



HAL
open science

New scheduling problems with perishable raw materials constraints

Jean-Charles Billaut

► **To cite this version:**

Jean-Charles Billaut. New scheduling problems with perishable raw materials constraints. 16th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2011), Sep 2011, Toulouse, France. 10.1109/ETFA.2011.6059024 . hal-01233559

HAL Id: hal-01233559

<https://hal.science/hal-01233559>

Submitted on 7 Feb 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

New scheduling problems with perishable raw materials constraints

Jean-Charles BILLAUT^{1,2}

¹ Université François Rabelais Tours

Laboratoire d'Informatique

64 avenue Jean Portalis, 37200 Tours, France.

jean-charles.billaut@univ-tours.fr

² CNRS, LAAS, 7 avenue du colonel Roche, 31077 Toulouse, France

Université de Toulouse; UPS, INSA, INP, ISAE; LAAS; 31077 Toulouse, France

Abstract

In production manufacturing systems, raw materials that are required for the production process are supposed to be present in sufficient quantities. The ordering of these products is given by an inventory management policy. We consider in this paper raw materials, that are present in sufficient quantities and that can be stored for a long time in containers under specific conditions (temperature, pressure, ...). The peculiarity is that the quality of these raw materials decreases as soon as they are open for being used. At this moment, the product has to be used before a given deadline, or it is lost. Two types of objective functions are associated to the use of such raw materials: the total cost of lost products (economic objective) and the total quantity of lost products (environmental objective).

We present in this paper this new scheduling framework, which brings new constraints and new objective functions. We show that the single machine problem with equal-length jobs and only one product is already NP-hard and we propose mathematical formulations for two single-machine problems and an MIP model, which is briefly evaluated.

1. Introduction

In production manufacturing systems, raw materials that are required for the production process are supposed to be present in sufficient quantities. Generally, the raw materials are split into different categories according to ABC inventory classification techniques (see [10] for a recent study, where a new criterion for ranking items is proposed), and from this classification, the supply is managed according to a given inventory management policy. This problem falls into the field of inventory management, in which a lot of constraints are considered like for example quantity discount, permissible delays in payments or perishable products. The last case generally concerns items that are produced and that cannot be stored infinitely without deterioration or devaluation. These specific con-

straints tend to modify the expression of the cost functions and an important literature deals with these problems (see [3] for a survey on deteriorating inventory).

However, the constraint of the availability of raw materials for scheduling activities has not received a great attention in the scheduling literature. Ouenniche and Bector [9] integrate the holding cost of work-in-process inventory into the constraints and the objective function, in a context of flowshop scheduling (see [7, 6] for an extension of this model). In [2], Cai, Chen, Xiao and Xu consider that a seafood manufacturer receives raw materials, which can be processed for giving different types of products. But for freshness reasons, the raw materials have to be used before a deadline. The manufacturer has to decide which type of product will be produced, in which quantity, and the production sequence. The authors consider the double problem of selection and scheduling (on a single machine), under the constraints of limit on raw materials and random processing times. They show the relations between their problem and the stochastic knapsack and propose optimal policies for the different problems considered. In [5], Grigoriev, Holthuijsen and van de Klundert integrate the raw material constraints into the scheduling problems. Three models of raw materials usage are identified: each customer order requires its own raw material type and it is supposed that the customer supplies the raw materials in sufficient quantity; all customer orders require a single common raw material; and finally many raw materials can be required by each of the customer orders. The authors consider a single machine environment, jobs with unit or equal processing times, known arrival times of raw materials. The objective is to find a schedule that meets the due dates.

The problem that we consider in this paper is different. We have to schedule a set of jobs, each job being characterized by a processing time, a due date, and a required amount of one or several raw materials. We consider that raw materials (called products in the following) are present in sufficient quantities. We assume that under certain storage conditions, they can be stored in con-

tainers for a long time period (several weeks). However, once a container has been opened, the use of the product is limited in time (at most a few days, at least around one hour), because the contact of ambient air or ambient temperature starts its degradation. Containers can be boxes, cans, or vials of different volumes. This study is motivated by the production of anti-cancer chemotherapy drugs [8], where doses to prepare require the use of some quantities of actives ingredients [1]. These products are stored into a fridge, and after being opened, have to be used before a given deadline, which can vary from one hour to several days. After the deadline, the product is lost. If necessary, a new vial of the same product may be opened for processing another job. Of course, this problem is applicable to other production contexts like food or chemical production.

The major difference between this model and the models of the literature is that the limit of product consumption depends on the schedule. The schedule imposes the decisions to open a new container (in the following, a vial is supposed to be an elementary unit of raw material). Two objectives are considered in this study: a “*job-oriented*” objective, which is the minimization of a tardiness measure of the product delivery, and a “*product-oriented*” objective, which is the minimization of the cost (economic point of view) or of the quantity (environmental aspect) of lost product. The approach which is used consists in minimizing a product-oriented measure, subject to a bounded job-related measure.

We show in this paper that the problem of minimizing the quantity of lost product is already strongly NP-hard with a unique product, a single machine and identical jobs. Several new scheduling problems are identified. An MIP model is proposed in the case of a single machine.

The rest of the paper is organized as follows. In Section 2, the general framework is explained and the notations are introduced. The complexity of the single machine and single product case is presented in Section 3 and a mathematical formulation is presented. The MIP model is tested and results are presented in Section 4.

2 New scheduling problems, notations

2.1 Notations

We consider a set \mathcal{J} of n jobs to schedule on a set \mathcal{M} of m machines. A set \mathcal{R} of r products are available for use, in sufficient quantity. Each product ℓ is stored into vials of capacity V_ℓ and can be used only during the T_ℓ next time units after being opened. Each job J_j is characterized by a processing time p_j , a due date d_j and the need of $b_{j,\ell}$ units of product ℓ . It is assumed w.l.g. that $b_{j,\ell} < V_\ell, \forall j, \forall \ell$ (index ℓ is omitted in case of one product).

We denote by C_j the completion time of J_j . We use the classical notations for the “*job-oriented*” criteria:

$$\text{the makespan: } C_{\max} = \max_{1 \leq j \leq n} C_j$$

$$\text{the maximum lateness: } L_{\max} = \max_{1 \leq j \leq n} (C_j - d_j)$$

$$\text{the maximum tardiness: } T_{\max} = \max_{1 \leq j \leq n} \max(0, C_j - d_j)$$

We denote by Q_ℓ the quantity of product ℓ that has been lost and by Z_ℓ its cost. Notice that the cost of lost product is related to the quantity, and we have $Z_\ell = w_\ell \cdot Q_\ell$, with w_ℓ the cost of losing one unit of product ℓ . The new objective functions that are “*product-oriented*” are:

$$\text{the quantity of lost products: } \sum Q_\ell = \sum_{\ell=1}^r Q_\ell$$

$$\text{the cost of lost products: } \sum w_\ell Q_\ell = \sum_{\ell=1}^r w_\ell \cdot Q_\ell$$

We assume that the remaining quantities of products in vials at the end of the planning horizon are lost. Notice that the minimization of the total quantity of lost product is equivalent to the minimization of the number of vials that are opened.

2.2 Problem framework

The problem is to assign each job to a performing machine, to determine a starting time for each job, and to determine the opening dates for the vials of each required product, in order to respect a bound on a job-oriented objective function, and to minimize a product-oriented objective function.

This is a general framework, where:

- jobs can have a routing if the machines constitute a shop environment (job shop, flow shop, etc.),
- jobs can be independent and machines can be duplicated (identical, uniform or unrelated parallel machines), and be subject to precedence constraints, etc.

and:

- products can be unique or multiple,
- products can be stored in a unique type of elementary container or in several types, of several volumes. In this case, an additional problem is to decide which type of elementary container has to be opened,
- the product consumption by the job can be continuous, discrete or instantaneous (two of these three types of consumption are illustrated in Fig. 1, where $I_\ell(t)$ indicates the inventory level of product ℓ at time t).

We introduce a notation to indicate the presence of the perishable raw materials constraints rm_p , with an indication on the consumption type: rm_p^i for instantaneous, rm_p^c for continuous and rm_p^d for discrete consumptions (notice that rm was introduced in [5] for “*raw materials*”). The notation $rm_p^i = 1$ indicates that one raw material product is required and that its consumption by the job is immediate.

(a) continuous consumption (b) immediate consumption

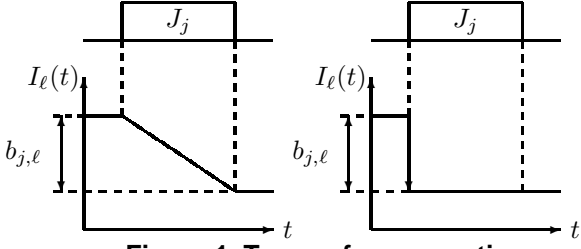


Figure 1. Types of consumptions

2.3 Examples

According to the three-field notation introduced in [4], the scheduling problems with raw materials constraints can be denoted as follows:

- $1|rm_p^i|\sum Q_\ell$ when there is no due date associated to the jobs,
- $1|rm_p^i = 1, L_{\max} \leq H|\sum Q_\ell$ when there is only one product, maximum lateness bounded by a given value H , and total quantity minimization,
- $P|rm_p^i, r_j, L_{\max} \leq H|\sum w_\ell Q_\ell$ for a parallel machines environment, jobs subject to release dates, bounded maximum lateness and total cost minimization.

We illustrate the problem definition by the following example. We consider a single machine problem with four jobs and one product. Processing times are equal to $p = (4, 3, 5, 2)$, product consumptions are equal to $b = (2, 1, 4, 3)$, the volume of one vial of the product is equal to $V = 5$ and its life time after being opened is equal to $T = 6$. We assume that all the jobs can be finished on time (no tardiness consideration) and we focus on the product consumption.

We consider that the product is consumed instantaneously by the job when its processing starts, that is we consider problem $1|rm_p^i = 1|\sum Q_\ell$.

The sequence of jobs (J_1, J_2, J_3, J_4) is illustrated in Fig. 2. One vial is open at time 0, immediately, job J_1 takes $b_1 = 2$ product units. 3 units remain available in the vial. At time 4, job J_2 starts and consumes $b_2 = 1$ product unit. At time 6 the product perishes and 2 units are lost. A new vial is opened at time 7 for job J_3 , which consumes immediately $b_3 = 4$ units. At time 12, job J_4 starts. However, job J_4 requires $b_4 = 3$ units of product, thus a new vial is opened, and it remains 3 product units at the end. These units are lost and at the end, 5 product units have been lost.

The sequence of jobs (J_3, J_2, J_4, J_1) is illustrated in Fig. 3. One vial is opened at time 0 and 4 units are consumed by job J_3 . At time 5, 1 unit is consumed by job J_2 and the remaining quantity is equal to 0. At time 8, a new vial is opened and job J_4 consumes 3 units. At time 10, job J_1 takes the 2 remaining units. At the end, the vial is empty and the quantity of lost product is equal to 0.

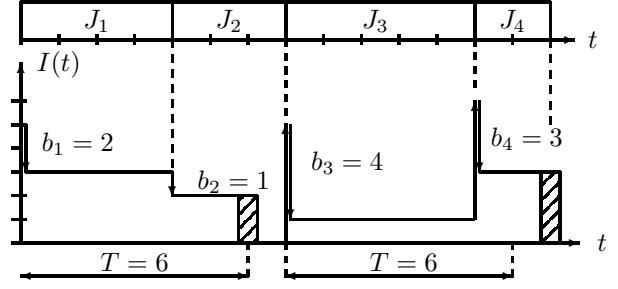


Figure 2. Example: 4 jobs, immediate consumption - sequence (J_1, J_2, J_3, J_4)

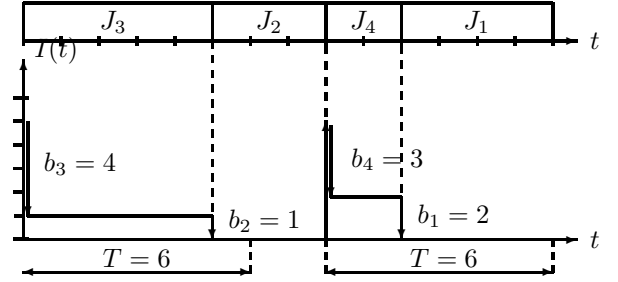


Figure 3. Example: 4 jobs, immediate consumption - sequence (J_3, J_2, J_4, J_1)

We consider now that the product is consumed continuously during the job processing, that is we consider problem $1|rm_p^c = 1|\sum Q_\ell$.

Fig. 4 illustrates the product consumption for sequence (J_1, J_2, J_3, J_4) . During the 4 first time units, 2 product units are consumed by job J_1 . Then, during the process of job J_2 , the product perishes and $2+1/3$ of product units are lost. A new vial has to be open during the job processing. The product is used for processing job J_3 but at the end of J_3 , the remaining product is lost (the quantity is equal to $2/3$). A new vial is open for job J_4 and only 3 units of the product are used. At the end, 5 product units are lost.

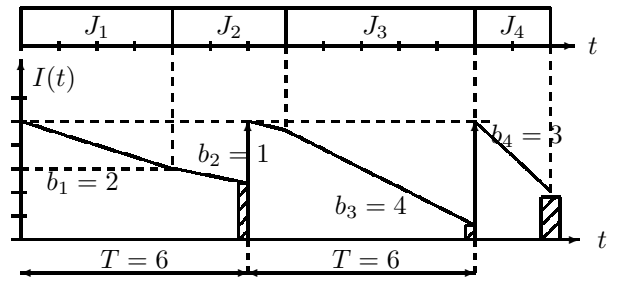


Figure 4. Example: 4 jobs, continuous consumption - sequence (J_1, J_2, J_3, J_4)

Fig. 5 illustrates the product consumption for sequence (J_3, J_2, J_4, J_1) . At time 6, only one third of job J_2 has been performed and therefore one third of b_2 has been consumed. $2/3$ of a product unit is lost, a new vial is opened. At time 12, the product perishes during the pro-

cess of job J_1 and again a new vial has to be open. At the end, the quantity of lost product is equal to $2/3$ plus $1/3$ plus 4, that is 5 product units.

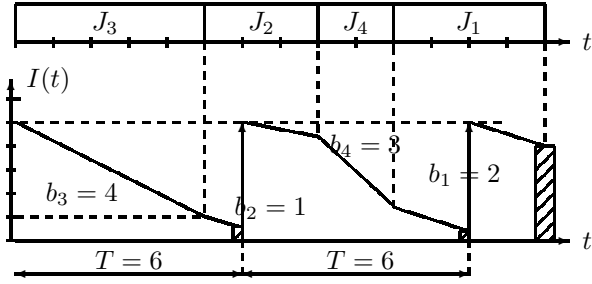


Figure 5. Example: 4 jobs, continuous consumption - sequence (J_3, J_2, J_4, J_1)

3 Single machine case

In this section, we show that problem $1|rm_p^i = 1, p_j = p|\sum Q_\ell$ is strongly NP-hard. Then, we give a mathematical formulation of problems $1|rm_p^i, L_{\max} \leq H|\sum Q_\ell$ and $1|rm_p^c, L_{\max} \leq H|\sum Q_\ell$.

3.1 Complexity

Proposition Problem $1|rm_p^i = 1, p_j = p|\sum Q_\ell$ is strongly NP-hard. The decision version of this problem is called 1M_PRM (single machine with perishable raw materials).

Proof. We show that the problem is strongly NP-hard by reduction from 3-PARTITION problem.

3-PARTITION:

Data: a collection of $n = 3m$ items, s_i the size of item i , B an integer. We assume that $\forall i, B/4 < s_i < B/2$.

Question: is it possible to partition the items into m triples of size B ?

1M_PRM:

Data: n' jobs, p the common processing time and (b_j) , consumption of job J_j ($1 \leq j \leq n'$), T the delay after opening, V the volume of one vial and Y three integers.

Question: is it possible to schedule the jobs so that the total quantity of lost product is less than or equal to Y ?

1M_PRM is in \mathcal{NP} (clear). We build the following instance to 1M_PRM: $n' = n, V = B, T = 5, Y = 0, b_j = s_j, \forall j, p = 2$.

Suppose that the answer to problem 3-PARTITION is 'yes'. Then, it is possible to partition the items into m triples S_1, \dots, S_m , each one of size B . Let schedule the subsets obtained in any order (see Fig. 6). For each subset S_h ($1 \leq h \leq m$), one has to open a new vial, of size $V = B$, the vial does not perish during the five times units and the jobs consume the whole quantity of the vial. At the end, m vials have been open and all the products

have been consumed. The quantity of lost product is equal to 0.

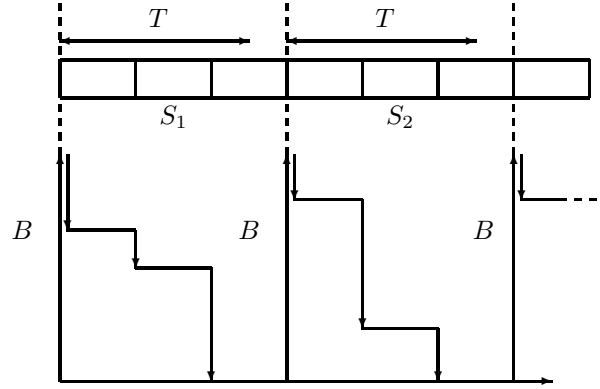


Figure 6. Answer to 3-PARTITION is 'yes'

Suppose now that the answer to 1M_PRM is 'yes'. We assume w.l.g. that $b_j \neq 0, \forall j$. Because the quantity of lost product is equal to 0, the number of vials that are needed is equal to m . The length of the schedule is equal to $\sum_{i=1}^n p_i = 2n = 6m$. Because $T = 5$, a new vial has to be open after the execution of only two jobs. If a new vial has to be open after the execution of only two jobs, then a new vial will have to be open for the last job, which will generate a loss of product. The consequence is that a vial has to be open every three jobs exactly. Therefore, the whole quantity in the vial has to be consumed and no backorder is allowed. Considering the triplets of jobs $(J_{3h+1}, J_{3h+2}, J_{3h+3}), \forall h, 0 \leq h \leq m-1$, the consumptions $(b_{3h+1}, b_{3h+2}, b_{3h+3})$ constitute a partition into m subsets of size V . Therefore, the answer to problem 3-PARTITION is 'yes'.

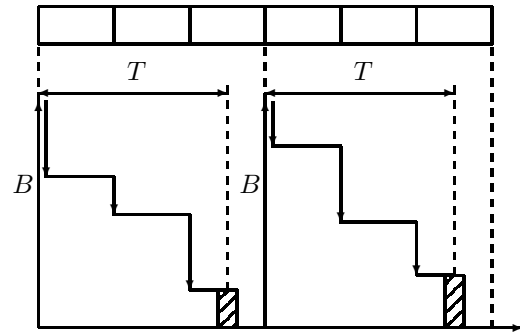


Figure 7. The whole quantity V has not been consumed

3.2 Mathematical formulations

We propose in this section a mathematical programming formulation of problems $1|rm_p^c, L_{\max} \leq H|\sum Q_\ell$ and $1|rm_p^i, L_{\max} \leq H|\sum Q_\ell$.

We define $x_{j,k} = 1$ if job J_j is in position k , 0 otherwise. The notation $p_{[k]}$ and $b_{[k],\ell}$ denote respectively the processing time of the job in position k and the consump-

tion of product ℓ of the job in position k . We have:

$$p_{[k]} = \sum_{j=1}^n p_j \cdot x_{j,k}$$

$$b_{[k],\ell} = \sum_{j=1}^n b_{j,\ell} \cdot x_{j,k}$$

We introduce the following continuous variables:

- $q_{[k],\ell} \geq 0$, quantity of inventory of product ℓ before the execution of the job in position k .
- $\delta_{[k],\ell} \geq 0$ is the time during which the product ℓ remains usable when the job in position k starts.
- $loss_{[k],\ell} \geq 0$ is the quantity of product ℓ which is lost during the execution of the job in position k .

As we will see, the expressions of these quantities depend on the hypotheses about the type of consumption (continuous or instantaneous). Four cases are identified:

1. $q_{[k],\ell} \geq b_{[k],\ell}$ and $\delta_{[k],\ell} \geq p_{[k]}$: in this case, there is enough product ℓ and it will not perish,
2. $q_{[k],\ell} < b_{[k],\ell}$ and $\delta_{[k],\ell} \geq p_{[k]}$: in this case, there is a backorder (not enough product ℓ),
3. $q_{[k],\ell} \geq b_{[k],\ell}$ and $\delta_{[k],\ell} < p_{[k]}$: in this case, the product will perish during the execution of the job in position k ,
4. $q_{[k],\ell} < b_{[k],\ell}$ and $\delta_{[k],\ell} < p_{[k]}$: in this case, there is not enough product and the product will perish before the end of the job in position k .

3.2.1 Case of continuous consumption

The inventory level of the product has the evolution illustrated in Fig. 1(a). We consider the four cases:

1. *Easy case*: If $q_{[k],\ell} \geq b_{[k],\ell}$ and $\delta_{[k],\ell} \geq p_{[k]}$, then

$$\begin{cases} q_{[k+1],\ell} = q_{[k],\ell} - b_{[k],\ell} \\ \delta_{[k+1],\ell} = \delta_{[k],\ell} - p_{[k]} \\ loss_{[k],\ell} = 0 \end{cases}$$

2. *Case of backorder*: If $q_{[k],\ell} < b_{[k],\ell}$ and $\delta_{[k],\ell} \geq p_{[k]}$, then

$$\begin{cases} q_{[k+1],\ell} = q_{[k],\ell} + V_\ell - b_{[k],\ell} \\ \delta_{[k+1],\ell} = T_\ell - (p_{[k]} - q_{[k],\ell} \frac{p_{[k]}}{b_{[k],\ell}}) \\ loss_{[k],\ell} = 0 \end{cases}$$

Indeed, the consumption rate is equal to $\frac{b_{[k],\ell}}{p_{[k]}}$. The backorder of product ℓ occurs $q_{[k],\ell} \frac{p_{[k]}}{b_{[k],\ell}}$ time units after the beginning of the job in position k . A new vial is open and the remaining quantity is consumed. This quantity is equal to $(p_{[k]} - q_{[k],\ell} \frac{p_{[k]}}{b_{[k],\ell}}) \frac{b_{[k],\ell}}{p_{[k]}} = b_{[k],\ell} - q_{[k],\ell}$.

3. *Case of perish*: If $q_{[k],\ell} \geq b_{[k],\ell}$ and $\delta_{[k],\ell} < p_{[k]}$, then

$$\begin{cases} q_{[k+1],\ell} = V_\ell - (p_{[k]} - \delta_{[k],\ell}) \frac{b_{[k],\ell}}{p_{[k]}} \\ \delta_{[k+1],\ell} = T_\ell - (p_{[k]} - \delta_{[k],\ell}) \\ loss_{[k],\ell} = q_{[k],\ell} - \delta_{[k],\ell} \frac{b_{[k],\ell}}{p_{[k]}} \end{cases}$$

4. *Case of backorder and perish*: The product ℓ will perish after $\delta_{[k],\ell}$ time units and the backorder will occur $q_{[k]} \frac{p_{[k]}}{b_{[k]}}$ time units after the beginning of the job in position k . Therefore, if $\delta_{[k],\ell} \geq q_{[k]} \frac{p_{[k]}}{b_{[k]}}$, we are in case 2, otherwise we are in case 3.

Notice that these expressions of the continuous variables are not linear. Problem $1|rm_p^c, L_{\max} \leq H| \sum Q_\ell$ cannot be formulated with a linear program, at least by using this definition of variables.

3.2.2 Case of instantaneous consumption

The inventory level of the product has the evolution illustrated in Fig. 1(b). It is assumed that if the product perishes exactly when a job starts, the consumption is realized by the job first, and the remaining quantity of product is lost. We consider the four cases:

1. *Easy case*: If $q_{[k],\ell} \geq b_{[k],\ell}$ and $\delta_{[k],\ell} \geq p_{[k]}$, then

$$\begin{cases} q_{[k+1],\ell} = q_{[k],\ell} - b_{[k],\ell} \\ \delta_{[k+1],\ell} = \delta_{[k],\ell} - p_{[k]} \\ loss_{[k],\ell} = 0 \end{cases}$$

2. *Case of backorder*: If $q_{[k],\ell} < b_{[k],\ell}$ and $\delta_{[k],\ell} \geq p_{[k]}$, then

$$\begin{cases} q_{[k+1],\ell} = q_{[k],\ell} + V_\ell - b_{[k],\ell} \\ \delta_{[k+1],\ell} = T_\ell - p_{[k]} \\ loss_{[k],\ell} = 0 \end{cases}$$

3. *Case of perish*: If $q_{[k],\ell} \geq b_{[k],\ell}$ and $\delta_{[k],\ell} < p_{[k]}$, then

$$\begin{cases} q_{[k+1],\ell} = 0 \\ \delta_{[k+1],\ell} = 0 \\ loss_{[k],\ell} = q_{[k],\ell} - b_{[k],\ell} \end{cases}$$

The product has perished and a new vial has to be open, if needed.

4. *Case of backorder and perish*: Again, for the same reasons, we are either in case 2 or in case 3.

3.2.3 Objective functions

The expressions of the objective functions are the following.

- $\sum Q_\ell$, the total quantity of product ℓ which is lost:

$$Q_\ell = \sum_{k=1}^n loss_{[k],\ell} + q_{[n+1],\ell}$$

$$\text{MIN} \sum Q_\ell = \sum_{\ell=1}^r Q_\ell$$

where $q_{(n+1),\ell}$ denotes the remaining quantity of product ℓ at the end of the horizon.

- $\sum w_\ell Q_\ell$, the total cost of product ℓ which is lost:

$$\text{MIN} \sum w_\ell Q_\ell = \sum_{\ell=1}^r w_\ell Q_\ell$$

The expression of the completion time of the job in position k , is:

$$C_{[k]} = \sum_{h=1}^k \sum_{j=1}^n p_j x_{j,h}, \forall k \in \{1, \dots, n\}$$

The expression of the maximum lateness is given by:

$$L_{\max} \geq C_{[k]} - \sum_{i=1}^n d_i x_{i,k}, \forall k \in \{1, \dots, n\}$$

and the bound on L_{\max} is given by:

$$L_{\max} \leq H$$

3.2.4 MILP formulation for problem $1|rm_p^i, L_{\max} \leq H|\sum Q_\ell$

This problem can be formulated by a linear program. We introduce the following binary variables:

- $v_{[k],\ell} = 1$ if a new vial of product ℓ is open for processing the job in position k because of a backorder, 0 otherwise.

$$v_{[k],\ell} = 1 \Leftrightarrow q_{[k],\ell} \leq b_{[k],\ell} - 1$$

- $w_{[k],\ell} = 1$ if a new vial of product ℓ is open for processing the job in position k because the product has perished, 0 otherwise.

$$w_{[k],\ell} = 1 \Leftrightarrow (q_{[k],\ell} \geq b_{[k],\ell}) \wedge (\delta_{[k],\ell} \leq p_{[k],\ell} - 1)$$

Notice that these expressions impose that we cannot have $v_{[k],\ell} = w_{[k],\ell} = 1$. The expressions of the objective functions are given in Section 3.2.3.

$$\sum_{i=1}^n x_{i,k} = 1, \forall k \in \{1, \dots, n\} \quad (1)$$

$$\sum_{k=1}^n x_{i,k} = 1, \forall i \in \{1, \dots, n\} \quad (2)$$

$$q_{[k],\ell} \leq b_{[k],\ell} - 1 + (1 - v_{[k],\ell}) \times (V_\ell + 1), \quad (3)$$

$$\forall k \in \{1, \dots, n\}, \forall \ell \in \{1, \dots, r\}$$

$$q_{[k],\ell} \geq b_{[k],\ell} - v_{[k],\ell} \times V_\ell, \quad (4)$$

$$\forall k \in \{1, \dots, n\}, \forall \ell \in \{1, \dots, r\}$$

$$\delta_{[k],\ell} \leq p_{[k]} - 1 + (1 - w_{[k],\ell}) \times (T_\ell + 1) + 2v_{[k],\ell}(T_\ell + 1), \quad (5)$$

$$\forall k \in \{1, \dots, n\}, \forall \ell \in \{1, \dots, r\}$$

$$\delta_{[k],\ell} \geq p_{[k]} - w_{[k],\ell} \times T_\ell - 2v_{[k],\ell}T_\ell \quad (6)$$

$$\forall k \in \{1, \dots, n\}, \forall \ell \in \{1, \dots, r\}$$

$$\text{loss}_{[k],\ell} \geq q_{[k],\ell} - b_{[k],\ell} - (1 - w_{[k],\ell}) \times V_\ell, \quad (7)$$

$$\forall k \in \{1, \dots, n\}, \forall \ell \in \{1, \dots, r\}$$

$$q_{[k+1],\ell} \geq q_{[k],\ell} - b_{[k],\ell} + V_\ell \times v_{[k],\ell} - 2V_\ell \times w_{[k],\ell}, \quad (8)$$

$$\forall k \in \{1, \dots, n-1\}, \forall \ell \in \{1, \dots, r\}$$

$$q_{[k+1],\ell} \leq q_{[k],\ell} - b_{[k],\ell} + V_\ell \times v_{[k],\ell} + 2V_\ell \times w_{[k],\ell}, \quad (9)$$

$$\forall k \in \{1, \dots, n-1\}, \forall \ell \in \{1, \dots, r\}$$

$$q_{[k+1],\ell} \geq q_{[k],\ell} - b_{[k],\ell} + V_\ell \times v_{[k],\ell} - 2V_\ell \times (2 - w_{[k],\ell} - v_{[k],\ell}), \quad (10)$$

$$\forall k \in \{1, \dots, n-1\}, \forall \ell \in \{1, \dots, r\}$$

$$q_{[k+1],\ell} \leq q_{[k],\ell} - b_{[k],\ell} + V_\ell \times v_{[k],\ell} + 2V_\ell \times (2 - w_{[k],\ell} - v_{[k],\ell}), \quad (11)$$

$$\forall k \in \{1, \dots, n-1\}, \forall \ell \in \{1, \dots, r\}$$

$$q_{[k+1],\ell} \leq 2V_\ell \times (1 - w_{[k],\ell} + v_{[k],\ell}), \quad (12)$$

$$\forall k \in \{1, \dots, n-1\}, \forall \ell \in \{1, \dots, r\}$$

$$q_{[k+1],\ell} \geq 2V_\ell \times (w_{[k],\ell} - 1 - v_{[k],\ell}), \quad (13)$$

$$\forall k \in \{1, \dots, n-1\}, \forall \ell \in \{1, \dots, r\}$$

$$q_{[1],\ell} = 0, \forall \ell \in \{1, \dots, r\} \quad (14)$$

$$\delta_{[k+1],\ell} \geq \delta_{[k],\ell} - p_{[k]} - 2T_\ell(v_{[k],\ell} + w_{[k],\ell}), \quad (15)$$

$$\forall k \in \{1, \dots, n-1\}, \forall \ell \in \{1, \dots, r\}$$

$$\delta_{[k+1],\ell} \leq \delta_{[k],\ell} - p_{[k]} + 2T_\ell(v_{[k],\ell} + w_{[k],\ell}), \quad (16)$$

$$\forall k \in \{1, \dots, n-1\}, \forall \ell \in \{1, \dots, r\}$$

$$\delta_{[k+1],\ell} \geq -T_\ell(1 - w_{[k],\ell} + v_{[k],\ell}), \quad (17)$$

$$\forall k \in \{1, \dots, n-1\}, \forall \ell \in \{1, \dots, r\}$$

$$\delta_{[k+1],\ell} \leq T_\ell(1 - w_{[k],\ell} + v_{[k],\ell}), \quad (18)$$

$$\forall k \in \{1, \dots, n-1\}, \forall \ell \in \{1, \dots, r\}$$

$$\delta_{[k+1],\ell} \geq T_\ell - p_{[k]} - 2T_\ell(1 + w_{[k],\ell} - v_{[k],\ell}), \quad (19)$$

$$\forall k \in \{1, \dots, n-1\}, \forall \ell \in \{1, \dots, r\}$$

$$\delta_{[k+1],\ell} \leq T_\ell - p_{[k]} + 2T_\ell(1 + w_{[k],\ell} - v_{[k],\ell}), \quad (20)$$

$$\forall k \in \{1, \dots, n-1\}, \forall \ell \in \{1, \dots, r\}$$

$$\delta_{[k+1],\ell} \geq T_\ell - p_{[k]} - 2T_\ell \times (2 - v_{[k],\ell} - w_{[k],\ell}), \quad (21)$$

$$\forall k \in \{1, \dots, n-1\}, \forall \ell \in \{1, \dots, r\}$$

$$\delta_{[k+1],\ell} \leq T_\ell - p_{[k]} + 2T_\ell \times (2 - v_{[k],\ell} - w_{[k],\ell}), \quad (22)$$

$$\forall k \in \{1, \dots, n-1\}, \forall \ell \in \{1, \dots, r\}$$

$$\delta_{[1],\ell} = 0, \forall \ell \in \{1, \dots, r\} \quad (23)$$

Constraints (1) and (2) ensure that there is one job per position and one position per job. Constraints (3) and (4) set the variables $v_{[k]}$. Constraints (5) and (6) set the variables $w_{[k]}$. Constraints (7) set the variables $loss_{[k]}$. Constraints (8) to (14) set the variables $q_{[k]}$. Constraints (15) to (23) set the variables $\delta_{[k]}$ (easy to check).

4 Computational experiments

The model has been tested with GLPK package (Gnu Linear Programming Kit) on a PC under Windows XP with Core 2 Duo, 2.26 GHz and 3.45 Go RAM. Processing times have been randomly generated in $[1,100]$. The due date of job J_j is equal to $\sum_{i=1}^n p_i, \forall j$, the number of products is equal to $r = 1$, the maximum volume V_{\max} is equal to 12, and for each instance, the volume has been generated in $[V_{\max}/2, V_{\max}]$. The consumption of each job has been generated in $[0, V_{\max}]$ and the life duration has been generated in $[100, 250]$. 20 instances have been generated per value of n . Table 1 gives the results. Column n is the number of jobs, columns ‘min’, ‘median’, ‘3rdq’, ‘avg’ and ‘max’ are respectively the minimum, the median, the third quartile, the average and the maximum of the computation times. Column ‘tml’ indicates the number of instances that have not been solved in less than 120 seconds. The values about CPU time are related to the instances that have been solved within the time limit.

n	CPU (s)					tml
	min	median	3rdq	avg	max	
6	0.00	5.10	10,15	5,18	18,30	0
8	0.10	0.10	0.20	0.16	0.50	7
10	0.10	0.20	0.28	0.45	1,90	10
12	0.20	0.60	4.70	4.44	26.70	11

Table 1. Results of experiments

The aim of these experiments was to validate the model. The results show that the problems can be difficult to solve with integer programming for a small number of jobs. This tends to indicate that the problems are difficult to handle in practice. We guess that the results will be certainly better with another solver like CPLEX or GUROBI optimizer, but dedicated methods like branch-and-bound will certainly perform better.

5 Conclusion

In this paper, we contribute to the area of scheduling problems with raw materials requirements, by considering that raw materials perish a certain time after being opened. We have proposed a general framework, which

allow considering a new set of difficult scheduling problems, depending on the product consumption type and on the considered objective function, which opens a wide area of new difficult scheduling problems. We propose a first linear programming formulation for the case of a single machine problem, where raw materials are consumed instantaneously.

An important problem, which seems more complex, consists in considering that the products are available under several packages, and that another decision variable is to choose which package – i.e. which volume – of the product, is the better to open, when needed.

References

- [1] J.-C. Billaut and P. Esquirol. Scheduling chemotherapy preparations with perishable raw materials constraints. In *INFORMS Health-Care, Montreal, Canada*, 2010.
- [2] X. Cai, J. Chen, Y. Xiao, and X. Xu. Product selection, machine time allocation, and scheduling decisions for manufacturing perishable products subject to a deadline. *Computers and Operations Research*, 35:1671–1683, 2008.
- [3] S. Goyal and B. Giri. Recent trends in modeling of deteriorating inventory. *European Journal of Operational Research*, 134(1):1–16, 2001.
- [4] R. Graham, E. Lawler, J. Lenstra, and A. R. Kan. Optimization and approximation in deterministic sequencing and scheduling: a survey. *Ann. Discrete Math.*, 4:287–326, 1979.
- [5] A. Grigoriiev, M. Holthuijsen, and J. van de Klundert. Basic scheduling problems with raw material constraints. *Naval research logistics*, 52(6):527–535, 2005.
- [6] J.-Y. Huang and M.-J. Yao. A new algorithm for determining production schedules when solving the multi-product economic lot sizing problem in flow shops. *International Journal of Advanced Manufacturing Technology*, 33(5-6):560–578, 2007.
- [7] T. Kim, Y. Hong, and S. Chang. Joint economic procurement-production-delivery policy for multiple items in a single-manufacturer, multiple-retailer system. *International Journal of Production Economics*, 103(1):199–208, 2006.
- [8] A. Mazier, J.-C. Billaut, and J.-F. Tournamille. Scheduling preparation of doses for a chemotherapy service. *Annals of Operations Research*, 178:145–154, 2010.
- [9] J. Ouenniche and F. Boctor. Two-group heuristic to solve the multi-product, economic lot sizing and scheduling problem in flow shops. *European Journal of Operational Research*, 129(3):539–554, 2001.
- [10] R. Teunter, M. Babai, and A. Syntetos. Abc classification: Service levels and inventory costs. *Production and Operations Management*, 19(3):343–352, 2010.