Stochastic Optimization for Material Requirements Planning

Abstract

This paper investigates stochastic optimization methods for Material Requirements Planning (MRP) systems under demand uncertainty. MRP systems are widely used by manufacturers to determine the lot sizes of components. These lot sizes are typically computed based on deterministic and dynamic demand assumptions, whereas safety stocks, which hedge against demand uncertainty, are determined independently based on different assumptions. As the lot sizes and safety stocks are not determined simultaneously, sub-optimal decisions are used in practice. The critical impact of inventories and service levels in manufacturing motivates the study of stochastic optimization methods for MRP. In this paper, a two-stage and a multi-stage model are proposed to deal with the static-static and static-dynamic decision frameworks, respectively. We first derive structural properties of the two-stage and multi-stage models, in particular when the two-stage model is used in the static-dynamic setting, to provide theoretical insights on when the multi-stage model can yield improved results compared to the two-stage model. As stochastic programming models are far from being convenient in real-world applications, several practical enhancements are proposed. First, to address scalability issues, a fix-and-optimize approach is proposed in combination with advanced scenario sampling methods. Second, to allow real-time static-dynamic decisions, a policy is derived from the solution of the multi-stage model. Third, to tackle large-scale problems, a rolling horizon heuristic is employed. Extensive computational experiments show that the stochastic optimization approach presented in this paper outperform the approaches in the literature and the common approaches used in practice. These experimental results and structural analyses allow us to give useful managerial insights on the value of the stochastic optimization for MRP.

Keywords: Material requirements planning; stochastic optimization; lot-sizing; uncertain demand

1 Introduction

This paper investigates the use of stochastic optimization in the context of Material Requirements Planning (MRP) systems. MRP software has been adopted by the majority of manufacturers to plan the production in short or medium-term horizons. Given the demands for end items, MRP systems compute the sizes of the lots to produce (or to order) for each component in each period. These calculations are based on the bills of materials (BOM) which indicate the hierarchy of components (i.e., the number of components required to produce each end item or component). The problem solved by MRP systems is a multi-echelon multi-item lot-sizing problem (Tempelmeier and Derstroff 1996). Early implementations of MRP solved this problem item by item with heuristic lot-sizing rules based on simple logic. For instance, the lot-for-lot rule sets the production quantities to the requirements of each period. As these rules perform poorly in the presence of shared resource capacities, recent implementations of MRP allow automatic planning by solving the lot-sizing problem with a mixed-integer linear program (MILP). As the MILP approach is flexible, it allows to model multiple extensions and constraints (e.g., Balakrishnan and Geunes 2000). These problems are commonly solved under the assumption of deterministic demand, since safety stocks are calculated separately to hedge against uncertainty. However, the existing safety stock computation methods for multi-echelon production systems do not consider the key decision components of MRP systems. In fact, the applicable methods (e.g., Graves and Willems 2008) are often designed for base stock policies. As the application of a base stock policy follows simple rules, the level of stock can be written as a function of the stochastic demand, and the safety stock is computed to meet a given service level. This approach is not suitable for advanced MRP systems, where the lot sizes must be determined in a complex environment with multiple echelons, setup costs, capacities, flow conservation constraints, etc. (see Section 2). Consequently, sub-optimal safety stock levels, which are not determined in conjunction with the lot sizes, are often used in practice. These safety stock levels are computed either manually, at the master production schedule (MPS) level, or under the assumption of a base stock policy (e.g., Graves and Willems 2008).

The contributions of this paper are threefold. First, we design scenario based stochastic optimization approaches for the *multi-echelon multi-item capacitated lot-sizing problem* (MMCLP) with lead times and stochastic demand. As these methods take into account the demand's stochasticity in the lot-sizing model, they remove the boundary between the lot sizes and safety stocks computations. We derive theoretical insights on when the multi-stage model, including its fix-and-optimize implementation, can yield improved results compared to the two-stage model. Second, the scalability problems arising with conventional stochastic programming approaches are addressed here with a heuristic, advanced scenario sampling methods, an execution policy derived from the solution of the model, as well as a rolling horizon solution framework to deal with the problems with a long planning horizon and to determine a solution in a dynamic-dynamic decision framework, the most challenging variant of the stochastic multi-stage lot-sizing model. Therefore, the proposed approaches allow to solve practical size instances. Third, this paper is the first to present an extensive computational evaluation of stochastic optimization approaches for the practical multi-echelon MRP setting. Our experiments show that stochastic optimization approaches reduce the costs significantly when compared to the classical methods. In addition, these approaches perform well in dynamic environments, whereas re-planning with classical methods leads to larger costs than freezing the decisions. Since the MRP logic is also used in the Distribution Resources Planning (DRP) systems, the solution approaches presented in this study can also be applied directly in the DRP systems under demand uncertainty. The rest of this section gives more details regarding each of these contributions.

This paper introduces two stochastic programming formulations representing two decision frameworks, referred to as *static-static*, and *static-dynamic* in the lot-sizing literature (e.g., Tempelmeier 2013). The static-static environment is encountered when a frozen period is considered. That is, the production quantities and setups are decided at time zero and fixed for the entire time horizon. The static-dynamic situation occurs when setups are linked to long-term decisions and must be fixed, whereas production quantities can be adjusted in each period. More precisely, the setups are decided at time zero for the entire planning horizon, whereas the production quantities for period $t + 1$ are decided after having observed the demands of period t. Examples of long-term decisions linked with the setups include the planning of secondary resources, such as the models in 3D printing, the molds in injection moldings, or the technicians who set up the production lines in various manufacturing sectors. Indeed, planning these secondary resources requires knowledge of the items produced in each period.

A two-stage (resp. multi-stage) stochastic program based on scenario sampling is proposed to model the static-static (resp. static-dynamic) environment. As the multi-stage model is difficult to solve (due to a large number of variables and constraints), a fix-and-optimize heuristic is proposed. Furthermore, to avoid solving the mathematical model in each period in the static-dynamic decision framework, an order-up-to-level policy (denoted S-Policy) is derived from the solution of the multi-stage model. The S-Policy is compared with a policy (denoted Q-Policy) where the ordered quantities are fixed to the values computed with the two-stage model. A theoretical analysis supports the design of the fix-and-optimize method and S-policy. In addition, three scenario sampling methods are investigated to find good quality solutions with a smaller number of scenarios, namely, crude Monte Carlo (CMC), quasi-Monte Carlo (QMC), and randomized quasi-Monte Carlo (RQMC). The present work is the first to consider the adaptations of these methods to solve the multi-echelon multi-item capacitated lot sizing problem under stochastic demands and its performance in this important application. More importantly, we demonstrate that QMC is inferior to RQMC and even to the simple CMC in the multi-stage setting. We also present the results of the scenario-generation methods in comparison with other applicable approaches.

The stochastic optimization approaches are compared with classical methods, namely, the deterministic mathematical model and lot-sizing rules (lot-for-lot, economic order quantity, economic order period, Silver-Meal) equipped with safety stocks. The use of the considered methods is simulated with a large number of scenarios using well-known academic benchmarks. Besides the static-static and static-dynamic decision frameworks, the considered models are also evaluated with a rolling horizon simulation in the *dynamic-dynamic* environment, where the plan is re-optimized in each period. Table 1 gives the model, solution approaches, and evaluation methodology for each decision framework. Note that the simulations in the static-dynamic and dynamic-dynamic decision frameworks are computationally intensive, since they require to re-solve the problem in each period. In contrast, with the static-static decision framework, the cost of a solution is simply observed under each scenario. These experiments complement the theoretical insights gained from the structural properties of the problem. They show that the multi-stage model slightly outperforms the two-stage model, but the two-stage model requires less computation effort. However, the multi-stage model significantly outperforms the two-stage model when the demand uncertainty is large, when the value added at each production step is large, and when components can be transformed into end items in few periods. The experiments also show that solving static-dynamic MMCLP with a sufficient number of scenarios leads to a good approximation of the stochastic process. In addition, advanced scenario sampling techniques (such as RQMC) allow to reduce the number of required scenarios. Consequently, the use of RQMC in conjunction with the fix-and-optimize heuristics allows to get a good approximation of the stochastic process, despite the large number of possible stochastic states. Finally, the results show that the execution policy derived from the multi-stage model (which allows to make the recourse decisions on a real time basis) outperforms other methods which do not require to solve a mathematical model during the execution.

Decision framework	**Model**	**Solution approaches**	**Evaluation**
Static-static	Two-stage	Two-stage	Observe (No-replanning)
Static-dynamic	Multi-stage	Two-stage, Multi-stage, Fix-and-optimize, S-Policy	Re-solve
Dynamic-dynamic	-	Two-stage, Multi-stage, Fix-and-optimize, S-Policy, Q-Policy	Rolling horizon simulation

Table 1: Considered decision framework.

This paper is organized as follows. Section 2 gives a review of previous work on stochastic MRP and stochastic multi-echelon lot-sizing problems. Section 3 formally describes the considered problem and presents the stochastic optimization models along with the structural analyses. Section 4 presents the scenario sampling approaches, the proposed heuristic, and the order-up-to-level policy. Section 5 presents the methods used to benchmark stochastic optimization approaches, and the simulation framework. Section 6 reports the experimental results. Finally, the conclusion follows in Section 7.

2 Literature Review

Mathematical models for material requirement planning have mostly been studied in a deterministic context (e.g., Zahorik et al. 1984, Billington et al. 1983, Clark and Armentano 1995). However, in practice, MRP systems are subject to diverse forms of uncertainty: demand, lead times, production yields, production capacity, among others (Guide and Srivastava 2000, Dolgui and Prodhon 2007). This literature review focuses on demand uncertainty in multi-echelon production systems, where the following three topics are successively covered: simulation studies on MRP in a stochastic demand context, safety stocks for MRP, and stochastic optimization approaches for multi-echelon MRP.

Several authors (e.g., Bai et al. 2002, Zhao and Lee 1993, Zhao et al. 2001, Enns 2002, Kadipasaoglu and Sridharan 1995, Ho and Ireland 1998) evaluate by simulation the impact of demand uncertainty on MRP systems for multi-echelon production problems. These studies also evaluate how the parameters (safety stocks, safety lead times, re-planning frequencies, frozen periods, lot-sizing rules) of classical MRP systems protect against demand uncertainty. The parameters considered by each of these papers are summarized in Table 2. The main conclusions of these works are the following. First, demand uncertainty and forecast errors have a significant impact on the costs and service levels. Second, safety stocks and safety lead times (i.e., considering buffer lead times in addition to the expected lead times) are efficient ways to protect against stochastic demand, but the choice of one versus the other depends on the considered system. In addition, most studies (e.g., Lagodimos and Anderson 1993, Bai et al. 2002, Zhao et al. 2001, Boulaksil 2016) advise to place safety stocks at the end item level, but some studies disagree. For instance, Carlson and Yano (1986) suggest to hold some safety stocks for components with large setup costs. Third, frequent re-planning with classical MRP systems is undesirable, because MRP systems are prone to nervousness (i.e., a minor change in the data leads to large modifications of the plan), and users tend not to trust a nervous system (Blackburn et al. 1985). Kadipasaoglu and Sridharan (1995) and Zhao and Lee (1993) showed that frequent re-planning leads to larger costs than infrequent re-planning, and that freezing a part of the master production schedule is the most efficient way to reduce nervousness.

Paper	**Counter measures**
Bai et al. (2002)	Frozen period, lot-sizing, safety stocks, planning horizon
Zhao and Lee (1993)	Frozen period, planning horizon, re-planning frequency
Zhao et al. (2001)	Safety stocks
Enns (2002)	Safety stocks, safety lead times, lot-sizing
Kadipasaoglu and Sridharan (1995)	Frozen period, safety stocks, lot-sizing
Ho and Ireland (1998)	Lot-sizing

Table 2: Previous studies on multi-echelon MRP systems with stochastic demand.

Although multiple studies suggest using safety stocks in MRP systems with demand uncertainty, to the best of our knowledge, no analytical method exists to directly determine the safety stocks in an MRP environment. In fact, the few works on safety stocks for MRP systems focus on special cases because the behavior of an MRP system is hard to model analytically (Benton 1991). For instance, Lagodimos and Anderson (1993) propose a safety stock computation approach for an MRP system with a lot-for-lot policy, constant demand, serial network, and no holding cost. Zijm and Van Houtum (1994) consider an MRP system with an order-up-to-level policy in assembly systems. Inderfurth (2009) studies a single-echelon MRP system with critical stock policy, where the production quantities are computed to bring the inventory levels above some critical thresholds. Other works on safety stocks for multi-echelon systems with non-stationary and uncertain demand (e.g., Inderfurth and Minner 1998, Graves and Willems 2008, Graves and Schoenmeyr 2016) focus on base stock policies. These approaches do not directly apply to the context of MRP, where the lot-sizing decisions have a significant impact on the risk of shortage.

Instead of computing the safety stock levels analytically, Benton (1991) and Boulaksil (2016) propose to use simulation methods. The lot-sizing problem is first solved based on the expected demand. Then, a simulation is run and the production quantities are adjusted to meet the desired service level. In the same vein, Sali and Giard (2015) revised the lot-for-lot rule to deal with uncertain demand. In their approach, lot sizes are computed to achieve the desired service level according to the cumulative distribution of the projected inventory levels.

Stochastic optimization models remove the need to use safety stocks because they account for the demand's probability distributions implicitly. In other words, the computation of safety stock levels and lot sizes are no longer isolated. Tempelmeier (2013) and Aloulou et al. (2014) review stochastic optimization methods for lot-sizing problems with uncertain demand. However, most studies (e.g., Brandimarte 2006), consider a single-echelon production system. To the best of our knowledge, the only work considering stochastic optimization for multi-echelon systems is presented in Grubbström and Wang (2003). Grubbström and Wang (2003) propose a dynamic programming approach to minimize the net present value in the capacitated multi-echelon lot-sizing problem with stochastic demand, but without lead times. However, stochastic optimization approaches have been proposed for related problems such as in aggregated production (e.g., Kaminsky and Swaminathan 2004) or supply chain management problem (e.g., Lin and Uzsoy 2016). In addition, other approaches than stochastic optimization have been proposed for the single item lot-sizing problem with stochastic demand such as distributionally robust optimization (e.g., Zhang et al. 2016).

Our work differs from the above literature in several aspects. First, to the best of our knowledge, this paper is the first to investigate scenario based multi-stage stochastic optimization for capacitated multi-echelon MRP systems with lead times and stochastic dynamic demand. These methods are useful for practitioners since they remove the boundary between safety stock and lot-size computations in MRP systems. Consequently, the proposed approaches remove the problems associated with safety stock computations in MRP systems. In addition, we compare the performance of a two-stage and a multi-stage formulation in static-static, static-dynamic, and dynamic-dynamic decision framework. To alleviate the scalability issues of stochastic optimization approaches, advanced sampling methods are considered, as well as a fix-and-optimize heuristic, and an execution policy. Finally, computationally intensive simulations are performed to compare (in terms of costs and KPIs) stochastic models with classical approaches such as lot-sizing rules, and deterministic models. The results show that stochastic optimization leads to a significant costs saving in MRP systems. In addition, the proposed methods are efficient and scalable.

3 Problem Formulation

This section describes the considered problem (Section 3.1), and the proposed stochastic optimization formulations modeling the static-static (Section 3.2) and static-dynamic (Section 3.3) decision frameworks.

3.1 Problem Description

The MMCLP is used to determine the production quantities Q_{it} for each item i in a set $\mathcal{I}$ and for each period t in the time horizon $\mathcal{H} = \{1, \dots, T\}$. The inputs of the model include BOM, lead times, probability distributions of the demands, and production capacities. We denote by $\mathcal{I} = \mathcal{I}_e \cup \mathcal{I}_c$ the set of items, where $\mathcal{I}_e$ and $\mathcal{I}_c$ are the sets of end items and of components, respectively. We assume (without loss of generality) that components have no external demand, whereas the probability distribution $\widetilde{D}_{it}$ of the demand is known for each end item $i \in \mathcal{I}_e$ and each period t. The BOM gives the hierarchy of components required for each end item, that is, the number R_{ij} of item i required to produce one unit of j. In addition, each item i has a lead time L_i, i.e., the production quantity Q_{it} is available for the next production step in period $t + L_i$. Finally, the production plan must respect the capacity C_k of each resource, given the resource consumption K_{ik} per unit of item i for each resource k in the set $\mathcal{K}$.

For each item i, inventory holding costs (h_i), fixed setup costs (s_i), and unit production costs (v_i) are considered. In addition, backlog costs (b_i), and lost sales costs (e_i) are also considered for the end items. As lead times are considered, the unit production cost of item i must account for the inventory holding costs of its components during the lead time (L_i). The unmet demand of item

i in period t is backlogged (it can be fulfilled in subsequent periods), but a penalty b_i is incurred in each period for each unit of backlog. In addition, a lost sale penalty e_i is incurred for each unit not delivered at the end of the horizon. Note that there is no constraint on the ordering quantities of raw materials (we assume that suppliers have an infinite capacity).

3.2 Two-stage Formulation for the MMCLP in a Static-Static Environment

The stochastic formulations of MMCLP are based on the set Ω of all possible demand scenarios. Given the probability p_w of each scenario ω (with $p_\omega > 0$ and $\sum_{\omega\in\Omega} p_\omega = 1$), the problem is to find the solution with the minimum expected total cost.

The static-static decision framework can be represented by a two-stage stochastic optimization model. The first-stage variables correspond to the decisions made in period 0 while the demand is unknown. In the static-static MMCLP, these decisions are the setup Y_{it} (variable equal to 1 if there is a setup, and 0 otherwise) and quantity Q_{it} for item i in period t. Second stage variables correspond to the inventory I_{it}^{ω} and backlog level B_{it}^{ω} of item i at the end of period t, observed after the realization of the demands to compute the cost for each scenario ω. B_{iT}^{ω} indicates the total remaining backlog quantity of item i for scenario ω at the end of the horizon, and this quantity can be interpreted as the lost sale. As the setup and quantity decisions are made before observing the demand, they are the same for all scenarios. On the contrary, the second stage variables can be different for each scenario. The deterministic capacitated multi-echelon lot-sizing problem is NP-hard, as it extends the capacitated lot-sizing problem which is itself NP-hard (Bitran and Yanasse 1982). As the stochastic versions extend the problem with multiple scenarios, MMCLP is also NP-hard. The problem can be formulated as the following MILP:

$$
\begin{aligned}
\min \quad & \sum_{\omega\in\Omega} p_\omega \left(\sum_{t\in\mathcal{H}} \sum_{i\in\mathcal{I}} (h_i I^\omega_{it} + s_i Y_{it} + v_i Q_{it}) + \sum_{i\in\mathcal{I}_e} \left(\sum_{t=1}^{t=T-1} b_i B^\omega_{it} + e_i B^\omega_{iT} \right) \right) && && (1)\\
\text{s.t.} \quad & \sum_{\tau=1}^{t-L_i} Q_{i\tau} + I_{i0} - \sum_{\tau=1}^{t} D^\omega_{i\tau} - I^\omega_{it} + B^\omega_{it} \quad = \quad 0 && i\in\mathcal{I}_e, \quad t\in\mathcal{H}, \quad \omega\in\Omega && (2)\\
& \sum_{\tau=1}^{t-L_i} Q_{i\tau} + I_{i0} - \sum_{\tau=1}^{t} \left(\sum_{j\in\mathcal{I}} R_{ij}\cdot Q_{j\tau} \right) - I^\omega_{it} \quad = \quad 0 && i\in\mathcal{I}_c, \quad t\in\mathcal{H}, \quad \omega\in\Omega && (3)\\
& Q_{it} \quad \le \quad M_i Y_{it} && i\in\mathcal{I}, \quad t\in\mathcal{H} && (4)\\
& \sum_{i\in\mathcal{I}} K_{ik} Q_{it} \le \quad C_k && t\in\mathcal{H}, \quad k\in\mathcal{K} && (5)\\
& B^\omega_{it} \quad \ge \quad 0 && i\in\mathcal{I}_e, \quad t\in\mathcal{H}, \quad \omega\in\Omega && (6)\\
& I^\omega_{it} \quad \ge \quad 0 && i\in\mathcal{I}, \quad t\in\mathcal{H}, \quad \omega\in\Omega && (7)\\
& Q_{it} \quad \ge \quad 0 \quad \text{and} \quad Y_{it} \quad \in \quad \{0,1\} && i\in\mathcal{I}, \quad t\in\mathcal{H}. && (8)
\end{aligned}
$$

The objective function (9) is the expected total cost over all scenarios, including inventory costs, setup costs, unit production costs, backlog costs, and end-of-horizon lost sales costs. Constraints (2) set the value of the backlog and inventory quantities for the end items. These values depend on the produced quantities and external demands. Constraints (3) set the inventory levels of components, which depend on the internal demands. Note that backlogs are not allowed for components since they are required for the planned production. Constraints (4) set the variable Y_{it} to 1 if the quantity of item i produced in period t is greater than 0. The value of M_i in constraints (4) is an upper bound of the production quantity of item i. This upper bound can be set to the minimum between the upper bound M^1_i (defined in Equation (9)) inferred from the demands of item i, and the upper bound M^2_i (defined in Equation (10)) inferred from the production capacities:

$$
M^1_i = \begin{cases} \max_{\omega\in\Omega} \sum_{t\in\mathcal{H}} D^\omega_{it} & \text{if } i\in\mathcal{I}_e \\ \sum_{j\in\mathcal{I}} R_{ij}\cdot M^1_j & \text{if } i\in\mathcal{I}_c \end{cases} \qquad (9) \qquad\qquad M^2_i = \min_{k\in\mathcal{K}|K_{ik}>0} \frac{C_k}{K_{ik}}. \qquad (10)
$$

The value of M^1_i states that the production quantity cannot be larger than the maximum total demand. Constraints (5) ensure that production capacities are respected.

3.3 Multi-Stage Formulation for the MMCLP in a Static-Dynamic Environment

The static-dynamic environment corresponds to a multi-stage stochastic optimization model. This model is similar to model (1)-(8), but the production quantities are scenario-dependent, and non-anticipativity constraints are included in the model.

In the static-dynamic decision framework, the production quantities in period t depend on the realizations of the demands at periods $1, \ldots, t-1$. Therefore, the quantity Q^{ω}_{it} of item i produced in period t depends on the scenario ω. The *non-anticipativity* constraints (11) ensure that identical decisions are made at stage t in all scenarios indistinguishable up to stage t. Accordingly, constraints (11) enforce equal production quantities in period $t+1$ for all scenarios ω with identical demands $D^{1 \ldots t}_{\omega}$ in periods 1 to t. Contrarily to the production quantities, the inventory and backlog levels are observed once the demands are known. Therefore, the index t (and not $t+1$) is used in the non-anticipativity constraints:

$$Q^{\omega}_{it+1} = Q^{\omega'}_{it+1}, \quad I^{\omega}_{it} = I^{\omega'}_{it}, \quad B^{\omega}_{it} = B^{\omega'}_{it} \qquad \forall\, i \in \mathcal{I},\ t \in \mathcal{H},\ \omega, \omega' | D^{1 \ldots t}_{\omega} = D^{1 \ldots t}_{\omega'}. \tag{11}$$

3.4 Structural properties and theoretical insights

To get some theoretical insights on the differences between the static-static and static-dynamic models, this section gives some structural properties of both models. The proofs of Proposition 1-3 are provided in the Electronic Companion.

Proposition 1. *The optimal solution of the static-static MMCLP has zero inventory of components in the last period, except for the components whose initial stock level is more than the amount required for the production of the end items during the complete planning horizon.*

Proposition 2. *The optimal solution of the static-dynamic MMCLP can have positive component inventory levels at the end of the planning horizon.*

Proposition 2 shows that the static-dynamic MMCLP model does not necessarily transform all components into end items. In other words, contrarily to the two-stage model, the multi-stage model takes into account the possibility of keeping components (instead of transforming them to end items) when the demand is low. Therefore, the additional benefit of the multi-stage model (which is generally more complex than the two-stage model) in a static-dynamic environment is more pronounced when the difference between the holding costs of end items and components is large. Such situations are typically encountered for products with high added value in the supply chain.

Proposition 3. *For the special case of MMCLP with a single item and infinite capacity, a reduction of the lead time has no impact on the total cost in the static-static framework (the production quantities are only shifted), as long as the initial inventory covers the demand from period* 1 *to the lead time* L*. On the contrary, a reduction of the lead times impacts the production quantities and the total cost in the static-dynamic decision framework.*

Although Proposition 3 concerns a specific case of MMCLP, the experiments presented in Section 6 show that the multi-stage model significantly outperforms the two-stage model for the generic MMCLP with short lead times in the static-dynamic decision framework (the two-stage model becomes a heuristic in such a decision framework).

4 Solution Methods

This section presents computational enhancements to address the scalability issues of the mathematical formulations presented in Section 3. Section 4.1 discusses a formulation of the multi-stage model where the non-anticipativity constraints are modeled implicitly, as well as a fix-and-optimize heuristic developed for this formulation. The resulting approaches significantly reduce the memory consumption and the computation time. In addition, Section 4.1 introduces an order-up-to-level policy conditional on production setups, which is derived from the multi-stage model. This approach alleviates the cumbersome static-dynamic decision process, since it requires no computation in the subsequent decision stages. As the complete set Ω of scenarios is usually too large, scenario sampling methods are provided in Section 4.3. Finally, Section 4.4 presents a rolling-horizon heuristic to solve the problem in a dynamic-dynamic decision framework.

4.1 Optimization Methods for Static-Dynamic MMCLP

As the number of scenarios of the multi-stage model grows exponentially with the number of periods (see Section 4.3) , solving the multi-stage model is very challenging. To reduce the size of the model, the formulation with *implicit non-anticipativity* uses a single variable for each group of equal variables. More precisely, the variable $Q_{it+1}^{D(1\ldots t)}$ replaces the set $\{Q_{it+1}^{\omega} \mid D_{\omega}^{1\ldots t} = D(1\ldots t)\}$ of variables, which represents the quantity produced in period t in different scenarios ω with identical demands $D(1\ldots t)$ from periods 1 to t. Similarly, variables $I_{it}^{D(1\ldots t)}$ and $B_{it}^{D(1\ldots t)}$ replace the sets $\{I_{it}^{\omega} \mid D_{\omega}^{1\ldots t} = D(1\ldots t)\}$ and $\{B_{it}^{\omega'} \mid D_{\omega}^{1\ldots t} = D(1\ldots t)\}$, respectively. Also, constraints (1)-(8) are generated for each possible realization of the demands in period 1 to t (and not for each scenario), item, and period. In addition, to further speed up the solution of the problem, the solution of the two-stage model is used as a warm start for the multi-stage model.

Finally, a *fix-and-optimize* heuristic is proposed. This approach has two steps. The first step determines the setups by solving the two-stage model. In the second step, the multi-stage formulation is solved, but the setups are fixed to the values found in the first step. As the non-fixed variables in the second step (production quantities, inventory levels, backlogs, lost sales) are continuous, the resulting model is a linear program.

Proposition 4 shows that fix-and-optimize yields a better solution than the two-stage model, and Propositions 5-7 show that the use of two-stage model as a heuristic for the static-dynamic MMCLP can lead to over production.

Proposition 4. *Given a set of scenarios Ω, for the static-dynamic MMCLP (multi-stage model), the cost of the solution s_1 using fix-and-optimize is lower than or equal to the cost of the solution s_2 using the two-stage model for this given set of scenarios.*

Proof. The two-stage model is equivalent to adding the following set of constraints to the multi-stage model:

$$Q_{i,t}^{\omega} = Q_{i,t}^{\omega'} \qquad \forall i \in \mathcal{I}, \quad t \in \mathcal{T}, \quad \omega,\ \omega' \in \Omega.$$

As the two-stage model is more restricted, s_2 is a feasible (but not necessarily optimal) solution of the linear program solved in the second step of the fix-and-optimize method. Therefore, the cost of the solution s_1 is lower than or equal to the cost of the solution s_2. $\square$

In the rest of this section, we show that using the two-stage model in a static-dynamic context leads to overproduction for the special case of MMCLP with a single item (and thus a single level) and infinite capacity. To simplify the equations, we refer to the quantity X_t *completed* in period t rather than the quantity Q_t produced in period t.

In the fix-and-optimize heuristic, once the step 1 is performed, the setups are fixed to the values of the solution of the two-stage model, and we denote $\sigma(k)$ the period in which the k^{th} production lot is completed. Given two periods t and $t'+1$ in which two consecutive lots of production are completed (i.e., one lot completed in period t, and the subsequent lot is completed in period $t'+1$), the cost $\bar{f}_{t\to t'}(\mathcal{X}_t)$ incurred from periods t to t' depends only on the total cumulative quantity $\mathcal{X}_t$ completed until period t ($\mathcal{X}_t = \sum_{\tau=0}^{\tau=t} X_\tau$). That is,

$$\bar{f}_{t\to t'}(\mathcal{X}_t) = \sum_{\tau=t}^{\tau=t'} \sum_{\omega\in\Omega} p_\omega f_\tau^\omega(\mathcal{X}_t) \tag{12}$$

where

$$f_t^\omega(\mathcal{X}_t^\omega) = g_t^\omega \left(\mathcal{X}_t^\omega - \sum_{\tau=1}^{\tau=t} D_\tau^\omega \right) \tag{13}$$

and

$$g_t^\omega(x) = \max\left(hx, -bx\right). \tag{14}$$

The optimal quantity for the first production lot denoted by $\mathcal{X}^\star_{\sigma(1)}$ (i.e., completed in period $\sigma(1)$) can be expressed as follows:

$$\mathcal{X}^\star_{\sigma(1)} = \underset{\mathcal{X}_{\sigma(1)}>I_0}{\arg\min} \left(\bar{f}_{\sigma(1)\to[\sigma(2)-1]}(\mathcal{X}_{\sigma(1)}) + f^\star_{\sigma(2)\to T}(\mathcal{X}_{\sigma(1)}) \right),$$

where $\bar{f}_{\sigma(1)\to[\sigma(2)-1]}(\mathcal{X}_{\sigma(1)})$ represents the total cumulative costs encountered from period $\sigma(1)$ until the period $\sigma(2)-1$ as a function of the total quantity $\mathcal{X}_{\sigma(1)}$ completed in period $\sigma(1)$, and $f^\star_{\sigma(2)\to T}(\mathcal{X}_{\sigma(1)})$ represents the optimal cost-to-go from periods $\sigma(2)$ to T as a function of the total quantity $\mathcal{X}_{\sigma(1)}$ completed in period $\sigma(1)$.

Proposition 5. *For the special case of static-static MMCLP with a single item and infinite capacity, the optimal cost-to-go from period $\sigma(k+1)$ is independent of the quantity completed in period $\sigma(k)$, and*

$$\mathcal{X}^\star_{\sigma(k)} = \underset{\mathcal{X}_{\sigma(k)}}{\arg\min}\, \bar{f}_{\sigma(k)\to[\sigma(k+1)-1]}(\mathcal{X}_{\sigma(k)}). \tag{15}$$

Proof. For the two-stage model, the optimal cost-to-go from period $\sigma(k+1)$ can be expressed as follows:

$$f^{\star}_{\sigma(k+1)\to T}(\mathcal{X}_{\sigma(k)}) \quad = \quad \min_{\mathcal{X}_{\sigma(k)}\le\mathcal{X}_{\sigma(k+1)}\le\ldots\le\mathcal{X}_{\sigma(n)}} \quad \sum_{l=k+1}^{n} \bar{f}_{\sigma(l)\to[\sigma(l+1)-1]}(\mathcal{X}_{\sigma(l)}), \tag{16}$$

where n is the period when the last production lot is completed, and $\sigma(n+1) = T+1$. In equation (16), the inequalities $\mathcal{X}_{\sigma(k)} \le \mathcal{X}_{\sigma(k+1)} \le \ldots \le \mathcal{X}_{\sigma(n)}$ state that the total cumulative quantity is a non-decreasing function of the time period. The rest of the proof shows that $\arg\min_{\mathcal{X}_{\sigma(l+1)}} \bar{f}_{\sigma(l+1)\to[\sigma(l+2)-1]}(\mathcal{X}_{\sigma(l+1)}) \ge \arg\min_{\mathcal{X}_{\sigma(l)}} \bar{f}_{\sigma(l)\to[\sigma(l+1)-1]}(\mathcal{X}_{\sigma(l)})$.
As shown in equation (12), $\bar{f}_{\sigma(l)\to[\sigma(l+1)-1]}(\mathcal{X})$ can be expressed as a weighted sum of piece-wise linear functions $f^{\omega}_{t}(\mathcal{X})$. As the total demand is a non-decreasing function of the number of periods,

$$\arg\min_{\mathcal{X}} f^{\omega}_{t+1}(\mathcal{X}) \ge \arg\min_{\mathcal{X}} f^{\omega}_{t}(\mathcal{X}).$$

Therefore,

$$\arg\min_{\mathcal{X}} \bar{f}_{[t+1]\to[t+1]}(\mathcal{X}) \ge \arg\min_{\mathcal{X}} \bar{f}_{t\to t}(\mathcal{X}),$$

since $\bar{f}_{[t+1]\to[t+1]}(\mathcal{X})$ is a weighted sum of piece-wise linear functions $f^{\omega}_{t+1}(\mathcal{X})$, and $\arg\min_{\mathcal{X}} f^{\omega}_{t+1}(\mathcal{X}) \ge \arg\min_{\mathcal{X}} f^{\omega}_{t}(\mathcal{X})$ for each scenario ω. Finally,

$$\arg\min_{\mathcal{X}} \bar{f}_{\sigma(l+1)\to[\sigma(l+2)-1]}(\mathcal{X}) \ge \arg\min_{\mathcal{X}} \bar{f}_{\sigma(l)\to[\sigma(l+1)-1]}(\mathcal{X}),$$

since $\bar{f}_{\sigma(l)\to[\sigma(l+1)-1]}(\mathcal{X})$ is a sum of piece-wise linear functions $\bar{f}_{\sigma(l)\to\sigma(l)}(\mathcal{X}), \ldots, \bar{f}_{[\sigma(l+1)-1]\to[\sigma(l+1)-1]}(\mathcal{X})$, and $\arg\min_{\mathcal{X}} \bar{f}_{t'\to t'}(\mathcal{X}) \ge \arg\min_{\mathcal{X}} \bar{f}_{t\to t}(\mathcal{X})$ for any t' in $\{\sigma(l+1), \ldots, \sigma(l+2)-1\}$ and t in $\{\sigma(l), \ldots, \sigma(l+1)-1\}$. Therefore,

$$\arg\min_{\mathcal{X}_{\sigma(k)}} f_{\sigma(k)\to[\sigma(k+1)-1]}(\mathcal{X}_{\sigma(k)}) \le \arg\min_{\mathcal{X}_{\sigma(k+1)}} f_{\sigma(k+1)\to[\sigma(k+2)-1]}(\mathcal{X}_{\sigma(k+1)}) \le \ldots \le \arg\min_{\mathcal{X}_{\sigma(n-1)}} f_{T}(\mathcal{X}_{\sigma(n-1)}),$$

and the constraints $\mathcal{X}_{\sigma(k)} \le \mathcal{X}_{\sigma(k+1)} \le \ldots \le \mathcal{X}_{\sigma(n)}$ can be omitted in the computation of the cost-to-go in equation (16), since they are redundant. This implies that the cost-to-go from period $\sigma(k+1)$ is independent of the quantity completed in period $\sigma(k)$, and equation (15) holds.

□

Proposition 6. *For the special case of static-dynamic MMCLP with a single item and infinite capacity, the total quantity* $\arg\min_{\mathcal{X}_{\sigma(1)}} f_{\sigma(2)\to T}(\mathcal{X}_{\sigma(1)})$ *minimizing the cost-to-go can be lower than the quantity* $\arg\min_{\mathcal{X}_{\sigma(1)}} f_{\sigma(1)\to[\sigma(2)-1]}(\mathcal{X}_{\sigma(1)})$ *minimizing the total cumulative cost from periods* $\sigma(1)$ *to* $\sigma(2)-1$*. In other words, unlike the case of the static-static MMCLP in Proposition 5, the inequality* $\arg\min_{\mathcal{X}_{\sigma(1)}} f_{\sigma(1)\to[\sigma(2)-1]}(\mathcal{X}_{\sigma(1)}) \leq \arg\min_{\mathcal{X}_{\sigma(1)}} f_{\sigma(2)\to T}(\mathcal{X}_{\sigma(1)})$ *does not hold for the static-dynamic MMCLP.*

Proof. An example is given below with the parameters $h = 1$, $b = 2$, $s = 0$, $v = 0$, setup in each period, and the two demand scenarios given in Table 3 with probability 0.5 each.

Period	1	2	3
Scenario 1	0	0	0
Scenario 2	10	0	0

Table 3: Demands in each scenario

The cost in period 1 is given by

$$f_{1\to1}(\mathcal{X}_1) = 0.5\max(\mathcal{X}_1, -2\mathcal{X}_1) + 0.5(\max(\mathcal{X}_1 - 10, 2(10 - \mathcal{X}_1)),$$

which is minimized at 10. Table 4 gives the calculations of the cost-to-go $f_{2\to3}(\mathcal{X}_1)$ for two separate cases where $\mathcal{X}_1 = 10$ and $\mathcal{X}_1 = 0$. Note that Table 4 shows only data for period 2 and 3, since period 1 is not considered in the cost-to-go $f_{2\to3}(\mathcal{X}_1)$. $f_{2\to3}(10)$ equals 10 with 10 (resp. 0) units of stock during periods 1 and 2 in scenario 1 (resp. 2). However, $f_{2\to3}(0)$ equals 0, with a quantity of 0 (resp. 10) completed in period 2 for scenario 1 (resp. 2), leading to no stock and no backlog during periods 1 and 2 for both scenarios. In this example, $\arg\min_{\mathcal{X}_1} f_{2\to3}(\mathcal{X}_1) < \arg\min_{\mathcal{X}_1} f_{1\to1}(\mathcal{X}_1)$ which contradicts the condition mentioned in Proposition 5.

□

Proposition 7. *For the special case of static-dynamic MMCLP with a single item, infinite capacity, and* $\sigma(1)$ *equal to the lead time* L*, the two-stage model leads to a larger than or equal production quantity completed in period* $\sigma(1)$ *compared to fix-and-optimize.*

Proof. Proposition 5 shows that in the static-static case, the quantity completed in period $\sigma(1)$ can be computed by

$$\mathcal{X}^\star_{\sigma(1)} = \arg\min_{\mathcal{X}_{\sigma(1)}} \bar{f}_{\sigma(1)\to\sigma(2)-1}(\mathcal{X}_{\sigma(1)}).$$

$\mathcal{X}_0$		Period	2	3
0	Quantity	Scenario 1	0	0
	Inventory		0	0
	Backlog		0	0
	Quantity	Scenario 2	10	0
	Inventory		0	0
	Backlog		0	0
10	Quantity	Scenario 1	0	0
	Inventory		10	10
	Backlog		0	0
	Quantity	Scenario 2	0	0
	Inventory		0	0
	Backlog		0	0

Table 4: Cost-to-go computation in mutli-stage

On the contrary, Proposition 6 shows that the condition $\mathcal{X}_{\sigma(2)} \geq \mathcal{X}_{\sigma(1)}$ is necessary in the optimal cost-to-go function of the multi-stage model. Therefore, the cost-to-go cannot be ignored in the static-dynamic, and

$$\mathcal{X}^{\star}_{\sigma(1)} = \arg\min_{\mathcal{X}_{\sigma(1)}} \left(\bar{f}_{\sigma(1)\to\sigma(2)-1}(\mathcal{X}_{\sigma(1)}) + f^{\star}_{\sigma(2)\to T}(\mathcal{X}_{\sigma(1)}) \right). \tag{17}$$

As $f^{\star}_{\sigma(2)\to T}(.)$ is an non-decreasing function, the quantity produced in period 0 by the fix-and-optimize heuristic can only be lower than or equal to the quantity produced by the two-stage model.

□

When the two-stage model and the fix-and-optimize heuristic are used in a rolling horizon framework, only the decisions of stage 0 (i.e., the lots completed in period $\sigma(1) = L$) are implemented. As a consequence, using the two-stage model in a static-dynamic decision framework can lead to over-production.

4.2 Execution Policy

The use of the multi-stage model in the static-dynamic decision framework is cumbersome because it requires to re-solve the model at each period. To ease the process, an order-up-to-level policy (denoted S-policy), which is conditional on the production setups, is derived from the solution of the multi-stage model. The S-policy is based on the notion of echelon stock E_{it}, that denotes the total quantity of item i in the system in period t. The echelon stock includes the stock of item i, the components i in the stocks of downstream items, and the quantities ordered in previous periods but not yet produced. Before applying the policy, the values of the replenishment level S_{it} and of the setup Y_{it} of each item i and period t are inferred from the solution of the multistage model (as explained in the next paragraph). For each period t with setup (i.e., $Y_{it} = 1$), the ordered quantity Q_{it} of each item i is computed to bring the echelon stocks E_{it} to the replenishment level S_{it} (i.e., $Q_{it} = S_{it} - E_{it}$ if $Y_{it} = 1$, and 0 otherwise). However, the quantities must respect the production capacities and the flow conservation constraints (i.e., the stocks of components must be large enough to allow the production of the item). If a quantity violates the flow conservation constraint, it is reduced to the largest feasible quantity. If a resource capacity is violated, the quantities of all items processed with the resource are reduced by the same percentage.

The values of S_{it} are inferred from the solution of the multi-stage model as follows. First, for each optimization scenario ω, the replenishment level S^{ω}_{it} is calculated as $S^{\omega}_{it} = E^{\omega}_{it} + Q^{\omega}_{it}$. To compute S_{it}, we consider only the scenarios where Q^{ω}_{it} is not constrained by the capacities or flow conservation constraints (but if the quantity is constrained in all the scenarios, then all the scenarios are considered). S_{it} corresponds to the average value of S^{ω}_{it} in these scenarios. The strategy of averaging the quantities over the scenarios with non-binding capacity constraints performs well, since the echelon stock levels are not ideal when the quantities are reduced by the capacity constraints. Preliminary experiments (omitted here) showed that the latter strategy performs better than using the maximum quantity, or averaging over all scenarios.

Note that two nearest scenario policies were also investigated. In each period t, the nearest scenario policy implements the decisions Q^{ω}_{it} associated with the optimization scenario ω with the demands $D^{1,..t}_{\omega'}$ in periods $1 \dots t$ nearest to the actual demands. A variant of this policy is to follow a path in the scenario tree. In each period t, this variant implements the decisions associated with the branch of the current path with demands in period t nearest to the actual demands. Preliminary experiments (not presented here) showed that both of these policies perform poorly. For this reason, they were not further considered.

To justify the proposed S-policy, Proposition 8 shows that this policy is optimal for the special case of the MMCLP with a single item and infinite capacity.

Proposition 8. *For the special case of the static-dynamic MMCLP with a single item and infinite capacity, there exists an optimal S-policy.*

Proof. For the sake of clarity, the vectors $Y_t, \dots, Y_{t'}$ and $Q_t, \dots, Q_{t'}$ are denoted by $Y_{t,\dots,t'}$ and $Q_{t,\dots,t'}$, respectively.

Since the probability distributions of the demands are independent in each period, given the state of the system in period t, the optimal cost-to-go $\Psi_{t\to T}(I_{t-1},\ Q_{t-L,\dots,t-1},\ Y_{1,\dots,t})$ from period t to T are identical for all scenarios: $\Psi_{t\to T}(I_{t-1},\ Q_{t-L,\dots,t-1},\ Y_{1,\dots,t})$ is a function of leftover inventory (I_{t-1}), production quantities $(Q_{t-L,\dots,t-1})$, and setups $(Y_{1,\dots,T})$. Note that this function $\Psi_{t\to T}(I_{t-1},\ Q_{t-L,\dots,t-1},\ Y_{1,\dots,t})$ differs from the function $f^{\star}_{\sigma(2)\to T}(\mathcal{X})$ defined in the proof of Proposition 5 since it also includes the setup decisions

The optimal cost-to-go can be expressed as follows:

$$\begin{aligned}\Psi_{t\to T}\left(I_{t-1}, Q_{t-L\dots t-1}, Y_{1\dots T}\right) = \sum_{\omega\in\Omega}\Bigg(&\sum_{t'=t}^{t'=t+L-1} g_t^{\omega}\left(I_{t-1} + \sum_{\tau=t}^{\tau=t'} Q_{\tau-L} - \sum_{\tau=t}^{\tau=t'} D_{\tau}\right)\\ &+\Psi_{t+L\to T}\left(I_{t-1} + \sum_{\tau=t}^{\tau=t+L-1} Q_{\tau-L} - D^{\omega}_{\tau}, Y_{1\dots T}, Q_{t\dots t+L}\right)\Bigg).\end{aligned}$$

As, $\sum_{t'=t}^{t'=t+L-1} g_t^{\omega}\left(I_{t-1} + \sum_{\tau=t}^{\tau=t'} Q_{\tau-L} - \sum_{\tau=t}^{\tau=t'} D_{\tau}\right)$ does not depend on the production quantities in period t, the production quantity in period t must be chosen to minimize

$$\sum_{\omega\in\Omega}\Psi_{t+L\to T}\left(I_{t-1} + \sum_{\tau=t}^{\tau=t+L-1} Q_{\tau-L} - D^{\omega}_{\tau}, Y_{1\dots T}, Q_{t\dots t+L}\right).$$

As S_t is computed by $S_t = I_{t-1} + \sum_{\tau=t}^{\tau=t+L} Q_{\tau-L}$, the quantity Q_t must minimize

$$\sum_{\omega\in\Omega}\Psi_{t+L\to T}\left(S_t - D^{\omega}_{\tau},\ Y_{1\dots T},\ Q_{t\dots t+L}\right).$$

Given the optimal setups, this function is convex and thus there exists a single global minimum in S_t, $Q^{\star}_{t+1}\dots\ Q^{\star}_{t+l-1}$.

□

While the S-policy is optimal for the special case of MMCLP with a single item and infinite capacity, it cannot be applied directly to the multi-echelon capacitated problem. Indeed, if the optimal replenishment quantities violate the capacity (or if they lead to negative levels of components inventory), a decision must be made to split the replenishment among the items. The experimental results presented in Section 6 show that the S-policy performs very well on the uncapacitated MMCLP but not on the capacitated MMCLP.

4.3 Scenario Sampling

Solving the two-stage (resp. multi-stage) model with the set Ω of all possible scenarios leads to the true optimal solution in the static-static (resp. static-dynamic) decision framework. However, Ω is usually large (sometimes infinite), and solving the resulting MILP is often impossible in practice. Consequently, the problem is approximated with samples of scenarios. This section describes the tree structure required to generate the scenarios in the multi-stage model, before exposing three scenario sampling techniques (used for the multi-stage and two-stage model), namely, crude Monte Carlo (CMC), quasi-Monte Carlo (QMC), and randomized quasi-Monte Carlo (RQMC).

The representation of the scenarios are different in the two-stage and multi-stage models. In the two-stage model, a scenario is a vector whose components are the demands for each end item in each period. In the multi-stage model, the scenarios are generated with a scenario tree, as shown in Figure 1. Each level of the tree corresponds to a period, the children of a node at level t are possible realizations of the demands in period $t+1$, and each path in the tree corresponds to a scenario. Using a scenario tree ensures that the decided production quantities in period t (given the demands in periods $1, \ldots, t-1$) account for the stochastic demands in period $t, \ldots, T$. Indeed, in a scenario tree, multiple demand realizations for periods $t, \ldots, T$ are available for each demand's realization of periods $1, \ldots, t-1$. The scenario tree structure is denoted by $[N_1, N_2, \ldots, N_T]$, where N_t is the number of branches of the nodes at level t. For instance, the structure of the tree in Figure 1 is $[2, 2]$. The demand realizations are sampled independently at each node of the tree. Consequently, the multi-stage model requires to sample vectors whose components are the demands for a single period (i.e., the dimension of the vectors is the number of end items). Note that advanced sampling techniques are crucial for the multi-stage model (see Section 6). Indeed, as the size of the tree is exponential in the number of periods, only a few demand realizations can be sampled at each node.

For more information on scenario generation in multi-stage stochastic optimization, the interested reader is referred to Dupačová et al. (2003), Keutchayan et al. (2017), and Kaut and Wallace (2003).

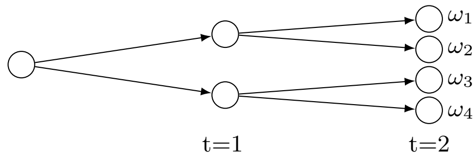


Figure 1: Example of scenario tree [2, 2]

CMC samples n vectors randomly (according to the probability distribution of the demand) with equal probability $1/n$. The CMC approximation is on average equal to the true optimal expected cost when the number of scenarios is sufficiently large. However, the variance of the approximation is σ^2/n, where σ^2 is the variance of the expected optimal cost. On the other hand, RQMC finds samples leading to approximations with theoretically lower variances than CMC. QMC and RQMC first select a set V_n of vectors in $[0,1]^d$, where d is the dimension of the sampled vectors, to cover evenly the unit cube. From V_n, the demand vectors are generated using the inverse of the cumulative probability distribution of the demands. Like CMC, all vectors have equal probability $1/n$. Two methods exist to generate V_n, namely, lattice rules and digital nets. Rank-1 lattice rules are used in this paper (higher rank lattices are uncommon in practice). These rules are formally defined as $V_n = \{i \cdot \boldsymbol{\alpha}/n + \boldsymbol{\delta} \bmod 1 \quad \forall\ i \in 1 \ldots n\}$, where $\boldsymbol{\alpha}$ is a generator vector, and $\boldsymbol{\delta}$ is a random point allowing to shift the lattice. Using $\boldsymbol{\delta} = 0$ leads to QMC which is a deterministic sampling technique, whereas RQMC uses $\boldsymbol{\delta} > 0$ to generate a random sample. The quality of the lattice is determined by vector $\boldsymbol{\alpha}$, which is generated here by the Lattice Builder tool (L'Ecuyer and Munger 2016). Lattice Builder is a software which implements multiple algorithms to build good rank-1 lattice rules. Here, $\boldsymbol{\alpha}$ is generated with the component-by-component method (Cools et al. 2006) to get a *fully projection-regular lattice* minimizing the (weighted) P_2 *discrepancy measure*. These two notions are explained below. V_n is *fully projection-regular* if the projections of V_n on each subset of coordinates contain n distinct points. Such lattices usually lead to better approximations. One gets a fully projection-regular lattice if the greater common divisor between n and each component of $\boldsymbol{\alpha}$ is 1 (L'Ecuyer and Lemieux 2000). The P_2 *discrepancy measure*, which is extensively used in the RQMC literature, takes the form $P_2(V_n) = -1 + \frac{1}{n}\sum_{i=1}^{i=n}\left(\prod_{j=1}^{j=d} 1 - 2\pi^2(v_{ij}^2 - v_{ij} + 1/6)\right)$, where

v_{ij} is the j^{th} component of the i^{th} vector in V_n. P_2 measures the error made by approximating with V_n the integral of the worst case function in a set E_c (the integration error equals $c \cdot P_2$). The set E_c of functions is defined as $E_c = \{f : [0,1) \to \mathbb{R} : |\hat{f}(h)| \leq \frac{c}{\|h\|^2}\}$, where $\hat{f}(h)$ is the Fourier coefficient of f evaluated at vector h, and $\|h\| = \prod_{j=1}^{s} \max(1, |h_j|)$. In addition, the derivation of P_2 requires the functions in E_c to have absolutely convergent Fourier representations. The derivation of P_2 for QMC (when $\boldsymbol{\delta} = 0$) is given by Sloan and Joe (1994). The integration error is stochastic with non-null $\boldsymbol{\delta}$, but the variance of the RQMC approximation is bounded by $c \cdot P_2$ for all functions f in E_c (L'Ecuyer and Lemieux 2000). Even though the considered lot-sizing problem does not correspond to the integration of a function in E_c, a set of vectors minimizing the P_2 discrepancy measure results in a low variance estimator in practice.

The weighted (resp. un-weighted) P_2 measure is used for the two-stage (resp. multi-stage) model. In the weighted version of P_2, the discrepancy measure P_2^{μ} is computed on each projection μ (on each subset of coordinates) of the integration lattice, and $\boldsymbol{\alpha}$ is chosen to minimize the weighted average $(\sum_{\mu} \gamma_{\mu} P_2^{\mu})$. We use the weights $\gamma_{\mu} = 0.1^k$, where k is the number of coordinates in the projection μ. In this setting, low dimension (i.e., involving few coordinates) projections have low integration errors, which usually leads to better approximation. However, the un-weighted P_2 discrepancy measure is used for the multi-stage case, because the sampled vectors have low dimensions. More information on weighted discrepancy measures can be found in L'Ecuyer and Lemieux (2000).

Finally, as demands are integer, a QMC or RQMC sample can contain multiple occurrences of a vector (even if V_n is fully projection-regular). Identical vectors are aggregated into a single one by adding their probabilities. To better control the number of scenarios, the sample size is increased until a predefined number n of different vectors are obtained. Such aggregations are especially useful for lumpy demands, where the probability of having no demand is large.

4.4 Rolling horizon framework

In practice, production planing tools are often used in a rolling horizon framework (Venkataraman 1996). Using the proposed methods in a rolling horizon framework leads to heuristics for the dynamic-dynamic decision framework. In the rolling horizon framework, the plan is optimized at period 0 by considering the first $\mathcal{H}$ periods, and the decisions of period 0 are implemented. Then, the demands of period 0 are revealed, and the backlogs and inventories are observed. Considering this information, the plan is re-optimized on the horizon 1 to $\mathcal{H}+1$. This process continues until the last period. The efficiency of the rolling horizon heuristics is shown in Section 6.4.

5 Comparison Methods and Simulation Framework

In this section, we describe the methods considered to benchmark the proposed approaches (Section 5.1), and describes the simulation framework considered to evaluate these approaches (Section 5.2).

5.1 Classical MRP Approaches

This section describes the safety stock computation methods, the deterministic mathematical programs and classical lot-sizing rules equipped with safety stocks classically used in MRP systems.

5.1.1 Safety Stock computations

Two approaches are considered to include safety stocks.

As suggested in the literature (e.g., Zhao et al. 2001), the first approach assumes that the safety stocks are computed for end items only at the MPS level. In such case, the safety stock ss_{it} in period t must cover against the demand uncertainty of end item i between t and the last period t'_i where the production quantity of i has been adjusted according to the actual demand. As proposed in Bookbinder and Tan (1988), the value of t'_i is different for each decision framework: (1) In static-static, $t'_i = 0$ since the production plan is fixed; (2) In static-dynamic, t'_i is computed with the expected time between order $TBO_i = \sqrt{\frac{2s_i}{\bar{D}_i h_i}}$ where $\bar{D}_i$ is the average demand of item i over the planing horizon and $t'_i = t - TBO_i$; (3) In dynamic-dynamic, as an order can be triggered in case of stock out, t'_i is computed based on the minimum between TBO_i and the time required to produce i from raw components.

As this study considers backlog costs, the target inventory level X_{it} of item i in each period t is computed to minimize the expected inventory and backlog costs $f(X_{it})$ in period t, and the safety stock ss_{it} is equal to $X_{it} - \sum_{\tau=t'_i}^{t} \bar{D}_{i\tau}$, where $\bar{D}_{i\tau}$ is the average demand of item i in period τ. The computation of $f(X_{it})$ is provided in equations (18) and (19) where $E[.]$ denotes the expected value,

$I_{it}(X_{it})$ denotes the inventory level depending on X_{it}, $B_{it}(X_{it})$ denotes the backlogs, and $\widetilde{D}_{it'_i->t}$ is the probability distribution of the demand between periods t'_i and t:

$$f(X_{it}) = h_i \cdot E\left[I_{it}(X_{it})\right] + b_i \cdot E\left[B_{it}(X_{it})\right] \tag{18}$$

$$f(X_{it}) = h_i \int_0^{X_{it}} (X_{it} - x) \cdot \widetilde{D}_{it'_i \to t}(x) \cdot dx + b_i \int_{X_{it}}^{\infty} (x - X_{it}) \cdot \widetilde{D}_{it'_i \to t}(x) \cdot dx. \tag{19}$$

Because minimizing (19) corresponds to a newsvendor problem (see Khouja 1999), the minimum is achieved at $X_{it} = \widetilde{D}^{-1}_{it \to t'_i}\left(\frac{b_i}{b_i + h_i}\right)$, and $\frac{b_i}{b_i+h_i}$ can be interpreted as the service level.

The second approach computes the safety stocks with the method introduced in Graves and Willems (2008) for a base stock policy in multi-echelon supply chain with non-stationary demand. In other words, the authors assume a similar situation as this paper, except that the production is planned with a base stock policy rather than with an MRP system. The method of Graves and Willems (2008) computes the safety stocks to guarantee the delivery of the maximum demand within the service time S_{it} of each item i and period t. The value of S_{it} is determined by a mathematical model to minimize the inventory costs of the safety stocks. As the components of an item are delivered within its inbound service time SI_{it}, its safety stock ss_{it} corresponds to

$$ss_{it}(S_{it}, SI_{it}) = \widehat{D}_i(t - SI_{it} - L_i, t - S_{it}) - \sum_{\tau = t - SI_{it} - L_i}^{\tau = t - S_{it}} \bar{D}_{i\tau},$$

where $\widehat{D}_i(t_1, t_2)$ denotes the maximum demand for item i between periods t_1 and t_2. In this work, the maximum demand of end item i is set to $\widehat{D}_i(t_1, t_2) = \widetilde{D}^{-1}_{it_1 \to t_2}\left(\frac{b_i}{b_i+h_i}\right)$ (as in the first safety stocks computation method). For the components, the value of $\widehat{D}_i(t_1, t_2)$ is inferred from the maximal demand of the end items. Finally, as our experiments are performed in a finite horizon, the safety stock of a component is set to 0 (after solving the model) for the periods that are too late to allow the transformation of this component into an end item.

5.1.2 Deterministic Mathematical Model with Safety Stocks

The deterministic mathematical model corresponds to the two-stage formulation (1)-(8) with a single scenario. Although the deterministic mathematical model using the expected demand scenario is interesting for comparison purposes, practitioners use safety stocks to hedge against uncertainties. To prioritize demand fulfillment over the safety stock requirements, we incorporate the safety stock requirements as soft constraints. More precisely, constraints (20) compute the quantity P_{it} of missing safety stock for item i in period t, except for the periods in $\bar{\mathcal{H}}_i$, where $\bar{\mathcal{H}}_i$ includes the first periods where low initial inventories prevent the creation of the safety stocks:

$$I_{it} + P_{it} \quad \geq \quad ss_{it} \qquad t \in \mathcal{H} \backslash \bar{\mathcal{H}}_i, \ i \in \mathcal{I}. \tag{20}$$

Then, a penalty p_{it} is incurred for each unit of missing safety stock. In our experiments, for t in periods $0, \ldots, T-1$, the penalty p_{it} is set to $1.5h_i$, that is, larger than the holding cost and lower than the backlog cost. In the last period ($t = T$), unmet demand leads to lost sales, and p_{it} is set to $5h_i$. In addition, the value of the upper bounds M_{it} of the production quantities are modified to allow the production of the safety stocks.

5.1.3 Classical Lot-sizing Methods

This section describes classical lot-sizing heuristics, namely lot-for-lot, economic order quantity, economic order period, and Silver-Meal. These rules were designed for the uncapacitated version of the considered problem, and the eventual violations must be repaired in a post-processing step when these rules are used in a system with capacities. Even though some procedures were proposed to reallocate the excess quantities, these approaches do not yield good quality solutions (Pochet and Wolsey 2006). Consequently, the lot-sizing rules are only considered for the uncapacitated version of the considered problem.

Following the MRP logic, the considered multi-echelon multi-item lot-sizing problem is solved item by item with the chosen rule, starting from end items up to the raw components. The demands of components are set according to the requirements of downstream items, whereas the average demand is considered for end items.

- In the **Lot-for-Lot (LL)** method, the lot size in each period corresponds to the net requirements. The computation of Q_{it} is given by $Q_{it} = \bar{D}_{i,t+L_p} + ss_{i,t+L_p} - PI_{i,t+L_p}$, where PI_{it} is the projected inventory level of item i in period t ($PI_{it} = \sum_{\tau \in \{0,\ldots,t\}} (Q_{i\tau} - \bar{D}_{i\tau})$).
- In the **Economic Order Quantity (EOQ)**, the lot sizes EOQ_i are first computed to balance the inventory and setup costs. The computation of EOQ_i is given by $EOQ_i = \sqrt{\frac{2 \cdot s_i \cdot \bar{D}_i}{h_i}}$, where

$\bar{D}_i$ is the average demand of item i over the planning horizon. In each period, the production quantity is the smallest multiple of EOQ_i, such that the projected inventory is larger than the safety stock (possibly 0).

- In the **Economic Order Period (EOP)**, the periods with production are first decided based on EOQ_i. More precisely, a setup is performed every $EOQ_i/\bar{D}_i$ periods, starting from the first period where the projected inventory is lower than the safety stock. The production quantities are equal to the sum of demands until the next period with production.
- In the **Silver-Meal (SM)** rule, a production order is executed if the projected inventory is below the safety stock. The production quantity is the demand for the next P periods, and P is chosen based on the average setup and inventory costs per period:

$$f(P) = \frac{s_p + \sum_{t=1}^{t=P} t \cdot h_p \cdot (D_{p,t+L_p})}{P}.$$

  The Silver-Meal procedure starts with $P = 0$, and increments P until the average cost increases (i.e., until $f(P+1) > f(P)$). As shown in Blackburn and Millen (1980), this approach is relatively more robust than solving an exact model when demand is uncertain.

5.2 Evaluation Methodology

5.2.1 Evaluation framework

To estimate the expected total costs associated with the use of a method, a simulation is performed over 5000 scenarios. These evaluation scenarios are different from the scenarios used for optimization, but they are sampled from the same distributions.

The simulation is performed independently on each scenario ω, and it results in the implementation of a solution s with setups Y_s and quantities Q_s. The solution s is directly available in the static-static decision framework, whereas the simulation of the static-dynamic decisions plans the production in each period based on the latest information on the demand (as explained in the next section). The cost of s is computed using the deterministic model with scenario ω, where the setup and quantity variables are respectively fixed to Y_s and Q_s.

5.2.2 Re-planning procedure

This section explains the re-planning methodology for the static-dynamic decision framework.

The solution of an instance $\mathcal{P}$ of MMCLP gives the setups Y_{it} for each item i and period t, as well as the production quantity Q_{i0} in period 0. Given an evaluation scenario ω, the quantities $Q_{i\tau}$ to produce in periods $\tau > 0$ are decided sequentially by taking into account the information on the demands in periods $0, \ldots, \tau - 1$. More precisely, $Q_{i\tau}$ is computed from the instance $\mathcal{P}^{\omega}_{\tau}$ which differs from $\mathcal{P}$ as follows: (1) The planning horizon becomes $\tau, \ldots, T$; (2) The setups Y_{it} are given for t in $\tau, \ldots, T$; (3) The initial inventory $I^{\omega}_{i\tau-1}$ and the backlog levels $B^{\omega}_{it\tau-1}$ are computed based the previously decided quantities $Q_{i0}, Q^{\omega}_{i1}, \ldots, Q^{\omega}_{1\tau-1}$ and the demands $D^{\omega}_{i0}, \ldots, D^{\omega}_{i\tau-1}$.

The considered methods require some adjustments to solve the modified instances $\mathcal{P}^{\omega}_{\tau}$. The *lot-sizing rules* set the quantity to 0 if there is no setup ($Q_{it} = 0$ if $Y_{it} = 0$), but they are applied as described in Section 5.1 otherwise ($Y_{it} = 1$). In the two-stage, multi-stage and deterministic models, the setup variables are set to the given values, thus a linear program is solved to determine the remaining continuous variables. To speed up the evaluation, the linear programs LP_{τ} associated with each instances $\mathcal{P}^{\omega}_{\tau}$ are adjusted (and not completely re-built) to each evaluation scenario ω. More precisely, the value of the initial inventory levels (possibly negative) are updated, as well as the value of M_{it} to allow the production of the eventual backlogs. LP_{τ} is solved with the barrier method, because our preliminary experiments (not presented here) showed that barrier performs faster than other linear programming solvers implemented in CPLEX for this problem. In addition, the solution of the previous evaluation scenario is used as a warm start. The *S-Policy* requires no modification, the echelon stock is computed based on the initial state, and the value Q^{ω}_{i0} is computed as described in Section 4.2.

The representation of the stochastic demands does not require any adjustment for the modified instances $\mathcal{P}^{\omega}_{\tau}$. More precisely, the lot-sizing rules and the deterministic models consider the average demand with safety stocks, whereas the two-stage and multi-stage models use samples of scenarios. Note that the multi-stage model consider scenario trees with structure $[N_1, \ldots, N_{T-t}]$, where $[N_1, \ldots, N_T]$ refers to the structure of the optimization tree.

6 Experimental Results

This section introduces the considered instances (Section 6.1), before presenting the experimental results. The performance of the sampling techniques on stochastic optimization approaches is shown in Section 6.2, along with the performance of the fix-and-optimize heuristic for the static-dynamic framework. Then, Section 6.3 presents the simulation results for the static-static and static-dynamic decision framework. Finally, the experiments presented in Section 6.4 demonstrate the rolling horizon simulation.

6.1 Instances

The experiments are performed using instances derived from the series A of Tempelmeier and Derstroff (1996). As these instances were designed for the deterministic MRP with zero lead times, they are extended (as explained in the Electronic Companion) to include the demand's probability distributions and lead times. Three sets of instances are considered: (1) 1026 *classical instances* are generated with full factorial design from the parameters given in Table 5; (2) 48 *instances with small distribution support* are generated with an assembly BOM (see Figure 2), a binomial distribution, and with full factorial design from the parameters (other than BOM and distribution) indicated in Table 5. (3) 20 *instances with large planning horizon* are generated with lead times randomly chosen in $[0, 3]$, a time horizon of $\hat{T} + 10$ periods, and various values for the other parameters.

Parameters	**Values**
BOM	General; Assembly (as depicted in Figure 2)
Resource structure	The items at the same echelon share the same resource
Resource utilization	Uncapacitated; 90%; 50%
Time between orders	1; 4
Distribution	Lumpy; Slow Moving; or Non-Stationary with rate of known demand in $\{0.25; 0.5; 0.75\}$ and Coefficient of variation in $\{0.1; 0.4; 0.7\}$
Lead time	L_1 (all items have a lead time of 1 period) L_2 (the lead times are equal to 1 for components, and 0 for end items)
Cost structures (ratio backlog/inventory costs)	2; 4
Echelon holding costs	Constant; large added value at last steps

Table 5: Values of the studied parameters.

6.2 Evaluation of Sampling and Optimization Methods

This section evaluates the impact of the scenario sampling techniques and of the number of scenarios on the two-stage and multi-stage models. These experiments are performed with a subset (of size

(a) Assembly **(b)** General

Figure 2: Considered BOM.

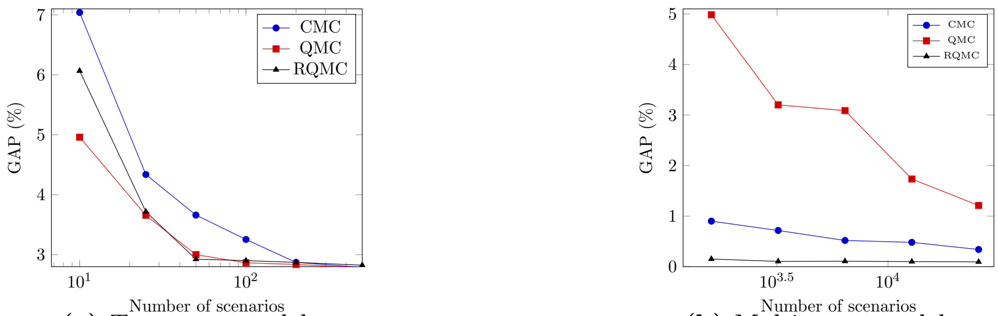
20) of the classical instances with various structures. Then, the solutions obtained with samples of scenarios are compared against the true optimal solutions for the instances with small support distribution, where the set of possible demand values is small enough to generate the full scenario set. Finally, the computation times required to solve the stochastic models and to run the fix-and-optimize heuristic are evaluated with the subset (of size 20) of the classical instances.

The performance of the methods is measured by the percentage gap (denoted GAP) between the expected total cost obtained with the method and the expected total cost obtained with the best performing method on the considered instances. Unless otherwise stated, this measure is used in all the experiments. The methods were implemented with Python and CPLEX 12.7, and run on an Intel(R) Xeon(R) X5675 3.07GHz processor.

6.2.1 Experiments on Sampling Methods

Figure 3a shows the GAP of the two-stage model in the static-static environment for different sample sizes $(10, 25, 50, 100, 200, \text{and } 500)$ and for the CMC and RQMC scenario generation methods. Figure 3a shows that a sufficiently large scenario sample is required to get a good approximation. For instance, 200 scenarios are required to get a good approximation with CMC. In addition, RQMC outperforms CMC (especially when few scenarios are used), whereas the performance of QMC and RQMC are similar. Therefore, advanced scenario sampling techniques (such as RQMC) allow to reduce the number of required scenarios. Indeed, using 50 scenarios sampled with RQMC or QMC leads to good approximations, which are comparable to the solution obtained with 200 scenarios sampled with CMC.

Regarding the multi-stage model, a first set of experiments (omitted here) showed that the scenario tree structure does not have a strong impact on the approximation quality, but scenario trees with a large number of branches at the early stages and a reasonable number of branches at the last stages perform better when the model is re-solved in each period (i.e., the S-policy is not used). However,



(a) Two-stage model. **(b)** Multi-stage model.

Figure 3: Approximation quality with CMC, QMC, or RQMC and various numbers of scenarios.

when the S-policy is used, the experiments show that a balanced scenario tree structure is more favorable than a scenario tree with a large number of scenarios in early stages. Figure 3b shows the GAP of the multi-stage model for various numbers (1600, 3200, 6400, 12800, and 25600) of scenarios sampled with CMC, QMC, and RQMC, and the computational performance is provided in the Electronic Companion. Figure 3b shows that RQMC leads to better solutions than CMC, especially when few scenarios are considered. In fact, 3200 scenarios sampled with RQMC generally lead to good approximation of the stochastic process, since considering a larger set of scenarios does not further reduce the GAP. Finally, QMC samples the same demands in all the nodes at the same level of the scenario tree. This leads to a poor representation of the stochastic demands compared to RQMC. Consequently, in the rest of the experiments the multi-stage model is solved with 6400 scenarios sampled with RQMC, since it is the largest number of scenarios which allows CPLEX to solve the entire problem to optimality within 10 hours (see the Electronic Companion). Even though instances with a relative short period are considered in the preliminary experiments, the multi-stage model can solve problems with a large time horizon by using a scenario tree with one branch at each node for the last periods which can be used in conjunction with a rolling horizon framework. This approach is explained in Section 6.4.

6.2.2 Experiments on Instances with a Perfect Set of Scenarios

The Electronic Companion reports the results of experiments performed on instances with small distribution supports, where the scenario set can be enumerated and the size is manageable. These experiments allow to compare the sampling approximations with the true optimal solutions, and they show that RQMC scenario sampling yields near-optimal solutions. In addition, these instances allow to compute the expected value of perfect information (EVPI) which is equal to 55.96% and 49.77% for the static-static and static-dynamic decision frameworks, respectively.

6.2.3 Performance Comparisons of the Stochastic Approaches

The Electronic Companion compares the performance (in terms of solution quality and computation times) of the multi-stage model, the two-stage model, and the fix-and-optimize heuristic in the static-dynamic environment. The results show that fix-and-optimize is efficient with a GAP of 0.30% versus 0.11% for the multi-stage model, whereas the multi-stage model is significantly slower to solve (241 seconds versus 3010 seconds on average). The two-stage model leads to slightly larger costs, with a GAP of 0.73%, but it requires only 10.67 seconds to solve on average. In addition, the Electronic Companion reports the results obtained by 5 different runs of the methods. These results show that the methods are very robust since different runs of the methods give similar results.

6.3 Simulation of the Static-Static and Static-Dynamic Decision Frameworks

This subsection gives the results of the considered approaches on the classical instances in the static-static and static-dynamic decision frameworks. The results on the uncapacitated instances are reported first to analyze the performance of the lot-sizing rules. Then, the impact of the instances' characteristics is analyzed using all the instances. According to the results presented in Section 6.2, the following methods are considered: the two-stage model (denoted by *2-stage*) with 500 scenarios sampled with $RQMC$; the multi-stage model (denoted by *M-stage*) and the fix-and-optimize heuristic (denoted by *Fix-&-Opt*) using a scenario tree with structure $[50, 8, 4, 4]$ (resulting in 6400 scenarios); the *S-policy* determined by the multi-stage model using a scenario tree structure $[10, 10, 8, 8]$ (resulting in 6400 scenarios); the deterministic model with the average demands scenario (denoted *Average*), as well as the deterministic model with safety stock computed at the MPS level (denoted *SS-MPS*), and safety stock computed using the guaranteed service time model (denoted *SS-GS*); the lot-sizing rules with safety stock computed at the MPS level (*LL-MPS, EOQ-MPS, EOP-MPS, SM-MPS*), and safety stock computed with the guaranteed service time model (*LL-GS, EOQ-GS, EOP-GS, SM-GS*). However, SS-GS is only considered in the static-dynamic framework because the safety-stock of components cannot be transformed into end items in the static-static framework. Also, the lot-sizing rules are only considered for uncapacitated instances, since preliminary experiments (not presented here) showed that they perform poorly on the capacitated instances.

6.3.1 Uncapacitated Instances

Table 6 reports the performance of the methods on uncapacitated instances in the static-static environment. Table 6 gives the CPU time required to solve the problem at period 0, the GAP, the expected cost per component (setup, inventory, backlog, lost-sale, and production), the proportion of demand delivered on time (fulfillment rate), backlogged, and in lost sale, as well as the average number of setups, and the expected number of periods covered by each lot. 2-stage leads to the lowest costs (with a GAP of 3.1%), followed by SS-MPS (11.3%), and SS-GS (12.2%). Among the considered lot-sizing rules, SM-MPS and SM-GS perform the best, with a GAP of 18.9% and 19.2%, respectively.

Table 7 reports the results for the static-dynamic environment in the same format as Table 6. The results show that Average, LL-GS, and LL-MPS lead to higher total costs in the static-dynamic environment than in the static-static environment. For instance, Average has a GAP of 49.6% in the static-static environment, versus 59.0% in static-dynamic environment. This well-known behavior is caused by nervousness (e.g., Zhao and Lee 1993). However, the use of safety stocks attenuates this problem, for instance, SS-MPS performs better in static-dynamic (8.9%) than in static-static (11.3%). Stochastic optimization approaches are superior in such circumstances, as 2-stage performs even better with the static-dynamic framework (0.6%) than with the static-static framework (3.1%). The results also show that M-stage slightly outperforms 2-stage (with a GAP of 0.1% versus 0.6%), and that Fix-&-Opt performs well (with a GAP of 0.2%).

	CPU Time (sec.)	***GAP*** **(%)**	**Setup ($)**	**Inventory ($)**	**Backlog ($)**	**Lost sales ($)**	**Production ($)**	**Fulfillment (%)**	**Backlog (%)**	**Lost sales (%)**	**No. setup**	**Coverage**
Average	0.0	49.6	7,151.6	3,528.2	1,856.1	9,707.5	3,370.3	82.13	10.69	7.18	24.59	2.51
SS-MPS	0.2	11.3	7,839.4	4,992.7	1,188.6	3,088.9	4,055.2	89.53	8.72	1.75	25.39	2.70
LL-MPS	0.2	58.3	23,761.8	2,565.3	614.2	2,093.5	4,033.4	92.97	5.23	1.80	39.47	1.26
EOQ-MPS	0.2	59.6	13,916.0	11,610.3	101.5	525.3	4,895.4	97.70	1.73	0.57	30.11	2.61
EOP-MPS	0.3	22.8	11,051.9	7,613.8	260.6	196.6	4,387.8	97.46	2.35	0.19	21.48	3.03
SM-MPS	0.5	18.9	12,523.7	5,621.6	307.7	196.6	4,387.8	96.37	3.43	0.20	29.88	2.04
LL-GS	3.7	60.1	23,775.1	2,315.0	642.8	2,554.2	3,952.3	92.36	5.25	2.39	39.47	1.21
EOQ-GS	3.7	59.3	13,766.1	11,454.8	103.0	700.6	4,822.3	97.31	1.67	1.01	29.88	2.58
EOP-GS	3.8	22.1	11,052.2	7,050.1	280.4	673.3	4,288.4	96.99	2.45	0.56	21.47	2.88
SM-GS	3.9	19.2	12,457.0	5,239.7	336.5	673.3	4,287.4	95.70	3.70	0.60	29.80	1.95
2-stage	7.7	3.1	7,077.2	5,071.0	1,807.7	1,048.8	3,983.2	86.83	12.18	0.99	23.16	3.12

Table 6: Results for the 352 non-capacitated classical instances in static-static environments.

	CPU Time (sec.)	*GAP* (%)	Setup ($)	Inventory ($)	Backlog ($)	Lost sales ($)	Production ($)	Fulfillment rate (%)	Backlog (%)	Lost sales (%)	No. setup	Coverage
Average	0.0	59.0	7,151.6	3,455.9	1,950.3	10,879.0	3,304.1	80.58	11.49	7.93	24.59	2.50
SS-GS	10.6	10.0	7,779.0	4,574.1	1,255.0	3,554.5	3,904.1	88.35	9.55	2.10	25.23	2.61
SS-MPS	0.1	8.9	7,843.9	4,723.7	1,208.1	3,242.2	3,890.8	89.00	9.13	1.87	25.45	2.64
LL-MPS	0.2	64.0	23,765.8	2,291.3	691.9	3,132.2	3,833.6	91.61	5.75	2.64	39.51	1.20
EOQ-MPS	0.2	58.2	13,923.9	11,388.5	103.6	599.0	4,778.9	97.48	1.72	0.81	30.10	2.55
EOP-MPS	0.3	20.2	11,055.6	7,188.4	290.8	452.8	4,169.1	97.03	2.55	0.41	21.51	2.93
SM-MPS	0.5	16.9	12,528.2	5,223.4	344.2	469.0	4,132.8	95.58	3.96	0.46	29.93	1.95
LL-GS	3.7	64.0	23,775.1	2,182.0	700.4	3,142.7	3,820.3	91.34	5.78	2.88	39.47	1.19
EOQ-GS	3.7	57.6	13,766.1	11,280.9	106.8	725.0	4,758.2	97.23	1.69	1.08	29.88	2.55
EOP-GS	3.8	21.3	11,052.2	6,851.3	308.3	803.5	4,178.3	96.73	2.61	0.66	21.47	2.85
SM-GS	3.9	18.5	12,457.0	5,048.9	369.9	841.2	4,143.7	95.16	4.10	0.74	29.80	1.91
2-stage	7.7	0.6	7,077.2	4,831.8	1,831.7	1,044.6	3,891.9	86.71	12.33	0.96	23.16	3.08
S-Policy	310.2	0.6	7,322.5	4,866.0	1,719.0	857.3	3,893.4	87.31	11.87	0.82	23.74	3.03
Fix-&-Opt	232.0	0.2	7,077.2	4,988.1	1,797.4	840.8	3,924.4	87.09	12.18	0.74	23.16	3.17
M-stage	841.7	0.1	7,323.8	4,838.4	1,772.3	757.6	3,913.3	87.33	12.00	0.67	23.89	3.04

Table 7: Results for the 352 non-capacitated instances in static-dynamic environments.

6.3.2 All Instances

This section compares the performance of the methods on all classical instances. Table 8 reports the average *GAP*s for each method on each instance type for the static-static and static-dynamic environment, as well as the CPU time required to solve the instance at period 0. As for the uncapacitated instances, Average performs better in the static-static environment than in static-dynamic. Moreover, 2-stage significantly outperforms deterministic methods with safety stocks in static-static with an average *GAP* of 2.62% versus 8.36% for SS-MPS, as well as in static-dynamic with an average 0.63%, versus 6.21% and 6.80% for SS-MPS and SS-GS respectively. In the static-dynamic decision framework, M-stage slightly outperforms 2-stage, with an average *GAP* of 0.07% versus 0.63%. These results also confirm that Fix-&-Opt is efficient (with an average *GAP* of 0.27% versus 0.07% for M-stage), whereas Fix-&-Opt requires only a few minutes of computation time versus almost an hour for M-stage.

The static-dynamic use of M-stage, 2-stage, Fix-&-Opt, Average, SS-MPS, and SS-GS are cumbersome since they require to solve a linear program in each period. On the contrary, the static-dynamic use of the S-policy as well as the static-static framework do not require additional computations. Among these real-time execution methods, the S-policy performs the best with a *GAP* of 1.78%, followed by 2-stage in the static-static environment with a *GAP* of 2.62%. However, investing more computation power allows to reduce the costs, since M-stage leads to a *GAP* of 0.07% in the static-dynamic decision framework.

The Electronic Companion provides an analysis of the plans created with the considered approaches. This analysis shows that the stocks of components are larger in the static-dynamic than in the static-

static decision framework. Indeed, the static-dynamic framework allows to keep the excess stock at the component level (with lower holding costs) when demand is low. Since 2-stage overlooks the possibility to react to low demand by keeping stock of components, it overestimates the holding costs. Therefore, 2-stage creates smaller lots than M-stage and Fix-&-Opt.

The gap between stochastic optimization methods and classical methods is exacerbated for instances with the following characteristics: (1) The demand uncertainty is large (that is, the instances with lumpy demands, slow-moving items, low rate of known demand). For instance, the GAP of SS-MPS increases from 4.57% to 7.70%, when the rate of known demand decreases from 75% to 25%. (2) The times required to transform the components into end items are short (that is, the instances with lead times of type L2). (3) The value of the items increases significantly at each production step (that is, the instances with large echelon costs).

In addition, M-stage significantly outperforms 2-stage in terms of solution quality in these situations. For instance, the GAP of M-stage and 2-stage are respectively 0.05% and 1.90% for lumpy demands, versus 0.02% and 0.26% for non-stationary demands. Finally, all the methods (except the S-policy) have lower GAPs when capacity is tight. This is not surprising since the capacities constrain the solution. For instance, if the capacity is tight, the best production plan would be to produce as much as possible. However, S-policy performs poorly under tight capacities because of the greedy procedure used to repair capacity violations. Indeed, S-policy has a GAP 3.83% for an expected utilization of 90% versus 0.90% for an expected utilization of 50%, and the S-policy outperforms 2-stage in the static-static environment with a GAP of 0.59% versus 3.05%. Typically, when capacity is violated, the repair procedure reduces the quantity in equal proportion for all the items, whereas prioritizing some items might be preferable.

6.4 Rolling horizon simulation

In practice, production planing tools are often used in a rolling horizon framework (Chand et al. 2002). This section presents a rolling horizon simulation, and studies the impact of the considered planning horizon. The rolling horizon simulation is performed in a dynamic-dynamic decision framework on instances with large time horizon ($\mathcal{T} = \hat{T} + 10$ periods), and different planning horizons $\mathcal{H} < \mathcal{T}$ are considered. At period 0, the plan is optimized by considering the first $\mathcal{H}$ periods, and the decisions of period 0 are implemented. Then, the demands of period 0 are revealed, and

		Static-static				Static-dynamic						
		Average	**SS-GS**	**SS-MPS**	**2-stage**	**Average**	**SS-GS**	**SS-MPS**	**S-Policy**	**2-stage**	**Fix-&-Opt**	**M-stage**
Distribution	**NonStationary**	30.53	6.45	6.34	1.39	38.82	5.31	5.00	0.93	0.26	0.03	**0.02**
	Lumpy	79.33	22.96	22.09	9.77	103.42	18.26	15.44	5.18	1.90	0.96	**0.05**
	SlowMoving	59.67	13.19	12.78	6.52	69.66	8.84	7.79	6.03	2.69	1.78	**0.55**
NonStationary Rate of known	**25%**	35.70	8.05	7.70	1.95	45.16	6.41	5.97	1.22	0.34	0.04	**0.02**
	50%	32.35	6.77	6.74	1.41	41.11	5.65	5.37	0.96	0.26	0.03	**0.01**
	75%	23.54	4.54	4.57	0.81	30.20	3.86	3.67	0.60	0.16	0.02	**0.02**
NonStationary Coefficient of Variation	**0.1**	29.57	5.74	5.67	1.77	38.10	4.37	3.90	0.81	0.31	0.01	**0.00**
	0.4	29.73	5.79	5.81	1.13	38.69	4.87	4.71	1.05	0.32	0.04	**0.01**
	0.7	32.30	7.82	7.53	1.27	39.68	6.67	6.40	0.91	0.14	0.04	**0.03**
BOM	**Assembly**	32.06	6.15	5.84	1.50	44.50	5.34	4.50	0.96	0.50	0.20	**0.12**
	General	43.17	10.98	10.87	3.74	50.50	8.27	7.92	2.60	0.75	0.35	**0.01**
Utilization	**50%**	19.77	5.15	5.81	1.72	29.19	3.98	4.08	3.83	0.56	0.27	**0.01**
	90%	43.48	8.36	7.96	3.08	54.27	6.49	5.61	0.90	0.73	0.33	**0.10**
	Uncapacitated	49.60	12.18	11.30	3.05	59.03	9.94	8.92	0.60	0.59	0.22	**0.09**
Time between orders	**1**	63.37	9.43	9.24	4.70	79.84	6.31	5.17	2.34	1.05	0.44	**0.07**
	3	11.87	7.70	7.47	0.54	15.16	7.29	7.25	1.22	0.20	0.11	**0.07**
Lead time	**L1**	34.71	6.39	6.67	1.05	44.12	5.80	5.61	1.16	0.29	0.17	**0.08**
	L2	40.52	10.74	10.04	4.18	50.88	7.81	6.80	2.39	0.96	0.38	**0.06**
Echelon cost	**Normal**	22.92	6.71	6.90	1.52	30.50	5.94	5.33	2.04	0.26	0.15	**0.08**
	Large	52.31	10.42	9.81	3.72	64.50	7.67	7.08	1.51	1.00	0.40	**0.05**
Cost structure	**2**	22.89	6.46	6.91	2.35	29.79	4.87	4.71	1.56	0.61	0.26	**0.05**
	4	52.34	10.67	9.80	2.88	65.21	8.74	7.70	1.99	0.64	0.29	**0.09**
All instances		37.62	8.57	8.36	2.62	47.50	6.80	6.21	1.78	0.63	0.27	**0.07**
CPU Time		0.10	10.59	0.17	8.95	0.10	10.59	0.17	1064.03	8.95	234.16	3240.36

Table 8: GAPs of the methods on the 1056 classical instances.

the backlogs and inventories are observed. Considering this information, the plan is re-optimized on the horizon 1 to $\mathcal{H}+1$. This process continues until the last period. To keep the duration of the evaluation process reasonable, the simulations are performed on 100 scenarios on each instance. Indeed, with a time horizon of 10 periods, the evaluation over 100 scenarios requires to resolve the problem 1000 times per instance.

The following methods are considered in the rolling horizon framework: *Average*, *SS-MPS*, and *SS-GS* with a planning horizon of $(\hat{T}+5)$ periods; *2-stage-H1*, *2-stage-H3*, and *2-stage-H5* denote respectively the two-stage model with time-horizons of $(\hat{T}+1)$, $(\hat{T}+3)$, and $(\hat{T}+5)$ periods; *Fix-&-Opt-H1*, *Fix-&-Opt-H3*, and *Fix-&-Opt-H5* denote the fix-and-optimize heuristic with time-horizons of $(\hat{T}+1)$, $(\hat{T}+3)$, and $(\hat{T}+5)$ periods. Following the results presented in Section 6.2, the fix-and-optimize heuristic uses a scenario tree structure $[50, 8, 4, 4]$ for the first four levels, whereas the nodes of subsequent levels have one branch. We also evaluate two policies which do not require to solve a mathematical model in each period. In *S-Policy-H6*, the S-Policy is inferred (as explained in Section 4.2) from the solution of the fix-and-optimize heuristic with a planning horizon of $(\hat{T}+6)$ periods. The model is first solved on the planning horizon $0 \dots (\hat{T}+6)$, and the resulting S-policy is executed in periods $0 \dots 2$. Then, the model is solved with the planning horizon $3 \dots (\hat{T}+9)$, and the S-Policy is executed in period $3 \dots 5$. This process continues until the end of the horizon. In *Q-Policy-H6*, the two stage model is solved over a planning horizon of $(\hat{T}+6)$ periods, and the resulting production quantities are implemented for the first 3 periods. Then, the two-stage model is re-solved by shifting the planning horizon of 3 periods (similarly to S-Policy-H6). In addition, to compare the static-static decision framework and the rolling horizon approach, *Average*, *SS-MPS*, and *2-stage* are simulated in the static-static decision framework (i.e., the plan remains fixed). However, *2-stage* is solved with 100 scenarios only at each iteration, because the two-stage model with 500 scenarios per iteration requires too much memory on these instances.

Though the multi-stage model could be applied in a rolling-horizon framework, the evaluation over a large number of scenarios is not considered because it requires a significantly higher computation time. Also, the lot-sizing rules are not included in the rolling horizon simulation because they do not perform as well as other methods based on the results of the previous experiments.

Table 9 reports the detailed costs and the KPIs (in the same format as Table 6). Due to nervousness, SS-MPS in the static-static decision framework performs better (with a GAP of 10.0%) than SS-MPS and SS-GS in the rolling horizon framework (with a GAPs of 24.2% and 18.7%). On the contrary, the results show that the stochastic approaches perform better in a rolling horizon framework than in the static-static framework. For instance, 2-stage-H5 in the rolling horizon framework outperforms 2-stage in the static-static framework with respective GAPs of 3.3% and 6.6%. Using the multi-stage model has a significant impact in the rolling horizon environment, as Fix-&-Opt-H5 outperforms 2-stage-H5 with respective $GAPS$ of 1.5% and 3.3%. As expected, using large planning horizon leads to lower costs. For instance, the GAP of the fix-and-optimize method decreases from 13.7% to 1.5% when the planning horizon increases from $(\hat{T}+1)$ to $(\hat{T}+5)$. Finally, Q-Policy-H6 outperforms S-Policy with respective GAPs of 3.6% versus 5.2%. As mention earlier, S-Policy does not perform well in presence of capacity. However, on the subset of 12 uncapacitated instances, S-Policy outperforms Q-Policy with a gap of 3.9% versus 4.4%.

Framework	Method	GAP (%)	Setup ($)	Inventory ($)	Backlog ($)	Lost sales ($)	Production ($)	Fulfillment rate(%)	Backlog (%)	Lost sales (%)	No. setup	Coverage
Static-static for entire horizon	**Average**	24.9	16,861.7	11,241.2	4,520.2	9,868.0	6,432.3	74.6	16.2	9.2	53.7	2.7
	SS-MPS	10.0	19,902.2	12,949.8	3,274.3	2,212.4	7,207.9	80.9	16.9	2.2	57.7	2.7
	2-stages	6.6	17,382.1	12,527.6	4,915.3	1,781.7	7,175.2	79.3	18.7	2.1	42.3	3.4
Rolling horizon	**Average H5**	43.8	20,422.0	8,382.4	7,639.3	15,626.8	6,115.2	64.0	26.7	9.3	50.6	2.2
	SS-GS H5	18.7	20,108.9	11,165.3	5,821.7	3,714.7	6,811.8	74.3	23.7	2.0	53.2	2.6
	SS-MPS H5	24.2	20,094.3	10,453.6	5,887.6	6,650.4	6,620.2	72.0	24.0	4.0	52.3	2.5
	2-stage H1	17.2	26,315.4	6,733.0	8,976.4	3,197.6	6,662.0	71.6	25.9	2.5	60.5	2.0
	Fix-&-opt H1	13.7	25,956.9	7,452.5	7,758.7	1,923.1	6,756.4	74.1	24.2	1.7	57.8	2.2
	2-stage H3	5.1	19,898.8	9,871.1	5,707.8	2,388.4	6,700.1	74.5	23.1	2.4	46.8	2.7
	Fix-&-opt H3	3.2	19,403.4	10,627.1	5,431.8	1,298.7	6,819.1	77.2	21.2	1.6	44.4	2.9
	2-stage H5	3.3	19,049.8	9,510.0	5,779.0	2,638.8	6,701.5	75.7	21.6	2.7	44.0	2.8
	Fix-&-opt H5	1.5	18,810.9	10,572.8	4,997.4	1,365.5	6,823.5	78.4	20.1	1.6	42.7	3.0
	S-Policy H6	5.2	18,523.0	11,241.4	4,855.2	1,913.3	6,769.3	77.7	19.9	2.4	44.6	2.9
	Q-Policy H6	3.6	18,774.0	10,684.3	4,994.1	1,926.2	6,816.5	78.7	19.3	2.0	43.9	2.9

Table 9: Results of the rolling horizon simulation on instances with large horizon

7 Conclusion

This paper provides a two-stage and a multi-stage formulation for the multi-echelon multi-item capacitated lot-sizing problem encountered in MRP systems. The two-stage model assumes that production quantities are fixed for the entire horizon. On the contrary, the multi-stage model represents the static-dynamic decision framework, where the production quantities are sequentially determined in each period. As solving the multi-stage model is challenging, we propose a fix-and-optimize heuristic, as well as an order-up-to-level policy which allows to make real-time recourse decisions during the static-dynamic execution. Our experiments show that stochastic optimization models significantly outperform the common methods used in practice as well as other methods adapted from the literature. The use of the two-stage formulation in a dynamic environment leads to slightly higher costs than the multi-stage model, but the two-stage model is much faster to solve (by a factor of 100). Consequently, the implementation of the two-stage stochastic optimization model should be considered in MRP systems, since this method would allow manufacturers to determine a plan that yields an optimal trade-off between the service levels and to the production costs in this complex environment. Future works include the design of scalable methods to solve large size instances of the multi-echelon multi-item capacitated lot-sizing problem with stochastic demand. In addition, an interesting extension of the proposed approaches (which can explicitly take into account stochastic dynamic demand) is to incorporate component substitution (see Balakrishnan and Geunes 2000), to derive robust plans by exploiting the flexibility of such systems.

References

Aloulou, M. A., A. Dolgui, M. Y. Kovalyov. 2014. A bibliography of non-deterministic lot-sizing models. *International Journal of Production Research* **52**(8) 2293–2310.

Bai, X., J. S. Davis, J. J. Kanet, S. Cantrell, J. W. Patterson. 2002. Schedule instability, service level and cost in a material requirements planning system. *International Journal of Production Research* **40**(7) 1725–1758.

Balakrishnan, A., J. Geunes. 2000. Requirements planning with substitutions: exploiting bill-of-materials flexibility in production planning. *Manufacturing & Service Operations Management* **2**(2) 166–185.

Benton, W.C. 1991. Safety stock and service levels in periodic review inventory systems. *Journal of the Operational Research Society* **42**(12) 1087–1095.

Billington, Peter J, John O McClain, L Joseph Thomas. 1983. Mathematical programming approaches to capacity-constrained mrp systems: Review, formulation and problem reduction. *Management Science* **29**(10) 1126–1141.

Bitran, G. R., H. H. Yanasse. 1982. Computational complexity of the capacitated lot size problem. *Management Science* **28**(10) 1174–1186.

Blackburn, J. D., D. H. Kropp, R. A. Millen. 1985. MRP system nervousness: Causes and cures. *Engineering Costs and Production Economics* **9**(1-3) 141–146.

Blackburn, J. D., R. A. Millen. 1980. Heuristic lot-sizing performance in a rolling-schedule environment. *Decision Sciences* **11**(4) 691–701.

Bookbinder, James H, Jin-Yan Tan. 1988. Strategies for the probabilistic lot-sizing problem with service-level constraints. *Management Science* **34**(9) 1096–1108.

Boulaksil, Y. 2016. Safety stock placement in supply chains with demand forecast updates. *Operations Research Perspectives* **3**(1) 27–31.

Brandimarte, P. 2006. Multi-item capacitated lot-sizing with demand uncertainty. *International Journal of Production Research* **44**(15) 2997–3022.

Carlson, R. C., C. A. Yano. 1986. Safety stocks in MRP–systems with emergency setups for components. *Management Science* **32**(4) 403–412.

Chand, S., V. N. Hsu, S. Sethi. 2002. Forecast, solution, and rolling horizons in operations management problems: A classified bibliography. *Manufacturing & Service Operations Management* **4**(1) 25–43.

Clark, Alistair R, Vinicius A Armentano. 1995. A heuristic for a resource-capacitated multi-stage lot-sizing problem with lead times. *Journal of the Operational research Society* **46**(10) 1208–1222.

Cools, R., F. Y. Kuo, D. Nuyens. 2006. Constructing embedded lattice rules for multivariate integration. *SIAM Journal on Scientific Computing* **28**(6) 2162–2188.

Dolgui, A., C. Prodhon. 2007. Supply planning under uncertainties in MRP environments: A state of the art. *Annual Reviews in Control* **31**(2) 269–279.

Dupačová, Jitka, Nicole Gröwe-Kuska, Werner Römisch. 2003. Scenario reduction in stochastic programming. *Mathematical programming* **95**(3) 493–511.

Enns, S. T. 2002. MRP performance effects due to forecast bias and demand uncertainty. *European Journal of Operational Research* **138**(1) 87–102.

Graves, S. C., T. Schoenmeyr. 2016. Strategic safety-stock placement in supply chains with capacity constraints. *Manufacturing & Service Operations Management* **18**(3) 445–460.

Graves, S. C., S. P. Willems. 2008. Strategic inventory placement in supply chains: Nonstationary demand. *Manufacturing & Service Operations Management* **10**(2) 278–287.

Grubbström, R. W., Z. Wang. 2003. A stochastic model of multi-level/multi-stage capacity-constrained production–inventory systems. *International Journal of Production Economics* **81** 483–494.

Guide, V. D. R., R. Srivastava. 2000. A review of techniques for buffering against uncertainty with MRP systems. *Production Planning & Control* **11**(3) 223–233.

Ho, C.-J., Tim C. Ireland. 1998. Correlating MRP system nervousness with forecast errors. *International Journal of Production Research* **36**(8) 2285–2299.

Inderfurth, K. 2009. How to protect against demand and yield risks in MRP systems. *International Journal of Production Economics* **121**(2) 474–481.

Inderfurth, Karl, Stefan Minner. 1998. Safety stocks in multi-stage inventory systems under different service measures. *European Journal of Operational Research* **106**(1) 57–73.

Kadipasaoglu, S. N., V. Sridharan. 1995. Alternative approaches for reducing schedule instability in multi-stage manufacturing under demand uncertainty. *Journal of Operations Management* **13**(3) 193–211.

Kaminsky, P., J. M. Swaminathan. 2004. Effective heuristics for capacitated production planning with multiperiod production and demand with forecast band refinement. *Manufacturing & Service Operations Management* **6**(2) 184–194.

Kaut, M., S. W. Wallace. 2003. Evaluation of scenario-generation methods for stochastic programming.

Keutchayan, J., M. Gendreau, A. Saucier. 2017. Quality evaluation of scenario-tree generation methods for solving stochastic programming problems. *Computational Management Science* 1–33.

Khouja, M. 1999. The single-period (news-vendor) problem: literature review and suggestions for future research. *Omega* **27**(5) 537–553.

Lagodimos, A. G., E. J. Anderson. 1993. Optimal positioning of safety stocks in MRP. *The International Journal Of Production Research* **31**(8) 1797–1813.

L'Ecuyer, P., C. Lemieux. 2000. Variance reduction via lattice rules. *Management Science* **46**(9) 1214–1235.

L'Ecuyer, P., D. Munger. 2016. Algorithm 958: Lattice builder: A general software tool for constructing rank-1 lattice rules. *ACM Transactions on Mathematical Software (TOMS)* **42**(2) 15.

Lin, P.-C., R. Uzsoy. 2016. Chance-constrained formulations in rolling horizon production planning: an experimental study. *International Journal of Production Research* **54**(13) 3927–3942.

Pochet, Y., L. A. Wolsey. 2006. *Production Planning by Mixed Integer Programming*. Springer Science & Business Media.

Sali, M., V. Giard. 2015. Monitoring the production of a supply chain with a revisited MRP approach. *Production Planning & Control* **26**(10) 769–785.

Sloan, I. H., S. Joe. 1994. *Lattice Methods for Multiple Integration*. Oxford University Press.

Tempelmeier, H. 2013. Stochastic lot sizing problems. *Handbook of Stochastic Models and Analysis of Manufacturing System Operations*. Springer, 313–344.

Tempelmeier, H., M. Derstroff. 1996. A Lagrangean-based heuristic for dynamic multilevel multiitem constrained lotsizing with setup times. *Management Science* **42**(5) 738–757.

Venkataraman, Ray. 1996. Frequency of replanning in a rolling horizon master production schedule for a process industry environment: A case study. *Production and Operations Management* **5**(3) 255–265.

Zahorik, Anthony, L Joseph Thomas, William W Trigeiro. 1984. Network programming models for production scheduling in multi-stage, multi-item capacitated systems. *Management Science* **30**(3) 308–325.

Zhang, Yuli, Zuo-Jun Max Shen, Shiji Song. 2016. Distributionally robust optimization of two-stage lot-sizing problems. *Production and Operations Management* **25**(12) 2116–2131.

Zhao, X., F. Lai, T. S. Lee. 2001. Evaluation of safety stock methods in multilevel material requirements planning (mrp) systems. *Production Planning & Control* **12**(8) 794–803.

Zhao, X., T.S. Lee. 1993. Freezing the master production schedule for material requirements planning systems under demand uncertainty. *Journal of Operations Management* **11**(2) 185–205.

Zijm, Henk, Geert-Jan Van Houtum. 1994. On multi-stage production/inventory systems under stochastic demand. *International Journal of Production Economics* **35**(1-3) 391–400.

Electronic Companion For Stochastic Optimization for Material Requirements Planning

The electronic companion presents the proofs of Propositions 1-3 in Section 1, the instances generation method in Section 2, followed by the detailed experimental results. First, the performance of the multi-stage model is exposed for various scenario tree structures in Section 3, and Section 4 presents the experiments conducted on instances with small distribution support. Then, Section 5 compares the performance of the multi-stage model, the two-stage model, and the fix-and-optimize heuristic. Finally, an analysis of the plans created with the proposed methods is given in Section 6.

1 Proofs of propositions 1-3

Proof of proposition 1: Assume that there exists an optimal solution S with a positive inventory of a component i at the end of the horizon ($I_{iT} > 0$). If the initial stock of component i does not cover the production of end items, there must be some periods with production. We have $I_{i\tau} \geq I_{iT} > 0$ for all τ in $\{t + l_i, \dots, T\}$, where t denotes the last period with production for item i. Consider the solution S' constructed from S by reducing the quantity Q'_{it} of item i produced in period t, with $Q'_{it} = \max\{0, Q_{it} - I_{iT}\}$. As the inventory $I_{i\tau}$ of component i in each period τ in $\{t + l_i, \dots, T\}$ is lower in solution S' than in solution S (note that the stock levels $I_{i\tau}$ remain positive), which contradicts the optimality of S.

Proof of proposition 2: We show this proposition using a simple example with the BOM given in Figure 1, the two demand scenarios given in Table 1, and the parameters: $L_A = 1$, $L_B = 0$, $h_A = 0$, $h_B = 1$, $b_B = 100$, $s_A = s_B = 0$, $v_A = v_B = 0$. Table 1 gives the optimal production quantities and stock levels. Table 1 shows that the multi-stage model leads to a stock level of 1 for item B in the last period of scenario 2.

Proof of proposition 3: In the static-static framework, the special case of MMCLP with a single item and infinite capacity can be reformulated as the following MIP:

$$\min \quad \sum_{\omega\in\Omega} p_\omega \left(\sum_{t\in\mathcal{H}} (hI_t^\omega + sY_t + vX_t) + \sum_{i\in\mathcal{I}_e} \left(\sum_{t=1}^{t=T-1} bB_t^\omega + eB_T^\omega \right) \right) \tag{1}$$

$$\text{s.t.} \quad \sum_{\tau=1}^{t} X_\tau + I_0 - \sum_{\tau=1}^{t} D_\tau^\omega - I_t^\omega + B_t^\omega = 0 \qquad t\in\mathcal{H}, \quad \omega\in\Omega \tag{2}$$

$$X_t \leq MY_t \qquad t\in\mathcal{H} \tag{3}$$

$$X_t = 0 \qquad t < L, \quad \omega\in\Omega \tag{4}$$

$$B_t^\omega \geq 0, \qquad I_t^\omega \geq 0 \qquad t\in\mathcal{H}, \quad \omega\in\Omega \tag{5}$$

$$X_t \geq 0 \quad \text{and} \quad Y_t \in \{0,1\} \qquad t\in\mathcal{H}. \tag{6}$$

The variables and parameters are similar to those of two-stage MMCLP given in equations (1)-(8) in the main document, but the i index is dropped, since there is a single item. In addition, Q_t is replaced by the variable X_t, which denotes the quantity of item received in period t. Constraints (2), (3), (5), (6) are similar to the constraints of the two-stage MMCLP model, and constraints (4) state that no end item can be produced before the lead time.

As the lead time only affects constraints (4), a modification of the lead time has no impact on the optimal solution if constraints (4) are not binding (i.e., the initial inventory covers the demand from period 1 to L). In other words, ignoring the start of horizon effect, a change in the lead time does not affect the solution in the static-static framework.

The formulation of the static-dynamic decision framework is similar to model (1)-(6). However, the quantity X_t^ω received in period t depends on the scenario ω, and the following non-anticipativity constraints are added:

$$X_t^\omega = X_t^{\omega'} \quad \forall\, t\in\mathcal{H},\ \omega,\omega' | D_\omega^{1\ldots t-L-1} = D_{\omega'}^{1\ldots t-L-1}. \tag{7}$$

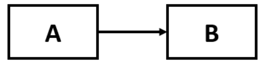


Figure 1: BOM of a small example with 2 items.

As a consequence, a reduction of the lead time removes some of the non-anticipativity contraints, which can result in a lower total cost.

Period			0		1	
Item			A	B	A	B
Demand		scenario 1	-	0	-	0
		scenario 2	-	0	-	1
Optimal quantity	**two-stage**	scenario 1	1	0	0	1
	multi-stage	scenario 1	0	0	0	1
		scenario 2	1	0	0	1
Inventory level	**two-stage**	scenario 1	0	0	0	0
	multi-stage	scenario 1	0	0	0	1
		scenario 2	0	0	0	0

Table 1: Demands scenario where the optimal solution of the multi-stage model has stock of components.

2 Instances Generation

As the instances from Tempelmeier and Derstroff (1996) were designed for the deterministic MRP with zero lead times, they are extended to include the demand's probability distribution and lead times. To demonstrate the performance and robustness of different approaches, we generate additional instances with different cost structures, varying echelon costs, large time horizons, and without capacity.

Three types of **lead times** are considered. In type L_1, all items have a lead time of 1 period. In type L_2, the lead times are equal to 1 for components, and to 0 for end items. In type L_3, the lead times are randomly chosen in $[0, 3]$, but they are generated such that the lead time from raw material to component does not exceed 5 periods. The consideration of lead times requires adding unit production costs and some periods with zero demand. The unit production cost v_i of item i is equal to the cost of holding components of i in stock during L_i periods ($v_i = \sum_{j \in \mathcal{I}} R_{ji} \cdot L_i \cdot h_j$). In addition, with non-zero lead times, a positive number $\hat{T}$ of periods is required to transform a raw component into an end item. As the initial inventory is null for the instances of Tempelmeier and Derstroff (1996), no demand can be fulfilled during the first $\hat{T}$ periods. Consequently, the demand is set to 0 in the first $\hat{T}$ periods (similarly to Almeder et al. (2015)).

The instances are either generated with a normal **time horizon** ($\hat{T} + 4$ periods) or a large time horizon ($\hat{T} + 10$ periods). For the instances with a normal time horizon, the demand forecast F_{it} of end-item i in period t (with $t \geq \hat{T}$) is the demand given in the instances of Tempelmeier and Derstroff (1996). For the instances with a large time horizon, the demand forecasts are generated with a normal distribution, with mean equal to the average demand in the instances of Tempelmeier and Derstroff (1996), and standard deviation equals to the coefficient of variation (in $\{0.1; 0.4; 0.7\}$) multiplied by the average demand.

Four **types of demand probability distributions** are considered in the experiments, namely, *Non-Stationary*, *Lumpy*, *Slow moving*, and *Binomial*. The *Non-stationary* distribution models the case where the demand's probability distribution is estimated based on known customer orders and some forecasts. The chosen probability distribution is motivated by two observations usually met in practice: (1) a larger fraction of the demand is known for the early periods; (2) the forecast errors are assumed to be normally distributed (as required by time series models). Consequently, the demand of item i in period t follows a Normal distribution with mean F_{it} and standard deviation $F_{it} \cdot (1-\alpha)^t \cdot e$, where e (set to 0.25) is the estimated forecast error, and $(1-\alpha)^t$ is the rate of unknown demands in period t (with α in $\{0.25, 0.5, 0.75\}$). The *Lumpy* distribution models items with erratic demands. Hence, the demand is null with probability 0.5 and follows a Poisson distribution with mean $2 \cdot F_i$ otherwise, where F_i is the average demand of item i in the instances of Tempelmeier and Derstroff (1996). *Slow moving* items are modeled by a Poisson distribution with mean 1. Finally, the *Binomial* distribution allows to compare the sampling approximation with the true optimal solution. Thus, the Binomial distribution allows to evaluate the quality of the sampling approximation. The demands follow binomial distributions with 7 trials, and success probability of 0.5. Since this distribution has a small support ($\{0, 1, 2, 3, 4, 5, 6, 7\}$), the complete set of scenarios is generated. Hence, the true optimal solution can be computed.

As the binomial and slow moving distributions modify the average demand, the setup costs and capacities are re-generated similarly to Tempelmeier and Derstroff (1996). The capacities are computed by dividing the average demand by the resource utilization (0.1, 0.5, or 0.9). The setup costs are generated as $s_i = \hat{h}_i \bar{D}_i \Theta^2/2$, where Θ is the time between orders (1, 2, 3, 4, or 5), and $\hat{h}_i$ is the echelon holding cost of item i (i.e., the incremental cost of holding one unit of item i rather than its components).

Two types of **cost structures** are considered. More precisely, the backlog costs are set to 2 or to 4 times the holding costs (given in the instances), whereas the lost sale costs are 10 times larger than backlog costs. Preliminary experiments show that the additional holding cost per production step is a sensitive parameter for the considered problem. As the instances of Tempelmeier and Derstroff (1996) assume a constant **echelon holding cost** (1 for all items), we generated some additional instances with large additional values at the last step. In these instances, the echelon holding costs $\hat{h}_i$ are set to 0.1, 1, and 10 for items at echelon 3, echelon 2 and end items, respectively. Since the echelon holding costs are modified, the setups costs must be re-generated. Finally, a set of instances are generated with unlimited **capacities**; that is, the capacity is set to a sufficiently large value, which allows to produce more items than the demand in the time horizon.

3 Solver Performance for the Multi-Stage Model

Table 2 gives the scenario tree structures associated with each number of scenarios, along with information on the models derived from these scenario trees (when demands are sampled with RQMC), namely, the *GAP*, the average and maximal computation times over all scenarios, the integrality gap reported by CPLEX after 10 hours of computation, and the number of rows and columns in the MILP reported by CPLEX after pre-solving the model. Note that the *GAP* is larger for 51,200 scenarios than for 25,600 scenarios because CPLEX is not able to solve to optimality the mathematical model with more than 25,600 scenarios (CPLEX reports an integrality gap of 0.02% with 25,600 scenarios).

Scenario Tree Structure	No. Scenarios	No. Instances	*Gap* (%)	Time (avg/max) (sec.)	Integrality Gap(%)	No. Columns	No. Rows
$[25, 4, 4, 4]$	1600	20	0.15	131.6/1042.05	0.00	8,402.45	11,825.70
$[25, 8, 4, 4]$	3200	20	0.10	573.65/4533.52	0.00	16,022.45	22,693.50
$[50, 8, 4, 4]$	6400	20	0.11	2862.54/17578.48	0.00	30,632.70	43,583.85
$[50, 8, 8, 4]$	12800	20	0.10	7198.68/36000.28	0.00	48,711.15	73,091.05
$[50, 16, 8, 4]$	25600	20	0.09	14102.6/36000.98	0.02	94,756.35	142,274.95
$[50, 32, 8, 4]$	51200	20	0.39	21453.45/36931.8	0.05	185,821.30	278,484.60

Table 2: Solver performance for the multi-stage model with different scenario tree structures.

4 Experiments on Instances with a Perfect Set of Scenarios

In this section, the quality of the sampling approximation is compared with the true optimal solutions of instances where the scenario set can be enumerated and the size is manageable. The considered instances have an assembly BOM (because it has a single end item), and binomial probability distributions (with a support containing only 8 values). As the time horizon contains 4 periods with stochastic demands (the first $\widehat{T}$ periods have 0 demand), the number of possible scenarios is limited to $8^4 = 4096$. Solving the two-stage and the multi-stage model over the 4096 scenarios leads to the true optimal solution in the static-static and static-dynamic environment, respectively.

Table 3 reports the GAPs of stochastic optimization approaches using either the perfect set of scenarios, or scenarios sampled with RQMC. Note that the GAPs reported in Table 3 are computed against the true optimal solution of the static-dynamic environment (i.e., the gap against the solution of the multi-stage model with the perfect set of scenarios). In addition, to compare the evaluation method presented in Section 5.2 of the main document with the perfect evaluation, the expected cost approximated with 5000 out-of-sample scenarios is reported in the column "Out-of-sample". Table 3 shows that RQMC scenario sampling yields good quality solutions. For instance, the GAP of the multi-stage model with RQMC is 0.04%. In addition, the out-of-sample evaluation is accurate, since the difference between the out of sample and perfect evaluation is never larger than 0.22%.

Environment	**Method**	**Scenario**	**No. Instances**	**Perfect evaluation (%)**	**Out-of-sample (%)**	**Difference (%)**
Static	Two-stage	RQMC	48	3.51	3.47	0.04
	Two-stage	Perfect	48	3.47	3.41	0.06
Static-dynamic	Two-stage	RQMC	48	1.34	1.15	0.19
	Two-stage	Perfect	48	0.90	0.68	0.22
	Multi-stage	RQMC	48	0.04	-0.15	0.19
	Multi-stage	Perfect	48	0.00	-0.19	0.19

Table 3: $\boldsymbol{GAP}$s of the stochastic optimization models on instances with small support.

Since the solution of the deterministic model on a specific scenario ω is in fact the perfect information solution for ω, the average costs of such solutions over the set Ω is a lower bound on the expected total cost. This lower bound corresponds to the expected cost obtained with knowledge of the future demand realization. The gap between the perfect information solution and the solution of the model is called expected value of perfect information ($EVPI$). $EVPI$ measures the potential gain associated with investments to get better demand forecasts. On the instance with small support, the average value of $EVPI$ are 55.96% and 49.77% for the static-static and static-dynamic decision frameworks, respectively.

5 Computational Comparisons of Different Models

Table 4 compares the performance (in terms of solution quality and computation times) of the multi-stage model, the two-stage model, and the fix-and-optimize heuristic in the static-dynamic environment, where the multi-stage model and the fix-and-optimize heuristic use the scenario tree structure $[50, 8, 4, 4]$. More precisely, Table 4 reports the GAP of each method, as well as the average and maximum computation times (over all instances), the integrality gap reported by CPLEX after 10 hours of computation, and the number of rows and columns reported by CPLEX after pre-solve. The results show that fix-and-optimize is efficient with a GAP of 0.30% versus 0.11% for the multi-stage model, whereas the multi-stage model is significantly slower to solve (241 seconds versus 3010 seconds on average). The two-stage model leads to slightly larger costs, with a GAP of 0.73%, but it requires only 10.67 seconds to solve on average.

	No. Instances	GAP (%)	Time avg/max (sec.)	Integrality gap (%)	No. Variables	No. Constraints
Two-stage	20	0.73	10.32/23.94	0.00	869.60	1,675.30
Multi-stage	20	0.11	3010.5/17711.38	0.00	30,632.70	43,583.85
Fix-and-optimize	20	0.30	241.19/284.39	0.00	9,991.25	24,939.30

Table 4: Performances of the two stage model, multi-stage model, and fix-and-optimize heuristic in the static-dynamic decision framework on a subset of 20 classical instances.

To evaluate the robustness of the approximation obtained by RQMC sampling, Table 5 shows the average, maximum and minimum GAP obtained by 5 different runs of the two-stage and multi-stage models. Table 5 shows that the methods are very robust since the difference between the maximum and minimum GAP is lower than 1%.

Environment	**Method**	**Average**	**Min**	**Max**
Static-static	Two-stage	2.98	2.72	3.33
Static-dynamic	Two-stage	1.00	0.58	1.43
	Multi-stage	0.25	0.02	0.53

Table 5: Average, minimum, and maximum GAP on 5 different runs with different seeds.

6 Analysis of the Plans

To analyze the differences between the plans created with stochastic optimization methods and the plans created with classical methods, Table 6 gives the average levels of stock at each echelon of the supply chain, and Table 7 shows the distribution of the expected cost per component (setup, inventory, backlog, lost sale, and production), the proportion of demand delivered on time (fulfillment rate), backlogged, and in lost sale, as well as the average number of setups, and the expected number of periods covered by each order. Four observations can be made from Tables 6 and 7:

- In the static-static environment, the surplus of end-item stocks created by 2-stage compared to Average (30.27 vs 20.96) can be considered as a safety stock.
- All methods hold higher amount of stocks at echelon 2 and 3 in the static-dynamic environment than in static-static (i.e, the echelon 2 items are not transformed to end-items). For instance, 2-stage has 9.28 units of stock at echelon 2 in static-dynamic framework, versus 7.28 units with static-static framework. This is due to the fact that if the demands observed in period t are lower than expected, keeping the additional stock at the higher echelon leads to lower inventory costs than keeping end-items stock.
- M-stage leads to slightly larger setup costs than 2-stage ($10,739.5\$$ vs $10,568.0\$$), as the additional setups allow to better adjust to the latest information.
- Larger lots are created by Fix-&-Opt than by the two-stage model (whereas production occurs at the same periods), with a total amount of stock of 47.89 versus 45.61. 2-stage overestimates the holding costs and creates smaller lots, because it does not consider the possibility of leaving the surplus of stock at higher echelon during optimization.

Framework	Method	Stock End-Item	Stock Echelon 2	Stock Echelon 3
static-static	**Average**	20.96	6.53	6.27
	SS-MPS	30.48	6.53	5.87
	SS-Grave	29.08	7.17	5.88
	2-stages	30.27	7.28	6.10
static-dynamic	**Average**	18.91	9.38	9.04
	SS-MPS	28.11	7.85	7.72
	SS-Grave	27.75	8.44	7.70
	2-stage	28.29	9.28	8.04
	S-Policy	28.20	11.02	9.59
	Fix-&-Opt	28.88	10.16	8.85
	M-stage	28.48	10.60	9.11

Table 6: Stock levels at each echelon.

References

Almeder, C., D. Klabjan, R. Traxler, B. Almada-Lobo. 2015. Lead time considerations for the multi-level capacitated lot-sizing problem. *European Journal of Operational Research* **241**(3) 727–738.

Framework	Method	Setup ($)	Inventory ($)	Backlogs ($)	Lost sales ($)	Production ($)	Fulfillment rate(%)	Backlogs (%)	Lost sales (%)	No. setup	Coverage
static-static	**Average**	10,359.6	3,126.7	2,130.2	9,725.1	3,369.2	81.6	11.1	7.3	27.4	2.0
	SS-MPS	10,938.7	4,389.6	1,456.1	4,292.2	3,840.4	88.1	8.8	3.1	27.9	2.1
	SS-Grave	10,907.9	4,204.6	1,502.7	4,536.5	3,798.8	87.4	9.2	3.4	27.8	2.1
	2-stages	10,568.0	4,448.6	1,815.1	2,801.1	3,813.8	86.7	10.7	2.6	26.9	2.3
static-dynamic	**Average**	10,359.6	3,044.4	2,236.8	11,481.6	3,277.2	79.6	11.9	8.5	27.4	2.0
	SS-MPS	10,942.4	4,182.7	1,479.1	4,301.8	3,735.5	87.7	9.2	3.1	27.9	2.1
	SS-Grave	10,907.9	4,103.8	1,498.7	4,431.5	3,751.3	87.4	9.4	3.3	27.8	2.1
	2-stages	10,568.0	4,262.7	1,822.1	2,739.7	3,742.8	86.7	10.8	2.5	26.9	2.3
	S-Policy	10,791.4	4,298.5	1,739.5	2,893.0	3,734.3	87.1	10.3	2.6	27.4	2.2
	Fix-&-Opt	10,568.0	4,353.8	1,799.6	2,594.5	3,765.5	86.9	10.7	2.4	26.9	2.3
	M-stage	10,739.0	4,300.8	1,765.3	2,479.5	3,765.1	87.3	10.5	2.2	27.5	2.3

Table 7: KPIs of the different models.

Tempelmeier, H., M. Derstroff. 1996. A Lagrangean-based heuristic for dynamic multilevel multiitem constrained lotsizing with setup times. *Management Science* **42**(5) 738–757.