A Flexible Model and a Hybrid Exact Method for Integrated Employee Timetabling and Production Scheduling

Christian Artigues, Michel Gendreau, Louis-Martin Rousseau

A Flexible Model and a Hybrid Exact Method for Integrated Employee Timetabling and Production Scheduling

Christian Artigues[1], Michel Gendreau[2] and Louis-Martin Rousseau[2]

[1] Université de Toulouse, LAAS-CNRS, 7 avenue du Colonel Roche,
31077 Toulouse, Cedex 4, France
`artigues@laas.fr`

[2] CRT, Université de Montréal, CP 6128, succ. Centre-ville,
Montreal, QC, Canada H3C 3J7 {`michelg,louism`}`@crt.umontreal.ca`

Abstract. We propose a flexible model and several integer linear programming and constraint programming formulations for integrated employee timetabling and production scheduling problems. A hybrid constraint and linear programming exact method is designed to solve a basic integrated employee timetabling and job-shop scheduling problem for lexicographic minimization of makespan and labor costs. Preliminary computational experiments show the potential of hybrid methods.

1 Introduction

In production systems, the decisions related to scheduling jobs on the machines and the decisions related to employee timetabling are often made in a sequential process. The objective of job scheduling is to minimize the production costs whereas the objective of employee timetabling is to maximize employee satisfaction (or to minimize labor costs). Either the employee timetabling is first established and then the scheduling of jobs must take employee availability constraints into account or the scheduling of jobs is done first and the employees must then adapt to cover the machine loads. It is well known that optimizing efficiently an integrated process would both improve production costs and employee satisfaction. However, the resulting problem has generally been considered as too complex to be used in practical situations. Some attempts have been made [1, 3, 7, 9–11, 21] but mostly considering an oversimplified version of the employee timetabling problem. Nevertheless the integration of task scheduling and employee timetabling has been successfully developed in complex transportation systems [6, 8, 13, 17, 19, 22, 23]. In this paper we propose a model of integrated production and employee scheduling that takes account of the following possible specific characteristics of the production context:

A) An employee that has started a task may be replaced at any moment by another employee (of the same skill) with no notable effect nor interruption of the processed task.

B) An employee is not necessarily needed during all the processing time of a task but only at some time periods that can occur before, during and after the processed task (setups, removals, transportation).
C) Because of the automated production process, or the nature of the tasks performed by the employee (e.g. supervision), an employee may perform several tasks simultaneously during a shift.
D) The production process can be quasi-continuous (on a 24 h basis) whereas the employee timetabling has to be discretized in periods (on an 8-hour basis for instance).
E) The duration of a task may change depending on the number or on the skill of the assigned workers.

In Section 2, we review the related work dealing with the integration of task and employee scheduling and we give the position of the considered problem among the various production scheduling and employee timetabling problems. In Section 3, we propose different ILP formulations of the considered problem. A constraint programming formulation is proposed in Section 4. In Section 5, we propose a hybrid framework to solve the lexicographic minimization of makespan and labor costs. In Section 6, we provide the results of a preliminary computational experiment carried out on a set of employee timetabling and job-shop scheduling instances. Concluding remarks are drawn in Section 7.

2   Literature Review and Position of the Considered Problem

We review some of the integrated vehicle and crew scheduling methods in Section 2.1 and the previous work on integrated production scheduling and employee timetabling in Section 2.2. We give the position of the considered problem in Section 2.3.

2.1   Vehicle and Crew Scheduling

Integrated vehicle and crew scheduling is an active research area in transportation systems, see [6, 8, 13, 17, 19, 22, 23, 28] among others.

We focus hereafter on some recent papers presenting different models and solution methods. Cordeau et al. [8] propose a Benders decomposition scheme to solve aircraft routing and crew scheduling problems. They use a set partitioning formulation for both the aircraft routing and the crew scheduling. In the first scheme, the primal subproblem involves only crew scheduling variables and the master problem involves only aircraft routing variables. Both the primal subproblem and master problem relaxation are solved by column generation. Integer solutions are found by a three-phase method, adding progressively the integrity constraints. More recently, Mercier et al. [23] have improved the robustness of the proposed model. Their method reverses the Benders decomposition proposed in [8] by considering the crew scheduling problem as the master problem.

Haase and Fridberg [19] propose a method to solve bus and driver scheduling problems. The problem is formulated as a set partitioning problem with additional constraints in which a column represents either a schedule for a crew or for a vehicle. The additional constraints are introduced to connect both schedule types. A branch-and-price-and-cut algorithm is proposed in which column generation is performed to generate both vehicle and crew schedules. The method is improved in [18] with a set partitioning formulation only for the driver scheduling problem that incorporates side constraints for the bus itineraries. These side constraints guarantee that a feasible vehicle schedule can be derived afterwards in polynomial time. Furthermore, the inclusion of vehicle costs in this extended crew scheduling formulation ensures the overall optimality of the proposed two-phase crew-first, vehicle-second approach.

Freling et al. [17] propose a method to solve bus and driver scheduling problems on individual bus lines. They propose a formulation that mixes the set partitioning formulation for crew scheduling and the assignment formulation for the vehicle scheduling problem. They compute lower bound and feasible solutions by combining Lagrangian relaxation and column generation. Columns correspond to crew scheduling variables. The constraints involving the current columns are relaxed in a Lagrangian way. The obtained Lagrangian dual problem is a single-depot vehicle scheduling problem (SDVSP). Once the Lagrangian relaxation is solved a new set of columns with negative reduced costs is generated. The method is iterated until the gap between the so-computed lower bound and an estimated lower bound is small enough. Feasible solutions are generated from the last feasible SDVSP and the current set of columns.

2.2   Production and Employee Scheduling

Specific employee scheduling problems involved in production scheduling are often tackled considering the job schedule is fixed. As a representative work in this area, Valls et al. [27] consider a fixed schedule in a multi-machine environment and consider the problem of finding the minimal number of workers. The problem is formulated as a restricted vertex coloring problem and a branch and bound algorithm is presented.

A large part of work involving both job scheduling and employee timetabling aims at keeping the number of required employees at each time period under a threshold, without considering the regulation constraints of employee schedules nor the individual preferences and skills of employees. Daniels and Mazzola [10] consider a flow-shop problem in which the duration of an operation depends on the selected mode to process an operation. Each mode defines a number of resources (workers) needed during the processing of the operation. The scheduling horizon is discretized in periods and at each time period, the number of workers cannot exceed a fixed number. Optimal and heuristics approaches are proposed. Daniels et al. [9] propose the same approach in a parallel machine context. Bailey et al. [3] and Alfares and Bailey [1] propose an integrated model and a heuristic for project task and manpower scheduling where the objective is to find a trade-off between labor cost and daily overhead project cost. The labor cost depends

on the number of employed workers at each time period. The daily overhead cost depends on the project duration. There are no machine constraints and the labor restrictions consist in setting a maximal number of workers per period. In [21], the authors propose a MILP to minimize the makespan in a flow-shop with multi-processor workstation as a primary objective and to determine the optimal number of workers assigned to each machine as a secondary objective. The sequence of jobs is fixed on each machine and the makespan is minimized through lot-streaming.

Faaland and Schmitt [16] consider an assembly shop with multiple workstations. Each task must be performed on a given workstation by a worker. There are production and late-delivery costs on one hand and labor cost linked to the total number of employees on the other hand. The authors study the benefits of cross-training which allows employees to have requisite skills for several work-centers. A heuristic based on a priority rule and on the shifting bottleneck procedure is proposed.

A more general problem (w.r.t. the timetabling problem) is studied by Daniels et al. [11]. They extend the model proposed in [10] to an individual representation of employees in a flow-shop environment. Each employee has the requisite skills for only a subset of machines and can be assigned to a single machine at each time period. The duration of a job operation depends on the number of employees assigned to its machine during its processing. The employees assigned to an operation are required during all its processing time. No schedule regulations are considered except unavailability periods. A branch and bound method is developed and the benefits of the level of worker flexibility for makespan minimization is studied.

In [20], Haït et al. propose a general model for integrating production scheduling and employee timetabling, based on the concepts of load center, configuration, employee assignment and sequence. A so-called load center is a subset of machines that can be managed simultaneously by a single employee. A configuration is a set of load centers defining a partition of a subset of machines. At each scheduling time period a single configuration is active. Hence, the number of load centers in a configuration gives the number of active employees. An employee assignment is an assignment of each load center of a configuration to a different employee. The authors define the configuration graph each node correspond to a possible configuration and there is an arc between two configurations that can be consecutive in time with a weight giving the cost of the configuration changeover. This model allows one to represent the simultaneous work of an employee on several machines. However, the computation method of the job durations performed simultaneously by the same operator is not provided. An example with two machines provided by the authors shows that the computation of this duration of a job amounts to solving a scheduling problem of the elementary tasks performed by the operator. Furthermore, it can happen in practice that more than one operator is needed during the processing of a job on a machine, which is not covered by the proposed model. In this model, a schedule is defined by the start time of the jobs and by a path (with possible

loops and cycles) in the graph of configurations with the employee assignment for each configuration of this path. The authors provide two examples of integrated resolution in a flow-shop context. In the first example, they propose a dynamic programming algorithm to find a feasible path in the configuration graph with a fixed number of equivalent operators and a fixed sequence of jobs. In the second example they propose a heuristic and a lower bound of the makespan in a flow-shop where the timetabling problem is reduced to the assignment of an employee to each machine, the duration of the jobs depending of the employee performance.

Drezet and Billaut [14] consider a project scheduling problem with human resources and time-dependent activities requirements. Furthermore, employees have different skills and the main legal constraints dictated by the workforce legislation have to be respected. The model is quite general. However, only human resources are considered since the considered context is not a production scheduling problem where machines are critical resources. A tabu search method is proposed as well as proactive scheduling techniques to deal with the uncertainty of the problem.

This brief summary of the state of the art reveals that, compared to the transportation domain, the integration of production scheduling and employee timetabling is in its earliest phase. Almost no existing approach tackles the complex regulation constraints of work nor the diversity of employee activities in modern production systems. Recently, more sophisticated models have been proposed but independently of the relevant literature in staff scheduling in other areas and without proposing a general solution methodology.

2.3   Position of the Considered Problem

There are several variants of the employee timetabling problem, see for instance the recent surveys [15, 26]. In this paper we focus on only one of the problems presented in [12] called individual shift scheduling where each employee (or team of employees) is considered individually with its own skills and preferences. The time horizon is discretized in elementary time periods (shifts). At each period, a set of activities has to be performed and each activity requires a specific number of workers. The objective of the employee timetabling problem is to assign a single activity to each employee at each time period ('rest' activity included) in order to cover the demand for all activities. Such an assignment is called a schedule. There are restrictions on the possible schedules due to regulation constraints and employee profiles. The objective of the timetabling problem is to maximize the employee satisfaction.

There is also a large number of different production scheduling problems [25]. In this paper we consider a rather general problem where a set of jobs linked by precedence constraints has to be scheduled on a set of machines. Each job has a processing time, a release date, a due date and is assigned to a unique machine. A job cannot be interrupted once started and each machine can process at most one job simultaneously. The job scheduling problem lies in assigning a start time to each job with the objective to minimize the production costs.

We propose to integrate the two problems by associating to each job (processed on a machine) a set of activities (performed by the employees) such that assigning a start time to a job determines the period of each associated activity. From the employee timetabling point of view, the demand profile is not known in advance but is determined by the job schedule. From the job scheduling point of view, the possibility to start a job is subject to the presence of the employees able to perform the activities generated by this job. The employee profile is determined by the selected employee schedules. We will give several mathematical formulations of variants of this problem in Section 3.

3 ILP Models of Integrated Employee Timetabling and Machine Scheduling Problems

The model of integration proposed by [20] is centered on the concept of configuration which is a partition of the machines at a given time period such that each subset is managed by a single operator. In this paper we propose to perform the integration through the concept of *activity* which is widely used in the employee timetabling literature. We first provide a model with a common time representation for timetabling and scheduling (Section 3.1). Then we extend this model to the case where there is a time representation for employee timetabling and another time representation for job scheduling (Section 3.2). We show how these models can be extended to tackle the variability in job durations and machine assignment through the concept of modes (Section 3.3). The three latter models are based on time indexed and assignment variable formulations. In Section 3.4 we show how the set covering formulation usually used in efficient employee scheduling methods can also be used in the production scheduling context.

3.1 Common Time Representation for Timetabling and Scheduling and Single-Mode Jobs

We consider the following employee timetabling and machine scheduling problem.

Let T denote a time horizon, discretized in a set of elementary time periods $t = 0, \ldots, T-1$. We consider an organization comprising a set of E employees $\mathcal{E} = \{1, \ldots, E\}$ and a set of m machines $\mathcal{M} = \{1, \ldots, m\}$. There is set of A activities $\mathcal{A} = \{1, \ldots, A\}$ where each activity may be required by a job and has to be performed by one or several employees. $\mathcal{A}_e$ is the set of activities employee e is able to perform.

The organization has to process a set of n jobs $\mathcal{J} = \{1, \ldots, n\}$ during the time horizon. Each job j has a known duration $p_j > 0$ and requires for its execution a precise machine m_j. A binary matrix $(b_{jk})_{1 \leq j \leq n, 1 \leq k \leq m}$ states if job j requires machine k, i.e. $b_{jm_j} = 1$ and $b_{jk} = 0$, $\forall k \neq m_j$. A matrix $(R_{ja})_{1 \leq j \leq n, 1 \leq a \leq A}$ is given where R_{ja} is the number of employees that have to perform activity a during the processing of job j. Each job j has a release date r_j and a due date d_j.

There are precedence constraints linking the jobs, represented by a directed graph $G = (V, U)$ where V is the set of nodes including one node per job plus a dummy start node denoted 0 and a dummy end node denoted $n+1$. U is the set of arcs representing the precedence constraints. Each arc (i, j) of U is valuated by a (positive or negative) time lag d_{ij}.

There are also specific constraints on the activities that can be assigned to a given employee over time which will be described below. The objective of the considered employee timetabling and machine scheduling problem is to assign a start time to each activity and to assign exactly one activity to each employee at each time period.

We assume that there is a production cost W_{jt} if job j starts at time t and an employee satisfaction cost C_{eat} if employee e is assigned to activity a at time t.

x_{jt} is a binary decision variable where $x_{jt} = 1$ if job j starts at time t and $x_{jt} = 0$ otherwise. y_{eat} is a binary decision variable such that $y_{eat} = 1$ if employee e is assigned to activity a at time t and $y_{eat} = 0$ otherwise. The problem can be formulated as follows:

$$\min \sum_{j=1}^{n} \sum_{t=0}^{T-1} W_{jt} x_{jt} + \sum_{e=1}^{E} \sum_{a=1}^{A} \sum_{t=0}^{T-1} C_{eat} y_{eat} \tag{1}$$

$$\sum_{t=0}^{T-1} x_{jt} = 1 \quad \forall j \in \mathcal{J} \tag{2}$$

$$x_{jt} = 0 \quad \forall j \in \mathcal{J}, \forall t \notin \{r_j, \ldots, d_j - p_j\} \tag{3}$$

$$\sum_{j=1}^{n} \sum_{\tau=t-p_j+1}^{t} b_{jk} x_{j\tau} \leq 1 \quad \forall t \in \{0, \ldots, T-1\} =, \forall k \in \mathcal{M} \tag{4}$$

$$\sum_{t=0}^{T-1} t x_{jt} - \sum_{t=0}^{T-1} t x_{it} \geq d_{ij} \quad \forall (i, j) \in U \tag{5}$$

$$\sum_{e=1}^{E} y_{eat} \geq \sum_{j=1}^{n} \sum_{\tau=t-p_j+1}^{t} R_{ja} x_{j\tau} \quad \forall a \in \mathcal{A}, \forall t \in \{0, \ldots, T-1\} \tag{6}$$

$$\sum_{a \in \mathcal{A}_e} y_{eat} = 1 \quad \forall e \in \mathcal{E}, \forall t \in \{0, \ldots, T-1\} \tag{7}$$

$$Fy \leq f \tag{8}$$

$$x_{jt} \in \{0, 1\} \quad \forall j \in \mathcal{J}, \forall t \in \{0, \ldots, T-1\} \tag{9}$$

$$y_{eat} \in \{0, 1\} \quad \forall e \in \mathcal{E}, \forall a \in \mathcal{A}, \forall t \in \{0, \ldots, T-1\}. \tag{10}$$

The objective of the problem is to minimize the total cost (1) subject to the following constraints. Each job has to be started exactly once: (2). Each job must be started a way that it is started and finished within its time window: (3). At most one job can be processed by a machine at each time period: (4). The precedence constraint must be satisfied: (5). The number of employees assigned to each activity at each time period has to cover the total demand of
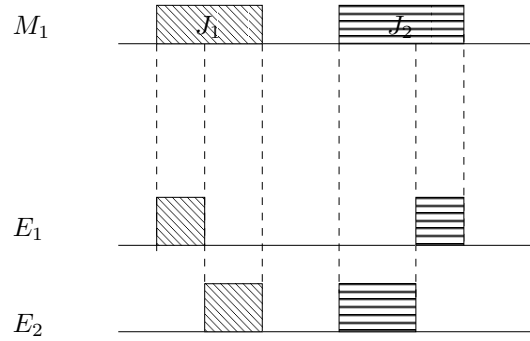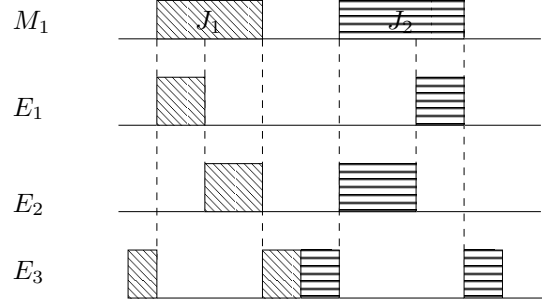
Fig. 1. A 1-machine and 2-employee example.

all jobs in process: (6). Each employee has to be assigned to exactly one activity (in set $\mathcal{A}_e$) at each time period: (7). We assume $\mathcal{A}$ contains also non working activities representing employee inactivity (break, lunch, etc.) gathered in set $\mathcal{P}$. Constraints (8) are specific constraints for each employee e of the form $\sum_{a \in \mathcal{A}} \sum_{t=0}^{T-1} F_{atq} y_{eat} \leq f_q$, with $F_{atq} \in \{-1, 0, 1\}$, which allow for instance the taking into account of minimum or maximum consecutive periods of work, and other complex regulation constraints. For instance, if no employee can work more than two consecutive shifts, the constraints of the form $\sum_{a \in \mathcal{A} \setminus \mathcal{P}} (y_{ea(t-1)} + y_{eat} + y_{ea(t+1)}) \leq 2$ can be defined for each time period $t \in [1, T-2]$ for each employee e. The main drawback of this formulation is that the number of these constraints can be huge in practical situations and in general a set covering formulation is preferred (see Section 3.4).

The main difference between the machines and the employee resource is that employee timetables are more flexible, as illustrated in the example displayed in Figure 1. In this example, the two jobs generate a single activity during their processing. If we suppose that the first employee E_1 allocated to this activity has to take a break while J_1 is in process, another employee can perform the activity until the break of E_1 is over which occurs in this example while J_2 is in process.

3.2 Different Time Representations for Timetabling and Scheduling and Single-Mode Jobs

We assume that, for practical reasons, there may be a different time representation for the machine scheduling problem and for the employee timetabling problem. Let T denote the time horizon for the scheduling problem and let Θ denote the time horizon for the timetabling problem. Furthermore, we assume that if a job j starts at time t, $0 \leq t < T$ then a number of employees $R_{jat\theta} \geq 0$ is required to perform activity a at each period θ, $0 \leq \theta < \Theta$.

Fig. 2. A 1-machine and 3-employee example.

It follows that demand covering constraints (6) can be generalized with constraints (16) below and the new model is

$$\min \sum_{j=1}^{n} \sum_{t=0}^{T-1} W_{jt} x_{jt} + \sum_{e=1}^{E} \sum_{a=1}^{A} \sum_{\theta=0}^{\Theta-1} C_{ea\theta} y_{ea\theta} \tag{11}$$

$$\sum_{t=0}^{T-1} x_{jt} = 1 \ \ \forall j \in \mathcal{J} \tag{12}$$

$$x_{jt} = 0 \ \ \forall j \in \mathcal{J}, \forall t \notin \{r_j, \ldots, d_j - p_j\} \tag{13}$$

$$\sum_{j=1}^{n} \sum_{\tau=t-p_j+1}^{t} b_{jk} x_{j\tau} \leq 1 \ \ \forall t \in \{0, \ldots, T-1\} =, \forall k \in \mathcal{M} \tag{14}$$

$$\sum_{t=0}^{T-1} t x_{jt} - \sum_{t=0}^{T-1} t x_{it} \geq d_{ij} \ \ \forall (i,j) \in U \tag{15}$$

$$\sum_{e=1}^{E} y_{ea\theta} \geq \sum_{j=1}^{n} \sum_{t=0}^{T-1} R_{jat\theta} x_{jt} \ \ \forall a \in \mathcal{A}, \forall \theta \in \{0, \ldots \Theta-1\} \tag{16}$$

$$\sum_{a \in \mathcal{A}_e} y_{ea\theta} = 1 \ \ \forall e \in \mathcal{E}, \forall \theta \in \{0, \ldots, \Theta-1\} \tag{17}$$

$$Fy \leq f \tag{18}$$

$$x_{jt} \in \{0,1\} \ \ \forall j \in \mathcal{J}, \forall t \in \{0, \ldots, T-1\} \tag{19}$$

$$y_{ea\theta} \in \{0,1\} \ \ \forall e \in \mathcal{E}, \forall a \in \mathcal{A}, \forall \theta \in \{0, \ldots, \Theta-1\}. \tag{20}$$

Such constraints allow one to consider the cases where the employees need not be present during all the processing of a job on its machine, or when the employee activity generated by the job is not simultaneous with the processing of the jobs. This feature takes place when employees have to perform setup or removal activities before and after the job processing, or when a control operation
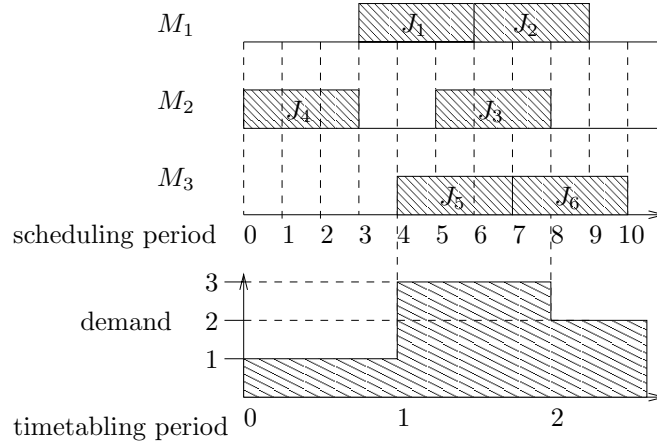
Fig. 3. A 1-machine and 2-employee example.

has to be carried out during a limited time while the job is in process. In Figure 2, a third employee is necessary only right before the start and right after the end of jobs J_1 and J_2.

This type of model allows one also to take account of a different time scale between the time horizon of the scheduling problem, with the time periods considered in the timetabling problem. Suppose the scheduling time period is one hour and the timetabling period is four hours, then an aggregated information of the activities to perform during each four-hour period has to be provided. For this purpose, values $R_{jat\theta}$ need not be integers if the activity a generated by job j in timetabling period θ occupies only a portion of an employee's work capacity. In Figure 3, each job is assumed to require 0.25 employees per time unit and generate a single activity. Then, the demand for employees able to perform this activity is displayed for each timetable period.

please check caption fig 3

3.3   Multi-mode Jobs

We consider the case where for each job j there is a number Q_j of different processing modes corresponding to different ways (durations, machine and activity requirements) to perform job j. Let p_j^q denote the duration of job j in mode q. Let $b_{jk}^q = 1$ if job j uses machine k in mode q and $b_{jk}^q = 0$ otherwise. x_{jt}^q is a binary decision variable such that $x_{jt}^q = 1$ if job j is started at time t in mode q and $x_{jt}^q = 0$ otherwise. $R_{jat\theta}^q$ now denotes the number of employees that must perform activity a at period θ if job j is started at time t in mode q. The model can be adapted as follows:

$$\min \sum_{j=1}^{n}\sum_{q=1}^{Q_j}\sum_{t=0}^{T-1} W_{jt}x_{jt}^q + \sum_{e=1}^{E}\sum_{a=1}^{A}\sum_{\theta=0}^{\Theta-1} C_{ea\theta}y_{ea\theta} \tag{21}$$

$$\sum_{q=1}^{Q_j}\sum_{t=0}^{T-1} x_{jt}^q = 1 \ \ \forall j \in \mathcal{J} \tag{22}$$

$$x_{jt}^q = 0 \ \ \forall j \in \mathcal{J}, \forall q \in \{1,\ldots,Q_j\}$$
$$\forall t \notin \{r_j,\ldots,d_j - p_j\} \tag{23}$$

$$\sum_{j=1}^{n}\sum_{q=1}^{Q_j}\sum_{\tau=t-p_j^q+1}^{t} b_{jk}^q x_{j\tau}^q \le 1 \ \ \forall t \in \{0,\ldots,T-1\} =, \forall k \in \mathcal{M} \tag{24}$$

$$\sum_{q=1}^{Q_j}\sum_{t=0}^{T-1} t x_{jt}^q - \sum_{s=1}^{Q_i}\sum_{t=0}^{T-1} t x_{it}^s \ge d_{ij} \ \ \forall (i,j) \in U \tag{25}$$

$$\sum_{e=1}^{E} y_{ea\theta} \ge \sum_{j=1}^{n}\sum_{q=1}^{Q_j}\sum_{t=0}^{T-1} R_{jat\theta} x_{jt}^q \ \ \forall a \in \mathcal{A}, \forall \theta \in \{0,\ldots\Theta - 1\} \tag{26}$$

$$\sum_{a \in \mathcal{A}_e} y_{ea\theta} = 1 \ \ \forall e \in \mathcal{E}, \forall \theta \in \{0,\ldots,\Theta - 1\} \tag{27}$$

$$Fy \le f \tag{28}$$

$$x_{jt}^q \in \{0,1\} \ \ \forall j \in \mathcal{J}, \forall q \in \{1,\ldots,Q_j\},$$
$$\forall t \in \{0,\ldots,T-1\} \tag{29}$$
$$y_{ea\theta} \in \{0,1\} \ \ \forall e \in \mathcal{E}, \forall a \in \mathcal{A}, \forall \theta \in \{0,\ldots,\Theta - 1\}. \tag{30}$$

3.4   Set Covering Formulations

Let $\mathcal{S}_e$ denote the set of valid schedules for an employee e. For each schedule $s \in \mathcal{S}_e$, each activity a and each timetabling period θ, binary value $y_{sa\theta}$ is such that $y_{sa\theta} = 1$ if the schedule performs activity a at time θ and $y_{sa\theta} = 0$ otherwise. C_s denotes the cost of a schedule $s \in \mathcal{S}_e$. In addition, a binary decision variable z_s is defined such that $z_s = 1$ if schedule s is selected and $z_s = 0$ otherwise.

A new model can then be proposed by including the set covering formulation of the timetabling constraints (we ignore the multi-mode characteristics):

$$\min \sum_{j=1}^{n}\sum_{t=0}^{T-1} W_{jt}x_{jt} + \sum_{e=1}^{E}\sum_{s \in \mathcal{S}_e} C_s z_s \tag{31}$$

$$\sum_{t=0}^{T-1} x_{jt} = 1 \ \ \forall j \in \mathcal{J} \tag{32}$$

$$x_{jt} = 0 \ \ \forall j \in \mathcal{J},$$
$$\forall t \notin \{r_j, \dots, d_j - p_j\} \tag{33}$$

$$\sum_{j=1}^{n} \sum_{\tau=t-p_j+1}^{t} b_{jk} x_{j\tau} \leq 1 \ \ \forall t \in \{0, \dots, T-1\} =, \forall k \in \mathcal{M} \tag{34}$$

$$\sum_{t=0}^{T-1} t x_{jt} - \sum_{t=0}^{T-1} t x_{it} \geq d_{ij} \ \ \forall (i,j) \in U \tag{35}$$

$$\sum_{e=1}^{E} \sum_{s \in \mathcal{S}_e} y_{sa\theta} z_s \geq \sum_{j=1}^{n} \sum_{t=0}^{T-1} R_{jat\theta} x_{jt} \ \ \forall a \in \mathcal{A}, \forall \theta \in \{0, \dots \Theta - 1\} \tag{36}$$

$$\sum_{s \in \mathcal{S}_e} z_s = 1 \ \ \forall e \in \mathcal{E} \tag{37}$$

$$x_{jt} \in \{0,1\} \ \ \forall j \in \mathcal{J}, \forall t \in \{0, \dots, T-1\} \tag{38}$$

$$z_s \in \{0,1\} \ \ \forall e \in \mathcal{E}, \forall s \in \mathcal{S}_e. \tag{39}$$

4   A Constraint Programming Model

Constraint programming formulations have been proposed for production scheduling [4] and for employee timetabling [12]. We present hereafter an integrated formulation which involves start time decision variables $S_j \in [r_i, d_i - p_i]$ for all jobs, an activity assignment variable $a_{\theta e} \in \mathcal{A}_e$ giving the activity assigned to employee e in period θ and a demand variable $\delta_{\theta a} \in \mathbb{N}$ giving the number of employees required for activity a during period θ. Consider the following constraint satisfaction problem (CSP):

$$S_j - S_i \geq d_{ij} \ \ \forall (i,j) \in U \tag{40}$$

$$S_j + p_j \leq S_i \vee S_i + p_i \leq S_j \ \ \forall i,j \in \mathcal{J}, m_i = m_j \tag{41}$$

$$\phi(\delta_{\theta a}, S) \ \ \forall \theta \in \{0, \dots, \Theta - 1\}, \forall a \in \mathcal{A} \tag{42}$$

$$\texttt{distribute}((\delta_{\theta a})_{a \in \mathcal{A}}, \mathcal{A}, (a_{\theta e})_{e \in \mathcal{E}}) \ \ \forall \theta \in \{0, \dots, \Theta - 1\} \tag{43}$$

$$\texttt{regular}((a_{\theta e})_{\theta \in \{0, \dots, \Theta - 1\}}, \Pi) \ \ \forall e \in \mathcal{E}. \tag{44}$$

Constraints (40) are the precedence constraints. Constraints (41) are the machine disjunctive constraints. Constraints (42) establish the link between the job start time variables S and the demand variable δ through generic constraint ϕ that needs to be specified for each specific problem. Constraints (43) represent demand satisfaction through the global cardinality constraint `distribute` which states that for a given period θ, $\delta_{\theta a}$ variables must have value a in the activity assignment vector $(a_{\theta e})_{e \in \mathcal{E}}$ of employees during period θ. Last, constraints (44) express the employee specific and regulation constraints through the global regular language membership constraints `regular` [24], restricting the sequence

of values taken by the assignment variables to belong to the regular language associated to Π.

The advantage of constraint programming is its high flexibility to model complex demand computations, as well as complex regulation constraints.

The above CSP can be transformed into an optimization problem by introducing cost variables. This can be done through the `element` global constraints (see next Section). As an alternative, in [12], a new global constraint $\mathtt{cost} - \mathtt{regular}(\mathtt{X}, \Pi, \mathtt{z}, \mathtt{C})$ extends the `regular` constraint by computing the cost z associated by an assignment of variables X given cost matrix C.

5 Solving a Lexicographic Makespan and Employee Cost Optimization Problem by a Hybrid LP-CP Method

In this section, we propose a hybrid CP-LP exact method to solve a lexicographic bicriteria optimization problem. The considered production cost is the makespan, denoted $C_{\max}$. Let C_{empl} denote the total satisfaction cost of employees. The considered problem can be denoted

$$\min Lex(C_{\max}, C_{\mathrm{empl}}) \tag{45}$$

$$C_{\max} \geq S_j + p_j \ \ \forall j \in \mathcal{J} \tag{46}$$

$$C_{\mathrm{empl}} = \sum_{e \in \mathcal{E}} \sum_{\theta=0}^{\Theta-1} C_{e\theta} \tag{47}$$

$$\mathtt{element}(C_{e\theta}, (C_{ea\theta})_{a \in \mathcal{A}_e}, a_{\theta e}) \ \ \forall e \in \mathcal{E}, \forall \theta \in \{0, \ldots \Theta - 1\} \tag{48}$$

$$(40)\ldots(44).$$

Constraints (46) enforce the makespan value. Constraint (47) defines the total cost C_{empl} as the sum of elementary employee/period costs represented by decision variables $C_{e\theta}$. `element` global constraints (48) simply enforce the implications $a_{\theta e} = v \implies C_{e\theta} = C_{ev\theta}$ for all $\theta \in \{0, \ldots \Theta-1\}$, $e \in \mathcal{E}$ and $v \in \mathcal{A}_e$. The problem can be solved by first finding the optimal makespan $C_{\max}^*$ (problem A) and, second, by finding the minimal employee cost C_{empl}^* compatible with $C_{\max}^*$ (problem B).

We propose to solve both problems A and B through implicit enumeration in a constraint programming framework. Hence $C_{\max}^*$ is found by iteratively searching the smallest V such that there is a feasible solution verifying $C_{\max} \leq V$ (problem A). C_{empl}^* is found by searching the smallest V' such that there is a feasible solution verifying $C_{\max} = C_{\max}^*$ and $C_{\mathrm{empl}} \leq V'$ (problem B).

At each node of each above-defined search trees, constraint propagation algorithms are performed to either reduce the domain of start time S and activity variables a or to detect an inconsistency and prune the node. The branching scheme first assigns values to start time variables and, once all start time variables are assigned, makes the remaining decisions on activity variables. Note

that constraints ϕ (42) have to ensure that once a complete assignment of the start time variables is computed, the demand variables δ are also completely assigned.

For both problems A and B, the makespan constraints set due dates on the job operations. Hence, standard scheduling constraint propagation algorithms can be used to reduce the start time domains. In the present work, we use precedence constraint propagation and edge-finding. We refer to [5] for a precise description of those algorithms.

For domain reduction of the demand and activity variables δ and a, besides the standard `distribute` and `regular` constraint propagation algorithm, we propose to embed the linear programming relaxation of the ILP formulation (21)...(30), limited to constraints involving $y_{ea\theta}$ assignment variables, into a global constraint. Let $\underline{\delta}_{\theta a}$ denote the smallest value in the domain of demand variable $\delta_{\theta a}$ for activity a during period θ at a given node of the constraint programming search tree. Then we consider the following LP relaxation, considering only labor costs:

$$\min \sum_{e=1}^{E} \sum_{a=1}^{A} \sum_{\theta=0}^{\Theta-1} C_{ea\theta} y_{ea\theta} \tag{49}$$

$$\sum_{e=1}^{E} y_{ea\theta} \geq \underline{\delta}_{\theta a} \quad \forall a \in \mathcal{A}, \forall \theta \in \{0, \ldots \Theta-1\} \tag{50}$$

$$\sum_{a \in \mathcal{A}_e} y_{ea\theta} = 1 \quad \forall e \in \mathcal{E}, \forall \theta \in \{0, \ldots, \Theta-1\} \tag{51}$$

$$Fy \leq f \tag{52}$$

$$0 \leq y_{ea\theta} \leq 1 \quad \forall e \in \mathcal{E}, \forall a \in \mathcal{A}, \forall \theta \in \{0, \ldots, \Theta-1\}. \tag{53}$$

At a given node, the relaxation is stronger if the lower bound $\underline{\delta}_{\theta a}$ on the demand is tight. This obviously depends on the definition and propagation of constraint ϕ. Each time the LP relaxation is unfeasible, which can occur due to both demand undercoverage or labor cost upper bound violation, the current node is pruned.

Last, whenever an upper bound Z on the total labor cost C_{empl} is known, the reduced cost based filtering technique can be applied. Let $\tilde{C}_{ea\theta}$ denote the reduced cost of an activity assignment variable $y_{ea\theta}$ and let $\underline{C}_{\mathrm{empl}}$ denote the current optimal LP solution value. If, $\underline{C}_{\mathrm{empl}} + \tilde{C}_{ea\theta} \geq Z$, a can be removed from the domain of $a_{\theta e}$.

6   Computational Results on a Basic Employee Timetabling and Job-Shop Scheduling Problem

In this section, we show the potential of hybrid methods to solve integrated employee timetabling and production scheduling problems, through the resolution of basic employee and job-shop scheduling instances. For constraint based

scheduling we use ILOG Solver 6.1 and Scheduler 6.1. For LP resolution we use ILOG Cplex 9.1. All programs are coded in C++ under Linux on a AMD x86-64 architecture.

We consider the standard job-shop scheduling problem in which a job is made of m operations which form a chain in the precedence graph. Each job has to be processed by all the machines successively. Hence the operations of the same jobs are all assigned to different machines.

We consider job-shop instances of six jobs and four machines, comprising 24 operations. We consider a set of 15 employees and a set of $4 + 1$ activities. The integer job operations processing times have been generated uniformly randomly between 1 and 10. We assume one time unit corresponds to one hour. We define a timetabling period as a eight-hour shift (i.e. $T = 8\Theta$). Each employee has to be assigned to one activity during each shift. We assume activity 5 corresponds to employee inactivity during the shift. Each employee has skills for two production activities out of four. Each break must be of at least two consecutive shifts (16-hour break). There is an integer cost (uniformly randomly generated from 1 to 5) for assigning a production activity to an employee during each shift. Furthermore, to ensure problem feasibility at minimal makespan, we consider an additional set of 10 extra employees having a greater assignment cost (equal to 9 for all extra employees and for all periods and all activities).

We now describe how constraint ϕ is implemented for the considered example. We simply assume there is a mapping between activities and machines. Hence, whenever a machine is in process during a shift, then an employee able to perform the corresponding activity is needed. It follows that at most four employees can be required simultaneously during a shift.

More precisely the link between the operation schedule S and the demand $(\delta_{\theta a})$ can be described by the following constraints. Let $D = T/\Theta$ and let $\mathcal{J}_k$ denote the set of operations scheduled on machine k. Let a_k denote the activity corresponding to machine k:

$$S_j + p_j > D\theta \land S_j < D(\theta + 1) \implies \delta_{\theta a_{m_j}} = 1$$
$$\forall j \in \mathcal{J}, \forall \theta \in [0, \Theta - 1]$$
$$(S_j + p_j \leq D\theta \lor S_j \geq D(\theta + 1), \forall j \in \mathcal{J}_k) \implies \delta_{\theta a_k} = 0$$
$$\forall k \in \mathcal{M}, \forall \theta \in [0, \Theta - 1].$$

We use the standard job-shop resolution method provided in the example library of ILOG scheduler for the scheduling constraint propagation parts. For the search part on the start time and activity variables, we use a simple backtracking on possible values (in a chronological way for the start times). All employee constraints have been coded by `distribute` constraints. The LP relaxation and the reduced cost-based filtering algorithms are embedded into a global constraint. These algorithms are called whenever the lower bound of an activity demand is increased for any period or when the domain of a variable $(a_{\theta e})$ is changed.

We have generated 10 instances having the above described characteristics. The results, comparing the hybrid method with and without reduced cost-based

Table 1. Method comparison on 10 basic employee and job-shop scheduling instances

Inst	Mks*	cost(M)	CPU(M)	#fails(M)	cost*	CPU(H)	#fails(H)	CPU(H⁻)	#fails(H⁻)
1	45	75	0.2s	3	29	0.8s	151	1.1s	438
2	56	69	0.2s	2	26	208s	27176	4099s	2459422
3	44	69	0.2s	2	26	2.2s	732	1.7s	1691
4	40	53	0.2s	3	23	0.5s	24	0.7s	183
5	40	63	0.2s	3	27	6.2s	4047	205s	117850
6	48	70	0.2s	7	28	0.9s	96	1.2s	371
7	43	67	0.2s	2	33	0.6s	83	0.8s	242
8	37	57	0.2s	3	22	28s	8185	400s	269799
9	49	69	0.2s	4	24(22)	3364s	340742	-	-
10	48	68	0.2s	3	23	4.1s	1140	408s	267695

filtering, are displayed in Table 1. Column Inst gives the instance number. Column Mks* gives the optimal makespan obtained by pure CP without considering employee cost minimization. Column cost(M) gives the employee cost of the obtained solution. Columns #fails(M) and CPU(M) give the total number of fails and the CPU times of this search process. Column cost* gives the minimal employee cost solution with a makespan equal to Mks*. Columns #fails(H) and CPU(H) give the total number of fails and the CPU times of the complete hybrid search method needed to find the optimal cost solution. Columns #fails(H⁻) and CPU(H⁻) give the same values for the hybrid method used without reduced cost-based filtering.

For the proposed instances, the makespan minimization problem is very easy since CP always solves the problem in less than 0.2s. Note that, in contrast, the hybrid methods outperform the standard constraint programming approaches for employee cost minimization since the latter is unable to find the optimal solution in a reasonable amount of time. Furthermore, while keeping the makespan optimal, the employee cost is significantly improved by the hybrid methods for all instances. One instance remains unsolved by all methods and the obtained lower and upper bounds are given as well as the total CPU time and number of fails needed to obtain them. This underlines the difficulty of the problem and shows the need for improvement of the proposed methods, considering also that the considered instances are small ones. The reduced cost-based filtering hybrid method outperforms the basic hybrid method on almost all instances showing the potential of high interaction between CP and LP for this kind of difficult integrated planning problem. In [2], enhanced hybrid methods and extended computational experiments are presented on the considered employee timetabling and job-shop scheduling problem.

7   Concluding Remarks

We have proposed a flexible model and several ILP and CP formulations for integrated employee timetabling and production scheduling. We have shown how the flexibility of constraint programming modeling can be used to represent complex relationships between schedules and activity demands. A hybrid exact method involving standard constraint programming-based scheduling and timetabling technique on the one hand, and a linear programming relaxation with reduced-cost based filtering on the other hand, has been used to solve to optimality instances of the problem which cannot be solved by standard constraint programming. We are planning to generate several other instances to study the behavior of the proposed method with different problem characteristics. The search algorithm has also to be refined since we have used only standard backtracking schemes without any particular rule for activity selection. More realistic employee timetabling constraints will have also to be considered. This may lead to an improvement of the results of pure constraint programming techniques. The search could also be guided by using the linear programming optimal solution. Decomposition methods such as Benders decomposition or column generation will have also to be tested.

References

1. Alfares, H., Bailey, J.: Integrated project task and manpower scheduling. IIE Transactions, 29 (1997) 711–717
2. Artigues, C., Gendreau, M., Rousseau, L., Vergnaud, A.: Solving an integrated employee timetabling and job-shop scheduling problem via hybrid branch-and-bound. Technical Report 06700, Université de Toulouse, LAAS-CNRS, Toulouse, France (2006)
3. Bailey, J., Alfares, H., Lin, W.: Optimization and heuristic models to integrate project task and manpower scheduling. Computers and Industrial Engineerings, 29 (1995) 473–476
4. Baptiste, P., Pape, C. L.: Disjunctive constraints for manufacturing scheduling: Principles and extensions. International Journal of Computer Integrated Manufacturing, 9 (1996) 306–310
5. Baptiste, P., Pape, C. L., Nuijten, W.: Constraint-Based Scheduling, Kluwer, Dordrecht (2001)
6. Bodin, L., Golden, B., Assad, A., Ball, M.: Routing and scheduling of vehicles and crews the state of the art. Computers and Operations Research, 10 (1983) 63–211
7. Chen, Z.: Simulataneous job scheduling and resource allocation on parallel machines. Annals of Operations Research, 129 (2004) 135–153
8. Cordeau, J. F., Stojković, G., Soumis, F., Desrosiers, J.: Benders decomposition for simultaneous aircraft routing and crew scheduling. Transportation Science, 35 (2001) 375–388
9. Daniels, R. L., Hoopes, B. J., Mazzolla, J. B.: Scheduling parallel manufacturing cells with resource flexibility. Management Science, 42 (1996) 1260–1276
10. Daniels, R. L., Mazzolla, J. B.: Flow shop scheduling with resource flexibility. Operations Research, 42 (1994) 504–522

11. Daniels, R. L., Mazzolla, J. B., Shi, D.: Flow shop scheduling with partial resource flexibility. Management Science, 50 (2004) 658–669
12. Demassey, S., Pesant, G., Rousseau, L. M.: Constraint programming based column generation for employee timetabling. In: 7th International Conference on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems (CPAIOR'05, Prague, 2005)
13. Desaulniers, G., Desrosiers, J., Ioachim, I., Solomon, M., Soumis, F., Villeneuve, D.: A unified framework for deterministic time constrained vehicle routing and crew scheduling problems. In: Crainic, T., Laporte, G. (eds.): Fleet management and logistics, Kluwer, Dordrecht (1998) 57–93
14. Drezet, L. E., Billaut, J. C.: Tabu search algorithms for a predictive and a reactive project scheduling problem. In: 6th Metaheuristics International Conference (Vienna, 2005)
15. Ernst, A., Jiang, H., Krishnamoorthy, M., Sier, D.: Staff scheduling and rostering: A review of applications, methods and models. European Journal of Operational Research, 153 (2004) 3–27
16. Faaland, B., Schmitt, T.: Cost-based scheduling of workers and equipment in a fabrication and assembly shop. Operations Research, 41 (1993) 253–268
17. Freling, R., Huisman, D., Wagelmanss, A.: Models and algorithms for integration of vehicle and crew scheduling. Journal of Scheduling, 6(1) (2003) 63–85
18. Haase, K., Desaulniers, G., Desrosiers, J.: Simultaneaous vehicle and crew scheduling in urban mass transit systems. Transportation Science, 35 (2001) 286–303
19. Haase, K., Friberg, C.: An exact algorithm for the vehicle and crew scheduling problem. In Wilson, N., ed.: Computer-Aided Transit Scheduling, Lecture Notes in Economics and Mathematical Systems, Vol. 471, Springer, Berlin (1999) 63–80
20. Haït, A., Baptiste, P., Brauner, N., Finke, G.: Approches intégrées à court terme. Chapter 6 in: Baptiste, P., Giard, V., Haït, A. (eds.): Gestion de production et resources humaines, Presse Internationales Polytechnique (2005)
21. Huq, F., Cutright, K., Martin, C.: Employee scheduling and makespan minimization in a flow shop with multi-processor work stations: a case study. Omega, 32 (2004) 121–129
22. Klabjan, D., Johnson, E., Nemhauser, G.: Airline crew scheduling with time windows and plane count constraints. Transportation Science, 36 (2002) 337–348
23. Mercier, A., Cordeau, J. F., Soumis, F.: A computational study of benders decomposition for the integrated aircraft routing and crew scheduling problem. Computers and Operations Research, 32 (2005) 1451–1476
24. Pesant, G.: A regular language membership constraint for finite sequences of variables. 10th International Conference on Principles and Practice of Constraint Programming (CP'04), Lecture Notes in Computer Science, Vol. 3258, Springer, Berlin (2004) 482–495
25. Pinedo, M. L.: Planning and Scheduling in Manufacturing and Services, Springer Series in Operations Research and Financial Engineering, Springer, Berlin (2005)
26. Soumis, F., Pesant, G., Rousseau, L. M.: Gestion des horaires et affectation du personnel. Chapter 4 in: Baptiste, P., Giard, V., Haït, A. (eds.): Gestion de production et resources humaines, Presse Internationales Polytechnique (2005)
27. Valls, V., Pérez, A., Quintanilla, S.: A graph colouring model for assigning a heterogeneous workforce to a given schedule. European Journal of Operational Research, 90 (1996) 285–302
28. Voß, S., Daduna, J. (eds.): Computer Aided Scheduling of Public Transport, Lecture Notes in Economics and Mathematical Systems, Vol. 505, Springer, Berlin (2001)