

# Optimizing Make-To-Stock policies through a robust lot-sizing model <sup>☆</sup>

Agostinho Agra

*CIDMA, Department of Mathematics, University of Aveiro, 3810-193 Aveiro, Portugal.  
Email: agra@ua.pt*

Michael Poss

*UMR CNRS 5506 LIRMM, Université de Montpellier 2, 161 rue Ada, 34392 Montpellier  
Cedex 5, France. Email: michael.poss@lirmm.fr*

Micael Santos

*Department of Mathematics, University of Aveiro, 3810-193 Aveiro, Portugal. Email:  
micael92@ua.pt*

---

## Abstract

In this paper we consider a practical lot-sizing problem faced by an industrial company. The company plans the production for a set of products following a Make-To-Order policy. When the productive capacity is not fully used, the remaining capacity is devoted to the production of those products whose orders are typically quite below the established minimum production level. For these products the company follows a Make-To-Stock (MTS) policy since part of the production is to fulfill future estimated orders. This yields a particular lot-sizing problem aiming to decide which products should be produced and the corresponding batch sizes. These lot-sizing problems typically face uncertain demands, which we address here through the lens of robust optimization.

First we provide a mixed integer formulation assuming the future demands are deterministic and we tighten the model with valid inequalities. Then, in order to account for uncertainty of the demands, we propose a robust approach where demands are assumed to belong to given intervals and the number of deviations to the nominal estimated value is limited. As the number of products can be large and some instances may not be solved to optimality, we propose two heuristics. Computational tests are conducted on a set of instances generated from real data provided by our industrial partner. The heuristics proposed are fast and provide good quality solutions for the tested instances. Moreover, since they are based on the mathematical model and use simple strategies to reduce

---

<sup>☆</sup>Published in *International Journal of Production Economics*, 200, 302–310, 2018. (<https://doi.org/10.1016/j.ijpe.2018.04.002>)

the instances size, these heuristics could be extended to solve other multi-item lot-sizing problems where demands are uncertain.

*Keywords:* Lot-sizing, Make-To-Stock, Robust optimization, Mixed-integer linear programming

---

## 1. Introduction

In this paper we consider a practical problem occurring in an aluminium extrusion industrial company. The company produces two main families of products: a family of products representing the main production activity of the company where a Make-To-Order (MTO) policy is followed (MTO family), and a family of products whose orders are typically quite below the established minimum production level. For this family, the company follows a Make-To-Stock (MTS) policy (MTS family). The production planning procedure for the MTO family is well established. However for the MTS family, as the orders are below the minimum production level, the company must find a solution between the two extreme cases: wait for new orders of the same product until the minimum production level is attained, or produce at least at the minimum production level of that item to satisfy the pending orders and store the leftovers in inventory. Both alternatives have their pros and cons. The first alternative has the advantage of avoiding stocks. On the other hand, the backlogging of demand orders may lead to intangible losses. Conversely, the second alternative has the advantage of a ready satisfaction of customer needs but generates high holding costs.

Currently, the company gives priority to the MTO family by planing its production first, and when extra production capacity is available, then it solves a lot-sizing problem to decide which products from the MTS family should be produced and defining the corresponding lot-sizes. This particular lot-sizing problem takes into account not only the pending orders of each product but also future ones, as the excess quantity produced will remain in stock until new orders are received. Therefore, it is necessary to estimate those future client orders. The uncertainty related to forecasting such future demands represents a risk for the planners since the inventory costs will depend greatly on such unknown demands. For industries where holding costs are high (as in the case of our industrial partner) it is desirable to derive robust solutions that take into account possible future deviations from the estimated demand values.

Here we address this lot-sizing problem defined for the MTS family of products, using the available production capacity. We consider both the deterministic and the robust cases where demands are assumed to belong to an uncertainty set and we look for the production plan that optimizes the worst-case scenario. For the production of the MTS family, we produce at most one batch of each product, hence, we allow at most one set-up. Therefore this particular lot-sizing problem is denoted by LS1S (Lot-Sizing with 1 Set-up). The robust problem is denoted by RLS1S.

Multi-product lot-sizing problems have been receiving a great attention, for recent publications, see e.g. (Cunha et al., 2017; Macedo et al., 2016; Sifaleras and Konstantaras, 2017). Frequently, due to the variety of products and their demand patterns, the companies follow different production policies for the different products. In some cases, different policies can even be considered for the same product (see (Zhang et al., 2013)) in order to satisfy the different demand streams. The decision between the MTO and the MTS policies was investigated by Zaerpour et al. (2008) and Altendorfer and Minner (2014). For an overview on comparison of such approaches see (Olhager and Prajogo, 2012). However, both MTO and MTS producing processes may share common resources forcing the production planners to coordinate the MTO and MTS policies (Rafiei and Rabbani, 2012). Examples of problems combining MTO–MTS policies can be found in different industries, such as food production systems (Soman et al., 2004) and steel plants (Zhang et al., 2015).

Several approaches have been proposed, mostly from last decade, regarding the integration of MTS and MTO policies. Beemsterboer et al. (2016) study the benefits of not prioritizing policies within a hybrid planning MTO–MTS approach. In (Beemsterboer et al., 2017a), the authors analyse the benefits of considering flexible lot sizing policies in a hybrid MTO–MTS approach for a two-product system. In (Beemsterboer et al., 2017b), the authors propose four methods of integrating make-to-stock items in the control of a job shop, which they evaluate using discrete event simulation. Kaminsky and Kaya (2009) propose heuristics for a multi-item problem where the manufacturer and the supplier have to decide which items to produce to stock and which to produce to order. Kalantari et al. (2011) present a decision support system for order acceptance/rejection in a hybrid MTO–MTS production environment. Perona et al. (2009) develop a decision-making approach to support inventory management decisions in a MTO–MTS environment for small and medium sized enterprises. Renna (2016) considers a multistage manufacturing serial system, where a production control strategy is performed to release MTO and MTS orders. Rafiei et al. (2013) propose a hierarchical production planning approach for a hybrid MTO–MTS system that includes both mid-term and short-term production planning levels. Rafiei et al. (2014) propose a genetic algorithm for a multi-site production planning of a hybrid MTO–MTS manufacturing system.

The MTS planning carries the risk that the forecasted orders may not materialize. Such risk has been identified before, see (Tang and Musa, 2011). When it is possible, delaying product differentiation can be an interesting intermediate solution (Gupta and Benjaafar, 2004), but that is not possible in most practical cases as the one faced by our industrial partner. For those cases, handling with uncertainty is of main relevance on MTS environments. To the best of our knowledge only Khakdaman et al. (2015) applied a robust multi-objective approach based on a set of scenarios to a hybrid MTO–MTS problem where uncertainty is considered in suppliers, processes and customers.

The problem considered in this paper occurs as a subproblem of a hybrid MTO–MTS manufacture system where a hierarchic approach is followed and priority is given to MTO. The problem focuses on solving the MTS planning

considering the remaining manufacturing capacity. From its nature, the MTS subproblem considers medium/long-term horizons where demand uncertainty plays a crucial role when defining lot-sizings.

A large number of publications has been devoted to the study of robust lot-sizing problems with demand uncertainty. One of the first papers on the topic is (Bertsimas and Thiele, 2006), which proposes a simple conservative approximation of the robust constraints and studies the structure of the optimal policies. In parallel to that work, another paper introduced affine decision rules (Ben-Tal et al., 2004), having the advantage of better approximating the robust constraints. The theoretical strength of affine decision rules has been studied in subsequent papers, among which (Iancu et al., 2013). More recent works have sought to solve the robust problem exactly, by using decomposition algorithms and dynamically adding constraints to the problem, see (Agra et al., 2016; Bienstock and Özbay, 2008; Gorissen and den Hertog, 2013). Robust lot-sizing problems and their variants are also addressed in more general papers dealing with multi-stage robust optimization, see (Delage and Iancu, 2015) for a survey on these problems. More generally, we refer to (Peidro et al., 2009) for a survey on papers dealing with uncertainty on supply chains.

Although motivated by a practical problem, we aim to incorporate the recent robust optimization techniques into this particular lot-sizing problem in order to close the gap between the robust techniques for classical lot-sizing problems and the robust techniques for MTS problems within hybrid MTO–MTS manufacture systems.

The contributions of this paper are more specifically detailed below. We introduce a mathematical model for the deterministic case where future demands are assumed to be known. Our model is different from the the classical ones (see for instance (Pochet and Wolsey, 2006)) mainly because we suppose that each product has at most one set-up. A proof that this particular problem is NP-hard is given. The model is tightened with valid inequalities.

We develop a robust mixed integer model where demands are considered uncertain and belong to intervals. The uncertainty set is further constrained by budget constraints that limit the number of possible periods where a demand can deviate from its nominal value preventing the solutions to be too conservative, obtaining the well-known budgeted uncertainty set introduced in (Bertsimas and Sim, 2004). We approximate the resulting robust constraints using the conservative approach of (Bertsimas and Thiele, 2006), rather than the computationally demanding affine decision rules from (Ben-Tal et al., 2004) or exact approaches used in (Agra et al., 2016; Bienstock and Özbay, 2008).

Since the problem is NP-hard, and we aim to develop approaches that can be used both with commercial and non-commercial (slower but free) solvers, we propose two heuristics. The first heuristic, called *Elite Heuristic*, is based on a pre-selection of a set of candidate products. The problem is solved for that restricted set of products using a mixed integer linear programming solver based on the strengthened formulation. The heuristic incorporates the practical rules used by the company to choose the products to produce. The second

heuristic, denoted as the *Tournament Heuristic*, runs in several iterations. At each iteration, the set of candidate products is partitioned into smaller subsets and the problem is solved optimally for each subset. Only the selected products of each subset are considered in the next iteration. The process is repeated until a final subset of products is solved or a number of iterations is attained.

To test the deterministic and robust formulations and the matheuristics we use the non-commercial solver Cbc from Coin-OR (2016), which is referred to as one of the fastest solvers among the non-commercial ones (Meindl and Templ, 2012). The test set was built from the real data provided by our industrial partner.

As the proposed heuristics use simple strategies to reduce the number of items and, consequently, the size of the instances, such heuristics can be easily adapted to other multi-item lot-sizing problems. It suffices to adapt the mathematical model to the particularities of the other problems. We also show, that in order to derive solutions that take into account future demands variations, robust strategies could be embedded into the mathematical model, and therefore into the heuristics, but of course such strategies would need further computational testing in other cases and contexts.

The outline of the paper is as follows. In Section 2 we introduce a mixed-integer formulation to model the practical LS1S problem assuming the demands are deterministic. The formulation is enhanced and a proof of NP-hardness is given. Then, in Section 3, we derive the robust model for the case where demands belong to an uncertainty set. In Section 4 we present the two heuristics. Computational experiments are reported in Section 5. Final conclusions are given in Section 6.

## 2. Formulation

In this section we introduce a mixed integer formulation for the LS1S problem. The formulation is presented in a generic format in order to establish connections to related models and existent literature. Let  $m$  denote the number of items considered and  $n$  denote the number of time periods of the planning horizon, and define the sets  $M = \{1, \dots, m\}$  and  $N = \{1, \dots, n\}$ . We split the time horizon into two sub horizons  $N_1 = \{1, \dots, n_1\}$  and  $N_2 = \{n_1 + 1, \dots, n\}$ . The first horizon is for production planning (where the extra production capacity is available) while the second horizon is considered for the inventory management aspects. The demand of item  $i \in M$ , in time period  $t \in N_1$ , denoted by  $d_{it}^e$ , is assumed to be known and, in our case, corresponds to pending orders. The demand for item  $i \in M$  in time period  $t \in N_2$  is denoted by  $d_{it}^p$  and is forecasted. If item  $i$  is produced, then the amount produced must be comprised between  $\underline{Q}_i$  and  $\overline{Q}_i$ . For each item  $i$ , parameters  $p_i$  and  $q_i$  represent the unit production cost and the fixed production cost, respectively. Such parameters may be negative if we allow them to incorporate, for instance, the selling price. For each product  $i \in M$ , and for each time period  $t \in N$ , parameters  $h_{it}$  and  $g_{it}$  are assumed to be nonnegative and model the unit inventory cost and the unit backlogging cost, respectively. Finally,  $\overline{S}$  is the maximum inventory capacity.

To formulate the problem, we define the following decision variables:  $x_{it}$  is the production of item  $i$  in period  $t$ ;  $z_{it}$  indicates whether there is production of item  $i$  in period  $t$ ;  $y_i$  is the set-up variable which is 1 if  $x_{it} > 0$  for some  $t \in N_1$ , and 0 otherwise;  $s_{it}$  is the inventory of item  $i$  at the end of time period  $t$ , and  $r_{it}$  is the backlogged demand of item  $i$  at the end of period  $t$ . The mixed integer programming formulation for LS1S is described below.

$$\min \quad \sum_{i \in M} \sum_{t \in N_1} p_i x_{it} + \sum_{i \in M} \sum_{t \in N} h_{it} s_{it} + \sum_{i \in M} \sum_{t \in N} g_{it} r_{it} + \sum_{i \in M} q_i y_i \quad (1)$$

$$s.t. \quad x_{it} + s_{i,t-1} + r_{it} = d_{it}^e + s_{it} + r_{i,t-1}, \forall i \in M, t \in N_1, \quad (2)$$

$$s_{i,t-1} + r_{it} = d_{it}^p + s_{it} + r_{i,t-1}, \quad \forall i \in M, t \in N_2, \quad (3)$$

$$\sum_{i \in M} s_{it} \leq \bar{S}, \quad \forall t \in N, \quad (4)$$

$$\underline{Q}_i z_{it} \leq x_{it} \leq \bar{Q}_i z_{it}, \quad \forall i \in M, t \in N_1, \quad (5)$$

$$y_i = \sum_{t \in N_1} z_{it}, \quad \forall i \in M, \quad (6)$$

$$z_{it} \in \{0, 1\}, \quad \forall i \in M, t \in N_1, \quad (7)$$

$$y_i \in \{0, 1\}, \quad \forall i \in M, \quad (8)$$

$$s_{it}, r_{it} \geq 0, \quad \forall i \in M, t \in N, \quad (9)$$

$$r_{i0} = s_{i0} = 0, \quad \forall i \in M, \quad (10)$$

$$\sum_{i \in M} x_{it} = V_t, \quad \forall t \in N_1. \quad (11)$$

The objective function (1) aims to minimize the sum of the production costs ( $\sum_{i \in M} \sum_{t \in N_1} p_i x_{it}$ ), the inventory cost ( $\sum_{i \in M} \sum_{t \in N} h_{it} s_{it}$ ), the backlogging cost ( $\sum_{i \in M} \sum_{t \in N} g_{it} r_{it}$ ), and the fixed production cost ( $\sum_{i \in M} q_i y_i$ ) over the planning horizon.

Constraints (2) are the inventory balanced constraints written for each item and each production period  $t \in N_1$ , while constraints (3) are the inventory balanced constraints for time periods  $t \in N_2$ . Constraints (4) impose an upper bound on the stock level. Constraints (5) are the variable lower and upper bound constraints. They impose a lower and an upper bound on the quantity produced of each product at each period and link the corresponding continuous variables to the set-up variables. Equations (6) establish the number of set-ups for each product. Together with (8) they ensure that at most one set-up can occur. Constraints (7) and (8) define the set-up variables as binary. Constraints (9) ensure non-negativity of the inventory and backlog variables. Constraint (11) represents the additional constraints related to the available production capacity.

Summing up the equations (2) from 1 to  $n_1$ , one obtains

$$\sum_{t \in N_1} x_{it} + s_{i0} - r_{i0} = \sum_{t \in N_1} d_{it}^e + s_{in_1} - r_{in_1}, \quad \forall i \in M \quad (12)$$

Denoting  $\sum_{t \in N_1} d_{it}^e$  by  $D_i^e$  and using  $s_{i0} = r_{i0} = 0$ , then (12) can be written as

$$\sum_{t \in N_1} x_{it} = D_i^e + s_{in_1} - r_{in_1}, \quad \forall i \in M. \quad (13)$$

Similarly, summing up equations (3) from  $n_1 + 1$  to  $\ell \in \{n_1 + 1, \dots, n\}$ , we obtain

$$s_{in_1} - r_{in_1} = \sum_{t=n_1+1}^{\ell} d_{it}^p + s_{i\ell} - r_{i\ell}, \quad \forall i \in M, \ell \in \{n_1 + 1, \dots, n\}, \quad (14)$$

Using (13) to eliminate  $s_{in_1}$  and  $r_{in_1}$ , then

$$s_{i\ell} - r_{i\ell} = \sum_{t \in N_1} x_{it} - D_i^e - \sum_{t=n_1+1}^{\ell} d_{it}^p, \quad \forall i \in M, \ell \in \{n_1 + 1, \dots, n\}. \quad (15)$$

Since  $s_{i\ell}$  and  $r_{i\ell}$  are nonnegative, we obtain

$$s_{it} \geq \sum_{t \in N_1} x_{it} - D_i^e - \sum_{\ell=n_1+1}^t d_{i\ell}^p, \quad \forall i \in M, t \in N_2, \quad (16)$$

$$r_{it} \geq - \sum_{t \in N_1} x_{it} + D_i^e + \sum_{\ell=n_1+1}^t d_{i\ell}^p, \quad \forall i \in M, t \in N_2. \quad (17)$$

As the holding and backlogging costs are assumed nonnegative, we may replace constraints (3) by (16) and (17). The resulting deterministic model for LS1S, defined by (1), (2), (4) – (11), (16), (17), will be denoted by DLS1S.

When  $N_2 = \emptyset$ , we obtain a classical lotsizing model with the additional constraint that at most one setup is allowed for each item. In a MTS environment the stocks may last for a longer time horizon than the planning one. Under deterministic assumptions the second time horizon can be easily dropped since the holding cost in the last time period can be easily computed. However, in a robust setting it is desirable to consider the second time horizon explicitly since it allows to incorporate different seasonal behaviours of demand and allow to account for different magnitudes of deviation to the estimated demands.

### 2.1. Strengthening the formulation

It is well known that the inclusion of valid inequalities can improve the model significantly, see (Pochet and Wolsey, 2006) for details. Constraints (16) can be

strengthened as follows.

$$s_{it} \geq \sum_{t \in N_1} x_{it} - D_i^e y_i - \sum_{\ell=n_1+1}^t d_{i\ell}^p y_i, \quad \forall i \in M, t \in N_2, \quad (18)$$

When  $y_i = 1$  inequality (18) coincides with (16) and when  $y_i = 0$  the right-hand side of (18) becomes negative. Thus (16) is valid for the set of feasible solutions. Moreover,

$$\begin{aligned} D_i^e y_i + \sum_{\ell=n_1+1}^t d_{i\ell}^p y_i &\leq d_i^e + \sum_{\ell=1}^t d_{i\ell}^p \\ \Leftrightarrow \sum_{t \in N_1} x_{it} - D_i^e y_i - \sum_{\ell=n_1+1}^t d_{i\ell}^p y_i &\geq \sum_{t \in N_1} x_{it} - D_i^e - \sum_{\ell=1}^t d_{i\ell}^p \end{aligned}$$

which implies that (18) is stronger than (16). In fact we can replace (16) by (18) in the formulation for LS1S.

A family of valid inequalities for the value of backlog variables follows.

$$r_{it} \geq D_i^e(1 - y_i) + \sum_{\ell=n_1+1}^t d_{i\ell}^p(1 - y_i), \quad \forall i \in M, t \in N_2. \quad (19)$$

If there is no setup for period  $i$ , i.e.  $y_i = 0$ , inequality (19) forces the demand  $D_i^e + \sum_{\ell=1}^t d_{i\ell}^p$  to be backlogged. Otherwise, if  $y_i = 1$ , inequality (19) simply imposes nonnegativity on the backlog variables. Contrary to the previous case, (19) cannot replace inequalities (17).

The strengthened deterministic model for LS1S, denoted by SDLS1S is given by (1), (2), (4) – (11), (17), (18), (19).

In the practical case provided to us by our industrial partner we have the following assumptions: (i) only one production period is considered ( $n_1 = 1$ ); (ii) only the backlog of the effective demand is penalized ( $g_{it} = 0, t \in N_2$ ); and (iii) the minimum production quantity is at least the effective demand ( $\underline{Q}_i \geq d_i^e$ ). The three assumptions imply that constraints (17) can be eliminated. For completeness we give below the resulting strengthened model, after simplification, for the practical case.

$$\min \quad \sum_{i \in M} \sum_{t \in N_1} p_i x_{it} + \sum_{i \in M} \sum_{t \in N} h_{it} s_{it} + \sum_{i \in M} g_{i1} r_{i1} + \sum_{i \in M} q_i y_i \quad (20)$$



$$s.t. \quad s_{it} \geq \sum_{t \in N_1} x_{it} - D_i^e y_i - \sum_{\ell=1}^t d_{i\ell}^p y_i, \quad \forall i \in M, t \in N, \quad (21)$$

$$r_{i1} \geq D_i^e (1 - y_i), \quad \forall i \in M, \quad (22)$$

$$s_{i1} \leq \bar{S}, \quad (23)$$

$$\underline{Q}_i y_i \leq \sum_{t \in N_1} x_{it} \leq \bar{Q}_i y_i, \quad \forall i \in M, \quad (24)$$

$$\sum_{i \in M} \sum_{t \in N_1} x_{it} = V_1, \quad \forall t \in N, \quad (25)$$

$$s_{it}, r_{it} \geq 0, \quad \forall i \in M, \quad (26)$$

$$y_i \in \{0, 1\}, \quad \forall i \in M. \quad (27)$$

Notice that constraints (23) are given just for  $t = 1$  since the stock level will decrease in subsequent periods. We denote model (20)-(27) by PLS1S.

## 2.2. Complexity analysis

The original problem LS1S is NP-hard as it generalizes the classical lot-sizing problem with varying capacities (Pochet and Wolsey, 2006). Here we focus on the particular case of the practical problem PLS1S. We show it is NP-hard for the simplified version with two time periods (implying it is NP-hard for the general case). The proof is done by reducing the partition problem to PLS1S.

**Proposition 2.1.** *The problem defined by PLS1S is NP-hard for the particular case  $n_1 = 1, n = 2$ .*

**Proof:** The decision problem, denoted by D-PLS1S, associated with the optimization problem asks whether there is a solution to (21)-(27) whose objective function value given by (20) is greater than  $L$ .

Next we reduce the partition problem to D-PLS1S. Recall that in the partition problem we are given  $k$  positive integers  $a_i, i \in K = \{1, \dots, k\}$  and wish to determine whether there exists a partition  $(S, K \setminus S)$  of  $K$  such that  $\sum_{i \in S} a_i = \sum_{i \in K \setminus S} a_i = \sum_{i \in K} a_i / 2$ .

For the reduction consider  $k = m, \underline{Q}_i = \bar{Q}_i = a_i$  and  $V_1 = \sum_{i \in K} a_i / 2$ . Further, for each  $i \in M$ , we set  $D_i^e = \min_{j \in K} a_j, d_{i1}^p = a_i - D_i^e, p_i = q_i = g_{i1} = 0, h_{it} = 0, t \in N, L = 0$ .

As  $\sum_{t \in N_1} x_{it} = a_i y_i$ , any feasible solution to D-PLS1S must satisfy  $\sum_{i \in M} a_i y_i = \sum_{i \in K} a_i / 2$ . Hence, there is a one to one correspondence between a feasible solution  $(x^*, y^*)$  of D-PLS1S, with the specified parameters, and a feasible solution of the partition problem, where  $S = \{j \in K | y_j^* = 1\}$ .  $\square$

### 3. Robust model for the demand uncertainty case

Clearly, it is not possible to know precisely the value of  $d_{it}^p$  for all  $i$  and  $t$  since these orders have not been made yet. At best, we can rely on historical data to draw a set of plausible values for these demands. A popular approach (see (Bertsimas and Sim, 2003, 2004; Bertsimas and Thiele, 2006)) considers the nominal value  $\bar{d}_{it}^p$  and the deviation  $\hat{d}_{it}^p$  for each  $i$  and  $t$  which could, for instance, be the mean value and the variance of the available historical data. The approach then supposes that the unknown parameter  $d_{it}^p$  can take any value in the interval  $[\bar{d}_{it}^p - \hat{d}_{it}^p, \bar{d}_{it}^p + \hat{d}_{it}^p]$  and that, for each item  $i$  and each time period  $t$ , the number of demands taking an extreme value is bounded by a given parameter  $\Gamma_t > 0$ :

$$\sum_{\ell \in N^t} \frac{|d_{i\ell}^p - \bar{d}_{i\ell}^p|}{\hat{d}_{i\ell}^p} \leq \Gamma_t. \quad (28)$$

where  $N^t = \{n_1 + 1, \dots, t\}$ . Formally, the uncertainty sets obtained with this approach can be written as

$$\mathcal{D}_{it} = \left\{ d_{i\ell}^p : d_{i\ell}^p = \bar{d}_{i\ell}^p + \delta_{i\ell}^+ \hat{d}_{i\ell}^p - \delta_{i\ell}^- \hat{d}_{i\ell}^p, \ell \in N^t, \delta_{i\ell}^+, \delta_{i\ell}^- \in [0, 1], \ell \in N^t, \right. \\ \left. \sum_{\ell \in N^t} (\delta_{i\ell}^+ + \delta_{i\ell}^-) \leq \Gamma_t \right\},$$

for each  $i \in M$  and  $t \in N$ , where  $\delta^+$  and  $\delta^-$  are auxiliary vectors that ease the linearization of constraint (28). Parameter  $\Gamma_t$  is often denoted as the *budget* of uncertainty. Taking a small value of  $\Gamma_t$  yields a small uncertainty set, while increasing  $\Gamma_t$  yields larger and larger uncertainty sets. The two extremes are  $\Gamma_t = 0$  for which  $\mathcal{D}_{it}$  is reduced to the singleton  $\{\bar{d}_i^p\}$  and  $\Gamma_t = t$  for which  $\mathcal{D}_{it}$  is equal to the box

$$\prod_{\ell \in N^t} [\bar{d}_{i\ell}^p - \hat{d}_{i\ell}^p, \bar{d}_{i\ell}^p + \hat{d}_{i\ell}^p].$$

In general, for each  $t$ ,  $\Gamma_t$  is comprised between 0 and  $t$ , and  $\Gamma_t \leq \Gamma_{t+1}$ . For simplicity we assume  $\Gamma_t$  is integer for each  $t$ , but all the results derived in this section could be extended to the case where parameters  $\Gamma_t$  are fractional.

Next we present a robust model for RLS1S. The model is based on the deterministic formulation DLS1S where constraints (16) are replaced by (18). For the remaining models discussed in the previous section (models D-LS1S, SDLS1S and PLS1S) the robust model can be derived in a similar way.

$$\min \sum_{i \in M} \sum_{t \in N_1} p_i x_{it} + \sum_{i \in M} \sum_{t \in N} h_{it} s_{it} + \sum_{i \in M} \sum_{t \in N} g_{it} r_{it} + \sum_{i \in M} q_i y_i \quad (29)$$

$$s.t. \quad s_{it} \geq \sum_{t \in N_1} x_{it} - D_i^e y_i - \sum_{\ell \in N^t} d_{i\ell}^p y_i, \forall i \in M, t \in N_2, d_{i\ell}^p \in \mathcal{D}_{it} \quad (30)$$

$$r_{it} \geq - \sum_{t \in N_1} x_{it} + D_i^e + \sum_{\ell \in N^t} d_{i\ell}^p, \forall i \in M, t \in N_2, d_{i\ell}^p \in \mathcal{D}_{it} \quad (31)$$

(2), (4) – (11), (17).

Constraints (30) and (31) are inspired by the approach used in (Bertsimas and Thiele, 2006). Their main advantage is to lead to a fairly simple robust counterpart. Namely, we show in the rest of the section that problem (2), (4) – (11), (29) – (31), can be reformulated as a compact MILP that is essentially of the same order of difficulty as the deterministic version LS1S. In contrast, the more advanced methods used in (Agra et al., 2016; Ben-Tal et al., 2004) provide more accurate solutions however at a high computational cost.

Formulation RLS1S contains an infinite number of constraints, yielding a semi-infinite MILP. We show next how to reformulate the problem as a compact MILP by using a well-known technique from robust optimization (e.g. (Ben-Tal and Nemirovski, 1998; Bertsimas and Sim, 2004; Bertsimas and Thiele, 2006)). First, we realize that for (30), positive deviations of  $d_{i\ell}^p$  are not increasing the cost of the solution since they would only decrease the stock more rapidly; hence, positive deviations can be neglected in (30). Similarly, we can restrict ourselves to negative deviations of  $d_{i\ell}^p$  in (31) since the positive deviations will never increase the amount of unmet demands. Formally, we can introduce the smaller uncertainty sets

$$\mathcal{D}_{it}^+ = \left\{ d_{i\ell}^p : d_{i\ell}^p = \bar{d}_{i\ell}^p + \delta_{i\ell} \hat{d}_{i\ell}^p, \ell \in N^t, \delta_{i\ell} \in [0, 1], \ell \in N^t, \sum_{\ell \in N^t} \delta_{i\ell} \leq \Gamma_t \right\},$$

$$\mathcal{D}_{it}^- = \left\{ d_{i\ell}^p : d_{i\ell}^p = \bar{d}_{i\ell}^p - \delta_{i\ell} \hat{d}_{i\ell}^p, \ell \in N^t, \delta_{i\ell} \in [0, 1], \ell \in N^t, \sum_{\ell \in N^t} \delta_{i\ell} \leq \Gamma_t \right\},$$

and we replace constraints (30) and (31) with the equivalent constraints

$$s_{it} \geq \sum_{t \in N_1} x_{it} - D_i^e y_i - \sum_{\ell \in N^t} \bar{d}_{i\ell}^p y_i, \quad \forall i \in M, t \in N_2, d_{i\ell}^p \in \mathcal{D}_{it}^-, \quad (32)$$

$$r_{it} \geq - \sum_{t \in N_1} x_{it} + D_i^e + \sum_{\ell \in N^t} \bar{d}_{i\ell}^p, \quad \forall i \in M, t \in N_2, d_{i\ell}^p \in \mathcal{D}_{it}^+. \quad (33)$$

Next we see that the infinite numbers of constraints (32) and (33) can be sub-

stituted by the following non-linear constraints

$$s_{it} \geq \sum_{t \in N_1} x_{it} - D_i^e y_i - \min_{d_{i\ell}^p \in \mathcal{D}_{it}^-} \left( \sum_{\ell \in N^t} d_{i\ell}^p y_i \right), \quad \forall i \in M, t \in N_2, \quad (34)$$

$$r_{it} \geq - \sum_{t \in N_1} x_{it} + D_i^e + \max_{d_{i\ell}^p \in \mathcal{D}_{it}^+} \left( \sum_{\ell \in N^t} d_{i\ell}^p \right), \quad \forall i \in M, t \in N_2. \quad (35)$$

Next we follow the classical dualization approach introduced in (Ben-Tal and Nemirovski, 1998) and used to lot-sizing problems in (Bertsimas and Thiele, 2006). Let  $i \in M$  and  $t \in N$  be fixed and let us focus on constraint (34). The inner minimization problem over variables  $d_{i\ell}^p$  in (34) can be replaced by a minimization problem over variables  $\delta_{i\ell}$ . Moving the term  $y_i \sum_{\ell \in N^t} \bar{d}_{i\ell}^p$  outside of the minimization because it does not involve the variable  $\delta_{i\ell}$ , and changing the minimization problem by a maximization problem, we obtain

$$y_i \sum_{\ell \in N^t} \bar{d}_{i\ell}^p - \max \sum_{\ell \in N^t} \delta_{i\ell} \hat{d}_{i\ell}^p y_i$$

$$s.t. \quad \sum_{\ell \in N^t} \delta_{i\ell} \leq \Gamma_t, \quad (36)$$

$$0 \leq \delta_{i\ell} \leq 1, \quad \forall \ell \in N^t. \quad (37)$$

Let us denote the dual variables of constraints (36) and (37) as  $z_{it}^-$  and  $u_{i\ell}^{t-}$ , respectively. Since constraints (36) and (37) define a bounded and non-empty polytope, we can apply strong linear programming duality to replace the maximization problem by its dual:

$$y_i \sum_{\ell \in N^t} \bar{d}_{i\ell}^p - \min \Gamma_t z_{it}^- + \sum_{\ell \in N^t} u_{i\ell}^{t-}$$

$$s.t. \quad z_{it}^- + u_{i\ell}^{t-} \geq \hat{d}_{i\ell}^p y_i, \quad \forall \ell \in N^t, \quad (38)$$

$$z_{it}^-, u_{i\ell}^{t-} \geq 0 \quad \forall \ell \in N^t. \quad (39)$$

Plugging the above minimization problem into the original constraint (34) for the fixed  $i$  and  $t$ , we obtain

$$s_{it} \geq \sum_{t \in N_1} x_{it} - D_i^e y_i - y_i \sum_{\ell \in N^t} \bar{d}_{i\ell}^p + \left( \begin{array}{l} \min \Gamma_t z_{it}^- + \sum_{\ell \in N^t} u_{i\ell}^{t-} \\ s.t. \quad z_{it}^- + u_{i\ell}^{t-} \geq \hat{d}_{i\ell}^p y_i, \quad \forall \ell \in N^t \\ z_{it}^-, u_{i\ell}^{t-} \geq 0 \quad \forall \ell \in N^t \end{array} \right). \quad (40)$$

Finally, notice that (40) is feasible if and only if there exist vectors  $z_{it}^- \geq 0$  and  $u_{i\ell}^{t-} \geq 0$  that satisfy the dual constraints (38) and such that

$$s_{it} \geq \sum_{t \in N_1} x_{it} - D_i^e y_i - y_i \sum_{\ell \in N^t} \bar{d}_{i\ell}^p + \Gamma_t z_{it}^- + \sum_{\ell \in N^t} u_{i\ell}^{t-}. \quad (41)$$

Therefore, the robust constraint (34) is equivalent to the constraints (38), (39), and (41).

Introducing dual variables  $z_{it}^+$  and  $u_{it}^{\ell+}$  to handle with constraints (35), we can reformulate RLS1S as the following compact MILP, denoted by RLS1S.

$$\begin{aligned}
\min \quad & \sum_{i \in M} \sum_{t \in N_1} p_i x_{it} + \sum_{i \in M} \sum_{t \in N} h_{it} s_{it} + \sum_{i \in M} \sum_{t \in N} g_{it} r_{it} + \sum_{i \in M} q_i y_i \\
s.t. \quad & s_{it} \geq \sum_{t \in N_1} x_{it} - D_i^e y_i - y_i \sum_{\ell \in N^t} \bar{d}_{i\ell}^p + \Gamma_t z_{it}^- + \sum_{\ell \in N^t} u_{i\ell}^{t-}, \forall i \in M, t \in N_2 \\
& z_{it}^- + u_{i\ell}^{t-} \geq \hat{d}_{i\ell}^p y_i, \quad \forall i \in M, t \in N_2, \ell \in N^t \\
& r_{it} \geq - \sum_{t \in N_1} x_{it} + D_i^e + \sum_{\ell \in N^t} \bar{d}_{i\ell}^p + \Gamma_t z_{it}^+ + \sum_{\ell \in N^t} u_{i\ell}^{t+}, \forall i \in M, t \in N_2 \\
& z_{it}^+ + u_{i\ell}^{t+} \geq \hat{d}_{i\ell}^p, \quad \forall i \in M, t \in N_2, \ell \in N^t \\
& (2), (4) - (11), (17), \\
& z_{it}^-, u_{i\ell}^{t-}, z_{it}^+, u_{i\ell}^{t+} \geq 0, \quad \forall i \in M, t \in N_2, \ell \in N^t.
\end{aligned}$$

The above approach is easy to apply because it amounts to solve a unique problem. Yet, the dimension of the new MILP is larger than the dimension of the original problem.

An alternative approach to the dualization is to compute *a priori* the minimum and maximum occurring in the right-hand side of inequalities (34) and (35), respectively. As explained above, these optimization problems can be converted into a maximization problem where the feasible set is defined by (36)-(37). We can observe that the extreme solutions of this set satisfy  $\delta_{i\ell} \in \{0, 1\}$ . Thus, the sets of extreme feasible solutions correspond to uniform matroids. Consequently, these maximization problems can be solved by a greedy algorithm that chooses the highest deviations. Hence, the following equalities hold.

$$\begin{aligned}
\min_{d_{i\ell}^p \in \mathcal{D}_{it}^-} \left( \sum_{\ell \in N^t} d_{i\ell}^p y_i \right) &= \left( \sum_{\ell \in N^t} \bar{d}_{i\ell}^p - \sum_{r=1}^{\Gamma_t} \hat{d}_{i\ell_r}^p \right) y_i \\
\max_{d_{i\ell}^p \in \mathcal{D}_{it}^-} \left( \sum_{\ell \in N^t} d_{i\ell}^p \right) &= \sum_{\ell \in N^t} \bar{d}_{i\ell}^p + \sum_{r=1}^{\Gamma_t} \hat{d}_{i\ell_r}^p
\end{aligned}$$

where  $\hat{d}_{i\ell_r}^p$  is the  $r^{th}$  largest deviation among the first  $t$  deviations  $\hat{d}_{i1}^p, \dots, \hat{d}_{it}^p$ .

Hence, (34) and (35) are replaced by

$$s_{it} \geq \sum_{t \in N_1} x_{it} - D_i^e y_i - \left( \sum_{\ell \in N_t} \bar{d}_{i\ell}^p - \sum_{r=1}^{\Gamma_t} \hat{d}_{i\ell^t}^p \right) y_i, \quad \forall i \in M, t \in N_2, \quad (42)$$

$$r_{it} \geq - \sum_{t \in N_1} x_{it} + D_i^e + \sum_{\ell \in N_t} \bar{d}_{i\ell}^p + \sum_{r=1}^{\Gamma_t} \hat{d}_{i\ell^t}^p, \quad \forall i \in M, t \in N_2. \quad (43)$$

For the computational results we use this second approach since it is more efficient computationally.

#### 4. Heuristics

Companies seek for quick approaches to find good solutions to their problems. While compact, formulations SDLS1S and RLS1S can take too much time to be solved to optimality using exact algorithms such as the Branch and Cut implemented in both commercial and open-source optimization software. The running time is even more relevant in our case since the number of items considered can be quite large, up to few hundreds. Hence, for a practical use, we present two heuristics to solve problems LS1S and RLS1S approximately. A first one, called *Elite Heuristic* extends the current practice of the company which is based on the selection of items accordingly to some criteria. Here we select a larger number of items accordingly to the same criteria and solve the models restricted to the selected items. The second heuristic, called *Tournament Heuristic*, selects iteratively small subsets of items until it reaches a final and small subset.

##### 4.1. Elite Heuristic

The first heuristic is to select a specific subset of items and solve the proposed formulation considering this subset of items.

The choice of the subset is as follows. Choose

- $\bar{m}_1$  items with the highest known demand values,  $D_i^e$ ;
- $\bar{m}_2$  items with the highest values of the known demand plus the forecasted demand for the first time period,  $(D_i^e + d_{i,n_1+1}^p)$ ;
- $\bar{m}_3$  items with the highest values of the known demand plus the forecasted demand for the first two time periods,  $(D_i^e + \sum_{t=n_1+1}^{n_1+2} d_{it}^p)$ ;
- $\bar{m}_4$  items with the highest values of the known demand plus the forecasted demand for the first three time periods,  $(D_i^e + \sum_{t=n_1+1}^{n_1+3} d_{it}^p)$ ;

Here we consider  $\bar{m}_1 = \bar{m}_2 = \bar{m}_3 = \bar{m}_4$ .

After selecting and joining these four list of items, the restricted model (the deterministic SDLS1S or the robust RLS1S) is solved. As the number of elite

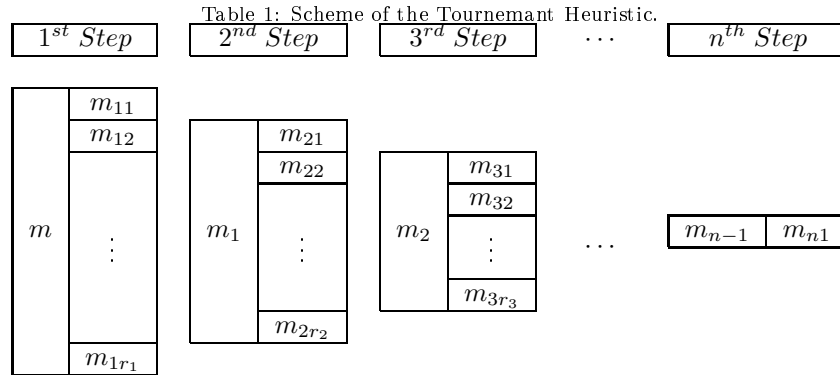
items is small the resulting model can be solved to optimality easily. As example, if we are given a set with 200 items and take  $\bar{m}_1 = \bar{m}_2 = \bar{m}_3 = \bar{m}_4 = 5$ , then five items will be selected from the initial set of 200 items using each one of the four criteria given above. The resulting restricted model with the selected items is solved using a solver.

Notice that the selection criteria do not take into account the costs. If the costs vary from product to product significantly, which is not the case of the instances we consider, then other criteria taking into account the cost structure (production costs, inventory costs, backloging costs, fixed production cost) should be used. For instance, if the backloging costs vary significantly, a fifth criteria should be considered which consists in selecting the  $m_5$  items with the highest backloging cost. Similarly, to the remaining cost components.

#### 4.2. Tournament Heuristic

The second heuristic is to decompose the original problem into smaller and easier subproblems which can be solved quickly. The items selected from these subproblems are the input to the subproblems of the next step. The process is repeated until a final and smaller subset of items is obtained. Next we detail this heuristic.

The first step is to divide the set of  $m$  items into  $r_1$  subsets with cardinalities  $m_{11}, m_{12}, \dots, m_{1r_1}$ , respectively, where  $m_{11} + m_{12} + \dots + m_{1r_1} = m$ . Then solve the restricted problem for each one of the subsets. As the resulting subproblems are simpler they are solved to optimality. The items that are produced in the optimal solution of each one of the subproblems are selected to the next iteration. Let  $m_1$  denote the number of items selected in the first step. Then split this set into  $r_2$  subsets with cardinalities  $m_{21}, m_{22}, \dots, m_{2r_2}$ , respectively, where  $m_{21} + m_{22} + \dots + m_{2r_2} = m_1$ . Each one of the subproblems is solved to optimality again in order to identify the new selected (produced) items. The process is repeated until a final subset with cardinality less or equal to a threshold is obtained or a maximum number of iterations is attained. The final solution is the optimal solution to the problem restricted to the final set of items. Table 1 outlines Heuristic 2.



In order to exemplify the decomposition process, assume we are given a set with 200 items and consider  $r_1 = 6$ . Thus, the set of items is split into six subsets as follows: four subsets with 33 elements ( $m_{11} = m_{12} = m_{13} = m_{14} = 33$ ) and two subsets with 34 elements ( $m_{15} = m_{16} = 34$ ). Then, each problem corresponding to a different subset is solved to optimality and the items produced in each one of the six problems are selected to the next round.

Notice that the solution to each subproblem is feasible to the original problem. Hence, it is expected that the quality of the solutions obtained will improve in each iteration, culminating in the solution of the final iteration.

The quality of the final solution may depend on the successive partitions of the set of items. The selection of items for each subset can be done randomly and the process can be repeated several times. Here we will not explore such possible improvement.

## 5. Numerical experiments

In this section we report the computational tests. The objectives of these numerical experiments are three-fold. First, we assess the difficulty of the deterministic and robust instances, reporting also the gains obtained by using the strengthened models proposed in Section 2.1. Second, we evaluate the objective function values for the robust and deterministic models to test the importance of using robust approaches. Last, we test the efficiency of the two proposed heuristics.

All tests were conducted on a computer Intel(R) Core(TM) i3-3250 CPU, 3.50GHz with 4 cores, using the open-source solver Cbc 2.9 (Coin-or branch and cut) from Coin-OR (2016). The heuristics were implemented in Julia, using the package JuMP (Lubin and Dunning, 2015).

Instances were generated using data provided by our industrial partner for the MTS family of products. Since the number of items considered depends on the pending order quantities (as items with few orders are not considered), the number of items varies weekly and can go up to a few hundred products (the complete MTS family). Also, the production capacity,  $V_1$ , varies weekly as it represents the residual capacity after the production of MTO products have been considered. Based on the information provided by our partner, suggesting to pick-up 80 items and to use a residual capacity around 2000, we generate fifteen instances for the deterministic problem by varying the number of items and the residual capacity. In relation to these two parameters (number of products and capacities), the fifteen instances aim to simulate realistic instances faced by the company currently and in the future. For the number of items we consider five possible values 80, 100, 150, 200, 300. Notice that for testing purposes it is not interesting to consider small size and, therefore, easy instances. For the production capacities,  $V_1$ , three values are considered: 1000, 2000, 3000 (since both sides with higher and lower capacities are relevant). The remaining data is taken as follows. The known demands  $D_i^e$  are given by real data and the forecasted demands  $d_{it}^p$  are obtained from the historical average demand of each item. The following additional parameters are considered (established by our industrial



partner, see (Santos, 2015) for details):  $T$  is set to 24,  $Q_i = \max\{250, D_i^e\}$ ,  $h_{it} = H * (1 + J)^t$ , where  $H = 3.52$ ,  $J = 0.001651$ ,  $p_i = 0.991003$ ,  $q_i = 669.11$ . A unit penalty cost of 1 is assumed for each item ( $g_{i1} = 1$ ). For the robust settings,  $\bar{d}_{i\ell}^p$  is set to the deterministic value  $d_{i\ell}^p$ . For  $\hat{d}_{i\ell}^p$ , two possible values are considered  $\hat{d}_{i\ell}^p = 0.2\bar{d}_{i\ell}^p$  and  $\hat{d}_{i\ell}^p = 0.4\bar{d}_{i\ell}^p$ .  $\Gamma$  varies in  $\{0, 1, 2, 3\}$ . This gives a total of 120 instances for the robust problem. Notice that the deterministic instances ( $\Gamma = 0$ ) are considered for both levels of deviations  $\hat{d}_{i\ell}^p$  as the assessment of the price of robustness is different for both levels.

### 5.1. Testing formulations and accessing instances difficulty

In the first experiments the instances are solved with a time limit of 1800 seconds. Table 2 gives the number of instances that were not solved within the given time limit using the model PLS1S with the improvements discussed in Section 2.1. Column capacity gives the production capacity divided by 1000, the second column with  $\Gamma = 0$  is for the deterministic case, columns 3-8 consider the robust case where  $\Gamma_t = \min\{t, \Gamma\}$ . The first three columns assume a maximum deviation of 20% from the nominal value and the next three columns assume a maximum deviation of 40% of that value. The last column gives the total number of instances that were not solved to optimality. Table 3 provides similar information however for the model PLS1S without the improvements (weak model).

Table 2: Number of unsolved instances using the strong formulation.

capacity	$\Gamma = 0$	Deviation=0.2			Deviation=0.4			Sum
		$\Gamma = 1$	$\Gamma = 2$	$\Gamma = 3$	$\Gamma = 1$	$\Gamma = 2$	$\Gamma = 3$	
1	0	0	0	0	0	0	0	
2	0	2	2	2	2	2	2	12
3	2	1	1	1	0	0	0	5
Sum	2	3	3	3	2	2	2	17

Table 3: Number of unsolved instances using the weak formulation.

capacity	$\Gamma = 0$	Deviation=0.2			Deviation=0.4			Sum
		$\Gamma = 1$	$\Gamma = 2$	$\Gamma = 3$	$\Gamma = 1$	$\Gamma = 2$	$\Gamma = 3$	
1	0	0	0	0	1	1	1	3
2	1	3	3	3	2	2	2	16
3	3	2	2	2	2	2	2	15
Sum	4	5	5	5	5	5	5	34

We can see that the number of unsolved instances drops from 34 to 17 by using the strengthened formulation instead of the weak one. There is no clear correlation between the difficulty of the instances and the value of  $\Gamma$  parameter. In relation to the production capacity, we can observe that the medium capacity instances seem to be a bit harder than the other ones.

Although not reported in the tables, the average of root gap is 60% for the weak formulation and 5% for the strong formulation. Nevertheless, all unsolved instances have a final integrality gap less than or equal to 3%. For the strong formulation the lower bound for the geometric average running time of the solved instances is 2.94 (it is a lower bound because instances faster than 1 second are set to 1 second).

We also solved these instances using CPLEX 12.7 rather than Cbc 2.9, keeping the time limit of 1800 seconds. Unsurprisingly, the former is much faster than the latter, the weak formulation solving already all but 11 instances (vs 34 for Cbc) while the strong formulation leaves only 3 instances unsolved (vs 17 for Cbc). Fortunately, we will show below that our heuristics, based on Cbc perform very well on our instances, solving nearly all of them to optimality.

### 5.2. Determining the price of robustness

Here we discuss the price for considering robust solutions. As explained in the introduction, estimating the future demands by the historic average values doesn't immunize the solution for possible deviations in the demand values that can lead to higher costs than the estimated ones. For the practical case considered here, the costs that may be underestimated are the inventory costs which are based on the estimated future demands, since for the backlogged demand only the pendent orders are penalized.

In order to report the computational results we define  $C_{(i,j)}$  as the cost of the solution obtained for  $\Gamma = i$  assessed when  $\Gamma = j$ . For instance,  $C_{(0,2)}$  is the cost obtained for the optimal deterministic solution (obtained for  $\Gamma = 0$ ) when facing an uncertainty level of  $\Gamma = 2$ , that is, when we allow the demand values for two time periods to suffer a maximum deviation, either of 20% or 40%, from the estimated nominal values.

In tables 4 and 5 we report for the two deviation levels the value  $\overline{C}_{(i,j)}$ , which is the average of the percentage gaps between the parameters  $C_{(i,j)}$  and  $C_{(i,i)}$ , defined formally by

$$\overline{C}_{(i,j)} = \frac{C_{(i,j)} - C_{(i,i)}}{C_{(i,i)}}.$$

For example,  $\overline{C}_{(0,2)} = 1.76$  means that the deterministic optimal solutions are, on average, 1.76% more expensive than the optimal solutions obtained for the robust model with  $\Gamma = 2$  when facing an uncertainty level of  $\Gamma = 2$ .

Table 4: Costs  $\overline{C}_{(i,j)}$  (expressed in %) considering a deviation 0.2.

	0	1	2	3
0	0	1.19	1.76	0
1	1.49	0	0.81	0
2	1.20	0	0	0
3	1.19	0	0	0

Table 5: Costs  $\overline{C}_{(i,j)}$  (expressed in %) considering a deviation 0.4.

	0	1	2	3
0	0	2.64	3.67	0
1	2.96	0	0.88	0
2	2.99	0.33	0	0
3	2.99	0.33	0	0

We see from these tables the robust solutions are more expensive, on average, than the deterministic solutions in the deterministic context. For the case where the deviation is allowed to be 40% we can see that protecting a solution for  $\Gamma = 3$  deviation periods will increase the cost by 3% in relation to the deterministic solution. The deterministic solution is more expensive when  $\Gamma \in \{1, 2\}$  but not when  $\Gamma = 3$ . In the worst case (two deviations), not protecting the deterministic solution will imply an increase of the cost of 3.67%.

We can also observe that robust solutions with  $\Gamma = 3$  are not interesting. This justifies why we have not included results for larger values of  $\Gamma$ . Of course, the final choice of which model to use (which value for  $\Gamma$ ) depends on the risk-averseness of the decision maker.

### 5.3. Heuristic performances

Finally we test the two proposed heuristics. As discussed above, running a MILP solver for a given time limit acts as a heuristic for those instances that were not solved to optimality. However, from a practical viewpoint, it is more appealing to have a tool which enables the decision maker to derive good solutions very quickly in order to allow him/her to test different parameters before taking a decision.

Table 6 summarizes the results obtained with the Elite heuristic. For each test we consider  $\overline{m}_1 = \overline{m}_2 = \overline{m}_3 = \overline{m}_4 = \hat{m}$ . The first line gives the value of  $\hat{m}$ , the second line gives the average of the optimality gaps. The optimality gap is defined as  $100 * ((ZH - Z^*)/Z^*)$ , where  $Z^*$  is the optimum value and  $ZH$  is the solution cost returned by the heuristic. The third line gives the average number of items selected when combining the four lists, and the last line gives the average running times in seconds.

Table 6: Average statistics for the Elite heuristic.

Value of $\hat{m}$	5	6	7	8	9	10
Optimality gap	5.6	0.2	0.04	0	0	0
Number of items selected	8.6	11	12	13.4	15.4	17
Running time	0.04	0.07	0.09	0.11	0.12	0.13

We can observe that when  $\hat{m}$  increases the optimality gap decreases as expected, being zero for  $\hat{m}$  greater or equal to 8. The running times are very small indicating that this heuristic performs very well for such values of  $\hat{m}$ . This

means that selecting at least 32 items and solving the restricted model is fast and gives good quality solutions.

Table 7 reports the results obtained with the Tournament heuristic. This heuristic is run until a final set of  $m \leq 50$  items is obtained or until a maximum of 6 iterations is attained. In each iteration the original set of items is split into  $r$  subsets. The first line gives the value of  $r$ . The second line gives the average number of iterations. The third line gives the average optimality gap and the last line gives the average running time in seconds.

Table 7: Average statistics for the Tournament heuristic.

Value of $r$	6	5	4
Number of iterations	2	2.62	3.33
Optimality gap	0.35	0.49	0.12
Running time	0.12	1.81	0.14

Although the heuristic is fast and gives solutions that in average have very small optimality gaps, we could not find parameters that lead to optimality in all the tested cases. Overall, the Tournament heuristic was better than the Elite heuristic only for the case where the number of elite elements selected was small ( $\hat{m} = 5$ ), otherwise the Elite heuristic outperformed the Tournament heuristic.

## 6. Conclusions

We consider a practical problem faced by a company that plans the production for a set of products following a Make-To-Order policy and uses the remaining production capacity to produce items for which the quantities ordered are small. The problem is concerned with the use of this extra production capacity in a given time period. Namely, deciding which items to produce and the corresponding production level. A penalty is associated with the backlogged demands of the items that are not produced. For the produced items, as the amount produced is in general greater than the pending orders, a holding cost is incurred.

We present a general model and establish the relation between this model and the classical lot-sizing models. We propose several enhancements for the formulations. As the holding costs depend on the value of the estimated future demands, we derive a robust model from the enhanced model where the future demands are considered uncertain and can vary in a given interval centered in the historical average values. The uncertainty set is the classical budget polytope introduced by Bertsimas and Sim (2004).

For practical purposes we propose two heuristic schemes that are based on the enhanced models (deterministic and robust). The models and the heuristics are tested using a set of instances generated from real data provided by our industrial partner. The computational experiments show that most instances cannot be solved to optimality within reasonable running time limit. In particular, they show that the Elite heuristic, which selects a small subset of items

(elite items), following the company criteria, is quite fast and in general obtains the optimal solution.

It is well-known that robust optimization approaches can generate too conservative solutions. That is, solutions that are good when the worst case scenario occurs, in our case when the orders for future demands are lower than expected, but their quality may not be so good when other demand scenarios are observed. While this falls beyond the scope of the current work, it could be interesting to compare our robust approach with the optimal solutions of stochastic programming approaches where one also takes the probabilities of the scenarios into account. Similarly, it would be interesting to test the effect of our model and the quality of the heuristics with different cost structures and/or on more complex problems where the set of feasible production plans is restricted by additional constraints (e.g. storage capacity, inclusion of set-up times). In particular, the heuristics would need to be adapted to these other contexts and further computational testing would be required.

### Aknowledgements

The work of the first author was partially funded by FCT and CIDMA within project UID/MAT/04106/2013.

Agra, A., Santos, M. C., Nace, D., Poss, M., 2016. A dynamic programming approach for a class of robust optimization problems. *SIAM Journal on Optimization*, 1799 – 1823.

Altendorfer, K., Minner, S., 2014. A comparison of make-to-stock and make-to-order in multi-product manufacturing systems with variable due dates. *IIE Transactions* 46 (3), 197–212.

Beemsterboer, B., Land, M., Teunter, R., 2016. Hybrid mto-mts production planning: An explorative study. *European Journal of Operational Research* 248 (2), 453 – 461.

Beemsterboer, B., Land, M., Teunter, R., 2017a. Flexible lot sizing in hybrid make-to-order/make-to-stock production planning. *European Journal of Operational Research* 260 (3), 1014 – 1023.

Beemsterboer, B., Land, M., Teunter, R., Bokhorst, J., 2017b. Integrating make-to-order and make-to-stock in job shop control. *International Journal of Production Economics* 185, 1 – 10.

Ben-Tal, A., Goryashko, A., Guslitzer, E., Nemirovski, A., 2004. Adjustable robust solutions of uncertain linear programs. *Math. Program.* 99 (2), 351 – 376.

Ben-Tal, A., Nemirovski, A., 1998. Robust convex optimization. *Mathematics of Operations Research* 23, 769 – 805.

- Bertsimas, D., Sim, M., 2003. Robust discrete optimization and network flows. *Math. Program.* 98, 49 – 71.
- Bertsimas, D., Sim, M., 2004. The price of robustness. *Operations Research* 52, 35 – 53.
- Bertsimas, D., Thiele, A., 2006. A robust optimization approach to inventory theory. *Operations Research* 54, 150 – 168.
- Bienstock, D., Özbay, N., 2008. Computing robust basestock levels. *Discrete Optimization* 5, 389 – 414.
- Coin-OR, 2016. Computational Infrastructure for Operations Research. Software.
- Cunha, J. O., Konstantaras, I., Melo, R. A., Sifaleras, A., 2017. On multi-item economic lot-sizing with remanufacturing and uncapacitated production. *Applied Mathematical Modelling* 50, 772 – 780.
- Delage, E., Iancu, D., 2015. Robust multi-stage decision making. *INFORMS Tutorials in Operations Research*, 20–46.
- Gorissen, B. L., den Hertog, D., 2013. Robust counterparts of inequalities containing sums of maxima of linear functions. *European Journal of Operational Research* 227 (1), 30–43.
- Gupta, D., Benjaafar, S., 2004. Make-to-order, make-to-stock, or delay product differentiation? A common framework for modeling and analysis. *IIE Transactions*, 529 – 546.
- Iancu, D. A., Sharma, M., Sviridenko, M., 2013. Supermodularity and affine policies in dynamic robust optimization. *Operations Research* 61 (4), 941–956.
- Kalantari, M., Rabbani, M., Ebadian, M., 2011. A decision support system for order acceptance/rejection in hybrid mts/mto production systems. *Applied Mathematical Modelling* 35 (3), 1363 – 1377.
- Kaminsky, P., Kaya, O., 2009. Combined make-to-order/make-to-stock supply chains. *IIE Transactions*, 103 – 119.
- Khakdaman, M., Wong, K. Y., Zohoori, B., Tiwari, M. K., Merkert, R., 2015. Tactical production planning in a hybrid make-to-stock-make-to-order environment under supply, process and demand uncertainties: a robust optimisation model. *International Journal of Production Research* 53 (5), 1358–1386.
- Lubin, M., Dunning, I., 2015. Computing in operations research using julia. *INFORMS Journal on Computing* 27 (2), 238–248.

- Macedo, P. B., Alem, D., Santos, M., Junior, M. L., Moreno, A., 2016. Hybrid manufacturing and remanufacturing lot-sizing problem with stochastic demand, return, and setup costs. *The International Journal of Advanced Manufacturing Technology* 82 (5), 1241–1257.
- Meindl, B., Templ, M., 2012. Analysis of commercial and free and open source solvers for linear optimization problems. Eurostat and Statistics Netherlands within the project ESSnet on common tools and harmonised methodology for SDC in the ESS.
- Olhager, J., Prajogo, D., 2012. The impact of manufacturing and supply chain improvement initiatives: A survey comparing make-to-order and make-to-stock firms. *Omega*, 159 – 165.
- Peidro, D., Mula, J., Poler, R., Lario, F.-C., 2009. Quantitative models for supply chain planning under uncertainty: A review. *Int J Adv Manuf Technol*, 400 – 420.
- Perona, M., Saccani, N., Zanoni, S., 2009. Combining make-to-order and make-to-stock inventory policies: An empirical application to a manufacturing SME. *Production Planning & Control*, 559 – 575.
- Pochet, Y., Wolsey, L., 2006. *Production Planning by Mixed Integer Programming*. Springer, New York.
- Rafei, H., Rabbani, M., 2012. Capacity coordination in hybrid make-to-stock/make-to-order production environments. *International Journal of Production Research* 50 (3), 773–789.
- Rafei, H., Rabbani, M., Alimardani, M., 2013. Novel bi-level hierarchical production planning in hybrid mts/mto production contexts. *International Journal of Production Research* 51 (5), 1331–1346.
- Rafei, H., Rabbani, M., Kokabi, R., 2014. Multi-site production planning in hybrid make-to-stock/make-to-order production environment. *Journal of Industrial Engineering International* 10 (3), 1–9.
- Renna, P., 2016. Production control policies for a multistage serial system under mto-mts production environment. *The International Journal of Advanced Manufacturing Technology* 83 (1), 449–459.
- Santos, M., 2015. Otimização do dimensionamento de lotes na extrusão. Master's thesis, University of Aveiro, <http://hdl.handle.net/10773/16293>.
- Sifaleras, A., Konstantaras, I., 2017. Variable neighborhood descent heuristic for solving reverse logistics multi-item dynamic lot-sizing problems. *Computers & Operations Research* 78, 385 – 392.
- Soman, C. A., van Donk, D. P., Gaalman, G., 2004. Combined make-to-order and make-to-stock in a food production system. *International Journal of Production Economics* 90 (2), 223 – 235.

- Tang, O., Musa, S. N., 2011. Identifying risk issues and research advancements in supply chain risk management. *Int. J. Production Economics*, 25 – 34.
- Zaerpour, N., Rabbani, M., Gharehgozli, A. H., Tavakkoli, R., 2008. Make-to-order or make-to-stock decision by a novel hybrid approach. *Advanced Engineering Informatics*, 186 – 201.
- Zhang, T., Zheng, Q. P., Fang, Y., Zhang, Y., 2015. Multi-level inventory matching and order planning under the hybrid make-to-order/make-to-stock production environment for steel plants via particle swarm optimization. *Computers & Industrial Engineering* 87, 238 – 249.
- Zhang, Z. G., Kim, I., Springer, M., Cai, G., Yu, Y., 2013. Dynamic pooling of make-to-stock and make-to-order operations. *International Journal of Production Economics* 144 (1), 44 – 56.