Priority Allocation Decisions in Large Scale MTO/MTS Multi-product Manufacturing Systems : Technical report

K. Hadj Youssef [*], Ch. van Delft [†], Y. Dallery [‡]

October 17, 2008

Abstract

We consider a single stage multi-product manufacturing facility producing a large number of end-products for delivery within a service constraint for the customer lead-time. The manufacturing facility is modeled as a multi-product, multi-priority queuing system. In order to reduce inventory costs, an efficient priority allocation between items consists in producing some items according to a Make-To-Stock (MTS) policy and others according to a Make-To-Order (MTO) policy depending on their features (costs, required lead-time, demand rates). We propose a general optimization procedure that gives a near-optimal flow control (MTO or MTS) to associate with each product and the corresponding near-optimal priority strategy. We illustrate efficiency of our procedure via several examples and by a numerical analysis. In addition, we show numerically that a small number of priority classes is sufficient to obtain near-optimal performances.

KEY-WORDS: *Make-to-Stock (MTS); Make-to-Order (MTO); Priority allocation; Scheduling rule; Heterogeneous multi-product queuing system.*

1 Introduction

An important issue in production management is whether products should be manufactured according to a Make-To-Order or a Make-To-Stock policy, depending on their main characteristics (costs, customer lead-time, demand parameters, ...). By definition, under make-to-order (MTO) management, a production order is released to the manufacturing facility only after a firm demand has been received from the customer. On the contrary, under make-to-stock (MTS) management, products are manufactured in anticipation of future demands. They are stored in the finished goods

[*] Laboratoire Genie Mecanique - Ecole Nationale d'Ingenieurs de Monastir, Tunisie, Khaled.HajYoussef@enim.rnu.tn

[†] HEC School of Management, Paris (GREGHEC), 78 351 Jouy-en-Josas Cedex, France. E-mail: vandelft@hec.fr

[‡] Laboratoire Génie Industriel - Ecole Centrale Paris, HEC School of Management, Paris (GREGHEC), 78 351 Jouy-en-Josas Cedex, France.

inventory (FGI), from which the orders will be directly delivered. The advantage of the MTS policy, with respect to the MTO policy, is to allow immediate reactivity to external demands, but its drawback is the inventory holding costs.
Clearly, adopting a pure MTS policy or a pure MTO policy for all products can be considered an extreme choice as the option of combining these two policies exists. In a context where a single manufacturing facility produces several classes of end-products, it seems appealing to try to produce some products (for example, the products with a low holding cost) according to a MTS policy and the others (typically, the products with a very high holding cost) according to a MTO policy. However, for general situations mixing several product types, deciding which products to manage according to each policy is a complex issue. As a matter of fact, the optimal MTO/MTS allocation depends not only on the parameters of each product type (demand behavior, customer lead-time, holding costs...) but also on the scheduling policy of the manufacturing facility. The simplest approach in scheduling a manufacturing facility is to use a first-in-first-out (FIFO) scheduling rule under which all production orders are processed according to their release dates. Another approach that may be more efficient is to give priority (PR) to some product types in order to give them more reactivity which would allow producing them under a MTO policy (i.e. without any inventory), while still guaranteeing the reactivity required by customers.

Several papers have addressed the performance analysis and the optimization strategy of combined MTS and MTO inventory/production systems. Different assumptions in terms of scheduling policy were considered. Arreola-Risa et al. [1] studied the optimality of MTO versus MTS policies for a multi-product manufacturing system where the products were scheduled according to a FIFO policy. Carr and Duenyas [3], Ha [7] and Veatch and Wein [16] studied the structure of optimal dynamic priority allocation rules in the setting of optimal control theory for queuing systems. Several other papers considered static (and actually simpler) priority allocation rules. Sox [14], Federgruen et al. [5, 6], Williams [17] and Rajagopalan [12] analyzed, under different assumptions, the system performances and the structure of the optimal MTO/MTS decision policy. These papers make the assumption that a static priority can be given to the MTO products. In the same context, Hadj Youssef et al. [8] considered a manufacturing facility producing two classes of end-products: a small number of high demand rate products, and a large number of low demand rate products. Static priority can be given to the low demand rate products. These authors provide analytical expressions that describe the structure of the optimal priority decision and the associate optimal MTO/MTS decision policy. Sleptchenko et al. [15] analyzed the interest of using repair priorities to reduce stock investment in spare part networks. In their case, due to the high complexity of the considered system, it was necessary to rely on a simplistic heuristic for the priority allocation.

The purpose of this paper is to give a general optimization procedure that gives near-optimal flow control (MTO or MTS) to associate to each product and the corresponding priority allocation strategy. The question of optimizing the flow control (MTO vs MTS) and the priority allocation for multi-item manufacturing systems is known to be significant with respect to the inventory costs (see for example [8] or [12]). Such an analysis, and an optimization issue, requires a performance evaluation model for the industrial system. We thus provide a multi-item queuing model for the considered class of manufacturing systems, which encompasses the customer lead-time concept. The first part of the paper is thus dedicated to the definition and the theoretical analysis of this queuing system, which in fact generalizes the models described in [2] and [8] to include the customer lead-time issue. This is the first contribution of the paper. In the second part of the paper, with respect to this general performance evaluation model, we consider the optimization of the flow control (MTO or MTS) and the priority allocation strategy. We focus on providing an implementable solution technique for general unstructured large-scale problems. It is easily seen that any direct formulation leads to very complex non-linear integer programming models, impossible to solve by a direct brute force approach. We thus propose an approximate optimization scheme relying on two fundamental theoretical properties which characterize the class of problems under consideration. First, it can be shown that for a fixed workload of prioritized items, the underlying optimization problem is a linear integer programming problem. Second, it is known that for this class of problem the relaxation of the integrity constraints provides near-optimal solution (i.e. a near integer solution). We show how these properties can be used to build near-optimal heuristics and theoretical bounds, from which it can be seen that the solutions can be expected to be excellent and near-optimal for typical industrial applications.

This paper is structured as follows. In section 2, we present the model, the main assumptions and the fundamental cost/service level formulation. In section 3, we present the performance analysis indices corresponding to the production/inventory model. Section 4 presents the optimization method and the bounds computation. Numerical studies are provided in section 5, focusing on the case with two priority indices (namely high priority and low priority). Section 6 extends the approach to a general number of priority indices. In this setting, we illustrate, with numerical examples, that for typical industrial applications near-optimal performances can be obtained with a small number of priority classes (i.e. 2 or 3 classes). Our conclusions are given in section 7.

2 The performance evaluation model

2.1 General assumptions for the production/inventory system

We consider a single-stage manufacturing facility producing k heterogeneous product types.

The arrivals of demands occur according to independent Poisson processes with rate λ_i for product type i. The delivery of parts, corresponding to a given demand for product type i, has to be completed in a given time period L_i after the order is received, called the customer lead-time. The customer lead-time is thus the admissible delay (i.e. which does not induce backorder costs) between the time the order is placed and the time it has to be delivered to the customer. As in ([2], chapter 4.5), we assume that the physical delivery cannot be made to the customer before this time period. If, at the due date, there are no parts in the finished good inventory, demands are then backordered. Furthermore, in the considered problem formulation, the delivery process has to satisfy fill rate constraints for the different products, which bound the probability that an order will be delivered after the required lead-time. The processing time is exponentially distributed with a mean value parameter $1/\mu$, which does not depend on the product index i. Furthermore, no setup is required to switch from one product-type to another, and the manufacturing process is totally reliable.

The flow in this single stage manufacturing facility is controlled by a base-stock control system (BSCS) [2, 8, 10]. The single-stage BSCS is a control mechanism which consists of releasing raw parts into the manufacturing process each time a new customer demand for finished products is issued. Such a control mechanism is defined by base-stock levels, which can be interpreted as the target inventory levels for the different items. It should be noted that the optimal values of these base-stocks depend on the priority allocation policy. For a given product, this generic policy can be interpreted as a MTS policy when the base-stock level is positive, and as a MTO policy when the base-stock level is equal to zero. The unit inventory holding cost rate associated to product type i is denoted by h_i.

The production orders for the items are scheduled according to their priority levels. We propose a theoretical analysis for a model with N priority classes. In this setting, products with a priority level l have a higher priority than products with priority levels j, with $j > l$. We also assume that the priority is preemptive, i.e. if a j-level priority order is being manufactured while a new l-priority level order has to be produced by the system (with $l < j$), the current j-level-priority order is stopped and delayed. The l-priority level order is then produced first. In addition, it is assumed that orders with equal priority level are scheduled according to a FIFO rule. We show via several examples that for typical large-scale industrial cases near-optimal performances can be obtained with a small number of priority classes (i.e. 2 or 3 classes).

In order to model this BSCS in presence of required customer lead-times, we propose a multi-product queuing model with time lags, which is an extension of the model discussed in ([2], chapter 4.5). The structure of this basic model is depicted below,

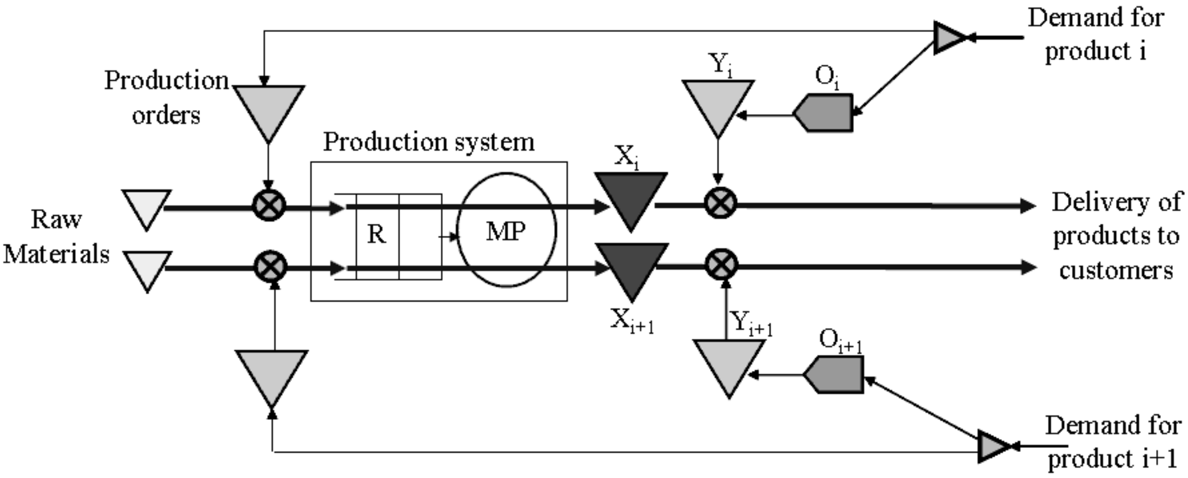

Figure 1: Queuing network model of a multi-product single stage BSCS with time lags.

This basic model with time lags and several products proceeds as follows. MP represents the considered manufacturing process. It contains parts being processed or waiting to be processed, which are referred to as work-in-process. Let us define the production lead-time as the time elapsed between the release of a raw part in the manufacturing facility (according to the BSCS) and the physical delivery of the corresponding finished part into the FGI.

O_i represents the order time lag process: it contains the list of time lagged orders. Queue R contains the raw parts waiting for processing in MP. Queue I_i is the product type-i FGI, i.e. the output buffer of the system containing type-i parts that have completed processing and are ready for delivery to customers. Queue Y_i contains backordered demands (i.e. orders delivered after the requisition date). R, I_i, and Y_i are unlimited.

Initially, before any customer demands arrive in the system, I_i contains a base-stock of s_i parts, while R, MP, O_i and Y_i are empty. When a customer demand for type-i product arrives in the system, it triggers a production order of a new part by MP and a demand for the delivery of a type-i finished part to the customer. The new raw part is directly stored in R while the new demand is stored in O_i for a deterministic time period equal to the time lag L_i, after which it will be transferred to queue Y_i. Queues I_i and Y_i are linked into a synchronization station: if there are simultaneously a customer demand in Y_i and an available part in I_i, the part is immediately delivered to the customer and the demand in Y_i is satisfied. If there is no available part in I_i, the customer demand continues to wait in queue Y_i until the arrival of a new finished product in queue I_i. The sojourn time in Y_i is called the backorder time, taking into account the time lag L_i between orders and requisitions. The case with time lags equal to zero has been considered in [8].

It is worth noting that the time lag L_i impacts both the backordered demands queue Y_i and the physical inventories I_i. The global effect of such a time lag is quite intuitive: if the time lag increases, it will progressively become superior to the production lead-time, in such a way that the backordered demands will decrease while the finished parts inventory will increase, as the parts will have to wait for the delivery requisition date.

2.2 Notations for the model

For this BSCS model, we adopt the following notations:

k : the number of end-products,

$\mathcal{C}$: the sets of indices corresponding to all products, i.e. $\mathcal{C} = \{1, ..., k\}$,

N : the number of priority classes (also called priority levels),

$\mathcal{C}_j$: the set of item indices corresponding to priority level j, for $j \in \{1, ..., N\}$.

For each product type $i \in \mathcal{C}$, we define:

λ_i : the average demand rate,

h_i : the inventory holding cost rate,

L_i : the required customer lead-time,

γ_i^r : the required fill rate associated to product type i. This fill rate is defined as the probability that the customer lead-time will be satisfied for a given delivery.

Furthermore, we have

μ : the average production rate of the manufacturing facility,

λ : the total workload, defined as $\lambda = \sum\limits_{i \in \mathcal{C}} \lambda_i$,

ρ : the utilization factor of the manufacturing facility, given by $\rho = \lambda/\mu$.

For every product type $i \in \mathcal{C}$, we also introduce the decision variables,

s_i : the base-stock level for product type i,

and the following random variables:

I_i : the number of type-i parts in the finished product inventory,

D_i: the backorder time for demand of type-i products.

The performance indices can now be defined as:

γ_i : the effective fill rate associated to product type i,

according to

$$\gamma_i = Prob\{D_i = 0\}, \tag{2.1}$$

Z_i : the average holding cost induced by product type i, according to

$$Z_i = h_i E[I_i]. \tag{2.2}$$

2.3 The fundamental cost/service level formulation

The optimization problem considered here consists of finding the optimal priority allocation and the corresponding values for the base-stock levels, which minimize the average holding cost, under a delivery fill rate constraint for each product type. For every product type $i \in \mathcal{C}$, we thus introduce the decision variables,

$$X_{i,j},$$

with $X_{i,j} = 1$ if product type i has the priority level j and $X_{i,j} = 0$ if not. In order to simplify the notations, these priority decision variables and the base-stock levels are aggregated into the following vectors

$$\mathcal{X} = X_{1,1}, ..., X_{1,N}, X_{2,1}, ..., X_{2,N}, ..., X_{k,1}, ..., X_{k,N} \text{ and } \mathcal{S} = s_1, s_2, ..., s_k.$$

This holding cost/service level formulation can then be expressed as the following nonlinear integer program

$$\begin{aligned}
Min_{\{\mathcal{X},\mathcal{S}\}} \quad & Z(\mathcal{X},\mathcal{S}) = \sum_{i \in \mathcal{C}} Z_i(\mathcal{X},\mathcal{S}) && (2.3)\\
\text{s.t.} \quad & \gamma_i(\mathcal{X},\mathcal{S}) \geq \gamma_i^r, \quad \text{with} \;\; i \in \mathcal{C}, && (2.4)\\
& \sum_{j=1}^{N} X_{i,j} = 1, \quad \text{with} \;\; i \in \mathcal{C}, && (2.5)\\
& X_{i,j} \in \{0,1\}, \quad \text{with} \;\; i \in \mathcal{C}, j = 1, ..., N, && (2.6)\\
& s_i \in \mathbb{N}, \quad \text{with} \;\; i \in \mathcal{C}. && (2.7)
\end{aligned}$$

Solving problem (2.3)-(2.7) is a complex issue. First, obtaining the expressions of the performance indices $Z_i(\mathcal{X},\mathcal{S})$ and $\gamma_i(\mathcal{X},\mathcal{S})$, for $i \in \mathcal{C}$, as functions of all the model parameters, is not direct and approximations have to be made. Some important structural properties can be exhibited, which allow us to simplify the expression of the main optimization problem. We first show how the initial problem with N priority classes can be approximatively decomposed into a set of $N - 1$ sub-problems with two priority classes. For solving such two-priority class models, we extend some theoretical results of [2] and [8] and provide an efficient approximate analysis scheme. Once the performance indices are known, the second difficulty directly arises from the structure of problem (2.3)-(2.7) which is a non-linear integer programming problem. We provide a solution procedure (both from the theoretical and numerical point of view). This approach is first illustrated for the two-priority class model. We show efficiency of the proposed approach with several examples. Then, we extend the method to the general multi-priority class models. Simultaneously, we exploit this new solution approach in order to get some managerial insight in the considered MTO/MTS issue.

3 Performance analysis model

We develop in this section the expression of the performance analysis indices, issued from the time-lagged BSCS, that are used in the optimization problem (2.3)-(2.7). This section is mainly focused on the situation with two priority indices (namely high priority and low priority). The theoretical expressions will be generalized to a general number of priority indices in section 6.

3.1 Decomposition property for multi-item time-lagged BSCS under FIFO rule

For a time-lagged BSCS under FIFO rule for all items, specific theoretical properties hold and explicit analytical formulas are available and define the major significant performance indices.

Property 1 : For $i \in \mathcal{C}$, under a FIFO rule, the BSCS random variables I_i, Y_i, D_i and the fill rates γ_i are independent both of the base-stock levels s_j and of the lead-times L_j, for all $j \in \mathcal{C} \setminus \{i\}$.
Proof : see Appendix 1.

Property 2: The multi-product BSCS under FIFO rule can be independently decomposed into a set of i (with $i \in \mathcal{C}$) independent time-lagged auxiliary single-product BSCS's with an arrival rate λ_i, a time-lag L_i and a service rate given by

$$\mu_i = \mu - [\sum_{m \in \mathcal{C}} \lambda_m] + \lambda_i = \mu(1-\rho) + \lambda_i. \tag{3.1}$$

.

Proof : see Appendix 2.

This result is a generalization of Buzacott et al. [2], who considered a multi-product queuing model of a BSCS without time-lags. This fundamental property underlies the overall solving approach for the flow/priority problem (2.3)-(2.7) considered in this paper. The main consequence of this property is that a decomposition property holds true under which the state variables related to each product type can be separately analyzed.

It is known (see [2], pp.138-143) that for such a single-product BSCS with time-lag, the average inventory is given by

$$E[I_i(s_i, \rho)] = s_i + \lambda_i \, L_i - \frac{\lambda_i}{\mu(1-\rho)} \left[1 - \left(\frac{\lambda_i}{\mu(1-\rho)+\lambda_i} \right)^{s_i} e^{-\mu(1-\rho)L_i} \right]. \tag{3.2}$$

Expression (3.2) makes the link between the average inventory $E[I_i(s_i, \rho)]$ and the global utilization factor ρ explicit. This feature is important because this dependence will be directly used in the solution procedure. It is worth noting that in this flow model, the average inventory level (3.2) increases with the base-stock, as usual, but it increases also with the time lag, because delivery of parts before the requisition date increases the effective inventory level.

Property 3: The expression of the corresponding fill rate is given by

$$\gamma_i(s_i, \rho) = 1 - \left(\frac{\lambda_i}{\mu(1-\rho)+\lambda_i} \right)^{s_i} e^{-\mu(1-\rho)L_i}. \tag{3.3}$$

Proof: See Appendix 3.

It is worth noting that, by applying the above equations, one finds that the inventory cost for product type i, expressed in (2.3), can be rewritten as a function of s_i and ρ as follows

$$Z_i(s_i, \rho) \quad = \quad h_i \left(s_i + \lambda_i \, L_i - \frac{\lambda_i}{\mu(1-\rho)} \left[1 - \left(\frac{\lambda_i}{\mu(1-\rho)+\lambda_i} \right)^{s_i} e^{-\mu(1-\rho)L_i} \right] \right). \tag{3.4}$$

The conclusion is that under a FIFO rule for the items, the multi-product BSCS with time lags can be fully characterized with a decomposition approach. Analytical expressions are available for the performance indices.

3.2 Performance analysis for the two-priority class model

This section proposes a performance analysis which is focused on the case of two priority indices (namely high priority and low priority). The approach will be generalized to a general number of priority indices in section 6.

3.2.1 The decomposition principle

The basic principles underlying the analysis are the following. We first divide the item indices in two groups : the high priority item indices $\mathcal{C}_1 := \{i : X_{i,1} = 1\}$ and low priority item indices $\mathcal{C}_2 := \{i : X_{i,2} = 1\}$. The performance evaluation approach consists in decomposing the multi-product queuing system into two systems: one separate queuing system for the high priority items, i.e. for items i with $i \in \mathcal{C}_1$ and one separate system for the low priority items, i.e. for items i with $i \in \mathcal{C}_2$.

Under the preemptive assumption, it is easy to see that the high priority orders are not affected by those of low priority. As a consequence, performance analysis of the high priority order class is straightforward, according to properties 1-3 given above. On the contrary, the low priority orders are significantly affected by the high priority orders. For the considered setting no closed form formula can be expected, and we therefore provide an approximate performance approach. It is worth noting that in fact this section generalizes the results presented in Hadj Youssef et al. [8] to the case of positive customer lead-times (i.e. $L_i \geq 0$, for all $i \in \mathcal{C}$).

3.2.2 Exact analysis for the high priority items

Under the preemptive assumption, it is clear that the production of high priority orders is not affected by low priority orders. The family of high priority items (i.e. the items $i \in \mathcal{C}_1$) can thus be analyzed as a multi-item BSCS with time lags, scheduled by a FIFO policy.

The multi-product BSCS restricted to the high priority level items can be decomposed into a set of independent time-lagged auxiliary single-product BSCS's with a service rate given by, for $i \in \mathcal{C}_1$,

$$\mu_{i,1}(\mathcal{X}) = \mu(1 - \rho_1(\mathcal{X})) + \lambda_i, \tag{3.5}$$

with the corresponding utilization factor for high priority items given by

$$\rho_1(\mathcal{X}) = \sum_{m \in \mathcal{C}} X_{m,1}\lambda_m/\mu. \tag{3.6}$$

For given numerical values of s_i and ρ_1, equations (3.3)-(3.4) can be reformulated as

$$\begin{aligned}
\gamma_{i,1}(s_i, \rho_1(\mathcal{X})) &= 1 - \left(\frac{\lambda_i}{\mu(1-\rho_1(\mathcal{X})) + \lambda_i}\right)^{s_i} e^{-(1-\rho_1(\mathcal{X}))\,\mu L_i}, &(3.7)\\
Z_{i,1}(s_i, \rho_1(\mathcal{X})) &= h_i\Bigg(s_i + \lambda_i\, L_i \\
&- \frac{\lambda_i}{\mu(1-\rho_1(\mathcal{X}))}\left[1 - (\frac{\lambda_i}{\mu(1-\rho_1(\mathcal{X})) + \lambda_i})^{s_i} e^{-(1-\rho_1(\mathcal{X}))\,\mu L_i}\right]\Bigg). &(3.8)
\end{aligned}$$

3.2.3 The low priority class: a heuristic approach

The analysis of the low priority items is more complex because the production facility becomes unavailable for these products each time a high priority order is issued. As the demand process for each high priority product is a Poisson process, the rate at which the production facility becomes unavailable for the low priority family is given by

$$\sum_{m \in \mathcal{C}_1} \lambda_m = \sum_{m \in \mathcal{C}} X_{m,1}\lambda_m, \tag{3.9}$$

which is the aggregate arrival rate for the high priority products. Once unavailable, the process remains in this state during a random time period which can be viewed as the busy period of a production facility loaded exclusively by the high priority family. Due to the assumptions stated in the above sections, this random unavailability time period can be interpreted as the busy period of an M/M/1 queue. Unfortunately, no closed-form formulas exist for the corresponding sojourn time probability density function. However, it is shown in [8] that the average sojourn time in the real system can be computed as

$$\frac{1}{\mu - \sum\limits_{m \in \mathcal{C}_2} \lambda_m - \sum\limits_{m \in \mathcal{C}_1} \lambda_m (2-\rho)} = \frac{1}{\mu - \sum\limits_{m \in \mathcal{C}} X_{m,2}\lambda_m - \sum\limits_{m \in \mathcal{C}} X_{m,1}\lambda_m(2-\rho)}. \tag{3.10}$$

We propose, as a simple heuristic, to approximate the different low priority items service time by simple exponentially distributed random variables, with rate denoted by $\frac{1}{\mu_{i,2}(\mathcal{X})}$ for every product type i (with $i \in \mathcal{C}_2$). In order to be consistent with the global multi-product model, we choose this rate in such a way that the average sojourn time in this fictitious queue is equal to the average sojourn time for the real system given in (3.10).

The demand processes are Poisson and the production rates are equal for all product types. This sojourn time is the same for every low priority item.

Under the exponential approximation, the average sojourn time associated with a production rate $\mu_{i,2}(\mathcal{X})$ is known to be given by

$$\frac{1}{\mu_{i,2}(\mathcal{X}) - \sum\limits_{m \in \mathcal{C}} X_{m,2}\lambda_m}, \tag{3.11}$$

and by equating (3.10)-(3.11), we find, for $i \in \mathcal{C}_2$,

$$\mu_{i,2}(\mathcal{X}) = \mu(1-\rho)(1-\rho_1(\mathcal{X})) + \lambda_i. \tag{3.12}$$

As a consequence, the low priority family can also be analyzed, in an approximate way, via a set of independent time-lagged auxiliary single-product BSCS's. For given numerical values of s_i and ρ_1, the fill rate and the cost for low priority item i, with $i \in \mathcal{C}_2$, can be found with equations (3.3)-(3.4) and amount to

$$\gamma_{i,2}(s_i, \rho_1(\mathcal{X})) \cong 1 - \left(\frac{\lambda_i}{\mu(1-\rho)(1-\rho_1(\mathcal{X}))+\lambda_i}\right)^{s_i} e^{-\mu(1-\rho)(1-\rho_1(\mathcal{X}))L_i}, \tag{3.13}$$

$$Z_{i,2}(s_i, \rho_1(\mathcal{X})) \cong h_i\Bigg(s_i + \lambda_i\, L_i - \frac{\lambda_i}{\mu(1-\rho)(1-\rho_1(\mathcal{X}))} \left[1 - \left(\frac{\lambda_i}{\mu(1-\rho)(1-\rho_1(\mathcal{X}))+\lambda_i}\right)^{s_i} e^{-\mu(1-\rho)(1-\rho_1(\mathcal{X}))L_i}\right]\Bigg) \tag{3.14}$$

Logically, one can introduce the utilization factor for low priority items

$$\rho_2(\mathcal{X}) = \sum_{m \in \mathcal{C}} X_{m,2}\lambda_m/\mu, \tag{3.15}$$

with the property that

$$\rho = \rho_1(\mathcal{X}) + \rho_2(\mathcal{X}). \tag{3.16}$$

4 The heuristic solution procedure

With respect to the general performance evaluation model exhibited in the previous section, we now consider the optimization of the flow control and the priority allocation strategy. It will be easily seen that any direct formulation leads to very complex non-linear integer programming models. As a consequence, we develop in this section an approximate optimization scheme relying on two fundamental theoretical properties. First, it will be shown that for a fixed workload of prioritized items, the underlying optimization problem simplifies itself to a linear integer programming problem. Second, we propose a relaxation of the integrity constraints which is known to provide in this case a near-optimal solution (i.e. a near integer solution).

4.1 The problem reformulation

It can be seen in above expressions that the utilization factor $\rho_1(\mathcal{X})$ defined in (3.6) plays a key role in that all the fill rates and the cost rates can be expressed as functions of this utilization factor. As a consequence, in the considered two priority class setting, the holding cost/service level formulation (2.3)-(2.7) can be rewritten as the following nonlinear integer program

$$Min_{\{\mathcal{X},\mathcal{S}\}} \quad Z(\mathcal{X},\mathcal{S}) = \sum_{i \in \mathcal{C}} \Big[X_{i,1}\, Z_{i,1}(s_i, \rho_1(\mathcal{X})) + X_{i,2}\, Z_{i,2}(s_i, \rho_1(\mathcal{X}))\Big] \tag{4.1}$$

$$\text{s.t.} \quad X_{i,1}\,\gamma_{i,1}(s_i, \rho_1(\mathcal{X})) + X_{i,2}\,\gamma_{i,2}(s_i, \rho_1(\mathcal{X})) \geq \gamma_i^r, \quad \text{for } i \in \mathcal{C}, \tag{4.2}$$

$$X_{i,1} + X_{i,2} = 1, \quad \text{for } i \in \mathcal{C}, \tag{4.3}$$

$$X_{i,1}, X_{i,2} \in \{0,1\}, \quad \text{for } i \in \mathcal{C}, \tag{4.4}$$

$$s_i \in \mathbb{N} \quad \text{for} \quad \text{for } i \in \mathcal{C}, \tag{4.5}$$

where $Z_{i,1}(\mathcal{X}, s_i)$, $Z_{i,2}(\mathcal{X}, s_i)$, $\gamma_{i,1}(\mathcal{X}, s_i)$ and $\gamma_{i,2}(\mathcal{X}, s_i)$ are calculated by equations (3.7)-(3.8) and (3.13)-(3.14).

4.2 Analysis for a fixed ρ_1 value.

Let us fix the value of the workload ρ_1. The holding cost/service level formulation amounts to the following nonlinear integer program

$$
\begin{aligned}
Min_{\{\mathcal{X},\mathcal{S}\}} \quad & Z = \sum_{i\in\mathcal{C}} \Big[X_{i,1}\, Z_{i,1}(s_i, \rho_1) + X_{i,2}\, Z_{i,2}(s_i, \rho_1) \Big] && (4.6)\\
\text{s.t.} \quad & X_{i,1}\, \gamma_{i,1}(s_i, \rho_1) + X_{i,2}\, \gamma_{i,2}(s_i, \rho_1) \geq \gamma_i^r, \quad \text{for } i \in \mathcal{C}, && (4.7)\\
& \sum_{i\in\mathcal{C}} X_{i,1} \frac{\lambda_i}{\mu} = \rho_1, && (4.8)\\
& X_{i,1} + X_{i,2} = 1, \quad \text{for } i \in \mathcal{C}, && (4.9)\\
& X_{i,1}, X_{i,2} \in \{0, 1\}, \quad \text{for } i \in \mathcal{C}, && (4.10)\\
& s_i \in \mathbb{N}, \quad \text{for } i \in \mathcal{C}. && (4.11)
\end{aligned}
$$

As a first step of the solution procedure, the property below establishes that the optimal base-stock values $s^*_{i,j}$ can be be easily found, as functions of the remaining variables and parameters of the problem.

Property 4: For a given ρ_1 value, the optimal base-stocks $s^*_{i,j}$ guaranteeing condition (4.7), for product type i with a priority level j, are given by the formulas

$$s^*_{i,1}(\rho_1) = \max\left(\left\lceil \frac{\mu(1-\rho_1)L_i + \ln(1-\gamma_i^r)}{\ln\left(\frac{\lambda_i}{\mu(1-\rho_1)+\lambda_i}\right)} \right\rceil \, , \; 0 \right), \tag{4.12}$$

$$s^*_{i,2}(\rho_1) = \max\left(\left\lceil \frac{(\mu(1-\rho)(1-\rho_1))L_i + \ln(1-\gamma_i^r)}{\ln\left(\frac{\lambda_i}{\mu(1-\rho)(1-\rho_1)+\lambda_i}\right)} \right\rceil \, , \; 0 \right). \tag{4.13}$$

Proof. It is obvious that $Z_{i,1}(\cdot, \rho_1)$, $Z_{i,2}(\cdot, \rho_1)$, $\gamma_{i,1}(\cdot, \rho_1)$ and $\gamma_{i,2}(\cdot, \rho_1)$, are increasing functions. The optimal $s^*_{i,j}$ values which satisfy the fill rate constraints and minimize the cost functions simultaneously are given by the solution of

$$\gamma_{i,j}(s^*_{i,j}, \rho_1) = \gamma_i^r, \quad \text{for } i \in \mathcal{C}. \tag{4.14}$$

By exploiting formulas (4.12)-(4.13) and by eliminating variables $X_{i,2}$ (with $i \in \mathcal{C}$) via constraint

(4.9), the above problem (4.6)-(4.11) can thus be simplified

$$Min_{\{\mathcal{X}\}} \qquad Z = \sum_{i \in \mathcal{C}} \left[\hat{Z}_{i,2}(\rho_1) + X_{i,1} \left(\hat{Z}_{i,1}(\rho_1) - \hat{Z}_{i,2}(\rho_1) \right) \right] \tag{4.15}$$

$$\text{s.t.} \qquad \sum_{i \in \mathcal{C}} X_{i,1} \frac{\lambda_i}{\mu} = \rho_1, \tag{4.16}$$

$$X_{i,1} \in \{0,1\} \quad \text{for } i \in \mathcal{C}, \tag{4.17}$$

with $\hat{Z}_{i,1}(\rho_1) = Z_{i,1}(s^*_{i,1}(\rho_1), \rho_1)$ and $\hat{Z}_{i,2}(\rho_1) = Z_{i,2}(s^*_{i,2}(\rho_1), \rho_1)$. It is easily seen that the optimal cost (4.15) is not a convex function of the decisions variables. As a consequence the problem can be expected to be hard to solve by a brute force approach. Problem (4.6)-(4.11) is a mixed binary/integer nonlinear program, which is a priori hard to solve. Furthermore, due to constraint (4.8) for most numerical values of ρ_1, there is no feasible solution to the problem. However, we show in the next subsections how this reformulation can be used in the approximate solution technique which precisely relies on a relaxation of the integrity in constraint (4.8).

4.3 The optimization procedure

The key element underlying the proposed approximate optimization procedure consists in observing that for a fixed value of ρ_1, the problem (4.15)-(4.17) becomes linear with respect to the decision variables $X_{i,1}$. Hence, the corresponding relaxation of the problem (4.15)-(4.17) to real variables is easily solved. Furthermore, it will be shown that the solution of the relaxed problem is known to be nearly integer. As a consequence, an efficient approximate integer optimal solution can be be built via a heuristic rounding procedure applied to the non-integer solution of the relaxed problem. In this way, the optimal solution can be approximated by considering a large number of values for ρ_1, or even by implementing some global search algorithms well suited for non-convex functions. We then provide some bounds enabling the measurement of the suboptimality of the procedure. We show that for typical industrial applications this heuristic can be expected to work extremely well, i.e. to provide a near optimal solution.

4.3.1 Definition of the basic subproblem

We focus first on the following sub-problem where the value of ρ_1 is fixed to a given numerical value α (clearly with $0 \leq \alpha \leq 1$). It is easy to see that for $\rho_1 = \alpha$, the problem (4.15)-(4.17) is rewritten as

$$Min_{\{\mathcal{X}\}} \qquad Z(\alpha) = \sum_{i \in \mathcal{C}} \left[\hat{Z}_{i,2}(\alpha) + X_{i,1} \Big(\hat{Z}_{i,1}(\alpha) - \hat{Z}_{i,2}(\alpha) \Big) \right] \tag{4.18}$$

$$\text{s.t.} \qquad \sum_{i \in \mathcal{C}} X_{i,1} \frac{\lambda_i}{\mu} = \alpha, \tag{4.19}$$

$$X_{i,1} \in \{0,1\}, \quad \text{for } i \in \mathcal{C}. \tag{4.20}$$

The constraint problem (4.18)-(4.20) is a binary linear program which can be interpreted as a special case of *knapsack* problem and efficient heuristics for solving this exist in the literature [9]. Unfortunately, for a given α value the feasibility of the problem seems tedious, i.e. it is absolutely not clear that one can exhibit integer solutions which satisfy constraint (4.19). Basically, the approximate solution procedure will be in two steps. First, for the given α value, one considers a relaxed real version of the problem (4.18)-(4.20). Second, one rounds the real solution to an integer solution. This integer solution is automatically feasible for a certain $\tilde{\alpha}$ value, computed from equation (4.19).

4.3.2 Analysis and solution of the relaxed subproblem

Let's consider the following relaxed problem

$$\begin{aligned} Min_{\{\mathcal{X}^r\}} \qquad & Z_{\{\mathcal{X}^r\}}(\alpha) = \sum_{i \in \mathcal{C}} \Big[\hat{Z}_{i,2}(\alpha) + X^r_{i,1} \big[\hat{Z}_{i,1}(\alpha) - \hat{Z}_{i,2}(\alpha) \big] \Big] && (4.21) \\ \text{s.t.} \qquad & \sum_{i \in \mathcal{C}} X^r_{i,1} \frac{\lambda_i}{\mu} = \alpha, && (4.22) \\ & X^r_{i,1} \in [0,1], \quad \text{for } i \in \mathcal{C}, && (4.23) \end{aligned}$$

with $\mathcal{X}^r = X^r_{1,1}, X^r_{2,1}, ..., X^r_{k,1}$.

Property 4: For any α value, with $0 \le \alpha \le 1$, $\mathcal{X}^{r*}(\alpha)$, the optimal solution of the relaxed problem (4.21)-(4.23) is a lower bound solution for the original problem (2.3)-(2.7).

This property will be useful in the heuristic solution procedure. Indeed, by implementing some global search algorithms well suited for non-convex function, a global lower bound for the optimal solution will be determined.

For most parameters, the optimal solution of the relaxed problem (4.21)-(4.23) is non-integer. Clearly, for the exceptional cases where this solution should be integer, i.e. $X^{r*}_{i,1}(\alpha) \in \{0,1\}$, it would also be the optimal solution of the initial problem (4.18)-(4.20).

Property 5: if the optimal solution $\mathcal{X}^{r*}(\alpha)$ of the relaxed problem (4.21)-(4.23) is not integer, then a unique component $i \in \mathcal{C} : X^{r*}_{i,1}(\alpha) \notin \{0,1\}$ exists and the other components $X^{r*}_{j,1}(\alpha)$, for $j \neq i$, are integer. *See [9].*

This property is important in that it demonstrates that the relaxed problem has a particular structure : the solution of this relaxed problem is nearly an integer solution. As a consequence, a simple rounding procedure should be very efficient and provide near-optimal solutions, especially if the number of products is large (as in typical industrial problems).

Corollary: For any α value, let us then define $\tilde{\alpha}(\alpha)$ as

$$\tilde{\alpha}(\alpha) = \sum_{i \in \mathcal{C} : X^{r*}_{i,1}(\alpha)=1} \frac{\lambda_i}{\mu}.$$

The restriction of $X^{r*}(\alpha)$ to integer values, i.e. to the vector $(\lfloor X^{r*}_{i,j}(\alpha) \rfloor)_{i \in \mathcal{C}, j \in \{1,2\}}$, is a feasible solution for the initial problem (4.18)-(4.20) with $\tilde{\alpha}(\alpha)$. The value of associated cost is thus an upper bound for the optimal cost function (2.3).

This corollary will be used in order to build an upper bound in the estimation procedure.

4.3.3 A heuristic solution procedure for the binary linear subproblem

For a given α value, the relaxed subproblem (4.21)-(4.23) exhibits a very simple structure. It is known that, instead of using a linear programming brute force approach, the optimal solution can be obtained by ordering item indices by decreasing values of the indicators

$$\frac{\hat{Z}_{i,2}(\alpha) - \hat{Z}_{i,1}(\alpha)}{\lambda_i}.$$

Let us define $\mathcal{L}(\alpha)$ the ordered list associated with α, where $\mathcal{L}(\alpha, i)$, with $i \in \mathcal{C}$, is interpreted as the i^{th}-product index ordered in decreasing values of the ratios

$$\frac{\hat{Z}_{i,2}(\alpha) - \hat{Z}_{i,1}(\alpha)}{\lambda_i}.$$

The solution procedure consists in setting the decision variables $X_{i,1}$ are equal to 1 for the first indices in the list as long as $\sum_i X_{i,1}\lambda_i/\mu \leq \alpha$ (see [9] for more details). At the end of the process, a variable will have to be set to a non-integer value in such a way that equation (4.22) is satisfied.

4.3.4 A heuristic solution procedure for the original problem

The optimization procedure for the general problem (2.3)-(2.7) which is based on the relaxed problem can then be summarized as follows:

```
Data: $\mu, k$, $\Delta\alpha$ and, for all $i \in \mathcal{C}$, $\lambda_i, L_i, h_i, \gamma_i^c$.
Initialization: Let the optimal cost upper bound $\overline{Z^*} = +\infty$ and the optimal cost lower bound $\underline{Z^*} = -\infty$.
For $\alpha = 0$ to $\frac{\lambda}{\mu}$ step $\Delta\alpha$ do:

    1- Solution of the relaxed subproblem (4.21)-(4.23) and lower bound computation:
        ▷ Let $\hat{\alpha}(\alpha)$=0.
        ▷ For $i = 1, ..., k$, do
            If $\hat{\alpha}(\alpha) + \frac{\lambda_{\mathcal{L}(\alpha,i)}}{\mu} \leq \alpha$ then $\mathcal{Y}^r_{\mathcal{L}(\alpha,i),1} = 1$ and $\hat{\alpha}(\alpha) = \hat{\alpha}(\alpha) + \lambda_{\mathcal{L}(\alpha,i)}$,
                                   else $\mathcal{Y}^r_{\mathcal{L}(\alpha,i),1} = \alpha - \hat{\alpha}(\alpha)$,
                        for $j = i+1, ..., k$, do $\mathcal{Y}^r_{\mathcal{L}(\alpha,j),1} = 0$.
        ▷ For $i = 1, ..., k$, do $\mathcal{X}^r_{i,1}(\alpha) = \mathcal{Y}^r_{\mathcal{L}(\alpha,i),1}$.

          If $\sum_{i \in \mathcal{C}} [\hat{Z}_{i,2}(\alpha) + \mathcal{X}^r_{i,1}(\alpha)(\hat{Z}_{i,1}(\alpha) - \hat{Z}_{i,2}(\alpha)] > \underline{Z^*}$

                then $\underline{Z^*} = \sum_{i \in \mathcal{C}} [(\hat{Z}_{i,2}(\alpha) + \mathcal{X}^r_{i,1}(\alpha)(\hat{Z}_{i,1}(\alpha) - \hat{Z}_{i,2}(\alpha)))]$.

    2- Feasible solution for (4.18)-(4.20)
        ▷ For $i = 1, ..., k$, do $\tilde{X}_{i,1}(\hat{\alpha}(\alpha)) = \lfloor \tilde{Y}_{\mathcal{L}(\alpha,i),1} \rfloor$.

    3- Approximate optimal solution for (4.18)-(4.20) and upper bound computation:
          If $\sum_{i \in \mathcal{C}} [\hat{Z}_{i,2}(\hat{\alpha}(\alpha)) + \tilde{X}_{i,1}(\hat{\alpha}(\alpha))(\hat{Z}_{i,1}(\hat{\alpha}(\alpha)) - \hat{Z}_{i,2}(\hat{\alpha}(\alpha)))] < \overline{Z^*}$

          then for $i = 1, ..., k$, do $X^*_{i,1} = \tilde{X}_{i,1}(\hat{\alpha}(\alpha))$

              and $\overline{Z^*} = \sum_{i \in \mathcal{C}} [(\hat{Z}_{i,2}(\hat{\alpha}(\alpha)) + \tilde{X}_{i,1}(\hat{\alpha}(\alpha))(\hat{Z}_{i,1}(\hat{\alpha}(\alpha)) - \hat{Z}_{i,2}(\hat{\alpha}(\alpha)))]$.
    Next $\alpha$.
Results:
        ▷ $X^*_{i,1}$ (for all i $\in \mathcal{C}$) and $\alpha^* = \sum_{i \in \mathcal{C}} \frac{\lambda_i}{\mu} X^*_{i,1}$.
        ▷ Optimal cost upper bound $\overline{Z^*}$ and lower bound $\underline{Z^*}$.
        ▷ In addition, $s^*_{i,j}$ can be calculated by (4.12)-(4.13) for all $i \in \mathcal{C}$ and $j \in \mathcal{C}_{\mathcal{N}}$.
```

In summary, the main ideas underlying the heuristic and the bounds computations are the following. First, we reduce the optimization problem to the minimization of a univariate non-linear function which continuously depends on the parameter α. The optimal α value will be estimated via a classical one-dimensional search procedure (or more roughly via a simple grid on the interval $[0, 1]$). Second, for any α value, the relaxed non-integer solution is associated to a relaxed problem cost function which is a lower bound value for the cost objective function. As a consequence, the minimum for any α value of the relaxed objective cost function constitutes a lower bound for the original cost function objective (4.18). Third, any feasible integer solution induces a cost objective value which is an upper bound for the optimal solution (4.18). The optimal solution is approximated by the feasible solution with the lowest cost (i.e. which corresponds to the lowest upper cost).

5 Numerical studies

In this section, we first show efficiency of the proposed approach by a numerical analysis through a large number of randomly generated examples. Then, via several examples, we provide some managerial insight into the considered MTO/MTS choice and priority allocation strategy.

5.1 Heuristic efficiency: a numerical analysis

Logically, a key element explaining the efficiency of the proposed heuristic is the number of products having a significant relative workload. In order to test numerically the efficiency of the proposed heuristic as a function of this parameter, we have considered a large number of problems with randomly generated data and parameters. The numerical analysis plainly confirms that the heuristic will be efficient if the number of references having a significant workload both in the low priority set and in the high priority set are large.

The randomly generated parameters. For this numerical analysis, the various parameters (demand rates, holding cost rates, customer lead-times and fill rates) are randomly generated. This allows us to test the heuristic procedure with general, and unstructured, parameters. The numerical values for the parameters λ_i and h_i are uniformly randomly chosen, respectively in fixed intervals $[\lambda_{Min},\, \lambda_{Max}]$ and $[h_{Min},\, h_{Max}]$. The parameters γ_i^r are randomly chosen as γ^{c1}, γ^{c2} or γ^{c3} with a probability $\frac{1}{3}$ for each value. In order to generate problems of significant interest, it is clearly required that the numerical values of the parameters L_i not be too high. As a matter of fact, in such a situation (i.e. with large customer lead-times) the optimal solution structure is obvious: each product type i is produced according to an MTO mode, no matter the priority rule, and the associate inventory is zero. To determine values corresponding to interesting problems, L_i is assumed to be randomly fixed in the interval $[0, 1.1\,\tilde{L}_i]$. $\tilde{L}_i$ is the threshold value for L_i with which product type i can be produced under an MTO mode, with the assumption that it is the only product type with a low priority. By equation (4.13) it is easily seen (see [8]) that this threshold value is given by

$$\tilde{L}_i = \frac{-\ln(1-\gamma_i^r)}{\mu - \lambda(2-\rho) + \lambda_i(1-\rho)}.$$

The numerical values of these parameters are $\lambda_{Min} = 0.01$, $\lambda_{Max} = 1000$, $h_{Min} = 1$, $h_{Max} = 10$, $\gamma^{c1} = 0.95$, $\gamma^{c2} = 0.97$, $\gamma^{c3} = 0.99$.

The indicators. We consider several performance indicators in the analysis of the numerical results. First, we measure the mean and the standard deviation of the difference between the upper and the lower bounds for the optimal allocation costs. This difference is expressed in terms of a percentage of the upper bound costs value. This indicator is denoted as DIF_{up-low}. Then, we measure the mean and the standard deviation of the difference between the optimal allocation upper bound costs and the simple FIFO performance costs. Again, the difference is expressed in terms of a percentage of the FIFO performance costs value. This indicator is denoted as $DIF_{FIFO-up}$. The numerical results are given in Table 1.

This analysis confirms that the heuristic performs extremely well for significantly large k values, and in fact even for small k values. It is worth noting that near-optimal allocation is easily computed: with a 1,5GHZ (centrino) CPU and 500M RAM, the average computing run time is less than 3 seconds for the 100 items case and less than 20 seconds for the 1000 items case.

k	nbr of samples	% average$[DIF_{up-low}]$	% $\sigma[DIF_{up-low}]$	% average $[DIF_{FIFO-up}]$	% $\sigma[DIF_{FIFO-up}]$
10	10	32,0945	11,8391	12,4871	19,7193
10	100	24,4740	9,61899	23,0133	15,4345
10	1000	24,9263	10,8195	22,0312	16,9613
10	5000	24,5922	10,6070	22,7079	16,8053
10	10000	24,6884	10,6152	22,6956	16,7260
25	10	4,77869	3,43381	12,42700	6,62904
25	100	7,72487	6,06290	10,54090	10,75500
25	1000	7,78477	5,56865	9,70236	9,22648
25	5000	7,53498	5,50985	10,21540	9,01670
25	10000	7,59687	5,51156	9,99865	9,12569
50	10	0,53465	0,23777	17,08150	4,01117
50	100	0,51878	0,30174	15,71680	4,21199
50	1000	0,48984	0,39131	15,43090	4,43881
50	5000	0,52431	0,51299	15,64240	4,50850
50	10000	0,51963	0,53694	15,46310	4,50941
100	10	0,17263	0,08380	18,61360	3,25966
100	100	0,13683	0,09578	17,96200	3,44479
100	1000	0,14173	0,09915	18,56790	3,42680
100	5000	0,13647	0,09775	18,48460	3,32194
100	10000	0,13686	0,09625	18,52250	3,44849
250	10	0,01230	0,02160	22,90980	2,93407
250	100	0,01916	0,02679	23,05470	2,66771
250	1000	0,02086	0,27620	22,67270	2,72987
250	5000	0,02292	0,02846	22,82190	2,63421
250	10000	0,02231	0,02849	22,73670	2,63613
500	10	0,00260	0,00136	28,83340	2,54563
500	100	0,00199	0,00174	29,37930	2,36511
500	1000	0,00272	0,00521	29,05860	2,27550
500	5000	0,00248	0,00398	29,09470	2,36842
1000	10	0,00058	0,00044	39,19050	2,07630
1000	100	0,00060	0,00078	38,77840	2,00956
1000	1000	0,00069	0,00237	38,83270	2,01950

Table 1: Performance analysis: numerical analysis.

5.2 Qualitative insights for the optimal priority allocation decision

To provide qualitative insight into the structure of the optimal MTO/MTS decision and the corresponding optimal priority assignment, we consider here three particular examples. For each example, we compute $X^*_{i,1}$, $s^*_{i,j}$ and s_i^{FIFO} for all the product types and for several values of ρ, where s_i^{FIFO} is the optimal base stock level if a FIFO policy is used for all items (in other words $\rho_1 = 0$ and all items are low priority items). These results are displayed in three separate subtables.

As defined above, $X^*_{i,1} = 1$ means that it is optimal to give the product type i high priority while, $X^*_{i,1} = 0$ indicates that low priority must be given to this product. In the same way, $s^*_{i,j} = 0$ means that it is optimal under priority level j to produce the type i items with an MTO mode, while, $s^*_{i,j} > 0$ indicates that the product must be made with an MTS policy.

Preliminary remark : the non-smoothness of the optimal policies

It is important to remark that several discontinuous mechanisms are at work in the considered problem: first, the base-stocks are integers; second the priority allocation is discrete in that for a given item type, the entire workload is given the same priority level; and third an order cannot be delivered before the due date (corresponding to the customer lead-time) and, as a consequence, must be kept in inventory until this date if produced in advance. Such non-continuous mechanisms induce optimal policies that are non-smooth (slightly so for examples 1 and 2, and more significantly so for the third example). In order to exhibit clearly the origin of these discontinuities, we have considered auxiliary theoretical models in which some relaxations have been introduced. It can be seen that the most critical factor for this non-smoothness is the integrity of the base stocks. We display in figure 5, in appendix, the optimal policies for the relaxed model, in which the base-stocks and the priority allocation are assumed to be continuous. It can be observed that in this case the smoothness of the optimal policies is greatly improved.

Example 1: impact of the priority allocation for homogeneous products

In the first example, we consider a setting with $k = 50$ identical items. The numerical parameters are chosen as follows : $\lambda_i = 1$, $h_i = 1$, $\gamma_i^c = 0.95$, and $L_i = 0.2$, for $i \in \mathcal{C}$. The numerical results are given in Table 2 and can be summarized as follows.

For small ρ values (i.e. $\rho \in [0, 0.7]$), the whole system is MTO : the optimal base stock levels are equal to zero (either under a uniform FIFO rule, or under the optimal priority allocation which corresponds in this case to a uniform priority rule). For ρ values lying between $[0.8, 0.95]$, it can be seen that it is optimal to allocate priority to a large number of items, which can be produced according to MTO. A limited number of items are produced according to MTS, but with a relatively low base-stock level. In comparison, using a uniform FIFO rule requires the introduction of a strictly positive base-stock for every item (i.e. a base-stock at least equal to 1 according to the integrity nature of the problem). The inventory reduction associated with the optimal policy is thus significant with respect to FIFO. For very high ρ values (i.e. $\rho \in [0.95, 1]$), with the optimal priority allocation, only a small number of items are MTS (while they are all MTS under FIFO). However, it can be seen that the cumulated inventories are relatively equal for the two priority schemes.

Example 2: impact of the priority allocation when the inventory costs are non-homogeneous

We consider the same setting as in the first example except that the items are distinguished by the holding costs h_i which are assumed to be different from one to the next. We suppose that the items are ordered by increasing h_i values. For the example, we take $h_i = 0.039i$. The numerical results show that several mechanisms are at work in this case with varying holding cost rates, as appears in the basic formulas underlying the model. The optimal MTO/MTS decision and the corresponding optimal priority assignment are given in Table 3 and can be analyzed as follows.

First, for small ρ values (i.e. $\rho \in [0, 0.7]$), implementing an MTO mode for all the items is the starting point: indeed, the optimal base-stock levels are equal to zero under a uniform FIFO rule as well as under the optimal priority allocation. However, another interesting mechanism is at work in this case: when an item is produced before the due date, it is kept in inventory until the delivery at the due date, inducing inventory costs. The optimal priority allocation consists thus in giving high priority to the low inventory cost items and in giving low priority to the high inventory cost items. As a consequence the items susceptible being produced in advance (and kept in inventory) will be the low inventory cost items. For ρ values lying between $[0.8, 0.99]$, the MTO mode can only be implemented for items that receive high priority. The optimal priority allocation consists thus of giving priority to an increasing number of items as long as the priority effect remains efficient. Clearly, the items with a low priority level are produced under MTS, with the associated inventory costs. The optimal priority allocation consists thus in choosing the low inventory cost items for the MTS mode and the high inventory cost items for the MTO mode. Furthermore, it can be observed that the cumulated inventories cost difference between FIFO and the optimal priority allocation decreases for increasing ρ values. For very high ρ values (i.e. $\rho \in [0.999, 1]$), all items are MTS. However, it can be seen that the cumulated inventories are relatively equal for the two priority schemes.

Example 3: impact of the priority allocation when the demand rates and the customer lead-times are non-homogeneous

We consider now an example where the items are distinguished by their demand rates and by their customer lead times. Furthermore, we suppose that the admissible customer lead-times are decreasing functions of the demand rates, which is a frequent situation in practice. We assume that $\lambda_i = 0.039i$ and $L_i = 0.0078(50 - i + 1) + 0.09$.

The optimal MTO/MTS decision and the corresponding optimal priority assignment are given in Table 4.

Let us first consider the case of small ρ values (i.e. $\rho \in [0, 0.7]$). Globally, if one neglects the discontinuous effect mentioned earlier, the lead-time effect and the demand effect induce an identical priority allocation trend : high-demand rate and/or low customer lead-time items require a high priority level in order to be MTO (while low-demand rate and high customer lead-time items are MTO even with a low priority). Under the optimal priority allocation, it can be seen that all the items are managed under MTO which is not the case of FIFO. For ρ values lying between $[0.8, 0.999]$, the opposite scheme is at work. If one neglects the discontinuous effect mentioned earlier, the lead-time effect and the demand effect induce an identical priority allocation trend : high-demand rate and/or low customer lead-time items are MTS even with a high priority allocation (while low-demand rate and high customer lead-time items can be MTO if they are allocated a high priority level). Thus, the optimal priority allocation consists mainly in giving a high priority to the high to medium lead-time/low to medium demand rate items. It can be observed that this allocation is not entirely smooth following the item indices and some exceptions occur with respect to the global allocation rule.

The optimal priority allocation index (X*i,1)

							ρ							
0.5	0.6	0.7	0.8	0.9	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99	0.999
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	1	0	0	0	0	0	0	0
0	0	0	1	1	1	0	1	0	1	0	0	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	1	1	0
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1

The optimal base stock levels (the PR case)

							ρ							
0.5	0.6	0.7	0.8	0.9	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99	0.999
0	0	0	2	3	3	3	4	4	5	6	8	11	21	195
0	0	0	2	3	3	3	4	4	5	6	8	11	21	195
0	0	0	2	3	3	3	4	4	5	6	8	11	21	195
0	0	0	0	3	3	3	4	4	5	6	8	11	21	195
0	0	0	0	3	3	3	4	4	5	6	8	11	21	195
0	0	0	0	3	3	3	4	4	5	6	8	11	21	195
0	0	0	0	3	3	3	4	4	5	6	8	11	21	195
0	0	0	0	3	3	3	4	4	5	6	8	11	21	195
0	0	0	0	3	3	3	4	4	5	6	8	11	21	195
0	0	0	0	3	3	3	4	4	5	6	8	11	21	195
0	0	0	0	0	3	3	4	4	5	6	8	11	21	195
0	0	0	0	0	3	3	4	4	5	6	8	11	21	195
0	0	0	0	0	0	3	0	4	5	6	8	11	21	195
0	0	0	0	0	0	3	0	4	0	6	8	11	21	195
0	0	0	0	0	0	0	0	0	0	0	0	11	21	195
0	0	0	0	0	0	0	0	0	0	0	0	0	0	195
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The optimal base stock levels (the FIFO case)

							ρ							
0.5	0.6	0.7	0.8	0.9	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99	0.999
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
0	0	0	1	2	2	2	2	2	2	3	3	4	8	62

Table 2 : Numerical results for Example 1

	The optimal priority allocation index (X*i,1)															The optimal base stock levels (the PR case)															The optimal base stock levels (the FIFO case)														
				ρ															ρ															ρ											
Product	0.5	0.6	0.7	0.8	0.9	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99	0.999	0.5	0.6	0.7	0.8	0.9	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99	0.999	0.5	0.6	0.7	0.8	0.9	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99	0.999
1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	3	3	4	4	5	5	6	8	11	21	1997	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	3	3	4	4	5	5	6	8	11	21	1997	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
3	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	2	3	3	4	4	5	5	6	8	11	21	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
4	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	3	3	4	4	5	5	6	8	11	21	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
5	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	3	3	4	4	5	5	6	8	11	21	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
6	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	3	3	4	4	5	5	6	8	11	21	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
7	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	3	3	4	4	5	5	6	8	11	21	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
8	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	3	3	4	4	5	5	6	8	11	21	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
9	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	3	3	4	4	5	5	6	8	11	21	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
10	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	3	3	4	4	5	5	6	8	11	21	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
11	1	1	1	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	3	4	4	5	5	6	8	11	21	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
12	1	1	1	1	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	3	0	4	5	5	6	8	11	21	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
13	1	1	1	1	1	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	5	6	8	11	21	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
14	1	1	1	1	1	1	1	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	6	8	11	21	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
15	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	11	21	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
16	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
17	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
18	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
19	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
20	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
21	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
22	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
23	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
24	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
25	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
26	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
27	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
28	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
29	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
30	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
31	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
32	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
33	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
34	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
35	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
36	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
37	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
38	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
39	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
40	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
41	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
42	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
43	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
44	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
45	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
46	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
47	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
48	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
49	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62
50	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1	2	2	2	2	2	2	3	3	4	8	62

Table 3 : Numerical results for Example 2

The optimal priority allocation index (X*i,1)

	ρ														
Product	0.5	0.6	0.7	0.8	0.9	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99	0.999
1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
2	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
3	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
4	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
5	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
6	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
7	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
8	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
9	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
10	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
11	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
12	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
13	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
14	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
15	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
16	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
17	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
18	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
19	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
20	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
21	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
22	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
23	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
24	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
25	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
26	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
27	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
28	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
29	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
30	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
31	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1
32	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1
33	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
34	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1
35	0	1	1	1	1	0	1	1	1	1	1	1	1	1	1
36	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1
37	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
38	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
39	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
40	1	1	1	1	1	1	1	0	0	0	0	0	1	1	1
41	1	1	1	1	1	1	1	0	0	0	1	0	1	0	0
42	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
43	1	1	1	1	0	1	0	1	0	0	0	0	0	0	0
44	1	1	1	1	0	0	0	1	1	1	0	1	0	0	0
45	1	1	1	1	0	0	0	0	1	0	0	0	0	0	0
46	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0
47	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0
48	1	1	1	0	0	0	0	0	0	0	0	1	0	0	0
49	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
50	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

The optimal base stock levels (the PR case)

	ρ														
Product	0.5	0.6	0.7	0.8	0.9	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99	0.999
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
27	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
34	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0
35	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0
36	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0
37	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
40	0	0	0	0	0	0	0	5	6	7	8	11	1	0	1
41	0	0	0	0	0	0	0	5	6	7	1	11	1	28	286
42	0	0	0	0	0	0	1	1	6	7	9	11	16	29	293
43	0	0	0	0	4	0	5	1	6	7	9	11	16	30	300
44	0	0	0	0	4	4	5	1	1	1	9	1	17	30	307
45	0	0	0	0	4	4	5	6	1	8	9	12	17	31	314
46	0	0	0	2	4	4	5	6	1	8	9	12	17	32	321
47	0	0	0	2	4	4	5	6	7	8	1	12	18	32	328
48	0	0	0	2	4	4	5	6	7	8	10	1	18	33	335
49	0	0	0	2	4	4	5	6	7	8	10	13	18	34	342
50	0	0	0	1	3	3	4	5	6	0	9	12	18	34	348

The optimal base stock levels (the FIFO case)

	ρ														
Product	0.5	0.6	0.7	0.8	0.9	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99	0.999
1	0	0	0	0	1	1	1	1	1	1	1	1	1	2	4
2	0	0	0	0	1	1	1	1	1	1	1	1	1	2	7
3	0	0	0	0	1	1	1	1	1	1	1	1	2	2	9
4	0	0	0	0	1	1	1	1	1	1	1	1	2	2	11
5	0	0	0	0	1	1	1	1	1	1	1	2	2	3	14
6	0	0	0	0	1	1	1	1	1	1	1	2	2	3	16
7	0	0	0	0	1	1	1	1	1	1	1	2	2	3	18
8	0	0	0	0	1	1	1	1	1	1	2	2	2	3	21
9	0	0	0	0	1	1	1	1	1	1	2	2	2	4	23
10	0	0	0	0	1	1	1	1	1	1	2	2	3	4	25
11	0	0	0	0	1	1	1	1	1	2	2	2	3	4	28
12	0	0	0	0	1	1	1	1	1	2	2	2	3	4	30
13	0	0	0	0	1	1	1	1	1	2	2	2	3	5	32
14	0	0	0	0	1	1	1	1	1	2	2	2	3	5	35
15	0	0	0	0	1	1	1	1	1	2	2	2	3	5	37
16	0	0	0	0	1	1	1	1	2	2	2	2	3	5	39
17	0	0	0	0	1	1	1	1	2	2	2	3	3	6	42
18	0	0	0	0	1	1	1	1	2	2	2	3	3	6	44
19	0	0	0	0	1	1	1	1	2	2	2	3	4	6	46
20	0	0	0	0	1	1	1	2	2	2	2	3	4	6	49
21	0	0	0	0	1	1	1	2	2	2	2	3	4	6	51
22	0	0	0	0	1	1	1	2	2	2	2	3	4	7	53
23	0	0	0	0	1	1	1	2	2	2	2	3	4	7	56
24	0	0	0	0	1	1	2	2	2	2	3	3	4	7	58
25	0	0	0	0	1	1	2	2	2	2	3	3	4	7	60
26	0	0	0	0	1	1	2	2	2	2	3	3	4	8	63
27	0	0	0	0	1	1	2	2	2	2	3	3	5	8	65
28	0	0	0	0	1	2	2	2	2	2	3	3	5	8	67
29	0	0	0	0	1	2	2	2	2	2	3	4	5	8	70
30	0	0	0	0	1	2	2	2	2	3	3	4	5	9	72
31	0	0	0	1	1	2	2	2	2	3	3	4	5	9	74
32	0	0	0	1	2	2	2	2	2	3	3	4	5	9	77
33	0	0	0	1	2	2	2	2	2	3	3	4	5	9	79
34	0	0	0	1	2	2	2	2	2	3	3	4	5	9	82
35	0	0	0	1	2	2	2	2	2	3	3	4	6	10	84
36	0	0	0	1	2	2	2	2	2	3	3	4	6	10	86
37	0	0	0	1	2	2	2	2	3	3	3	4	6	10	89
38	0	0	0	1	2	2	2	2	3	3	3	4	6	10	91
39	0	0	0	1	2	2	2	2	3	3	4	4	6	11	93
40	0	0	0	1	2	2	2	2	3	3	4	4	6	11	96
41	0	0	0	1	2	2	2	2	3	3	4	5	6	11	98
42	0	0	0	1	2	2	2	3	3	3	4	5	6	11	100
43	0	0	0	1	2	2	2	3	3	3	4	5	7	12	103
44	0	0	1	1	2	2	2	3	3	3	4	5	7	12	105
45	0	0	1	1	2	2	2	3	3	3	4	5	7	12	107
46	0	0	1	1	2	2	3	3	3	3	4	5	7	12	110
47	0	0	1	1	2	2	3	3	3	4	4	5	7	13	112
48	0	0	1	1	2	2	3	3	3	4	4	5	7	13	114
49	0	0	1	1	2	2	3	3	3	4	4	5	7	13	117
50	0	0	0	0	1	1	1	2	2	3	3	4	6	13	118

Table 4 : Numerical results for Example 3

6 The case of a general number of priority indices

We propose in this section a generalization of the preceding results to the case of multi-level priority. We show, via numerical examples, that for the considered setting, two or three priority classes permit near-optimal performances.

6.1 Approximate performance analysis for the general case

We consider here a model with a N priority indices. The decomposition approach proceeds as follows. First, as in the two-priority case (i.e. the case with $N = 2$), under the preemptive assumption, it is clear that the production flow of the high priority orders (i.e. the items i with $i \in \mathcal{C}_1$) is not affected by the orders with a lower priority level (i.e. the items i with $i \in \mathcal{C}_2, \mathcal{C}_3, ...\mathcal{C}_N$). This family of items can thus be analyzed in the same way as for the two-priority case in section 4.

Let us now consider a general family of items corresponding to priority p (with $1 < p \leq N$). First, under the preemptive assumption, it is clear that the production flow of these p-priority orders (i.e. the items i with $i \in \mathcal{C}_p$) is not affected by the orders with a lower priority level (i.e. the items i with $i \in \mathcal{C}_{p+1}, \mathcal{C}_{p+2}, ...\mathcal{C}_N$). One only has to consider the interactions of these p-priority orders with the orders with a higher priority level (i.e. the items i with $i \in \mathcal{C}_1, \mathcal{C}_2, ...\mathcal{C}_{p-1}$).

As p-priority orders are preempted each time a higher priority level order is issued, the p-priority class can thus be viewed as the low priority class in a fictitious two-priority level system where the high priority family is composed of the aggregation of all the higher priority items (i.e. the items i with $i \in \mathcal{C}_1, \mathcal{C}_2, ...\mathcal{C}_{p-1}$).

In summary, if one tries to compare this with the two-priority case analyzed in section 4, the p-priority items have to be simultaneously analyzed as the high priority items with respect to items $i \in \mathcal{C}_{p+1}, \mathcal{C}_{p+2}, ...\mathcal{C}_N$ and as low priority items with respect to items $i \in \mathcal{C}_1, \mathcal{C}_2, ...\mathcal{C}_{p-1}$.

First, we introduce utilization factors for the different priority classes

$$\rho_p(\mathcal{X}) = \sum_{m \in \mathcal{C}} X_{m,p} \lambda_m / \mu, \tag{6.1}$$

with the property that

$$\rho = \sum_{p=1}^{N} \rho_p(\mathcal{X}). \tag{6.2}$$

By applying equations from the two-priority case to the decomposition scheme explained above, for given numerical values of s_i and ρ_j, the fill rate and the cost for item i can be found, for $p = 2, ..., N$, via equations (3.13)-(3.14) and amount to

$$\gamma_{i,p}(s_i, \rho_1(\mathcal{X}), ..., \rho_{p-1}(\mathcal{X}), \rho_p(\mathcal{X})) \cong$$
$$1 - \left[\frac{\lambda_i}{\mu(1 - \sum\limits_{j=1}^{p} \rho_j(\mathcal{X}))(1 - \sum\limits_{j=1}^{p-1} \rho_j(\mathcal{X})) + \lambda_i} \right]^{s_i} e^{-\mu(1 - \sum\limits_{j=1}^{p} \rho_j(\mathcal{X}))(1 - \sum\limits_{j=1}^{p-1} \rho_j(\mathcal{X}))L_i} \tag{6.3}$$

and

$$Z_{i,p}(s_i, \rho_1(\mathcal{X}), ..., \rho_{p-1}(\mathcal{X}), \rho_p(\mathcal{X})) \cong h_i\Bigg(s_i + \lambda_i\, L_i - \frac{\lambda_i}{\mu(1 - \sum\limits_{j=1}^{p} \rho_j(\mathcal{X}))(1 - \sum\limits_{j=1}^{p-1} \rho_j(\mathcal{X}))}$$
$$\Bigg[1 - \Bigg(\frac{\lambda_i}{\mu(1 - \sum\limits_{j=1}^{p} \rho_j(\mathcal{X}))(1 - \sum\limits_{j=1}^{p-1} \rho_j(\mathcal{X})) + \lambda_i}\Bigg)^{s_i} e^{-\mu(1-\sum\limits_{j=1}^{p} \rho_j(\mathcal{X}))(1-\sum\limits_{j=1}^{p-1} \rho_j(\mathcal{X}))L_i}\Bigg]\Bigg). \tag{6.4}$$

For given values $\rho_1, \rho_2, ..., \rho_N$, (i.e. in fact for a given priority scheme for the items), it is possible to compute analytically the optimal base-stocks $s^*_{i,j}$ guaranteeing condition (2.4), for product type i with a priority level j, by the formulas

$$s^*_{i,1}(\rho_1) \quad = \quad \max\left(\left\lceil \frac{\mu(1-\rho_1)L_i + \ln(1-\gamma_i^r)}{\ln\left(\frac{\lambda_i}{\mu(1-\rho_1)+\lambda_i}\right)} \right\rceil\ ,\ 0\right), \tag{6.5}$$

and for $p = 2, ..., N$,

$$s^*_{i,p}(\rho_1, ..., \rho_p) = \max\left(\left\lceil \frac{(\mu(1 - \sum\limits_{j=1}^{p} \rho_j)(1 - \sum\limits_{j=1}^{p-1} \rho_j) - \lambda)L_i + \ln(1 - \gamma_i^r)}{\ln\left(\frac{\lambda_i}{\mu(1-\sum\limits_{j=1}^{p} \rho_j)(1-\sum\limits_{j=1}^{p-1} \rho_j)-\lambda+\lambda_i}\right)} \right\rceil\ ,\ 0\right). \tag{6.6}$$

6.2 Problem reformulation with a general priority indices number

Along the lines of section 4, by exploiting the above formulas and by eliminating some variables by substitution, for given values $\rho_1, \rho_2, ..., \rho_N$, the initial problem (2.3)-(2.6) can thus be replaced by

$$Min_{\{\mathcal{X}\}} \qquad Z = \sum_{i\in\mathcal{C}} \sum_{j=1}^{N} \Big[X_{i,j} Z_{i,j}(s^*_{i,j}(\rho_1, ..., \rho_j), \rho_1, ..., \rho_j) \Big] \tag{6.7}$$

$$\text{s.t.} \qquad \sum_{i\in\mathcal{C}} X_{i,j} \frac{\lambda_i}{\mu} = \rho_j, \quad j = 1, ..., N, \tag{6.8}$$

$$\sum_{j=1}^{N} X_{i,j} = 1, \quad \forall\ i \in \mathcal{C}, \tag{6.9}$$

$$X_{i,j} \in \{0, 1\}, \quad \forall\ i \in \mathcal{C}, j = 1, ..., N. \tag{6.10}$$

The solution procedure is quite similar to the procedure developed in section 4 for the two-priority case. Note that for given numerical values of $\rho_1, \rho_2, ..., \rho_N$, the problem (6.7)-(6.10) becomes linear in the decision variables, as in the two-priority setting given in section 4. Clearly, this problem encounters the same feasibility question for given factors $\rho_1, \rho_2, ..., \rho_N$ as in the section 4 setting. We thus have to rely on a relaxation of the problem to real variables to guarantee existence of solutions. Then, an approximate integer optimal solution is obtained via a heuristic based on a rounding procedure applied to the non-integer solution. In this way, by considering a large number of values for $\rho_1, \rho_2, ..., \rho_N$, or even by implementing some global search algorithms well suited for non-convex functions, the optimal solution can be approximated. Clearly, the optimal solution of the relaxed problem can then be considered as a lower bound of the original problem.

6.3 Numerical studies

To study the structure of the optimal MTO/MTS decision and the corresponding optimal priority assignment as a function of N, i.e. the number of priority levels, we consider here an example with $k = 100$ items, with $\lambda_i = 0.1i$, $h_i = 1$, $\gamma_i^c = 0.95$ and $\rho = 0.95$. The customer lead times $L_i = L$ are equal for all product types. The curves exhibited in figure 6.3 give the optimal cost (corresponding to the optimal MTO/MTS decision and the corresponding optimal priority assignment) for different ρ values. This numerical example essentially confirms that even with a small number of priority classes (i.e. $N = 2, 3$), near-optimal performances are obtained for typical problems.

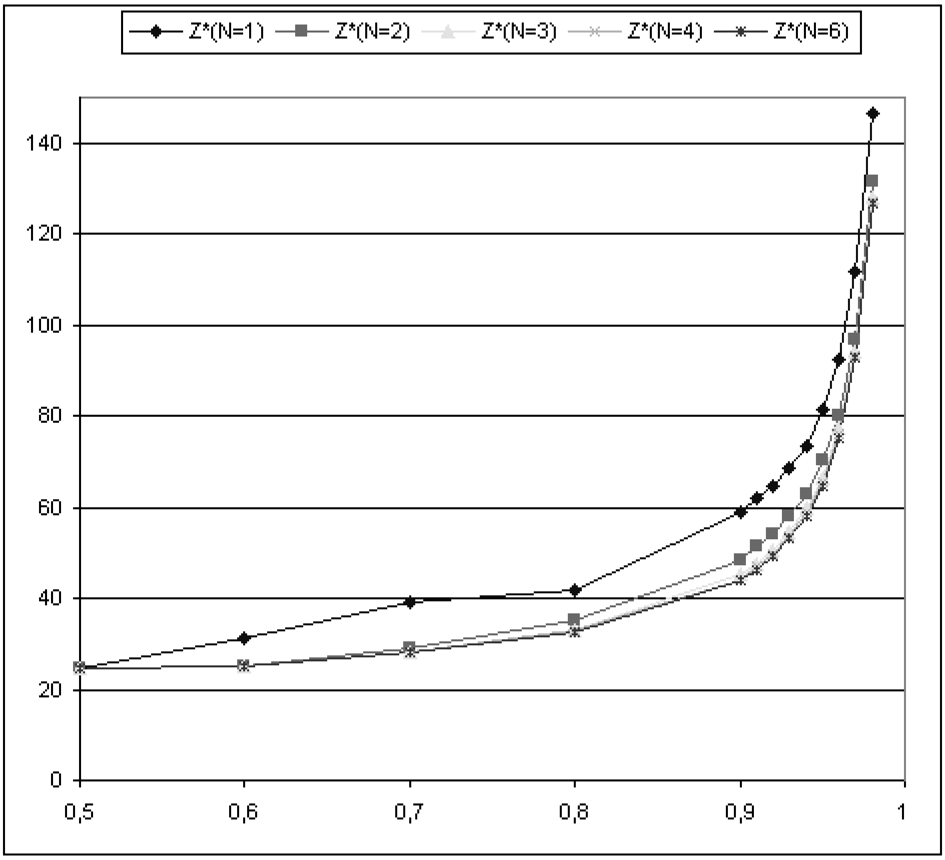

Figure 6.3 : General number of priority indices : numerical results.

7 Conclusion

This paper has analyzed priority allocation in a finite single-stage manufacturing facility producing multiple heterogeneous items. The production orders are issued according to a base-stock level policy. We have provided a heuristic optimization procedure to allocate priority to each product, in the setting of unstructured large-scale problems. Such problems lead to complex large-scale integer non-linear programs. We show how to exploit some specific properties in order to end with a tractable optimization problem. Through numerical examples we have illustrated these results and shown that the potential benefit of an optimal policy appears to be significant for practical problems. It is worth noting that the potential benefit can be expected to be even higher in case of fixed inventory costs linked to the presence/absence of inventory.

The extension of the approach to the consideration of robustness, with respect to some uncertainty on the parameters values, is ongoing research.

References

[1] Arreola-Risa A. and G.A. DeCroix, "Make-to-order versus Make-to-stock in a production-inventory system with general production times",*IIE Transactions*, (1998), 705-713.

[2] Buzacott J.A. and J.G. Shanthikumar, "Stochastic models of manufacturing systems", Prentice Hall,(1993).

[3] Carr S.and I. Duenyas.,"Optimal admission control and sequencing in a Make-to-stock/Make-to-order production system", *Operations Research*, (2000), 1–27.

[4] Dantzig G.B. "Discrete variable extremum problems". *Operations Research* 5, (1957), 266-277.

[5] Federgruen A. and Z. Katalan, "The impact of adding a MTO item to a MTS production system", *Management Sciences*, 45(7), (1999), 980-994.

[6] Federgruen A. and Z. Katalan, "Determining Production Schedules under Base-Stock Policies in Single Facility Multi-Item Production Systems", *Operations Research*, 46(6), (1998), 883-898.

[7] Ha A. "Optimal dynamic scheduling policy for a Make to Stock production system". *Operations Research* 45, (1997), 42-53.

[8] Hadj Youssef K., Ch van Delft and Y. Dallery,"Efficient scheduling rules in a combined make-to-stock and make-to-order manufacturing system", *Annals of Operations Research*, 126, (2004), 103-134.

[9] Kellerer H., U. Pferschy, and D. Pisinger, "Knapsack Problems", *Springer*, (2003).

[10] Liberopoulos G. and Y. Dallery,"A unified framework for pull control mechanisms in multi-stage manufacturing systems", *Annals of Operations Research*, (2000), 325-355.

[11] Perez A. and P. Zipkin,"Dynamic scheduling rules for a multi-product make-to-stock queue", *Operations Research*, 45(6),(1997), 919-930.

[12] S. Rajagopalan,"Make to Order or Make to Stock : Model and Application", *Management Sciences*, 48(2),(2002), 241-256.

[13] Soman C.A, D.P. Van Donk and G. Gaalman, "Combining make to order and make to stock in a food production system", *Working paper*, (2000).

[14] Sox C., L.J. Thomas, J.O. McClain, "Coordinating production and inventory to improve service", *Management Sciences*, 43(9),(1997), 1189-1197.

[15] A. Sleptchenko, M.C. van der Heijden and A. van Harten,"Using repair priorities to reduce stock investment in spare part networks", *European Journal of Operational Research*, 45(6),(1997), 919-930.

[16] Veatch M.H and L.M. Wein, "Scheduling a make to stock queue: index policies and hedging points", *Operations Research*, 44 (163),(2005),733-750.

[17] Williams T.M, "Special products and uncertainty in production/inventory systems", *European Journal of Operations Research*, 15,(1984),46-54.

[18] Zipkin P., "Performance analysis of a multi-item production-inventory system under alternative policies", *Management Sciences*, 38(2),(1995),182-197.

8 Appendix

Appendix 1 : proof of property 1. Let us consider, for $t \geq 0$, a general sample path P, for the considered time-lagged BSCS, with the corresponding variables $N_{P,i}(t)$, $I_{P,i}(s_i, t)$, $Y_{P,i}(s_i, t)$ and $D_{P,i}(s_i, t)$, with $N_{P,i}(t)$ the number of type-i parts in progress in the system (i.e. awaiting parts and work-in-process). For this sample path, let's introduce the following notations :

- $A_{P,i}(t)$: the corresponding total number of type-i demands arrived in the time interval $[0, t]$, which, by definition, have to be delivered in the time interval $[L_i, t + L_i]$,
- $B_{P,i}(t)$: the total number of type-i parts produced in the time interval $[0, t]$,
- $tA_{P,i,n}$: the arrival time of the n^{th} order of type-i products,
- $tB_{P,i,n}$: the completion time of the n^{th} order of type-i products,
- and $D_{P,i,n}$: the backorder time for of the n^{th} demand of type-i products.

By definition, along any sample path P, we have

$$\begin{aligned}
N_{P,i}(t) &= A_{P,i}(t) - B_{P,i}(t), && (8.1)\\
I_{P,i}(s_i,t) &= Max\{0, s_i + B_{P,i}(t) - A_{P,i}(t-L_i)\}\\
&= Max\{0, s_i - N_{P,i}(t) + (A_{P,i}(t) - A_{P,i}(t-L_i)\}, && (8.2)\\
Y_{P,i}(s_i,t) &= Max\{0, A_{P,i}(t-L_i) - [s_i + B_{P,i}(t)]\}\\
&= Max\{0, N_{P,i}(t) - s_i - (A_{P,i}(t) - A_{P,i}(t-L_i))\}, && (8.3)\\
D_{P,i,n}(s_i) &= Max\{0, tB_{_{P,i,n-s_i}} - [tA_{_{P,i,n}} + L_i]\}. && (8.4)
\end{aligned}$$

Note first that $A_{P,i}(t)$ and $tA_{P,i,n}$ depend exclusively on the product type i arrival process, and not on other parameters. Furthermore, due to the control policy, $B_{P,i}(t)$ and $tB_{_{P,i,n-s_i}}$ depend exclusively on the different arrival processes for all the products (which directly trigger the production orders) and on the production process (which realizes these orders). Thus, under the mild assumption of the existence of steady-state probability density functions for $N_i(.)$, $I_i(s_i,.)$, $Y_i(s_i,.)$ and $D_i(s_i)$, Property 1 is directly completed by equations (8.1)-(8.4).

Appendix 2 : proof of property 2. By definition of the BSCS model, the dynamics of order arrival processes and of the order production process are independent of the time-lag structure. As a consequence, the state variables N_i, the number of type-i parts in progress in the system (i.e. awaiting parts and work-in-process), and in fact the associate waiting or sojourn times), can be characterized as in [8] where a model without time lags has been studied in its entirety. In particular, it has been shown that a decomposition property under which the state variables related to each product families can be separately analyzed holds true. The probability distribution of the state variables N_i is explicitly given by

$$Prob\{N_i = n\} = (1-\rho_i)\rho_i^n \text{ for } n = 0, 1, 2, \ldots$$

with

$$\rho_i = \frac{\lambda_i}{\mu_i} \text{ and } \mu_i = \mu - \sum_{j \neq i} \lambda_j. \tag{8.5}$$

Now, according to (8.1)-(8.4), as for each product type-i, the corresponding variables X_i, Y_i, D_i and the fill rate γ_i depend on the variable N_i (and in fact on the associated waiting or sojourn times) and on the parameters s_i and L_i, the multi-product time-lag queuing model can be decomposed into independent single product models.

Appendix 3 : proof of property 3. In ([2], p106), it has been proved that one has

$$Pr\{\tilde{D}_i > d\} = \rho_i^{s_i}\, e^{-(\mu_i - \lambda_i)d},$$

where $\tilde{D}_i$ is the backordered time for a zero-time lag model (i.e. if L_i =0). Now, we define

- $t\tilde{A}_{_{i,n}}$: the random arrival time of the n^{th} order of type-i products (if $L_i = 0$),
- $t\tilde{B}_{_{i,n}}$: the random completion time of the nth order of type-i products (if $L_i = 0$),

- $\tilde{D}_{i,n}$: the random backorder time for of the n^{th} order of type-i products (if $L_i = 0$).

By definition, one has

$$\tilde{D}_{i,n} = Max\{0, t\tilde{B}_{i,n-s_i} - t\tilde{A}_{i,n}\}$$

and

$$Pr\{\tilde{D}_{i,n} > d\} = Pr\{t\tilde{B}_{i,n-s_i} - t\tilde{A}_{i,n} > d\}.$$

As $tA_{i,n}$ and $tB_{i,n}$ are independent from the time lag L_i, one has

$$\Pr\{D_{i,n} > d\} = \Pr\{t\tilde{B}_{i,n-s_i} - t\tilde{A}_{i,n} - L_i > d\} = \Pr\{\ \tilde{D}_{i,n} > d + L_i\}$$

and

$$\Pr\ \{D_i > d\} = \rho_i^{s_i}\, e^{-(\mu_i - \lambda_i)(L_i + d)}.$$

The optimal priority allocation index (X*i,1)

				ρ											
Product	0.5	0.6	0.7	0.8	0.9	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99	0.999
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0,5	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	1	0,56	0	0	0	0	0	0	0	0	0	0
11	0	0	0	1	1	0,95	0,35	0	0	0	0	0	0	0	0
12	0	0	0	1	1	1	1	0,76	0,19	0	0	0	0	0	0
13	0	0	0	1	1	1	1	1	1	0,68	0,14	0	0	0	0
14	0	0	0	1	1	1	1	1	1	1	1	0,6	0,07	0	0
15	0	0	0	1	1	1	1	1	1	1	1	1	1	0,56	0,085
16	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
17	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
18	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
19	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
20	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
21	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
22	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
23	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
24	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
25	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
26	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
27	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
28	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
29	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
30	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
31	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
32	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
33	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
34	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
35	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
36	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
37	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
38	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
39	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
40	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
41	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
42	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
43	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
44	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
45	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
46	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
47	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
48	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
49	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
50	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1

The optimal base stock levels (the PR case)

				ρ											
Product	0.5	0.6	0.7	0.8	0.9	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99	0.999
1	0	0	0	1,73	2,94	3,19	3,5	3,88	4,39	5,1	6,14	7,84	11,22	21,29	201,48
2	0	0	0	1,73	2,94	3,19	3,5	3,88	4,39	5,1	6,14	7,84	11,22	21,29	201,48
3	0	0	0	0,86	2,94	3,19	3,5	3,88	4,39	5,1	6,14	7,84	11,22	21,29	201,48
4	0	0	0	0	2,94	3,19	3,5	3,88	4,39	5,1	6,14	7,84	11,22	21,29	201,48
5	0	0	0	0	2,94	3,19	3,5	3,88	4,39	5,1	6,14	7,84	11,22	21,29	201,48
6	0	0	0	0	2,94	3,19	3,5	3,88	4,39	5,1	6,14	7,84	11,22	21,29	201,48
7	0	0	0	0	2,94	3,19	3,5	3,88	4,39	5,1	6,14	7,84	11,22	21,29	201,48
8	0	0	0	0	2,94	3,19	3,5	3,88	4,39	5,1	6,14	7,84	11,22	21,29	201,48
9	0	0	0	0	2,94	3,19	3,5	3,88	4,39	5,1	6,14	7,84	11,22	21,29	201,48
10	0	0	0	0	1,31	3,19	3,5	3,88	4,39	5,1	6,14	7,84	11,22	21,29	201,48
11	0	0	0	0	0	0,18	2,28	3,88	4,39	5,1	6,14	7,84	11,22	21,29	201,48
12	0	0	0	0	0	0	0	0,92	3,55	5,1	6,14	7,84	11,22	21,29	201,48
13	0	0	0	0	0	0	0	0	0	1,61	5,31	7,84	11,22	21,29	201,48
14	0	0	0	0	0	0	0	0	0	0	0	3,15	10,42	21,29	201,48
15	0	0	0	0	0	0	0	0	0	0	0	0	0,002	9,462	184,33
16	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
29	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
31	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
32	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
33	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
34	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
35	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
36	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
37	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
38	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
39	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
40	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
41	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
42	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
43	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
44	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
45	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
46	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
47	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
48	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
49	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0
50	0	0	0	0	0	0	0	0	0	0	0	0	0,002	0,002	0

The optimal base stock levels (the FIFO case)

				ρ											
Product	0.5	0.6	0.7	0.8	0.9	0.91	0.92	0.93	0.94	0.95	0.96	0.97	0.98	0.99	0.999
1	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
2	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
3	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
4	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
5	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
6	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
7	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
8	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
9	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
10	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
11	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
12	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
13	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
14	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
15	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
16	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
17	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
18	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
19	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
20	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
21	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
22	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
23	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
24	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
25	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
26	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
27	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
28	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
29	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
30	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
31	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
32	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
33	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
34	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
35	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
36	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
37	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
38	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
39	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
40	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
41	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
42	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
43	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
44	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
45	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
46	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
47	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
48	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
49	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135
50	0	0	0	0,2	1	1,13	1,27	1,44	1,65	1,91	2,29	2,87	3,97	7,08	61,135

Table 5 : Numerical results under smoothness assumptions