequalities* (4.1) *define facets of P.*

The validity of this theorem will also follow from the validity of the more general Theorem 4.3.

Observe that all (in)equalities (1.1), (1.2) and (4.1) are of the set-packing type, i.e., they only involve coefficients 0 or 1, and their right-hand side equals 1. In fact, the following holds.

Theorem 4.2. *All facets of P defined by set-packing inequalities are given by* (1.2) *and* (4.1).

Proof. Consider an arbitrary valid set-packing inequality I and define

$$t^* = \max_j \max_{t_2 \geqslant t_1} \{t_2 - t_1 \colon x_{j,t_2} \text{ and } x_{j,t_1} \text{ occur with coefficient 1 in } I\}.$$

Let i be the job which realizes t^*. We will make use of the following observation: no two variables x_{jt} and x_{ks}, with $k \neq j$, and $|s-t| \geqslant p$, can simultaneously occur with coefficient 1 in I.

Let us first consider the case $t^* \geqslant 2p-1$. Then, it is easy to verify that no variable, x_{it}, $i \neq j$, for any t, can occur in the inequality; thus I is implied by equalities (1.1), and cannot represent a facet.

Next, suppose $p \leqslant t^* \leqslant 2p-2$, i.e., $t^* = p+l-1$ for some $1 \leqslant l \leqslant p-1$. More specifically, suppose that x_{is} and $x_{i,\, s+p+l-1}$ have coefficient 1 in I. From our previous observation, it easily follows that, for any $j \neq i$, x_{jt} cannot occur in I if either $t \leqslant s+l-1$ or $t \geqslant s+p$. Hence, I is implied by (4.1).

Finally, when $t^* \leqslant p-1$, let s be the smallest index such that, for some k, x_{ks} occurs in I with coefficient 1. It follows again from our observation that, for all j and for all $t \geqslant s+p$, x_{jt} does not occur in I. Hence, I is implied by (1.2). □

(Van den Akker et al. (1993) have independently established that, for the more general scheduling problem S mentioned in Section 1, all facet-defining set-packing inequalities are given by Sousa and Wolsey (1992) (see also van den Akker (1994)).)

In the following we investigate generalizations of (4.1). To start with, (4.1) can be generalized to the following inequalities:

$$\begin{aligned} &\sum_{j \in J} \sum_{t=s}^{s+k\cdot p+l-1} x_{jt} + \sum_{j \notin J} \sum_{r=0}^{k-1} \sum_{t=s+l}^{s+p-1} x_{j,t+rp} \leqslant k, \\ &\text{for } J \subset \{1,\dots,n\} \text{ with } |J| = k > 0, \\ &1 \leqslant l \leqslant p-1 \text{ and } 1 \leqslant s \leqslant T-(k+1)\cdot p - l + 2. \end{aligned} \tag{4.2}$$

Notice that for $J = \{i\}$, (4.2) is equivalent to (4.1). The inequalities (4.2) are valid and even facet-defining as witnessed by the following theorem.

Theorem 4.3. *The inequalities* (4.3) *define facets of P.*

Proof. To facilitate the proof, we define subsets of periods which occur in (4.2). Let

$$\begin{aligned} A &= [s,\ s+k\cdot p+l-1], \\ B &= \{t+r\cdot p : r=0,\dots,k-1;\ t=s+l,\dots,s+p-1\} \\ &= [s+l,\ s+p-1] \cup [s+p+l,\ s+2p-1] \\ &\quad \cup \cdots \cup [s+(k-1)\cdot p+l,\ s+k\cdot p-1]. \end{aligned}$$

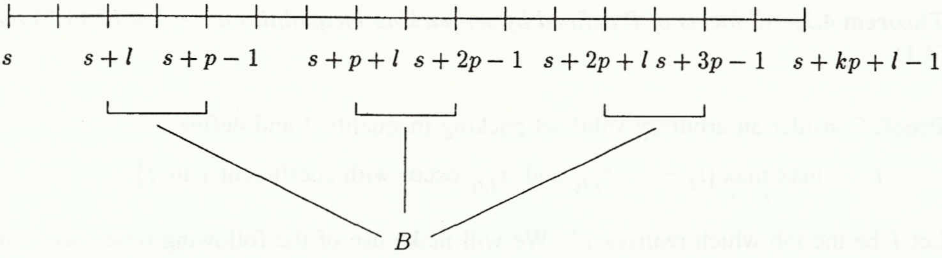

Fig. 2.

(see Fig. 2 for an illustration of the case $p = 5$, $k = 3$, $l = 2$.) With these notations, (4.2) can be rewritten as

$$\sum_{j \in J} \sum_{t \in A} x_{jt} + \sum_{j \notin J} \sum_{t \in B} x_{jt} \leqslant k. \tag{4.3}$$

First we show that these inequalities are valid. Suppose that $k+1$ jobs start in the interval $[s, s+k \cdot p + l - 1]$. The only way to achieve this is to start exactly one job in each of the intervals $[s, s+l-1], [s+p, s+p+l-1], \ldots, [s+k \cdot p, s+k \cdot p + l - 1]$ (this is easily checked by induction on k), i.e., to start the jobs in $A \setminus B$. However, the periods in $A \setminus B$ only occur in (4.3) for the k jobs in J. This implies that (4.3) is valid. Let us show now that (4.3) is facet-defining.

Let $F = \{x \in P\colon \sum_{j \in J} \sum_{t \in A} x_{jt} + \sum_{j \notin J} \sum_{t \in B} x_{jt} = k\}$ and suppose $\sum_{j=1}^{n} \sum_{t=1}^{T-p+1} \pi_{jt} x_{jt} = \pi_0$ for all $x \in F$.

Now, let $j \in J$ and $t \in A \setminus \{s + k \cdot p + l - 1\}$. Consider a solution with job j started at period t, and other jobs started at $t-p, t-2p, \ldots$ and $t+p+1, t+2p+1, \ldots$, in such a way that jobs in J are started in A (thus ensuring that $x \in F$). Shifting job j one period towards $t+1$ proves

$$\pi_{jt} = \pi_j^{\text{in}} \quad \text{for all } j \in J, \text{ for all } t \in A. \tag{4.4}$$

Let now $i \notin J$ and $t \in [s + r \cdot p + l, s + (r+1) \cdot p - 2]$, where $r \in \{0, \ldots, k-1\}$ (this is assuming $l \leqslant p-2$; else this step of the proof is not required). Consider the following solution: start job i at time t, start $k-1$ jobs from J in A, at periods $t-p$, $t-2p, \ldots$, and $t+p+1$, $t+2p+1, \ldots$, and start all other jobs outside A. Shifting job i one period proves

$$\pi_{it} = \pi_i^r, \quad \text{for all } i \notin J,\ t \in [s + r \cdot p + l, s + (r+1) \cdot p - 1],$$
$$r \in \{0, \ldots, k-1\}. \tag{4.5}$$

Also, interchanging job $i \notin J$ and $j \in J$ proves (for any $l \in \{1, \ldots, p-1\}$):

$$\pi_j^{\text{in}} + \pi_i^{r_1} = \pi_j^{\text{in}} + \pi_i^{r_2}, \quad \text{for all } r_1, r_2 \in \{0, \ldots, k-1\}. \tag{4.6}$$

(4.5) and (4.6) together imply

$$\pi_{it} = \pi_i^{\text{in}}, \quad \text{for all } i \notin J, \text{ for all } t \in B. \tag{4.7}$$

Now, a similar reasoning as in Theorem 3.3 ensures that

$$\pi_{it} = \pi_i^{\text{out}}, \quad \text{for all } i \notin J, \text{ for all } t \notin A. \tag{4.8}$$

Furthermore, consider a solution with the jobs from J at s, $s+p, \dots, s+(k-1)\cdot p$ and job i, $i \notin J$ at $s+k\cdot p$. Simple interchange arguments imply, together with (4.3) and (4.8),

$$\pi_{it} = \pi_i^{\text{out}}, \quad \text{for all } i \notin J,\ t \notin B. \tag{4.9}$$

Also, similar arguments imply

$$\pi_{jt} = \pi_j^{\text{out}}, \quad \text{for all } j \in J,\ t \notin A. \tag{4.10}$$

Moreover, it is easy to see that

$$\pi_j^{\text{in}} + \pi_i^{\text{out}} =_j^{\text{out}} + \pi_i^{\text{in}}, \quad \text{for all } i, j \in \{1, 2, \dots, n\}, \tag{4.11}$$

or equivalently $\beta = \pi_j^{\text{in}} - \pi_j^{\text{out}}$ for all $j = 1, \dots, n$.

Now (4.3), (4.7), (4.9)–(4.11) imply

$$\begin{aligned}
&\sum_{j=1}^{n} \sum_{t=1}^{T-p+1} \pi_{jt} x_{jt} \\
&= \sum_{j\in J} \sum_{t\in A} \pi_j^{\text{in}} x_{jt} + \sum_{j\in J} \sum_{t\notin A} \pi_j^{\text{out}} x_{jt} + \sum_{j\notin J} \sum_{t\in B} \pi_j^{\text{in}} x_{jt} + \sum_{j\notin J} \sum_{t\notin B} \pi_j^{\text{out}} x_{jt} \\
&= \sum_{j=1}^{n} \pi_j^{\text{out}} \sum_{t=1}^{T-p+1} x_{jt} + \beta \cdot \left(\sum_{j\in J} \sum_{t\in A} x_{jt} + \sum_{j\notin J} \sum_{t\in B} x_{jt} \right),
\end{aligned}$$

proving the theorem. □

Even though there is an exponential number of inequalities of type (4.2), the separation problem for this class of inequalities is polynomially solvable. Indeed, notice that (4.2) can be rewritten as

$$\sum_{j=1}^{n} \sum_{r=0}^{k-1} \sum_{t=s+l}^{s+p-1} x_{j,t+r\cdot p} + \sum_{j\in J} \sum_{r=0}^{k} \sum_{t=s}^{s+l-1} x_{j,t+r\cdot p} \leqslant k. \tag{4.12}$$

We want to check whether a given x^* violates one of these inequalities. Fix s, l and k (there are only $O(pnT)$ choices for these three values). Then, the first term of (4.12) is a constant. Pick the k values of j which maximize the second term and put them in a set J^*. If x^* violates (4.12) for any J, then it does so for J^*.

Another way of generalizing the inequalities (4.1) is the following. Choose a nonempty set $J \subset \{1, \dots, n\}$ and a nonempty set $S \subset \{1, \dots, T-2p+2\}$. For each $s \in \{1, \dots, T-p+1\}$, define $q_s = 1$ if $s \in S$ and $q_s = 0$ otherwise. Then, by adding the constraints (1.1) for $j \in J$ and the constraints (1.2) for $s \in S$, each with coefficient $1/2$, we obtain the following valid inequality:

$$\sum_{j\in J} \sum_{t=1}^{T-p+1} \left\lfloor \left(\tfrac{1}{2} + \tfrac{1}{2} \cdot \sum_{s=t-p+1}^{t} q_s \right) \right\rfloor \cdot x_{jt} + \sum_{j \notin J} \sum_{t=1}^{T-p+1} \left\lfloor \tfrac{1}{2} \cdot \sum_{s=t-p+1}^{t} q_s \right\rfloor \cdot x_{jt} \leqslant \left\lfloor \tfrac{1}{2}(|J| + |S|) \right\rfloor \tag{4.13}$$

We refer to these inequalities as (J, S) inequalities. The inequalities (4.1) are the special case of (4.13) obtained for $J = \{i\}$ and $S = \{s, s+l\}$. Of course, a more sophisticated choice for S could lead to other valid inequalities. Indeed, it is possible to generalize inequalities (4.1) by choosing S as k couples of periods in the following way: for some $k \geqslant 2$ and $l \in \{1, \ldots, p-1\}$,

$$S = \{s,\ s+l,\ s+p,\ s+p+l, \ldots, s+(k-1)\cdot p,\ s+(k-1)\cdot p + l\}.$$

However, the resulting (J, S) inequalities do not define facets of P. In fact, (when $k \geqslant 2$ of course), they can be strengthened by lifting certain coefficients to 2. The following inequalities result:

$$\sum_{j\neq i} \sum_{t=s+l}^{s+k\cdot p-1} x_{jt} + \sum_{t=s}^{s+k\cdot p+l-1} x_{it} + \sum_{r=1}^{k-1} \sum_{t=s}^{s+l-1} x_{i,t+r\cdot p} \leqslant k,$$
$$\text{for } i, k, l, s \text{ with } 1 \leqslant i \leqslant n,\ 1 \leqslant k \leqslant n,\ 1 \leqslant l \leqslant p-1 \text{ and } 1 \leqslant s \leqslant T-(k+1)\cdot p - l + 2. \tag{4.14}$$

(Observe that, when $k \geqslant 2$, then some variables occur with coefficient 2 in (4.14).) The following holds.

Theorem 4.4. *The inequalities* (4.14) *define facets of* P.

Proof. We first introduce some notation. With i, k, l, s as in (4.14), let

$$A = [s,\ s+k\cdot p+l-1],$$
$$C = [s+l,\ s+k\cdot p-1],$$
$$D = \{t + r\cdot p : r = 1, \ldots, k-1;\ t = s, \ldots, s+l-1\}$$
$$= [s+p,\ s+p+l-1] \cup [s+2p,\ s+2p+l-1] \cup \cdots \cup [s+(k-1)\cdot p,\ s+(k-1)\cdot p+l-1].$$

We can rewrite (4.14) as

$$\sum_{j\neq i} \sum_{t\in C} x_{jt} + \sum_{t\in A\setminus D} x_{it} + 2\cdot \sum_{t\in D} x_{it} \leqslant k \tag{4.15}$$

Let us first show that (4.15) is valid for P. Consider any feasible schedule. It is easy to see that the only way to start k jobs in C is to start them in $C \setminus D = [s+l,\ s+p-1] \cup \cdots \cup [s+(k-1)\cdot p+l,\ s+k\cdot p-1]$ (one job in each subinterval). But, if this is the case, then there is no room left to start job i in A, and hence (4.15) is satisfied. So, the only way to violate (4.15) is to start $k-1$ jobs in C, and job i in D. Let us suppose job i starts at $s+r\cdot p+q$, $r\in\{1,\dots,k-1\}$, $q\in\{0,\dots,l-1\}$. Then two intervals of consecutive periods remain for placing $k-1$ jobs in C:

$$[s+l,\ s+(r-1)\cdot p+q] \quad\text{and}\quad [s+(r+1)\cdot p+q,\ s+k\cdot p-1].$$

But it is easy to check that no $k-1$ jobs can start in these intervals. This establishes the validity of (4.15). Let us show now that (4.15) is facet-defining. Let

$$F=\left\{x\in P:\ \sum_{j\neq i}\sum_{t\in C}x_{jt}+\sum_{t\in A\setminus D}x_{it}+2\sum_{t\in D}x_{it}=k\right\},$$

and suppose $\sum_{j=1}^{n}\sum_{t=1}^{T-p+1}\pi_{jt}x_{jt}=\pi_0$ for all $x\in F$.

Consider a solution with job j, $j\neq i$, starting at period $t\in C\setminus\{s+k\cdot p-1\}$. Let the other jobs start at $t-p$, $t-2p,\dots,$ and at $t+p+1$, $t+2p+1,\dots,$ while ensuring that $x\in F$ (this is always possible). Shifting job j towards period $t+1$ proves that

$$\pi_{jt}=\pi_j^{\text{in}},\quad \text{for all } j\neq i,\ \text{for all } t\in C. \tag{4.16}$$

Now, consider a solution with job i placed at t, with $t\in[s,\ s+l-1]$, and all other jobs at $t-p$, $t-2p,\dots,$ and $t+p+1$, $t+2p+1,\dots$. This can be done in such a way that $x\in F$, since $t+p+1$, $t+2p+1,\dots,$ $t+(k-1)\cdot p+1$ are $k-1$ periods in C. Now, shifting job i from t to $t+1$ proves that

$$\pi_{it}=\pi_i^{\text{in1}},\quad \text{for all } t\in[s,\ s+l]. \tag{4.17}$$

A similar argument shows:

$$\pi_{it}=\pi_i^{\text{in2}},\quad \text{for all } t\in[s+k\cdot p,\ s+k\cdot p+l-1]. \tag{4.18}$$

Consider next a schedule x with job i starting at t, $t\in[s+r\cdot p+l,\ s+(r+1)\cdot p-2]$, for some $0\leqslant r\leqslant k-1$, and $k-1$ other jobs starting at $t-r\cdot p$, $t-(r-1)\cdot p,\dots,t-p$, $t+p+1,\dots,t+(k-r-1)\cdot p+1$. Notice that the latter periods are all in $C\setminus D$, and hence $x\in F$. Comparing x with another schedule in which job i starts at $t+1$ shows that $\pi_{it}=\pi_{i,t+1}$ for all $t\in[s+r\cdot p+l,\ s+(r-1)\cdot p-2]$. Also exchanging job i with one of the other jobs which start in $C\setminus D$ shows, in combination with our previous observations (4.16)–(4.18) that

$$\pi_{it}=\pi_i^{\text{in1}}=\pi_i^{\text{in2}}=\pi_i^1\quad \text{for all } t\in A\setminus D. \tag{4.19}$$

Now, consider a solution with job i starting at t, $t\in D$ and place the other jobs at $t-p, t-2p,\dots$ and $t+p$, $t+2p,\dots,$ ensuring that the solution is in F (notice that exactly $k-2$ of these periods are in D, and hence in C). Interchanging job i and job $j\neq i$ leads easily to

$$\pi_{it}=\pi_i^2\quad \text{for all } t\in D. \tag{4.20}$$

To prove $\pi_{jt} = \pi_j^{\text{out}}$ for all j, for all $t \notin C$, we refer to the construction used in Theorem 3.3. Moreover, simple interchange arguments imply:

$$\pi_{j_1}^{\text{out}} + \pi_{j_2}^{\text{in}} = \pi_{j_1}^{\text{in}} + \pi_{j_2}^{\text{out}}, \quad \text{for all } j_1, j_2 \neq i,$$

$$\pi_i^1 + \pi_j^{\text{out}} = \pi_i^{\text{out}} + \pi_j^{\text{in}}, \quad \text{for all } j \neq i,$$

$$\pi_i^2 + \pi_j^{\text{out}} = \pi_i^{\text{in}} + \pi_j^{\text{in}}, \quad \text{for all } j \neq i.$$

With these last equalities established and together with (4.16), (4.19) and (4.20) the theorem follows easily. □

Notice that there are $O(pn^2T)$ inequalities in the class (4.14). Hence, the separation problem for this class of inequalities can be solved in polynomial time.

5. A cutting-plane algorithm for SEL

In this section, we describe an unsophisticated cutting-plane algorithm for SEL, based on the results of Sections 3 and 4, and we report on its performance on randomly generated problem instances. We are mainly interested in the question whether the inequalities derived in Section 4 are of any practical relevance, that is whether they are able to cut off fractional solutions of the problems we generated and whether they are able to improve the LP lower bound. Therefore, no attempts were made to minimize or even record running times of the algorithm for the various problem instances. Concerning this topic of running times, we will restrict ourselves to some general remarks later in this section.

The cutting-plane algorithm works as follows. We start with a model consisting solely of the constraints (1.1). This model is solved to optimality (we used the LP-package LINDO). Then the following six classes of inequalities are searched successively in order to find violated inequalities (where R denotes the following set of periods (see Section 4):

$$R = \{s,\ s+l,\ s+p,\ s+p+l, \ldots, s+(k-1)\cdot p,\ s+(k-1)\cdot p+l\}):$$

Class 1: constraints (1.2),
Class 2: constraints (4.1),
Class 3: constraints (4.2) with $k > 1$,
Class 4: constraints (4.14) with $k > 1$,
Class 5: constraints (4.13) with $|J| = 2$, and $S = R \cup \{s_1\}$ with s_1 such that $1 \leqslant s_1 \leqslant s - p$ or $s + (k-1)\cdot p \leqslant s_1 \leqslant T - 2p + 2$,
Class 6: constraints (4.13) with $|J| = 3$, and $S = R \cup \{s_1, s_2\}$ with s_1, s_2 such that $1 \leqslant s_1, s_2 \leqslant s - p$ or $s + (k-1)\cdot p \leqslant s_1, s_2 \leqslant T - 2p + 2$ and $s_1 \leqslant s_2 - p$.

When violated inequalities are found, they are added to the model, the extended model is solved to optimality and the whole process is repeated. When no violated inequalities are detected or if an integral solution is found, the algorithm stops.

Table 1

		LP	CPA	OPT			LP	CPA	OPT
$p=2$	1	4 (i)	4 (i)		$p=3$	1	14.25	14.40	15
$n=20$	2	7 (i)	7 (i)		$n=30$	2	14.00	14.47	15
$T=46$	3	6.50	7 (i)		$T=102$	3	15.21	15.60	17
c_{jt} in	4	12.33	13 (i)		c_{jt} in	4	9 (i)	9 (i)	
[0–25]	5	9 (i)	9 (i)		[0–40]	5	10.00	10.67	11
$p=2$	1	7	7 (i)		$p=4$	1	15.00	15.81	17
$n=30$	2	10.5	11 (i)		$n=20$	2	19.50	22 (i)	
$T=66$	3	9	9.5	10	$T=93$	3	20.08	21.33	22
c_{jt} in	4	11.11	11.38	12	c_{jt} in	4	23.00	24.40	26
[0–25]	5	7.33	8 (i)		[0–60]	5	23.43	25 (i)	
$p=3$	1	4.88	5 (i)		$p=5$	1	12.62	14.5	16
$n=20$	2	8.17	9 (i)		$n=20$	2	11.00	11 (i)	
$T=67$	3	11.27	11.43	12	$T=114$	3	21.17	22.88	23
c_{jt} in	4	8 (i)	8 (i)		c_{jt} in	4	22.14	24.24	26
[0–25]	5	9.67	10 (i)		[0–60]	5	15.90	16 (i)	

A few implementation issues are worth mentioning. First, if violated inequalities in one of the six classes are found, then subsequent classes are not checked. Secondly, at each iteration, only those inequalities are maintained whose slack is smaller than 0.1; all other inequalities are removed from the model. Observe also that, for all classes of valid inequalities used in this algorithm, the separation problem is polynomially solvable.

The cutting-plane algorithm was tested on sixty problem instances divided over two types. We generated thirty problem instances from Type 1, distributed over six categories, where a category is determined by a specific choice of p and n (see Table 1 for the problem instances of Type 1). Problem instances of this type are such that each cost-coefficient c_{jt} is drawn from a uniform distribution whose range can also be found in Table 1.

Problem instances from Type 2 (see Table 2) represent the case of weighted start-times with job-dependent release dates and deadlines. Here, for each job j, the release date r_j is an integer drawn uniformly between 1 and $1/2\,pn$; the deadline d_j is an integer drawn uniformly between r_j and $0.6\,pn$, and a weight w_j is drawn from the uniform distribution between 1 and 10. The cost-coefficients of job j are now defined as follows:

$$c_{jt} = \begin{cases} w_j(t-r_j), & \text{if } r_j \leqslant t \leqslant d_j, \\ M, & \text{otherwise (where } M \text{ denotes a large integer).} \end{cases}$$

Similar cost functions are considered by Sousa and Wolsey (1992) and van den Akker (1994) for jobs having arbitrary lengths.

In Tables 1 and 2, LP denotes the value of the LP-relaxation of model (1.1)–(1.3). CPA denotes the value found by the cutting-plane algorithm described earlier, and OPT denotes the value of an optimal solution, which was found by applying the branch-and-bound algorithm implemented in LINDO (where only those variables which were

Table 2

		LP	CPA			LP	CPA
$p=2$	1	9016	9018 (i)	$p=3$	1	12258	12258 (i)
$n=20$	2	9039	9039 (i)	$n=30$	2	13110	13110 (i)
$T=46$	3	9007.5	9008 (i)	$T=102$	3	12221	12221 (i)
5 out	4	9048	9049 (i)	5 out	4	12142	12143 (i)
of 33	5	8048	8048 (i)	of 11	5	12112	12112 (i)
$p=2$	1	13035	13035 (i)	$p=4$	1	8597.5	9046 (i)
$n=30$	2	13037.5	13041 (i)	$n=20$	2	9098	9098 (i)
$T=66$	3	12081	12081 (i)	$T=93$	3	8035	8035 (i)
5 out	4	14034	14034 (i)	5 out	4	9030	9033 (i)
of 20	5	12145	12145 (i)	of 8	5	8062	8062 (i)
$p=3$	1	8071	8071 (i)	$p=5$	1	8307	8307 (i)
$n=20$	2	9026	9026 (i)	$n=20$	2	10103	10103 (i)
$T=67$	3	8079	8079 (i)	$T=114$	3	9116	9116 (i)
5 out	4	8052.5	8053 (i)	5 out	4	8127	8127 (i)
of 29	5	8050	8050 (i)	of 14	5	9063	9063 (i)

fractional in the solution of the LP-relaxation are forced to be 0 or 1). The symbol "(i)" denotes that the solution found is integral. Notice that all cost-coefficients are integral, so that all lower bounds computed can validly be rounded-up to the next integer.

Let us first comment on the results depicted in Table 1. Regarding the choice of T, preliminary experiments indicated that for relatively large values of $T(T \geqslant (p+1) \cdot n)$ as well as for minimal values of $T(T = p \cdot (n+1))$, the LP-relaxation of model (1.1)–(1.3) almost always has an integral optimal solution. So, we tried to choose T in such a way that fractional LP-relaxations arise.

For the thirty instances considered in Table 1, the cutting-plane algorithm finds seventeen times an integral solution (compared to five times for the LP-relaxation of (1.1)–(1.3)) and, for the remaining instances, it improves the lower bound nine times. Not surprisingly, the results indicate that the problems get harder when p and/or n increase. For the "easier" problems ($p=2$, $n=20$, 30 and $p=3$, $n=20$), the cutting-plane algorithm often finds integral optimal solutions. For the "harder" problems ($p=3$, $n=30$, and $p=4$, 5, $n=20$) the algorithm usually improves the lower bound obtained from the LP-relaxation of (1.1)–(1.3). In case the cutting-plane algorithm terminated with a fractional solution, it had used inequalities from all six classes. So, for this type of problems, it appears that the inequalities derived in Section 4 are quite useful. The running time of the cutting-plane algorithm largely depends on the number of LP's which have to be solved. Generally speaking, this number increases from 10–20 for the easy problems to 80–120 for the hard problems. Of course, one can influence this number by the strategy one employs in adding valid inequalities.

Consider now the problem instances of Type 2. For this type, the LP-solutions very often turned out to be integral. We employed the following strategy in order to get instances whose LP-solution was not integral. For each of the categories, we continued generating random problem instances until five problems were available whose LP-solution was fractional. The total number of instances we had to generate for each category to find those five instances can be found in Table 2. Next, we ran the cutting-plane algorithm on the thirty instances we had selected in this way.

The results summarized in Table 2 show that the algorithm works quite satisfactorily for this type of problem instances. In all cases, the algorithm finds an integral solution. The LP lower bound is improved eight times. For fourteen problem instances, inequalities from class 2 were used; six times inequalities from class 3 were used, and twice inequalities from class 4 and 5 were used. Except for two problem instances, the number of iterations was below twenty.

Acknowledgements

We are grateful to Antoon Kolen for pointing out the inequalities (4.13) to us, and to Hans-Jürgen Bandelt for his comments on an earlier version of this paper. The first author has been partially supported in the course of this research by ONR (grants N00014-92-J-1375 and N00014-92-J-4083) and by NATO (grant CRG 931531).

References

R. Aboudi and G.L. Nemhauser, "An assignment problem with side constraints: strong cutting planes and separation," in: J.J. Gabszewicz, J.-F. Richard and L.A. Wolsey, eds., *Economic Decision-Making: Games, Econometrics and Optimisation* (Elsevier, Amsterdam, 1990) pp. 457–471.

R. Aboudi and G.L. Nemhauser, "Some facets for an assignment problem with side constraints," *Operations Research* 39 (1991) 244–250.

J. Ahmadi, R. Ahmadi, H. Matsuo and D. Tirupati, "Component fixture positioning-sequencing for printed circuit board assembly with concurrent operations," *Operations Research* 43 (1995) 444–457.

M.O. Ball and M.J. Magazine, "Sequencing of insertions in printed circuit board assembly," *Operations Research* 36 (1988) 192–201.

G. Birkhoff, "Tres observaciones sobre el algebra lineal," *Revista Universidad Nacional de Tucumán, Series A* 5 (1946) 147–151.

Y. Crama, A.W.J. Kolen, A.G. Oerlemans and F.C.R. Spieksma, "Throughput rate optimization in the automated assembly of printed circuit boards," *Annals of Operations Research* 26 (1990) 455–480.

M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-completeness* (Freeman, New York, 1979).

M.R. Garey, D.S. Johnson, B.B. Simons and R.E. Tarjan, "Scheduling unit-time tasks with arbitrary release times and deadlines," *SIAM Journal on Computing* 10 (1981) 256–269.

G.L. Nemhauser and L.A. Wolsey, *Integer and Combinatorial Optimization* (Wiley, New York, 1988).

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

F.C.R. Spieksma and Y. Crama, "The complexity of scheduling short tasks with few starting times," Report M92-06, University of Limburg, The Netherlands (1992).

J.M. van den Akker, "LP-based solution methods for single-machine scheduling problems", Ph.D. Thesis, Eindhoven University of Technology, The Netherlands (1994).

J.M. van den Akker, C.P.M. van Hoesel and M.W.P. Savelsbergh, "Facet inducing inequalities for single-machine scheduling problems," Memorandum COSOR 93-27, Eindhoven University of Technology, The Netherlands (1993).

P.J.M. van Laarhoven and W.H.M. Zijm, "Production preparation and numerical control in PCB assembly," *International Journal of Flexible Manufacturing Systems* 5 (1993) 187–207.