



HAL
open science

A perfect information lower bound for robust lot-sizing problems

Marcio C Santos, Michael Poss, Dritan Nace

► **To cite this version:**

Marcio C Santos, Michael Poss, Dritan Nace. A perfect information lower bound for robust lot-sizing problems. *Annals of Operations Research*, 2018, 271 (2), pp.887-913. 10.1007/s10479-018-2908-x . hal-01409515v3

HAL Id: hal-01409515

<https://hal.science/hal-01409515v3>

Submitted on 31 Jan 2018

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

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

A perfect information lower bound for robust lot-sizing problems

Marcio Costa Santos · Michael Poss · Dritan Nace

Received: date / Accepted: date

Abstract Robust multi-stage linear optimization is hard computationally and only small problems can be solved exactly. Hence, robust multi-stage linear problems are typically addressed heuristically through decision rules, which provide upper bounds for the optimal solution costs of the problems. We investigate in this paper lower bounds inspired by the perfect information relaxation used in stochastic programming. Specifically, we study the uncapacitated robust lot-sizing problem, showing that different versions of the problem become tractable whenever the non-anticipativity constraints are relaxed. Hence, we can solve the resulting problem efficiently, obtaining a lower bound for the optimal solution cost of the original problem. We compare numerically the solution time and the quality of the new lower bound with the dual affine decision rules that have been proposed by Kuhn et al. (2011).

Keywords multi-stage robust optimization · perfect information · lot-sizing problem · complexity

1 Introduction

Lot-sizing optimization problems appear in a wide range of applications where products have to be made to attend demands along a planning horizon. In these problems, the future demands are usually not known with precision before being reached. To be relevant in practice, optimization models must model the uncertainty explicitly so that the decisions are taken to optimize the benefit over the whole planning horizon. Different models of uncertainty exist in the literature, each of which having its advantages and drawbacks. In this paper, we focus on robust lot-sizing and we assume that the demand uncertainty is modeled by a convex set. Thus, the objective is to optimize the cost of the production plan in the worst-case scenario represented by the set. This model, used in Agra et al (2016); Bienstock and Özbay (2008); Bertsimas and Dunning (2014); Bertsimas and Thiele (2006); Kuhn et al (2011); Ben-Tal et al (2004); Gorissen and den Hertog (2013), among others, is relevant when historical data are not accurate enough to draw probabilistic distributions of the uncertain demands.

Robust optimization is often known as being an easy approach to handle uncertainty since, for instance, a robust linear program with polyhedral uncertainty sets can be reformulated as a linear program whose dimension does not grow much with respect to the dimensions of the deterministic problem (Ben-Tal and Nemirovski, 1998). While this is true for *static problems*, where decisions are taken before revealing the uncertain parameters, the situations with *adjustable problems* is far more complex.

M. C. Santos

Department of Computer Science, Université Libre de Bruxelles, B-1050 Brussels, Belgium and INOCS, INRIA Lille Nord-Europe, France E-mail: mcs.marcio@gmail.com

M. Poss

UMR CNRS 5506 LIRMM, Université de Montpellier, 161 rue Ada, 34392 Montpellier Cedex 5, France
E-mail: michael.poss@lirmm.fr

D. Nace

Sorbonne universités, UMR CNRS 7253 Heudiasyc, Université de Technologie de Compiègne, Centre de Recherches de Royallieu, 60200 Compiègne, France
E-mail: dritan.nace@hds.utc.fr

Adjustable robust optimization problems suppose that the uncertainty is revealed as time goes and one can adjust the values of some of the decision variables according to the current knowledge of the uncertain parameters. Hence, the adjustable optimization variables become functions of the uncertain parameters. Robust lot-sizing problems can be modeled either as static or adjustable robust optimization problems. In the static version, it is assumed that the production plan is fixed before the planning horizon starts. In contrast, the adjustable version supposes that the production and other decisions can be adjusted according to the values taken by past demands, thus becoming *functions* of the past demands. In this paper, we consider the adjustable situation.

Adjustable robust optimization is known for being \mathcal{NP} -hard, even in the case of a linear program with only two decision stages (Ben-Tal et al, 2004). In spite of its theoretical difficulty, the problem can be solved exactly by decomposition approaches whenever some assumptions hold (Ayoub and Poss, 2016; Billionnet et al, 2014; Zeng and Zhao, 2013). These approaches consider finite subsets of the uncertainty sets and dynamically increase the number of elements in the subsets by solving separation problems. The numerical tractability of the resulting algorithms highly depends on the complexity of the separation problem. For instance, the separation problem for the robust vehicle routing problem can be solved in polynomial time (Agra et al, 2013), while those related to facility location or network design problems require solving MILP with big- M coefficients (Ayoub and Poss, 2016; Billionnet et al, 2014; Zeng and Zhao, 2013). These decomposition approaches do not extend to multi-stage problems, because of the non-anticipativity constraints present in these problems. Stated simply, non-anticipativity constraints model the fact that optimization variables can only depend on past realizations of the uncertain parameters; they cannot adjust their decision to unknown realizations, see for instance Birge and Louveaux (2011).

Given the difficulty of adjustable multi-stage robust problems, many researchers have developed heuristic approaches that try to provide feasible solutions for these problems. The bottom line of all these approaches is to restrict the set of feasible functions for the adjustable variables. The seminal paper in this line of research is Ben-Tal et al (2004) which restricts adjustable variables to affine functions of the uncertainties, which they call *affine decision rules*. Subsequent authors have studied more complex decision rules that offer more flexibility than affine decision rules while providing more or less tractable optimization problems. Among others, Chen and Zhang (2009) propose to define affine decision rules built from extended descriptions of the uncertainty set and Goh and Sim (2010) introduce complex piece-wise linear decision rules defined through the lifting of the uncertainty set. More complex decision rules have also been considered, such as Bertsimas and Georghiou (2015) which proposes piece-wise decision rules modeled with the help of binary optimization variables, or Bertsimas and Dunning (2014); Postek and den Hertog (2016) which dynamically partition the uncertainty set and selects constant policies for each element of the partition.

The heuristic solutions yield upper bounds for minimization problems. Since the optimal solution of the underlying optimization problem is unknown, one needs lower bounds to evaluate the quality of the aforementioned upper bounds. Up to our knowledge, the literature is scarce when it comes to proposing lower bounds and we are aware of only two previous methods along that direction. In the first one, the authors have considered the simple lower bound that consists of selecting a finite subset of the uncertainty set and solving exactly the resulting finite linear program. This approach has been used by Bertsimas and Dunning (2014), among others, in the course of their partitioning algorithm. In the second one, Kuhn et al (2011) have introduced dual affine decision rules, which provide lower bounds for multistage problems. While the latter work targets more specifically multistage stochastic optimization, their approach is also applicable to robust optimization.

The contributions of this paper follow that line of research by providing another way to compute lower bounds for multistage robust optimization problems. Our approach relaxes the non-anticipativity constraints of the problem, thus yielding a relaxation of the original problem. This relaxation is well-known in the stochastic programming literature as the *perfect information* relaxation. The first mention of the associated optimization problem in the stochastic programming literature, called the expected value of the perfect information, can be traced back to Avriel and Williams (1970). It has then been further studied in several papers (e.g. C. C. Huang (1977)) and has become a well-known concept in stochastic programming (Birge and Louveaux, 2011). However, up to our knowledge it has never been used in robust optimization. The interest of studying the perfect information relaxation in robust optimization is two-fold. First, it helps the decision maker in assessing how much gain could be obtained by reducing the uncertainty on the uncertain parameters (e.g. Oostenbrink et al (2008)). Second, it can be used as a lower bound for the optimal value of the true uncertain problem. We argue that this second aspect

is particularly relevant for robust optimization problems for which many papers have studied upper bounds while good lower bounds are rarely mentioned. As we show in the paper, the perfect information relaxation of the robust lot-sizing problems can be solved efficiently either through polynomial-time algorithms or MILP reformulations. Our experiments realized on lot-sizing instances inspired by the literature seem to indicate that the perfect information relaxation can be quite tight. Throughout the paper, we pay a particular attention to the budget uncertainty set introduced by Bertsimas and Sim (2004) and widely used in the mixed-integer linear robust optimization literature.

The remaining sections of this paper are organized as follows. In Section 2, we formally present the problems we tackle and the uncertainty model used. In Section 3, we recall the primal and dual affine decision rules from the literature, and introduce the problem obtained by relaxing the non-anticipativity constraints. In section 4, which contains the main methodological contributions of our paper, we present combinatorial algorithms and (integer or continuous) linear programming formulations to solve the relaxations based on perfect information. In Section 5, we present numerical experiments to evaluate the quality of the bounds empirically. We conclude the paper in Section 6 and delay the technical derivations to Appendix.

2 Problem description

2.1 Deterministic model

We describe below the problem studied in this paper. Let $H = \{1, \dots, n\}$ denote the planning horizon composed of n periods. In each period of time i , there is a demand that needs to be attended, which we denote d_i . The demand of period i can be attended in three different ways. First, the demand can be attended from the production at period i itself, which incurs only the production cost from period i , denoted c_i . Second, the demand can be attended from the production at the earlier period $j < i$, which incurs the production cost c_j plus the storage cost for each period $j \leq k < i$, denoted h_k . Last, the demand can be attended from the production at the later period $j > i$, which incurs the production cost c_j plus the backloging cost for each period $i \leq k < j$, denoted p_k . The objective of the lot-sizing problem, denoted by \mathcal{LS} , is to provide a *production plan* (information about the amount produced, stored and backlogged in each period $i \in H$) that fulfills the *client demand* d_i at each period i by combining the three aforementioned ways to attend the demand. Also, we have to pay a fixed cost g_i in each period i where production takes place. To keep notations simple, we consider herein problems with a single item and a single producer; one can readily generalize our approach to problems with multiple items and producers. Let us denote the the setup, production, stock and backlog variables as, respectively, y_i , x_i , s_i and r_i . The mathematical formulation of problem \mathcal{LS} follows.

$$\begin{aligned}
(\mathcal{LS}) \quad & \min \quad \kappa \\
& s.t \quad \kappa \geq \sum_{i \in H} (c_i x_i + g_i y_i + h_i s_i + p_i r_i) & (1) \\
& s_{i+1} = x_i - d_i + s_i - r_{i-1} + r_i & \forall i \in H, & (2) \\
& x_i \leq M y_i & \forall i \in H, & (3) \\
& y \in \{0, 1\}^n, x, s, r \geq 0, \\
& s_1 = r_n = 0.
\end{aligned}$$

Constraint (1) imposes that κ be not smaller than the cost of the production plan represented by x , which is formulated in the right-hand side of the constraint. Constraints (2) are equilibrium constraints linking the production, stock and backlog variables. Constraints (3) state that if we produce in a period, then we must pay a fixed cost (setup cost g_i). The other constraints state that y is a binary vector while the other variables are non-negative, and set initial and final conditions for s and r , respectively. Notice that the objective function could be substituted with the right-hand side of constraint (1) in the above formulation. However, we prefer to keep the formulation as it is presented here because it can be translated more naturally to the robust context described in the next subsection.

2.2 Robust model

We assumed so far that demands are known with precision when solving problem \mathcal{LS} described earlier. This is unrealistic in many applications where one has to face uncertainty about the exact values of the demands. To model this issue, we consider an *uncertainty polytope* Ξ and we suppose that the demand at time period i is defined by the affine function $d_i(\xi)$ defined over Ξ :

$$d_i(\xi) = \bar{d}_i + \sum_{j \in H} \hat{D}_{ij} \xi_j, \quad (4)$$

where \bar{d}_i can be seen as the mean value of the clients demands for time period i and \hat{D} is the *deviation matrix* (which can be estimated from historical data) that represents all temporal relations among demands. We typically have that $\hat{D}_{ij} = 0$ for $j > i$ since these relations link current demands to the previous ones. We also assume that the demand functions are non-negative, more precisely, $d_i(\xi) \geq 0$ for all $i \in H$ and $\xi \in \Xi$.

In the robust context, backlog and holding variables depend on the specific scenario ξ . Hence, they are represented by functions $s_i(\xi)$ and $r_i(\xi)$ for each time period i . The situation is more complex with production and setup variables. One could suppose that they are independent of ξ , which would model the fact that all *decisions* must be taken at the beginning of the planning horizon, see for instance Agra et al (2016); Bienstock and Özbay (2008); Bertsimas and Thiele (2006). In this paper, we consider a more subtle approach where the productions and setups can be adjusted according to *past* demand realizations. Hence, these decisions are modeled by functions $x_i(\xi)$ and $y_i(\xi)$ for each time period i . Notice that, for each time period i , these functions must depend only on the demand revealed up to time period i . This is modeled by the *non-anticipativity* constraints

$$\begin{aligned} x_i(\xi) &= x_i(\xi') & \forall \xi, \xi' \in \Xi, \text{Proj}_{[1\dots i]}(\xi) &= \text{Proj}_{[1\dots i]}(\xi'), \\ y_i(\xi) &= y_i(\xi') & \forall \xi, \xi' \in \Xi, \text{Proj}_{[1\dots i]}(\xi) &= \text{Proj}_{[1\dots i]}(\xi'), \\ r_i(\xi) &= r_i(\xi') & \forall \xi, \xi' \in \Xi, \text{Proj}_{[1\dots i]}(\xi) &= \text{Proj}_{[1\dots i]}(\xi'), \\ s_i(\xi) &= s_i(\xi') & \forall \xi, \xi' \in \Xi, \text{Proj}_{[1\dots i]}(\xi) &= \text{Proj}_{[1\dots i]}(\xi'). \end{aligned}$$

where $\text{Proj}_{[1\dots i]}(\xi)$ denotes the projection of ξ on its first i components. Said differently, the non-anticipativity constraints model the fact that the adjustable optimization variables do not depend on future knowledge of the uncertainty. The mathematical formulation for the robust model follows.

$$\begin{aligned} (\mathcal{P}) \quad \min \quad & \kappa \\ \text{s.t.} \quad & \kappa \geq \sum_{i \in H} (c_i x_i(\xi) + g_i y_i(\xi) + h_i s_i(\xi) + p_i r_i(\xi)) & \forall \xi \in \Xi, \end{aligned} \quad (5)$$

$$s_{i+1}(\xi) = x_i(\xi) - d_i(\xi) + s_i(\xi) - r_{i-1}(\xi) + r_i(\xi) \quad \forall i \in H, \forall \xi \in \Xi, \quad (6)$$

$$x_i(\xi) \leq M y_i(\xi) \quad \forall i \in H, \forall \xi \in \Xi, \quad (7)$$

$$x_i(\xi) = x_i(\xi') \quad \forall \xi, \xi' \in \Xi, \text{Proj}_{[1\dots i]}(\xi) = \text{Proj}_{[1\dots i]}(\xi'), \quad \forall i \in H, \quad (8)$$

$$y_i(\xi) = y_i(\xi') \quad \forall \xi, \xi' \in \Xi, \text{Proj}_{[1\dots i]}(\xi) = \text{Proj}_{[1\dots i]}(\xi'), \quad \forall i \in H, \quad (9)$$

$$s_i(\xi) = s_i(\xi') \quad \forall \xi, \xi' \in \Xi, \text{Proj}_{[1\dots i]}(\xi) = \text{Proj}_{[1\dots i]}(\xi'), \quad \forall i \in H \cup \{n+1\}, \quad (10)$$

$$r_i(\xi) = r_i(\xi') \quad \forall \xi, \xi' \in \Xi, \text{Proj}_{[1\dots i]}(\xi) = \text{Proj}_{[1\dots i]}(\xi'), \quad \forall i \in H, \quad (11)$$

$$y(\xi) \in \{0, 1\}^n, x(\xi), s(\xi), r(\xi) \geq 0 \quad \forall \xi \in \Xi,$$

$$s_1(\xi) = r_n(\xi) = 0 \quad \forall \xi \in \Xi.$$

Constraints (5)–(7) play the same role as constraints (1)–(3) for each element $\xi \in \Xi$. Constraints (8)–(11) are the non-anticipativity constraints mentioned previously. Notice that the above problem contains an infinite number of constraints and variables.

In this paper, we study bounding procedures for problem \mathcal{P} as well as for the following two simplifications of problem \mathcal{P} . The first one considers that the setup decisions must be taken before knowing anything about the demand; that is, y becomes a vector of optimization variables that are independent of ξ . The second one looks at the problem without setup costs, which can be modeled by setting all components of y and g to 1 and 0, respectively. We denote these simplifications as \mathcal{P}_1 and \mathcal{P}_0 , respectively. Each of the three models is relevant for specific applications. For instance, \mathcal{P}_0 is close to the classical supply chain model addressed in most papers from the robust lot-sizing literature (e.g. Bertsimas and Thiele (2006); Ben-Tal et al (2004); Gorissen and den Hertog (2013)). In contrast, models \mathcal{P}_1 and \mathcal{P} are relevant for applications that involve fixed costs for the production due, for instance, to machine configurations.

In general, we assume that Ξ can be any non-empty polytope, described by the matrix W with m rows and $|H| = n$ columns

$$\Xi = \{\xi \mid W\xi \leq q\}. \quad (12)$$

In addition to general polytopes, we will also take a closer look at the complexity of the optimization problems obtained when using the *budgeted uncertainty polytope* introduced in Bertsimas and Sim (2004). Given a positive real Γ , a compact description of the budgeted polytope is given by

$$\Xi_\Gamma = \left\{ \xi \mid \sum_{i \in H} |\xi_i| \leq \Gamma, \quad -1 \leq \xi_i \leq 1, \forall i \in H \right\}. \quad (13)$$

Polytope (13) can of course be written as a special case of (12) by replacing each absolute value $|\xi_i|$ by ξ_i and $-\xi_i$, thus rewriting the unique non-linear constraint $\sum_{i \in H} |\xi_i| \leq \Gamma$ as 2^n linear ones. Alternatively, the polytope can be described using an extended formulation where $\xi_i = \xi_i^+ - \xi_i^-$, $\xi_i^+, \xi_i^- \geq 0$ for each $i \in H$, in which case the non-linear constraint becomes $\sum_{i \in H} (\xi_i^+ + \xi_i^-) \leq \Gamma$. The main motivation behind Ξ_Γ , which explains its tremendous success in the robust optimization literature, is that it is unlikely that many parameters deviate simultaneously from their mean values (represented by $\xi = 0$). We refer to Bertsimas and Sim (2004); Poss (2013) for more details on how this intuitive can be formalized to approximate probabilistic constraints.

3 Bounds

We present in Subsections 3.1 and 3.2 approaches from the literature that provide upper and lower bounds for the optimal solution of the adjustable robust problems. Notice that these approaches cannot be applied to robust multi-stage optimization problems that contain adjustable integer variables, such as \mathcal{P} . Hence, in the following two sections, we assume that y does not depend on ξ , either because it is a nonadjustable vector of optimization variables (as in \mathcal{P}_0) or because each of its components has been fixed to 1 (as in \mathcal{P}_1).

3.1 Affine decision rules

The classical upper bound for multi-stage robust optimization problems is based on the so-called affine decision rules. The main idea of the approach is to impose that functions s_i, r_i and x_i depend affinely on ξ . Formally, these restrictions are modeled with constraints

$$x_i(\xi) = x_i^0 + \sum_{j=1}^i x_i^j \xi_j, \quad (14)$$

$$s_i(\xi) = s_i^0 + \sum_{j=1}^i s_i^j \xi_j, \quad (15)$$

$$r_i(\xi) = r_i^0 + \sum_{j=1}^i r_i^j \xi_j. \quad (16)$$

where x_i^0, s_i^0, r_i^0 and x_i^j, s_i^j, r_i^j for $i, j \in H$ are optimization variables. The right-hand side of (14)–(16) involves only the components of $\text{Proj}_{[1\dots i]}(\xi)$. Hence, the equation models implicitly the non-anticipativity constraints introduced in the previous section. Substituting $x_i(\xi), s_i(\xi)$, and $r_i(\xi)$ with the rhs of (14)–(16) for each $i \in H$, we obtain an upper bound for \mathcal{P}_0 and \mathcal{P}_1 , see Appendix A. Then, one can apply classical tools from robust optimization to the formulation from Appendix A to reformulate the upper bound as a compact linear program.

3.2 Dual affine decision rules

Recently, Kuhn et al (2011) have proposed lower bounds for problems \mathcal{P}_0 and \mathcal{P}_1 , which they call *dual affine decision rules*. To be more precise, their approach is developed to provide lower bounds for multi-stage stochastic linear programs. To apply the technique to robust multi-stage programs, one needs to introduce artificial probability weights for the scenarios in Ξ . These probability weights are then used to formulate a lower bounding problem where the robust constraints are relaxed to expectation constraints. Then, a subtle reformulation allows them to provide a compact linear mixed integer formulation for the lower bounding problem. The reformulation is based on the use of convex duality and probability theory. One of the main difficulties of the method relies in the computation of the expectation matrix $\mathcal{M} = \mathbb{E}(\xi\xi^T)$. The approach is sketched in Appendix B, we redirect the interested reader to Kuhn et al (2011) for full details.

3.3 Perfect information relaxation

The major impediment to the efficient solutions of problems \mathcal{P} , \mathcal{P}_0 and \mathcal{P}_1 lies in the presence of the non-anticipativity constraints. Expressing non-anticipativity constraints is not easy in general and strongly depends on the particular structure of the considered set Ξ . In what follows, we propose a lower bounding problem for \mathcal{P} that relaxes the non-anticipativity constraints from \mathcal{P} , which we call the *problem with perfect information*. Unlike the affine decision rules and the dual version presented in the previous sections, the perfect information relaxation can be applied to \mathcal{P} regardless to the dependency of y on ξ . The associated optimization problems are denoted by \mathcal{PI} , \mathcal{PI}_1 , and \mathcal{PI}_0 , for \mathcal{P} , \mathcal{P}_1 and \mathcal{P}_0 respectively.

$$\begin{aligned}
(\mathcal{PI}) \quad & \min && \kappa \\
& s.t && \kappa \geq \sum_{i \in H} (c_i x_i(\xi) + g_i y_i(\xi) + h_i s_i(\xi) + p_i r_i(\xi)) && \forall \xi \in \Xi \\
& && s_{i+1}(\xi) = x_i(\xi) - d_i(\xi) + s_i(\xi) - r_{i-1}(\xi) + r_i(\xi) && \forall i \in H, \forall \xi \in \Xi, \\
& && x_i(\xi) \leq M y_i(\xi) && \forall i \in H, \forall \xi \in \Xi, \\
& && y(\xi) \in \{0, 1\}^n, x(\xi), s(\xi), r(\xi) \geq 0 && \forall \xi \in \Xi, \\
& && s_1(\xi) = r_n(\xi) = 0 && \forall \xi \in \Xi.
\end{aligned}$$

Hence, a formulation for problem \mathcal{PI}_1 can be obtained from the above formulation by removing the dependency of ξ from y , while a formulation for \mathcal{PI}_0 is obtained by removing the variables y and constraints associated with it. This approach is well-known in stochastic optimization to examine the quality of proposed solutions. In particular, it is used to compute the so-called *expected value of perfect information* which defines the maximum price that one would be ready to pay to obtain perfect information about the actual scenario, see Birge and Louveaux (2011).

We discuss in the next section how to solve problems \mathcal{PI}_0 , \mathcal{PI}_1 and \mathcal{PI} .

4 Solving the problem with perfect information

Let us first introduce some useful definitions.

- The *cumulative cost* w_{ij} represents the unitary cost of producing at time period i to satisfy the demand of time period j :

$$w_{ij} = c_i + \sum_{l=i}^{j-1} h_l + \sum_{l=j}^{i-1} p_l.$$

Notice that the above is well-defined, since for a fixed period i , $i \neq j$, only one of the two summations is not empty.

- We denote by ω_j the *minimum cumulative cost* for period j , the smallest among values $\{w_{ij}, i \in H\}$.
- We denote by $\text{opt}(X)$ the optimal solution cost of any optimization problem X .

In the following, we discuss how to solve the optimization problems obtained by relaxing the non-anticipativity constraints. We first focus on problem \mathcal{PI}_0 , then we address problem \mathcal{PI}_1 , and we finish with problem \mathcal{PI} . For each problem, we present a generic solution algorithm that can handle general uncertainty polytopes and more efficient algorithms that are tailored for the budgeted uncertainty polytope.

4.1 No setup

We first deal with the robust problem without setup costs \mathcal{PI}_0 , which can be formulated as follows

$$\begin{aligned}
(\mathcal{PI}_0) \quad & \min && \kappa \\
& s.t && \kappa \geq \sum_{i \in H} (c_i x_i(\xi) + h_i s_i(\xi) + p_i r_i(\xi)) && \forall \xi \in \Xi \\
& && s_{i+1}(\xi) = x_i(\xi) - d_i(\xi) + s_i(\xi) - r_{i-1}(\xi) + r_i(\xi) && \forall i \in H, \forall \xi \in \Xi, \\
& && x(\xi), s(\xi), r(\xi) \geq 0 && \forall i \in H, \forall \xi \in \Xi, \\
& && s_1(\xi) = r_n(\xi) = 0 && \forall \xi \in \Xi.
\end{aligned}$$

We show that \mathcal{PI}_0 is equivalent to problem

$$\max_{\xi \in \Xi} \sum_{i \in H} \omega_i d_i(\xi),$$

where ω_i is the minimum cumulative cost of period i as defined previously. Hence, the result shows that the complexity of \mathcal{PI}_0 is related to the complexity of optimizing an affine function over Ξ .

Theorem 1 *Let Ξ be any uncertainty set. Then,*

$$\text{opt}(\mathcal{PI}_0) = \max_{\xi \in \Xi} \sum_{i \in H} \omega_i d_i(\xi)$$

Proof Let us introduce the optimization variable $v_{ij}(\xi)$ that denotes the fraction of demand $d_j(\xi)$ that is produced at period i . Using these variables and the vector w defined above, \mathcal{PI}_0 can be reformulated as follows:

$$\begin{aligned}
& \min && \kappa \\
& s.t && \kappa \geq \sum_{j \in H} d_j(\xi) \sum_{i \in H} w_{ij} v_{ij}(\xi) && \forall \xi \in \Xi \\
& && \sum_{i \in H} v_{ij}(\xi) = 1 && \forall j \in H, \forall \xi \in \Xi, \\
& && v(\xi) \geq 0 && \forall \xi \in \Xi.
\end{aligned}$$

For each $j \in H$, let $i^*(j) = \arg \min_{i \in H} w_{ij}$. We claim that the optimal solution of the above problem satisfies, for each $\xi \in \Xi$, $v_{ij}(\xi) = 1$ if $i = i^*(j)$ and $v_{ij}(\xi) = 0$ if $i \in H \setminus \{i^*(j)\}$. Hence, the above problem boils down to

$$\min \left\{ \kappa : \kappa \geq \sum_{j \in H} d_j(\xi) \omega_j, \forall \xi \in \Xi \right\},$$

obtaining the result.

We prove the claim by considering a solution v that satisfies $v_{i'j'}(\xi') = \epsilon > 0$ for some $i' \in H \setminus \{i^*(j')\}$ and $\xi' \in \Xi$. Then, we define the solution $\bar{v}_{ij}(\xi)$ equal to $v_{ij}(\xi)$ if $\xi \neq \xi'$ or $j \neq j'$, and satisfying $\bar{v}_{i'j'}(\xi') = 0$ and $\bar{v}_{i^*(j')j'}(\xi') = v_{i^*(j')j'}(\xi') + \epsilon$. By definition,

$$\sum_{i \in H} w_{ij} v_{ij}(\xi) = \sum_{i \in H} w_{ij} \bar{v}_{ij}(\xi) \quad (17)$$

if $\xi \neq \xi'$ or $j \neq j'$. Furthermore,

$$\begin{aligned} \sum_{i \in H} w_{ij} \bar{v}_{ij'}(\xi') &= w_{i^*(j')j'} \bar{v}_{i^*(j')j'}(\xi') + \sum_{i \in H \setminus \{i^*(j')\}} w_{ij'} \bar{v}_{ij'}(\xi') \\ &= w_{i^*(j')j'} (v_{i^*(j')j'}(\xi') + v_{i'j'}(\xi')) + \sum_{i \in H \setminus \{i^*(j')\}} w_{ij'} v_{ij'}(\xi') \\ &\leq w_{i^*(j')j'} v_{i^*(j')j'}(\xi') + w_{i'j'} v_{i'j'}(\xi') + \sum_{i \in H \setminus \{i^*(j')\}} w_{ij'} v_{ij'}(\xi') \end{aligned} \quad (18)$$

$$= \sum_{i \in H} w_{ij} v_{ij'}(\xi'), \quad (19)$$

where (18) follows from the definition of $i^*(j')$. Relations (17) and (19) lead to

$$\max_{\xi \in \Xi} \sum_{j \in H} d_j(\xi) \sum_{i \in H} w_{ij} \bar{v}_{ij}(\xi) \leq \max_{\xi \in \Xi} \sum_{j \in H} d_j(\xi) \sum_{i \in H} w_{ij} v_{ij}(\xi),$$

so that solution \bar{v} is not more expensive than solution v . \square

Theorem 1 implies that \mathcal{PI}_0 is polynomially solvable in the input since linear programming is polynomially solvable. We show in the next result that we can get faster algorithms for Ξ_Γ .

Corollary 1 *Let $\Xi = \Xi_\Gamma$ and define the subset $\Gamma(\omega^T \hat{D}) \subseteq H$ that contains the indices of the Γ largest elements of the vector $\omega^T \hat{D}$, and $\Gamma'(\omega^T \hat{D})$ that denotes the $(\Gamma+1)$ -largest element of that vector. The following holds:*

$$\text{opt}(\mathcal{PI}_0) = \sum_{i \in H} \omega_i \bar{d}_i + \sum_{i \in \Gamma(\omega^T \hat{D})} (\omega^T \hat{D})_i + (\Gamma - \lfloor \Gamma \rfloor) \omega^T \hat{D}_{\Gamma'(\omega^T \hat{D})}.$$

Moreover, $\text{opt}(\mathcal{PI}_0)$ can be computed in $O(n^2)$.

Proof We obtain immediately from Theorem 1 that

$$\begin{aligned} \text{opt}(\mathcal{PI}_0) &= \max_{\xi \in \Xi_\Gamma} \sum_{i \in H} \omega_i d_i(\xi) \\ &= \max_{\xi \in \Xi_\Gamma} \sum_{i \in H} \left(\omega_i \bar{d}_i + \sum_{j \in H} \omega_i \hat{D}_{ij} \xi_j \right) \\ &= \sum_{i \in H} \omega_i \bar{d}_i + \max_{\xi \in \Xi_\Gamma} \sum_{i,j \in H} \omega_i \hat{D}_{ij} \xi_j \\ &= \sum_{i \in H} \omega_i \bar{d}_i + \max_{\substack{i \in H \\ -1 \leq \xi_i \leq 1}} \sum_{\substack{\xi_i \leq \Gamma \\ i,j \in H}} \omega_i \hat{D}_{ij} \xi_j \\ &= \sum_{i \in H} \omega_i \bar{d}_i + \sum_{i \in \Gamma(\omega^T \hat{D})} (\omega^T \hat{D})_i + (\Gamma - \lfloor \Gamma \rfloor) \omega^T \hat{D}_{\Gamma'(\omega^T \hat{D})} \end{aligned}$$

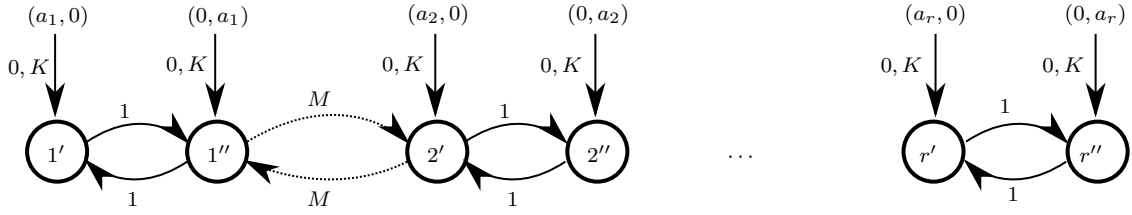


Fig. 1: Reduction of the partition problem to $\mathcal{P}\mathcal{I}_1$.

Regarding the complexity, we must first compute all cumulative costs, which takes $O(n^2)$. Then, we must compute the minimum cumulative cost for every period, which takes $O(n^2)$. Finally, we must compute the product $\omega^T \hat{D}$, which takes $O(n^2)$ and choose the Γ larger values of it, which takes $O(\Gamma \log n)$. Hence, the complexity of this strategy is $O(3n^2 + \Gamma \log n) = O(n^2)$. \square

4.2 Non-adjustable setup

In this section, we address problem $\mathcal{P}\mathcal{I}_1$, which can be formulated as

$$\begin{aligned}
 (\mathcal{P}\mathcal{I}_1) \quad & \min && \kappa \\
 & s.t. && \kappa \geq \sum_{i \in H} (c_i x_i(\xi) + g_i y_i + h_i s_i(\xi) + p_i r_i(\xi)) && \forall \xi \in \Xi \\
 & && s_{i+1}(\xi) = x_i(\xi) - d_i(\xi) + s_i(\xi) - r_{i-1}(\xi) + r_i(\xi) && \forall i \in H, \forall \xi \in \Xi, \\
 & && x_i(\xi) \leq M y_i && \forall i \in H, \forall \xi \in \Xi, \\
 & && x(\xi), s(\xi), r(\xi) \geq 0 && \forall \xi \in \Xi, \\
 & && y \in \{0, 1\}^n, \\
 & && s_1(\xi) = r_n(\xi) = 0 && \forall \xi \in \Xi.
 \end{aligned}$$

As in the previous section, we first address $\mathcal{P}\mathcal{I}_1$ for general polytopes and show that the problem is \mathcal{NP} -hard and can be reformulated as mixed-integer linear program. We present then a polynomial approach that is applicable to Ξ assuming that \hat{D} is a diagonal matrix. We address first the complexity of $\mathcal{P}\mathcal{I}_1$, using a reduction from the partition problem.

Theorem 2 *The problem $\mathcal{P}\mathcal{I}_1$ is \mathcal{NP} -hard.*

Proof Let $R = \{1, \dots, r\}$ and consider a set of positive integers $\{a_i, i \in R\}$. The partition problem looks for a subset $S \subset R$ such that

$$\sum_{i \in S} a_i = \sum_{i \in R \setminus S} a_i. \quad (20)$$

Finding out whether such a subset S exists is \mathcal{NP} -complete, see for instance Garey and Johnson (2002). We define the corresponding instance of $\mathcal{P}\mathcal{I}_1$ by considering $H = \{1', 1'', \dots, r', r''\}$, setting the production costs c to 0, the fixed costs g to K (to be defined later), the holding and backloging costs between i' and i'' to 1 for each $i \in R$, while those between periods i'' and $(i+1)'$ are set to a large number M (to be defined later), for each $i = 1, \dots, r-1$. Finally, $\bar{d} = 0$, \hat{D} is the identity matrix, and $\Xi \subset \mathcal{R}_+^{2r}$ is defined as the convex hull of the two vectors $\xi' = (a_1, 0, a_2, 0, \dots, a_r, 0)$ and $\xi'' = (0, a_1, 0, a_2, \dots, 0, a_r)$, see Figure 1. One readily verifies that the above polytope Ξ can be obtained through $4r + 2$ linear inequalities: the first $4r$ inequalities characterize the line joining ξ' and ξ'' while the last two inequalities bound the line to obtain the required segment. It is well-known (e.g. Ayoub and Poss (2016)) that we can restrict ourselves to the two extreme points of Ξ when analyzing the optimal solution of the problem and its cost.

Let $A = \sum_{i=1}^r a_i$. Choosing M large enough and $K = \max_{i=1, \dots, r} a_i + 1$, we prove below that there exists a subset S of R that satisfies (20) if and only if the optimal solution cost of the above instance of \mathcal{PT}_1 is equal to $rK + \frac{A}{2}$. Let us first show that, in any optimal solution to \mathcal{PT}_1 , either $y_{i'} = 1$ and $y_{i''} = 0$, or $y_{i'} = 0$ and $y_{i''} = 1$, for each $i \in R$. Specifically, setting $y_{i'} = 1$ and $y_{i''} = 0$ yields a cost of a_i for scenario ξ' and 0 for scenario ξ'' , in addition to the fixed cost K . Similarly, setting $y_{i''} = 1$ and $y_{i'} = 0$ yields a cost of 0 for scenario ξ' and a_i for scenario ξ'' , in addition to the fixed cost K . Hence, both approaches cost $K + a_i$ in the worst-case. This is always less costly than setting $y_{i'} = y_{i''} = 0$, which costs at least Ma_i in both scenarios, or setting $y_{i'} = y_{i''} = 1$, which costs $2K$ in both scenarios.

Let $S \subseteq R$ denote the elements for which $y_{i'} = 1$, so that $R \setminus S$ contains the elements of R for which $y_{i''} = 1$. We see that the cost of the solution described by S is equal to

$$rK + \max \left(\sum_{i \in S} a_i, \sum_{i \in R \setminus S} a_i \right).$$

Hence, the cost of S is minimized if and only if (20) holds, proving the result \square

The above result is in line with classical results on robust combinatorial optimization problems (e.g. Kouvelis and Yu (2013)) which show that the robust counterparts of polynomially solvable optimization problems turn \mathcal{NP} -hard for arbitrary uncertainty sets. This being said, it is possible to solve \mathcal{PT}_1 for general uncertainty polytopes through a mixed-integer linear programming reformulation. The first step to obtain the reformulations relies on reformulating \mathcal{PT}_1 as static robust optimization problem, hence, removing the adjustable variables from the problem.

Lemma 1 *Problem \mathcal{PT}_1 can be reformulated as*

$$\begin{aligned} \min & \left(\max_{\xi \in \Xi} \sum_{i,j \in H} w_{ij} v_{ij} d_j(\xi) + \sum_{i \in H} g_i y_i \right) \\ \text{s.t.} & \sum_{i \in H} v_{ij} = 1 \quad \forall j \in H, \end{aligned} \quad (21)$$

$$v_{ij} \leq y_i \quad \forall i, j \in H, \quad (22)$$

$$y \in \{0, 1\}^n, v \geq 0. \quad (23)$$

Proof Let us define the binary variable $v_{ij}(\xi)$ that is equal to 1 iff the demand of period j is produced at period i . Using a reasoning similar to the one used in the proof of Theorem 1, we can reformulate \mathcal{PT}_1 as

$$\begin{aligned} \min_{y \in \{0,1\}^n} \max_{\xi \in \Xi} \min_{v(\xi)} & \sum_{i,j \in H} w_{ij} v_{ij}(\xi) d_j(\xi) + \sum_{i \in H} g_i y_i \\ \text{s.t.} & \sum_{i \in H} v_{ij}(\xi) = 1 \quad \forall j \in H, \forall \xi \in \Xi, \end{aligned} \quad (24)$$

$$v_{ij}(\xi) \leq y_i \quad \forall i, j \in H, \forall \xi \in \Xi, \quad (25)$$

$$v_{ij}(\xi) \in \{0, 1\} \quad \forall i, j \in H, \forall \xi \in \Xi.$$

where the first term of the objective function represents the cost of attending the demand of all periods while the second term represents the fixed costs. Notice then that, for any $y \in \{0, 1\}^n$, we always have that $v_{ij}(\xi) = 1$ iff

$$w_{ij} = \min_{i' \in H} \{w_{i'j} : y_{i'} = 1\},$$

which does not depend on ξ . Hence, $v_{ij}(\xi) = v_{ij}(\xi')$ for each pair $\xi, \xi' \in \Xi$ and we can remove the dependency on ξ from $v_{ij}(\xi)$. Moreover, we can remove the binary restrictions on v because constraints (24) and (25) are formed by a totally unimodular matrix. Applying von Neumann's minmax theorem, we can exchange the maximization over ξ and the minimization over v obtaining the desired result. \square

Let $\theta \in \mathbb{R}^m$ denote the dual variables associated to the constraints that characterize Ξ . Dualizing the inner maximization problem obtained in Lemma 1, we obtain immediately a MILP reformulation for \mathcal{PT}_1 .

Corollary 2 Let Ξ be the uncertainty polytope defined in (12). The problem $\mathcal{P}\mathcal{I}_1$ can be solved by the following mixed integer linear program

$$\begin{aligned}
\text{opt}(\mathcal{P}\mathcal{I}_1) &= \min \sum_{l=1}^m q_l \theta_l + \sum_{i,j \in H} v_{ij} w_{ij} \bar{d}_j + \sum_{i \in H} g_i y_i \\
\text{s.t.} \quad & \sum_{l=1}^m W_{lk} \theta_l \geq \sum_{i,j \in H} v_{ij} w_{ij} \hat{D}_{jk} && \forall k \in H, \\
& \sum_{i \in H} v_{ij} = 1 && \forall j \in H, \\
& v_{ij} \leq y_i && \forall i, j \in H, \\
& y \in \{0, 1\}^n, v \geq 0.
\end{aligned}$$

We devote the rest of the section to the study of uncertainty set Ξ_Γ , further assuming that \hat{D} is a diagonal matrix. We show that, under these assumptions, problem $\mathcal{P}\mathcal{I}_1$ can be solved in polynomial time. Our approach requires the following extension of a classical result from Bertsimas and Sim (2003) that shows how some robust combinatorial optimization problems can be solved by solving a polynomial number of times the deterministic counterpart. The result considers generic optimization problems defined by the optimization variables $z \in \{0, 1\}^{|L|}$ and the uncertainty polytope $\Xi_\Gamma \subset \mathbb{R}^{|K|}$, where K and L are arbitrary index sets. Its proof is similar to the proof of Proposition 1 from Bougeret et al (2016) and is therefore deferred to Appendix C.

Lemma 2 Consider the index sets K and L , and consider the subset $K(l) \subseteq K$ for each $l \in L$. Let $\mathcal{Z} \subseteq \{0, 1\}^{|L|}$ be the feasibility set and $\alpha \in \mathcal{R}_+^{|L|}$ and $\beta \in \mathcal{R}_+^{|K| \times |L|}$ be the cost vectors that characterize the robust optimization problem

$$\min_{z \in \mathcal{Z}} \max_{\xi \in \Xi_\Gamma} \sum_{l \in L} \left(\alpha_l + \sum_{k \in K(l)} \beta_{kl} \xi_k \right) z_l. \quad (26)$$

Define the subset $L(k) = \{l \in L : k \in K(l)\}$ for each $k \in K$. If any element $z \in \mathcal{Z}$ satisfies the constraints

$$\sum_{l \in L(k)} z_l = 1, \quad k \in K,$$

then the optimal solution cost to problem (26) is given by

$$\min_{\{(k', l') : k' \in K, l' \in L(k)\} \cup \{(0, 0)\}} G_{k'l'}$$

where

$$G_{k'l'} = \Gamma \beta_{k'l'} + \min_{z \in \mathcal{Z}} \sum_{l \in L} \left(\alpha_l + \sum_{k \in K(l)} \max(0, \beta_{kl} - \beta_{k'l'}) \right) z_l$$

for each $\{(k', l') : k' \in K, l' \in L(k)\}$ and where $\beta_{00} = 0$.

To use Lemma 2, we should reformulate problem $\mathcal{P}\mathcal{I}_1$ as a special case of problem (26). In this aim, we use the shortest path formulation for the deterministic problem $\mathcal{L}\mathcal{S}$, presented in Pochet and Wolsey (2013), and modify that formulation to handle the robust aspect as required. This is presented in the following theorem.

Theorem 3 Let $\Xi = \Xi_\Gamma$ and assume that \hat{D} is a diagonal matrix. Then, problem $\mathcal{P}\mathcal{I}_1$ can be solved by solving $O(n^2)$ shortest path problems in an acyclic digraph that contains n nodes.

Proof Let us first consider the case without backlog. Consider the digraph $G = (V, A)$, where $V = \{1, \dots, n+1\}$ and $A = \{(i, j) | i < j; i, j \in V\}$. When the demand is known with precision $(d(\xi) = \bar{d})$, it is known (see Section 7.4 from Pochet and Wolsey (2013)) that the optimization problem stated in Lemma 1 can be reformulated as a shortest path problem between vertices 1 and $n+1$ in G where the cost of taking arc (i, j) is equal to $g_i + \sum_{k=i}^j w_{ik} \bar{d}_k$, which means that all demands $i \leq k \leq j$ are attended by producing at time period $i \in H$. Letting $\mathcal{Z} \subset \{0, 1\}^{|A|}$ be the set of incidence vectors of all paths from 1 to $n+1$ in G , the deterministic problem can thus be reformulated as

$$\min_{z \in \mathcal{Z}} \sum_{(i,j) \in A} \left(g_i + \sum_{k=i}^j w_{ik} \bar{d}_k \right) z_{ij}. \quad (27)$$

Since any demand $k \in H$ is attended by producing at a unique period $i \in H$, we see that any $z \in \mathcal{Z}$ must satisfy the constraints

$$\sum_{\substack{(i,j) \in A \\ i \leq k \leq j}} z_{ij} = 1, \quad \forall k \in H.$$

Thanks to Lemma 1, one readily verifies that the robust counterpart of the problem can be similarly reformulated as the min max robust counterpart of problem (27), namely

$$\min_{z \in \mathcal{Z}} \max_{\xi \in \Xi} \sum_{(i,j) \in A} \left(g_i + \sum_{k=i}^j w_{ik} d_k(\xi) \right) z_{ij} = \min_{z \in \mathcal{Z}} \max_{\xi \in \Xi} \sum_{(i,j) \in A} \left(g_i + \sum_{k=i}^j w_{ik} \bar{d}_k + \sum_{k=i}^j w_{ik} \hat{D}_{kk} \xi_k \right) z_{ij}, \quad (28)$$

where the equality follows from the diagonal assumption on \hat{D} . The result follows directly from applying Lemma 2 to (28).

To extend this reasoning to \mathcal{PL}_1 with backlog, we replace the simple digraph G defined above by the multi digraph $H = (V, A')$ where $A' = \{(i, \ell, j) | i < j; i, j \in V; i \leq \ell \leq j\}$. Then, the deterministic cost of each arc (i, ℓ, j) is given by $g_\ell + \sum_{k=i}^j w_{\ell k} d_k(\xi)$, which represents the fact that we fulfill the demands from period i to j by producing at period ℓ . The rest of the proof is similar, yielding the result. \square

4.3 Adjustable setup

In this section we present a linear program that computes the value of $opt(\mathcal{PL})$ for general polytopes Ξ . Similarly to Theorem 1, we show that the complexity of problem \mathcal{PL} is related to the complexity of optimizing a linear function over Ξ . This time however, the linear program to be solved contains $O(n^3)$ constraints in addition to those describing Ξ . Let us first recall a well-known dynamic programming algorithm to solve the deterministic problem \mathcal{LS} , taken from Pochet and Wolsey (2013).

Lemma 3 *Problem \mathcal{LS} can be solved by the following recursive function*

$$opt(\mathcal{LS}) = G(n)$$

where

$$G(j) = \min_{\{i,k | i < k \leq j\}} \left(G(i) + g_k + \sum_{l=i+1}^j w_{kl} d_l \right), \quad (29)$$

where $G_0 = 0$.

To compute \mathcal{PL} , it will be useful to reformulate the above dynamic program as the linear program provided in the next lemma.

Lemma 4 *The optimal solution cost of \mathcal{LS} is equal to the optimal solution cost of the following linear program, with optimization variables u_i for each $i \in H$*

$$\begin{aligned} \max \quad & u_n \\ \text{s.t.} \quad & u_j \leq u_i + g_k + \sum_{l=i}^j w_{kl} d_l \quad \forall i < k \leq j, i \in H \cup \{0\}, j \in H. \\ & u_0 = 0 \end{aligned}$$

Proof We introduce optimization variable u_i to represent the value of $G(i)$. The counterpart of equation (29) for u is

$$u_j = \min_{\{i,k \mid i \leq k \leq j\}} \left(u_i + g_k + \sum_{l=i}^j w_{kl} d_l \right). \quad (30)$$

Then, we relax the equality in (30) to

$$u_j \leq \min_{\{i,k \mid i \leq k \leq j\}} \left(u_i + g_k + \sum_{l=i}^j w_{kl} d_l \right), \quad (31)$$

and ensure that the objective function of the linear program maximizes the value of u_j . Constraint (31) can be linearized to

$$u_j \leq u_i + g_k + \sum_{l=i}^j w_{kl} d_l \quad \forall i \leq k \leq j, i \in H. \quad (32)$$

Combining linear constraints (32) with the objective function that maximizes u_n and adding the initial condition u_0 yields the result. \square

Using the above results, we propose a linear programming reformulation for \mathcal{PI} .

Theorem 4 *For any uncertainty set Ξ , problem \mathcal{PI} can be solved by the following linear program in optimization vectors u and ξ*

$$\begin{aligned} \mathcal{PI} = \max \quad & u_n \\ \text{s.t.} \quad & u_j \leq u_i + g_k + \sum_{l=i}^j w_{kl} d_l(\xi) \quad \forall i \leq k \leq j, i \in H, j \in H, \\ & u_0 = 0, \\ & \xi \in \Xi. \end{aligned}$$

Proof We define $\mathcal{PI}(\xi)$, as the problem \mathcal{PI} restricted to a fixed element $\xi \in \Xi$. One readily sees that

$$\text{opt}(\mathcal{PI}) = \max_{\xi \in \Xi} \text{opt}(\mathcal{PI}(\xi))$$

and moreover, $\mathcal{PI}(\xi)$ is a deterministic lot-sizing problem. Hence, we can apply Lemma 4 to solve $\mathcal{PI}(\xi)$ as the linear program

$$\begin{aligned} \text{opt}(\mathcal{PI}(\xi)) = \max \quad & u_n \\ \text{s.t.} \quad & u_j \leq u_i + g_k + \sum_{l=i}^j w_{kl} d_l(\xi) \quad \forall i \leq k \leq j, i \in H, j \in H, \\ & u_0 = 0. \end{aligned} \quad (33)$$

Now, as $\text{opt}(\mathcal{PI}) = \max_{\xi \in \Xi} \text{opt}(\mathcal{PI}(\xi))$ we have that

$$\begin{aligned} \text{opt}(\mathcal{PI}) = \max \quad & u_n \\ \text{s.t.} \quad & u_j \leq u_i + g_k + \sum_{l=i}^j w_{kl} d_l(\xi) \quad \forall i \leq k \leq j, i \in H, j \in H, \\ & \xi \in \Xi, \\ & u_0 = 0. \end{aligned}$$

Recalling that $d(\xi)$ is an affine function of ξ ends the proof. \square

5 Numerical experiments

In this section, we present the experimental results obtained with the relaxation presented in the last sections for problems \mathcal{P}_1 and \mathcal{P}_0 . We use the budgeted polytope Ξ_Γ as uncertainty set due to its importance in the literature and the identity as the deviation matrix, so that the demand can be expressed as $d_i(\xi) = \bar{d}_i + \hat{d}_i \xi_i$.

We compare the quality of our lower bound with the one provided by dual affine decision rules from Kuhn et al (2011), recalled in Section 3.2. We also compare these lower bounds with the upper bound provided by the affine decisions rules (called primal affine decision rules in Kuhn et al (2011)), recalled in 3.1. We do not carry out experiments for problem \mathcal{P} because the primal and dual affine decision rules methods cannot be applied to this problem so that we have no comparison possible for our method.

Concerning the probabilistic distribution used in dual affine decision rules, we use a uniform distribution over the extremes points of the uncertainty polytope. The reasons for that are two-fold: first we need a distribution that allows us to easily compute matrix \mathcal{M} (the expectation matrix); second, we do not have a dominating element among the extreme points of the considered polytope, contrasting with the problem studied in Kuhn et al (2011).

The tests were carried out on an Intel(R) Core(TM) i7 CPU M60, 2.6Hz 4GB Ram machine and all formulations and algorithms were coded in C++, compiled with a GNU G++ 4.5 compiler and IBM CPLEX 12.3. In the next subsection, we explain how the instances are built and which experiments are carried out.

5.1 Instances

We start with the description of the instances. We consider two sets of instances: DYN and DOWN. The instances in the set DYN represent lot-sizing problems in which the costs associated are seasonal. The instances of that set are inspired by Ben-Tal et al (2004) and they fulfill a criteria known as *Wagner-Within*, which has been introduced in Pochet and Wolsey (1994). Roughly speaking, this criteria implies that it is always cheaper to produce the client demands of each period at the period itself. More precisely, we have for each period i that $c_i = 20 + 5 \sin(\frac{i\pi}{12})$, $h_i = 5 + 2 \sin(\frac{i\pi}{12})$, $p_i = 7 + 2 \sin(\frac{i\pi}{12})$, and $g_i = 30 + 10 \sin(\frac{i\pi}{12})$. Notice that these instances are similar to the instances used in several papers on robust lot-sizing, e.g. Kuhn et al (2011); de Ruiter et al (2017). In particular, they model a product whose demand varies seasonally.

To contrast with that set of instances, we use a second set of instances that do not fulfill the Wagner-Within criteria. In the set of instances named DOWN, for each each period i is cheaper to produce the client demand at period $3\lfloor \frac{i}{3} \rfloor$. More precisely, we have that $c_i = 10 + 5(i \bmod 3)$, $h_i = 3$, $p_i = 4$, and $g_i = 50$.

We consider the deviations fixed as 20% of the nominal demand, and the nominal deviations are given by the formula $100 + 50 \sin(\frac{i\pi}{12})$ for each period i . We consider horizons of planning that have 20, 30, 40, 50, 60, 70, 80, 90 and 100 periods. We tested two different values of the parameter Γ for each number of periods in the horizon of planning, each of them inspired by the probabilistic bounds computed in Bertsimas and Sim (2004), see also Poss (2013). Table 1 presents the values used for the experiments.

H	$\Gamma^{0.01}$	$\Gamma^{0.1}$
10	5	8
20	7	11
30	8	14
40	9	16
50	10	17
60	11	19
70	12	20
80	12	22
90	13	23
100	14	24

Table 1: Values of parameter Γ inspired by the probabilistic bounds from Bertsimas and Sim (2004).

5.2 Lot-Sizing problem without setup costs

We report below a comparison of the two lower bounds and as well as their solution times. Let *AFFINE* stand for the affine decision rules presented in the Section 3.1, *PI* for the perfect information proposed and *DUAL* for the dual affine decision rules presented in Section 3.2. We compare the *optimality gaps* of *PI* and *DUAL* using the solution of the approach *AFFINE* as upper bound. Specifically, we define the *approximative optimality gap* for *PI* for the instance *I* as

$$\frac{\text{opt}(\text{AFFINE}(I)) - \text{opt}(\text{PI}(I))}{\text{opt}(\text{AFFINE}(I))}.$$

We define similarly the approximative approximation gap for the problem *DUAL* concerning the instance *I* as

$$\frac{\text{opt}(\text{AFFINE}(I)) - \text{opt}(\text{DUAL}(I))}{\text{opt}(\text{AFFINE}(I))}.$$

In the following, we report the approximative optimality gap and the solution times, computed for each number of periods and each set of instances.

5.2.1 Instances DOWN

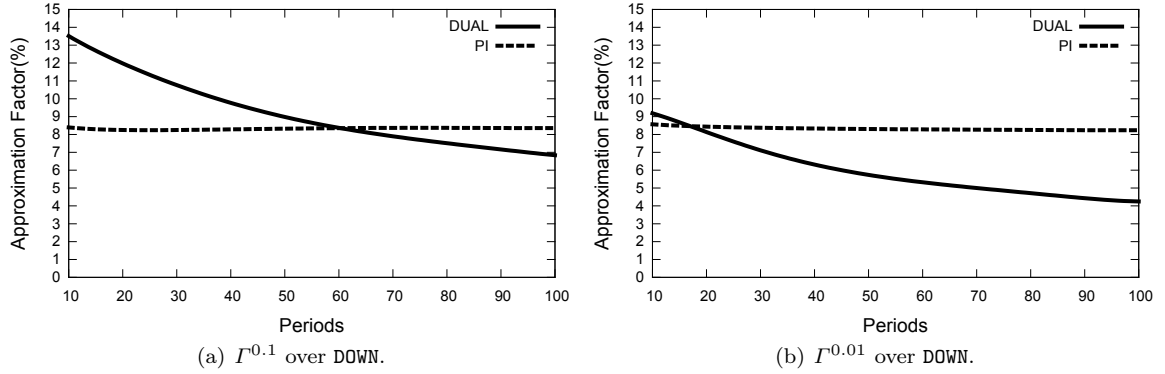


Fig. 2: The optimality gaps for \mathcal{PI}_0 for the instances in DOWN.

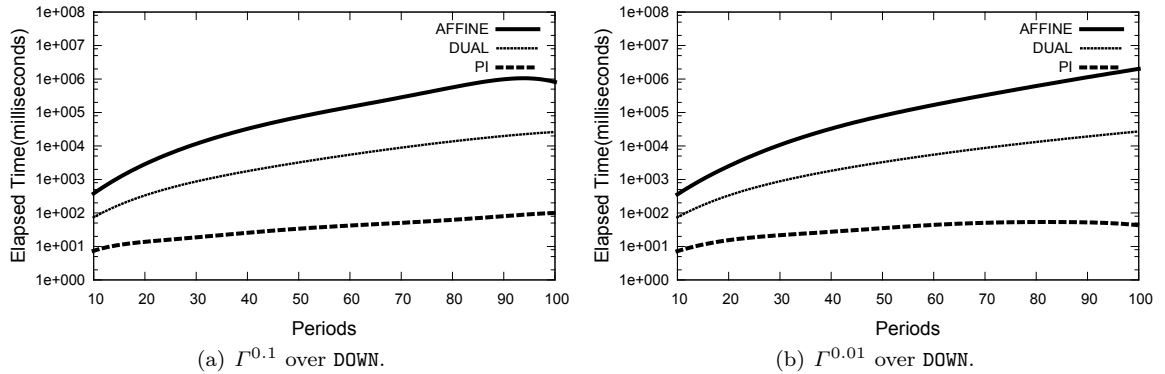


Fig. 3: Elapsed time of \mathcal{PI}_0 for the instances in DOWN.

The approximation obtained by *PI* is nearly constant at 8% while the one of *DUAL* is slightly decreasing with the number of periods, see Figure 2. In any case, approach *PI* still provides a reliable

approximation with a computation time much smaller than the one required by *DUAL* as presented in Figure 3

5.2.2 Instances DYN

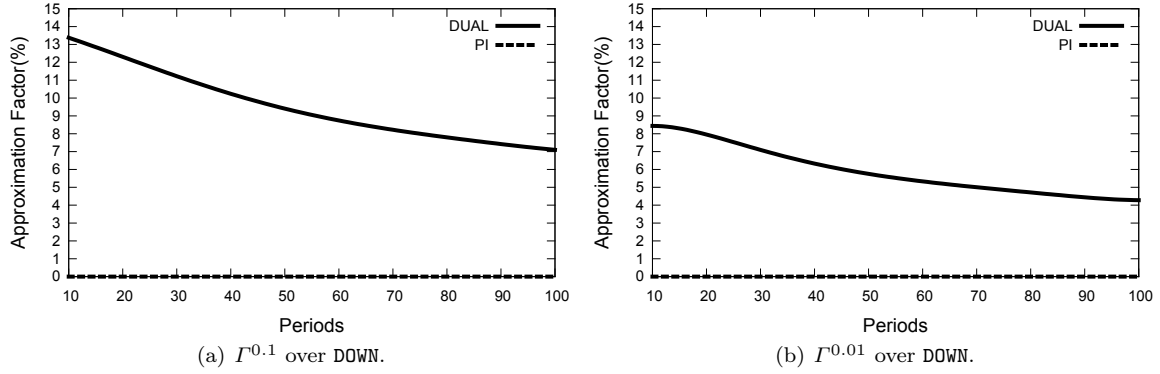


Fig. 4: The optimality gaps for \mathcal{PT}_0 for the instances in DYN.

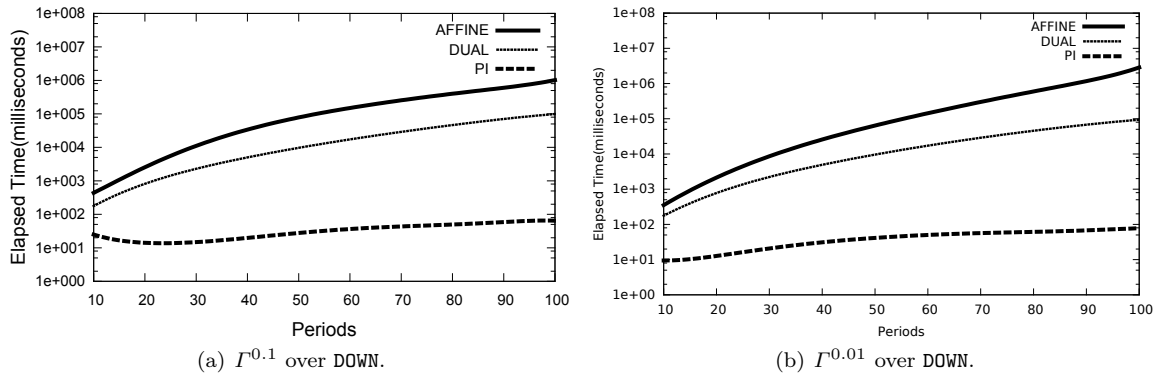


Fig. 5: Elapsed time of \mathcal{PT}_0 for the instances in DYN.

The bound provided by *PI* can prove the optimality of the affine decisions rules while the dual affine decision rules decrease to reach a gap of 5%, see in Figure 4. It seems that the Wagner-Within criteria, where it is also always cheaper to produce the demand of some period i at the period itself, strongly reduces the impact of the non-anticipativity constraints. As for the computation time, our approach is much faster than the other two, as reported in Figure 5. One can also notice that the dual affine decision rules are more time consuming for the instance set DYN than they are for the instance set DOWN.

5.3 Lot-Sizing problem with setup costs

As before, *AFFINE* stands for the affine decision rules presented in the Section 3.1, *PI* for the perfect information proposed in this paper and *DUAL* for the dual affine decision rules presented in Section 3.2. We compare the *optimality gaps* of *PI* and *DUAL* using the solution cost of *AFFINE* as upper bound.

In the following, we report the approximative optimality gap and the solution times, computed for each number of periods and each set of instances. Two remarks must be done at this point. First, we solve the problem *PI* with the MILP proposed in Corollary 2 instead of the polynomial algorithm described in

Theorem 3. We do that because unreported results show the MILP to be much faster than the dynamic programming algorithm proposed in Theorem 3 for our instances. Specifically, the dynamic programming algorithm roughly has the same running time as the dual affine decision rules, while the MILP is much faster, as reported below. Second, we impose a time limit of 20 minutes for all problems. The affine decision rules formulations are only able to solve problems with 50 periods or less in that amount of time, so that we only report the results for the instances having up to 50 periods.

5.3.1 Instances DOWN

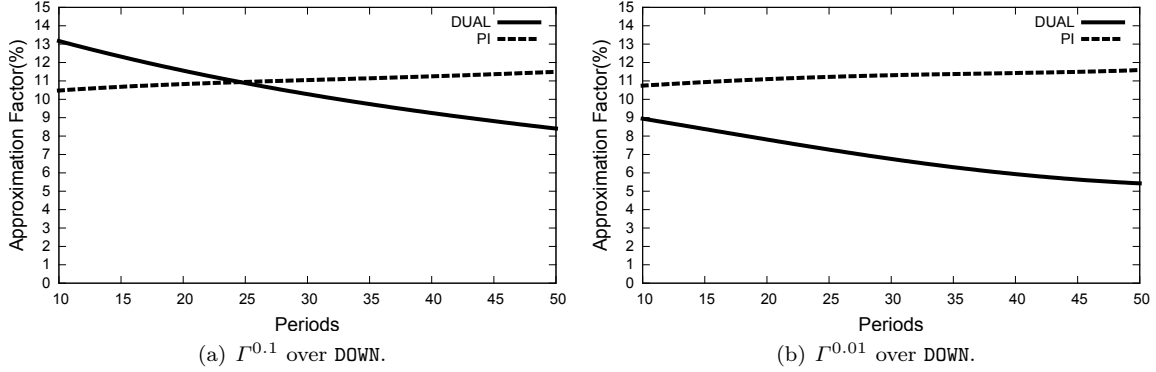


Fig. 6: The optimality gaps for \mathcal{PI}_0 for the instances in DOWN.

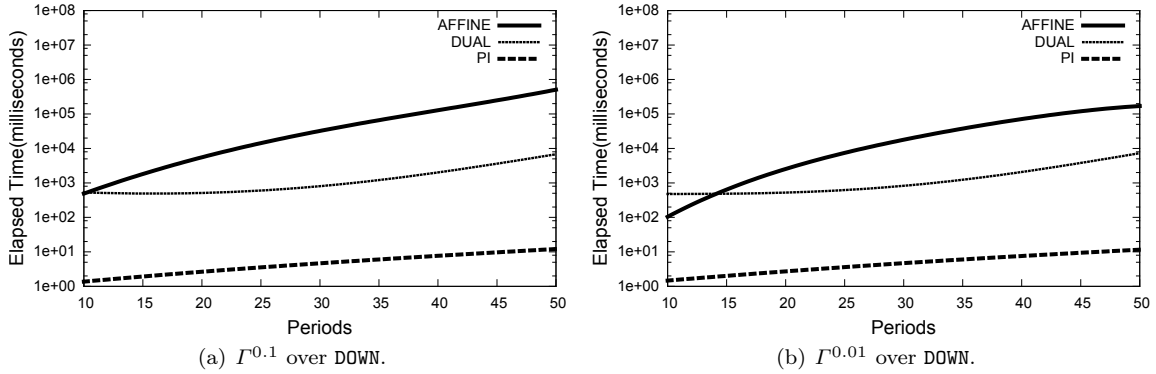


Fig. 7: Elapsed time of \mathcal{PI}_0 for the instances in DOWN.

The approximation obtained by PI is nearly constant at 11% while the one of $DUAL$ is decreasing with the number of periods, see Figure 6. Figure 7 then shows that PI can be solved much faster than $DUAL$.

5.3.2 Instances DYN

Although we are not able to prove the optimality of the affine decision rules, the lower bound provided by PI is really close to the solution cost provided by $AFFINE$ (roughly 1%). And again, concerning the elapsed time, the proposed method is 2 or 3 magnitude orders faster than the dual affine decision rules, see Figure 9.

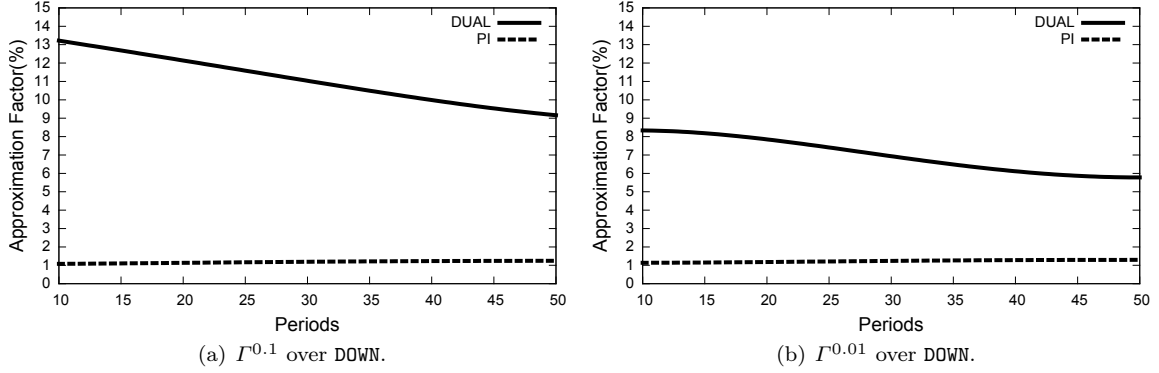


Fig. 8: The optimality gaps for \mathcal{PT}_0 for the instances in DYN.

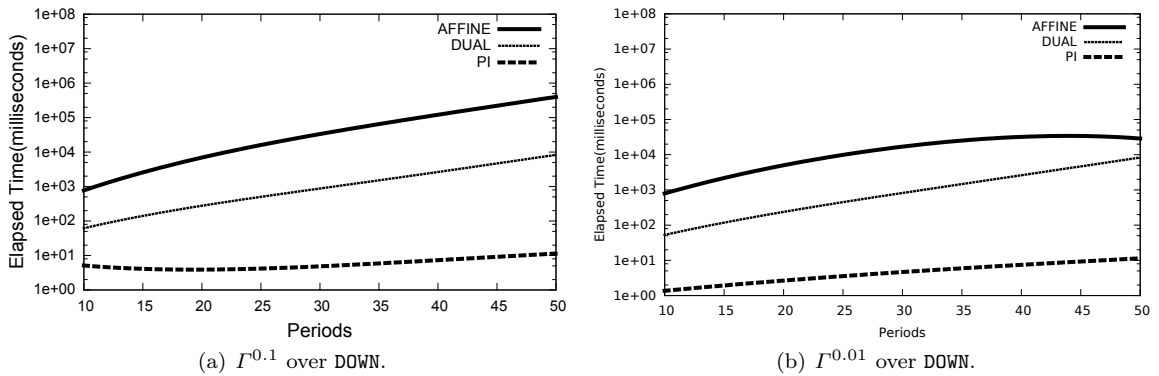


Fig. 9: Elapsed time of \mathcal{PT}_0 for the instances in DYN.

6 Concluding remarks

We have adapted in this paper the perfect information relaxation, well-known in the stochastic programming literature, to the robust lot-sizing problem, yielding a lower bound for the robust problem. We could prove that computing the bound can be done in polynomial time for several variants of the problem, using dedicated combinatorial algorithms or linear programs. Our numerical results, realized on instances inspired by the literature, suggest that the new lower bound can be tight and can be solved much faster than the bounds based on decision rules. In addition, our approach can handle binary adjustable variables, which is not the case of primal and dual affine decision rules.

The tractability of our lower bound is highly problem-dependent. Hence, it would be interesting to investigate its tractability for other class of multi-stage robust optimization problems and other uncertainty sets. Concerning the extensions of the problems under study, we think it may be interesting to consider problems with capacities on the production and the storage. An alternative extension would consider problems with multiple items. Concerning the uncertainty set, it could be interesting to look for dedicated algorithms for ellipsoidal uncertainty sets, which have recently been the topic of several papers in the combinatorial robust optimization literature (e.g. Baumann et al (2014)).

Another interesting aspect of the perfect information relaxation is that it turns multi-stage problems into two-stages problems, for which several papers have recently proposed exact solution algorithms based on variants of the Benders decomposition (e.g. Ayoub and Poss (2016); Billionnet et al (2014); Zeng and Zhao (2013)). Hence, one could use these algorithms to solve the relaxation for much more general problems than the variants of the lot-sizing considered herein, and assess the quality of the obtained lower bound.

References

- Agra A, Christiansen M, Figueiredo RMV, Hvattum LM, Poss M, Requejo C (2013) The robust vehicle routing problem with time windows. *Computers & OR* 40(3):856–866, DOI 10.1016/j.cor.2012.10.002, URL <http://dx.doi.org/10.1016/j.cor.2012.10.002>
- Agra A, Santos MC, Nace D, Poss M (2016) A dynamic programming approach for a class of robust optimization problems. *SIAM Journal on Optimization* 26(3):1799–1823
- Avriel M, Williams A (1970) The value of information and stochastic programming. *Operations Research* 18(5):947–954
- Ayoub J, Poss M (2016) Decomposition for adjustable robust linear optimization subject to uncertainty polytope. *Computational Management Science* 13(2):219–239
- Baumann F, Buchheim C, Ilyina A (2014) Lagrangean decomposition for mean-variance combinatorial optimization. In: *Combinatorial Optimization - Third International Symposium, ISCO 2014, Lisbon, Portugal, March 5-7, 2014, Revised Selected Papers*, pp 62–74
- Ben-Tal A, Nemirovski A (1998) Robust convex optimization. *Mathematics of Operations Research* 23(4):769–805
- Ben-Tal A, Nemirovski A (2000) Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical Programming* 88(3):411–424, DOI 10.1007/PL00011380, URL <http://dx.doi.org/10.1007/PL00011380>
- Ben-Tal A, Goryashko A, Guslitzer E, Nemirovski A (2004) Adjustable robust solutions of uncertain linear programs. *Math Program* 99(2):351–376
- Bertsimas D, Dunning I (2014) Multistage robust mixed integer optimization with adaptive partitions
- Bertsimas D, Georghiou A (2015) Design of near optimal decision rules in multistage adaptive mixed-integer optimization. *Operations Research* 63(3):610–627, DOI 10.1287/opre.2015.1365, URL <http://dx.doi.org/10.1287/opre.2015.1365>
- Bertsimas D, Sim M (2003) Robust discrete optimization and network flows. *Mathematical Programming* 98(1):49–71, DOI 10.1007/s10107-003-0396-4, URL <http://dx.doi.org/10.1007/s10107-003-0396-4>
- Bertsimas D, Sim M (2004) The price of robustness. *Operations Research* 52(1):35–53
- Bertsimas D, Thiele A (2006) A robust optimization approach to inventory theory. *Oper Res* 54(1):150–168
- Bienstock D, Özbay N (2008) Computing robust basestock levels. *Discrete Optimization* 5(2):389 – 414, in *Memory of George B. Dantzig*
- Billionnet A, Costa M, Poirion P (2014) 2-stage robust MILP with continuous recourse variables. *Discrete Applied Mathematics* 170:21–32
- Birge JR, Louveaux F (2011) *Introduction to stochastic programming*. Springer Science & Business Media
- Bougeret M, Pessoa A, Poss M (2016) Robust scheduling with budgeted uncertainty. *Discrete Applied Mathematics* Conditionally accepted. Available online at <https://hal.archives-ouvertes.fr/hal-01345283/document>
- C C Huang WTZ I Vertinsky (1977) Sharp bounds on the value of perfect information. *Operations Research* 25(1):128–139
- Chen X, Zhang Y (2009) Uncertain linear programs: Extended affinely adjustable robust counterparts. *Operations Research* 57(6):1469–1482
- Garey MR, Johnson DS (2002) *Computers and intractability*, vol 29. wh freeman New York
- Goh J, Sim M (2010) Distributionally robust optimization and its tractable approximations. *Oper Res* 58(4-Part-1):902–917
- Gorissen BL, den Hertog D (2013) Robust counterparts of inequalities containing sums of maxima of linear functions. *European Journal of Operational Research* 227(1):30–43
- Kouvelis P, Yu G (2013) *Robust discrete optimization and its applications*, vol 14. Springer Science & Business Media
- Kuhn D, Wiesemann W, Georghiou A (2011) Primal and dual linear decision rules in stochastic and robust optimization. *Mathematical Programming* 130:177–209
- Oostenbrink JB, Al MJ, Oppe M, Rutten-van Mölken MP (2008) Expected value of perfect information: an empirical example of reducing decision uncertainty by conducting additional research. *Value in Health* 11(7):1070–1080

- Pochet Y, Wolsey L (2013) *Production Planning by Mixed Integer Programming*. Springer Science & Business Media
- Pochet Y, Wolsey LA (1994) Polyhedra for lot-sizing with wagner-whitin costs. *Math Program* 67:297–323, DOI 10.1007/BF01582225, URL <http://dx.doi.org/10.1007/BF01582225>
- Poss M (2013) Robust combinatorial optimization with variable budgeted uncertainty. *4OR* 11(1):75–92, DOI 10.1007/s10288-012-0217-9, URL <http://dx.doi.org/10.1007/s10288-012-0217-9>
- Postek K, den Hertog D (2016) Multistage adjustable robust mixed-integer optimization via iterative splitting of the uncertainty set. *INFORMS Journal on Computing* 28(3):553–574, DOI 10.1287/ijoc.2016.0696, URL <http://dx.doi.org/10.1287/ijoc.2016.0696>
- de Ruiter FJCT, Ben-Tal A, Brekelmans R, den Hertog D (2017) Robust optimization of uncertain multistage inventory systems with inexact data in decision rules. *Comput Manag Science* 14(1):45–66, DOI 10.1007/s10287-016-0253-6, URL <https://doi.org/10.1007/s10287-016-0253-6>
- Zeng B, Zhao L (2013) Solving two-stage robust optimization problems by a constraint-and-column generation method. *Operations Research Letters* 41(5):457–461

A Reformulation of affine decision rules

Plugging (14)–(16) into \mathcal{P} and enforcing that y does not depend on ξ , we obtain the formulation below.

$$\begin{aligned}
\min_{\kappa} \quad & \kappa \\
s.t. \quad & \kappa \geq \sum_{i \in H} \left(g_i y_i + c_i (x_i^0 + \sum_{j=1}^i x_i^j \xi_j) + h_i (s_i^0 + \sum_{j=1}^i s_i^j \xi_j) + p_i (r_i^0 + \sum_{j=1}^i r_i^j \xi_j) \right) & \forall \xi \in \Xi \\
& s_i^0 + \sum_{j=1}^i s_i^j \xi_j \geq \sum_{k=1}^i \left(x_k^0 - \bar{d}_k + \sum_{j=1}^{l_k^x} x_k^j \xi_j - \sum_{l=1}^n \hat{D}_{kl} \xi_l \right) & \forall i \in H, \forall \xi \in \Xi \\
& r_i^0 + \sum_{j=1}^i r_i^j \xi_j \geq \sum_{k=1}^i \left(\bar{d}_k - x_k^0 - \sum_{j=1}^{l_k^x} x_k^j \xi_j + \sum_{l=1}^n \hat{D}_{kl} \xi_l \right) & \forall i \in H, \forall \xi \in \Xi \\
& x_i^0 + \sum_{j=1}^i x_i^j \xi_j \leq y_i M & \forall i \in H, \forall \xi \in \Xi \\
& x_i^0 + \sum_{j=1}^i x_i^j \xi_j \geq 0 & \forall i \in H, \forall \xi \in \Xi \\
& s_i^0 + \sum_{j=1}^i s_i^j \xi_j \geq 0 & \forall i \in H, \forall \xi \in \Xi \\
& r_i^0 + \sum_{j=1}^i r_i^j \xi_j \geq 0 & \forall i \in H, \forall \xi \in \Xi \\
& y_i \in \{0, 1\} & \forall i \in H,
\end{aligned}$$

Although the number of variables in the problem described above is polynomial, we still have to deal with an infinite number of constraints. As said already, one easily sees that we can restrict ourselves to the extreme points of Ξ ; yet, this typically leads to an exponential number of constraints. An alternative and more compact approach applies classical tools from robust optimization to reformulate each robust constraint as a polynomial number of deterministic constraints plus a polynomial number of additional real variables, see Ben-Tal and Nemirovski (2000). We skip the details of that approach and provide below the resulting MILP.

$$\begin{aligned}
\min_{\kappa} \quad & \kappa \\
s.t. \quad & \kappa - \sum_{i \in H} (g_i y_i + c_i x_i^0 + h_i s_i^0 + p_i r_i^0) \geq \sum_{i=1}^m q_i \theta_i \\
& \theta^T W_i \geq \sum_{j=i}^n (c_j x_j + h_j s_j + p_j r_j) \quad \forall i, 0 \leq i \leq m \\
& s_i^0 - \sum_{k=1}^i (x_k^0 - \bar{d}_k) \geq \sum_{j=1}^m q_j \alpha_j^i \quad \forall i \in H \\
& \alpha^T W_j \geq s_i^j + \sum_{k=i}^j x_k \quad \forall i, j, 0 \leq j \leq i, i \in H \\
& r_i^0 - \sum_{k=1}^i (\bar{d}_k - x_k^0) \geq \sum_{j=1}^m q_j \beta_j^i \quad \forall i \in H \\
& \beta^T W_j \geq r_i^j - \sum_{k=i}^j x_k \quad \forall i, j, 0 \leq j \leq i, i \in H \\
& x_i^0 + y_i M \geq \sum_{j=1}^m q_j \pi_j^i \quad \forall i \in H \\
& \pi^T W_j \geq x_i^j \quad \forall i, j, 0 \leq j \leq i, i \in H \\
& x_i^0 \geq \sum_{j=1}^m q_j \theta_j^{x,i} \quad \forall i \in H \\
& (\theta^x)^T W_j \geq x_i^j \quad \forall i, j, 0 \leq j \leq i, i \in H \\
& s_i^0 \geq \sum_{j=1}^m q_j \theta_j^{s,i} \quad \forall i \in H \\
& (\theta^r)^T W_j \geq x_i^j \quad \forall i, j, 0 \leq j \leq i, i \in H \\
& r_i^0 \geq \sum_{j=1}^m q_j \theta_j^{r,i} \quad \forall i \in H \\
& (\theta^s)^T W_j \geq x_i^j \quad \forall i, j, 0 \leq j \leq i, i \in H \\
& y_i \in \{0, 1\} \quad \forall i \in H,
\end{aligned}$$

We can obtain similarly the formulation for the problem without setup by removing the variable y from the above formulation.

B Reformulation of dual affine decision rules

In this appendix, we sketch how to obtain a MILP reformulation for the lower bound of the robust lot-sizing following the method presented in Kuhn et al (2011). The method requires to introduce an artificial probability measure with our uncertainty set Ξ . To keep the presentation simple, we will exemplify the framework from Kuhn et al (2011) with the discrete probability measure $\mathbb{P}(\xi)$ defined over the set of extreme points of Ξ . Hence, \mathbb{P} is any vector in $\mathbb{R}_+^{|\text{ext}(\Xi)|}$ that satisfies

$$\sum_{\xi \in \text{ext}(\Xi)} \mathbb{P}(\xi) = 1.$$

The first step of the approach of Kuhn et al (2011) relaxes the robust constraints to expectation constraints as follows.

$$\begin{aligned}
& \min_{\kappa} \quad \kappa \\
s.t. \quad & \mathbb{E} \left(\left[\kappa - \sum_{i \in H} (g_i y_i + c_i x_i(\xi) + h_i s_i(\xi) + p_i r_i(\xi) + \pi(\xi)) \right] \xi \right) = 0 \\
& \mathbb{E} \left(\left[s_i(\xi) - \sum_{k=1}^i (x_k(\xi) - d_k(\xi)) + \alpha_i(\xi) \right] \xi \right) = 0 \quad \forall i \in H, \\
& \mathbb{E} \left(\left[r_i(\xi) + \sum_{k=1}^i (x_k(\xi) - d_k(\xi)) + \beta_i(\xi) \right] \xi \right) = 0 \quad \forall i \in H, \\
& \mathbb{E} ([x_i(\xi) - y_i M + \psi_i(\xi)] \xi) = 0 \quad \forall i \in H, \\
& \mathbb{E} ([x_i(\xi) + \theta_i^x(\xi)] \xi) = 0 \quad \forall i \in H, \\
& \mathbb{E} ([s_i(\xi) + \theta_i^s(\xi)] \xi) = 0 \quad \forall i \in H, \\
& \mathbb{E} ([r_i(\xi) + \theta_i^r(\xi)] \xi) = 0 \quad \forall i \in H, \\
& y_i \in \{0, 1\} \quad \forall i \in H,
\end{aligned}$$

where ψ and θ are additional slack variables. One readily verifies that, for any probability distribution $\mathbb{P}(\xi)$, the above problem provides a lower bound for the optimal solution of problem \mathcal{PT}_1 .

Then, using advanced dualization techniques inspired by the dualization used in classical robust optimization, the authors of Kuhn et al (2011) are able to reformulate the above lower bound as the following mixed-integer linear program, where matrix \mathcal{M} is defined as $\mathcal{M} = \mathbb{E}(\xi\xi^T)$, \mathcal{W} is the matrix defined as $\mathcal{W} = (W - qe_1^T)\mathcal{M}$, and e_1 is the vector with all entries equals to one. Specifically, the following mixed-integer linear program is obtained by applying reformulation (4.6) from Kuhn et al (2011) to \mathcal{PT}_1 .

$$\begin{aligned}
& \min_{\kappa} \quad \kappa \\
s.t. \quad & \kappa - \sum_{i \in H} (g_i y_i + c_i x_i^0 + h_i s_i^0 + p_i r_i^0) + \pi_0 = 0 \\
& -\pi_j = \sum_{i \in H} (c_i x_i^j + h_i s_i^j + p_i r_i^j) \quad \forall j \in H \\
& s_i^0 = \sum_{k=1}^i (x_k^0 - \bar{d}_k) + \alpha_i^0 \quad \forall i \in H \\
& s_i^j = \sum_{k=1}^i x_k^j - \sum_{l=1}^n \hat{D}_{jl} + \alpha_i^j \quad \forall i, j \in H \\
& r_i^0 = \sum_{k=1}^i (\bar{d}_k - x_k^0) + \beta_i^0 \quad \forall i \in H \\
& r_i^j = \sum_{k=1}^i (-x_k^0) + \sum_{l=1}^n \hat{D}_{jl} + \beta_i^j \quad \forall i, j \in H \\
& My - x_i^0 + \pi_i^0 = 0 \quad \forall i \in H \\
& -x_i^j + \pi_i^j = 0 \quad \forall i, j \in H \\
& x_i^0 + \theta_i^{x,0} = 0 \quad \forall i \in H \\
& s_i^j + \theta_i^{s,j} = 0 \quad \forall i \in H, \forall j \in H \cup \{0\} \\
& r_i^j + \theta_i^{r,j} = 0 \quad \forall i \in H, \forall j \in H \cup \{0\} \\
& \mathcal{W}\alpha^T = 0 \\
& \mathcal{W}\beta^T = 0 \\
& \mathcal{W}\pi^T = 0 \\
& \mathcal{W}(\theta^x)^T = 0 \\
& \mathcal{W}(\theta^s)^T = 0 \\
& \mathcal{W}(\theta^r)^T = 0 \\
& y_i \in \{0, 1\} \quad \forall i \in H.
\end{aligned}$$

We can obtain similarly the formulation for the problem without setup by removing the variable y from the above formulation.

C Proof of Lemma 2

Let us detail the inner maximization of (26) as

$$\begin{aligned}
\sum_{l \in L} \alpha_l z_l + \max \quad & \sum_{k \in K} \xi_k \sum_{l \in L(k)} \beta_{kl} z_l \\
\text{s.t.} \quad & \sum_{k \in K} \xi_k \leq \Gamma, \\
& \xi_k \leq 1, \quad k \in K, \\
& \xi_k \geq -1, \quad k \in K.
\end{aligned} \tag{34}$$

By definition, $\beta_{kl} \geq 0$ for each $k \in K, l \in L(k)$, so that we can relax constraints (34) to $\xi \geq 0$ without affecting the optimal solution. Thanks to the strong duality in linear programming, the optimal solution cost of the above problem is equal to the optimal solution cost of its dual, given by

$$\begin{aligned}
\min \quad & \Gamma \theta + \sum_{k \in K} \varphi_k \\
\text{s.t.} \quad & \theta + \varphi_k \geq \sum_{l \in L(k)} \beta_{kl} z_l, \quad k \in K \\
& \theta, \varphi_k \geq 0.
\end{aligned}$$

Substituting φ_k by $\max(0, \sum_{l \in L(k)} \beta_{kl} z_l - \theta)$ for each $k \in K$, we can further reformulate (26) as

$$\min_{x \in \mathcal{Z}, \theta \geq 0} \Gamma \theta + \sum_{l \in L} \alpha_l z_l + \sum_{k \in K} \max \left(0, \sum_{l \in L(k)} \beta_{kl} z_l - \theta \right). \tag{35}$$

The crucial step of our proof (which differs from Theorem 3 from Bertsimas and Sim (2003)) is that, because the constraint $\sum_{l \in L(k)} z_l = 1$ holds for each $k \in K$, we can further reformulate (35) as

$$\min_{x \in \mathcal{Z}, \theta \geq 0} \Gamma \theta + \sum_{l \in L} \alpha_l z_l + \sum_{k \in K} \sum_{l \in L(k)} z_l \max(0, \beta_{kl} - \theta).$$

The rest of the proof is identical to the proof of Theorem 3 from Bertsimas and Sim (2003).