Production, Manufacturing and Logistics

Computing non-stationary (*s*, *S*) policies using mixed integer linear programming

Mengyuan Xiang [a,*], Roberto Rossi [a], Belen Martin-Barragan [a], S. Armagan Tarim [b]

[a] *Business School, University of Edinburgh, Edinburgh, United Kingdom*
[b] *Department of Management, Cankaya University, Ankara, Turkey*

binary search   3. 7%
18%   8

A B S T R A C T

This paper addresses the single-item single-stocking location non-stationary stochastic lot sizing problem under the (*s*, *S*) control policy. We first present a mixed integer non-linear programming (MINLP) formulation for determining near-optimal (*s*, *S*) policy parameters. To tackle larger instances, we then combine the previously introduced MINLP model and a binary search approach. These models can be reformulated as mixed integer linear programming (MILP) models which can be easily implemented and solved by using off-the-shelf optimization software. Computational experiments demonstrate that optimality gaps of these models are less than 0.3% of the optimal policy cost and computational times are reasonable.

© 2018 Elsevier B.V. All rights reserved.

1. Introduction

Stochastic lot sizing is an important research area in inventory theory. One of the landmark studies is Scarf (1960), which proved the optimality of (*s*, *S*) policies for a class of dynamic inventory models. The (*s*, *S*) policy features two control parameters: *s* and *S*. Under this policy, the decision maker checks the opening inventory level at the beginning of each time period: if it drops to or below the reorder point *s*, then a replenishment should be placed to reach the order-up-to-level *S*. Unfortunately, computing optimal (*s*, *S*) policy parameters remains a computationally intensive task.

Since Scarf's landmark study, the (*s*, *S*) policy has been object of extensive research. For instance, Johnson and Thompson (1975), Sethi and Cheng (1997), Chen and Song (2001), and Hu, Zhang, and Zhu (2016) investigated demand correlation; more recently, Qiu, Sun, and Lim (2017) and Lim and Wang (2017) investigated demand distributional ambiguity.

In the literature, studies on (*s*, *S*) policy can be categorized into stationary and non-stationary. A number of studies investigated the computation of stationary (*s*, *S*) policy parameters, e.g. (Archibald & Silver, 1978; Federgruen & Zipkin, 1984; Feng & Xiao, 2000; Iglehart, 1963; Sahin, 1982; Stidham, 1977; Veinott, Arthur, & Wagner, 1965; Zheng & Federgruen, 1991). However, there has

been an increasing recognition that lot-sizing studies need to be undertaken for non-stationary environments (Graves, 1999).

In this work, we focus on the single-item single-stocking location stochastic lot-sizing problem under non-stationary demand, fixed and unit ordering cost, holding cost and penalty cost. Only two studies investigated computations of (*s*, *S*) policy under non-stationary stochastic demand (Askin, 1981; Bollapragada & Morton, 1999).

Askin (1981) adopted the "least cost per unit time" approach in selecting order-up-to-levels and reorder points under a penalty cost scheme. Decision makers first determine desired cycle lengths and order-up-to-levels. Then, reorder points are decided by means of a trade-off analysis between expected costs per period in cases of ordering and not ordering.

As Bollapragada and Morton (1999) pointed out, the approach discussed by Askin (1981) is computationally expensive because of the need of convolving demand distributions. In contrast, Bollapragada and Morton (1999) proposed a stationary approximation heuristic for computing optimal (*s*, *S*) policy parameters. Firstly, decision makers precompute pairs of (*s*, *S*) values for various demand parameters and tabulate results. Here, a large number of efficient algorithms exist for generating the stationary table, e.g. (Federgruen & Zipkin, 1984; Feng & Xiao, 2000; Zheng & Federgruen, 1991). Secondly, order-up-to-levels and reorder points can be read from stationary tables by averaging the demand parameters over an estimate of the expected time between two orders. However, this algorithm relies upon complex code, particularly for generating stationary tables.

---

* Corresponding author.
   E-mail addresses: mengyuan.xiang@ed.ac.uk (M. Xiang), roberto.rossi@ed.ac.uk (R. Rossi), Belen.Martin@ed.ac.uk (B. Martin-Barragan), at@cankaya.edu.tr (S.A. Tarim).

Askin (1981) and Bollapragada and Morton (1999) do not provide a satisfactory solution to the problem of computing near-optimal (s, S) policy parameters: they rely on ad-hoc computer coding and provide relatively large optimality gaps. A recent computational study Dural-Selcuk, Kilic, Tarim, and Rossi (2016) estimated the optimality gap of Askin (1981) and Bollapragada and Morton (1999) at 3.9% and 4.9%, respectively; these figures are in line with those reported in the original works. These drawbacks motivate the investigation of simple and yet effective heuristic methods for computing (s, S) policy parameters; methods that do not need dedicated computer coding and that can provide better optimality gaps.

The aim of this paper is to introduce two new heuristics to compute near-optimal (s, S) policy parameters. We build upon Rossi, Kilic, and Tarim (2015), which discussed mixed-integer linear programming (MILP) heuristics for approximating optimal (R, S) policy parameters − under this policy, the replenishment intervals R and order-up-to-levels S are determined at the beginning of the planning horizon, while associated order quantities are decided only when orders are issued. The (R, S) policy is effective in dealing with system nervousness (Tunc, Kilic, Tarim, & Eksioglu, 2013), while the (s, S) policy is cost-optimal (Scarf, 1960). Our two mixed-integer nonlinear programming (MINLP)-based heuristics leverage two key building blocks: modeling techniques originally discussed in Rossi et al. (2015), and K-convexity of the problem cost function, originally discussed in Scarf (1960). In contrast to other approaches in the literature, our heuristics can be easily implemented and solved by using off-the-shelf mathematical programming packages such as IBM ILOG optimization studio.

Our contributions to literature on stochastic lot-sizing are the following.

- We introduce the first mixed integer non-linear programming (MINLP) model to compute near-optimal (s, S) policy parameters.
- We show that this model can be approximated as a mixed integer linear programming (MILP) model by piecewise linearising the cost function; this approximation can be solved by using off-the-shelf software.
- To tackle larger instances, we combine the previously introduced MINLP model and a binary search procedure; this latter approach requires dedicated code, but scales better than the previous one.
- Computational experiments demonstrate that optimality gaps of our models are tighter than existing algorithms (Askin, 1981; Bollapragada & Morton, 1999) in the literature, and computational times of our models are reasonable.

The rest of this paper is organized as follows. Section 2 describes the problem setting and a stochastic dynamic programming (SDP) formulation. Section 3 discusses the notion of K-convexity and introduces relevant K-convex cost functions which are approximated by an MINLP model in Section 4. Section 5 presents an MINLP heuristic for approximating (s, S) policy parameters. Section 6 introduces an alternative binary search approach for computing (s, S) policy parameters. A detailed computational study is given in Section 7. Finally, we draw conclusions in Section 8.

2. Problem description

We consider a single-item single-stocking location inventory management system over a T-period planning horizon. We assume that orders are placed at the beginning of each time period, and delivered instantaneously. Ordering costs $c(\cdot)$ comprise a fixed ordering cost K for placing an order, and a linear ordering cost c proportional to order quantity Q. Demands d_t in each period

$t = 1, \ldots, T$ are independent random variables with known probability distributions. At the end of period t, a linear holding cost h is charged on every unit carried from one period to the next; and a linear penalty cost b is occurred for each unmet demand at the end of each time period.

For a given period $t = \{1, \ldots, T\}$, let I_{t-1} denote the opening inventory level and Q_t represent the order quantity.

The immediate expected holding and penalty costs at period t can be expressed as

$$f_t(y) = \mathrm{E}[h \max(y - d_t, 0) + b \max(d_t - y, 0)], \tag{1}$$

where E denotes the expectation taken with respect to the random demand d_t.

The ordering cost $c(Q_t)$ is defined as:

$$c(Q_t) = \begin{cases} K + c\, Q_t, & Q_t > 0 \\ 0, & Q_t = 0 \end{cases}$$

Let $C_t(I_{t-1})$ represent the expected total cost of an optimal policy over periods $t, \ldots, T$ when the initial inventory level at the beginning of period t is I_{t-1}. We model the problem as a stochastic dynamic program (Bellman, 1957) via the following functional equation

$$C_t(I_{t-1}) = \min_{Q_t} \{ c(Q_t) + f_t(I_{t-1} + Q_t) + \mathrm{E}[C_{t+1}(I_{t-1} + Q_t - d_t)] \}$$

$$\tag{2}$$

where

$$C_T(I_{T-1}) = \min_{Q_t} \{ c(Q_T) + f_T(I_{T-1} + Q_T) \}$$

represents the boundary condition.

3. The optimality of (s, S) policies in stochastic lot sizing

Scarf (1960) proved that the optimal policy in the dynamic inventory problem is always of the (s, S) type based on a study of the function $G_t(y) + cy$, where

$$G_t(y) = f_t(y) + \mathrm{E}[C_{t+1}(y - d_t)], \tag{3}$$

and y is the stock level immediately after purchases are delivered (see Scarf, 1960, Eq. (4)).

Since we consider a non-stationary environment, values of the (s, S) policy parameters will depend on the given period t. Let (s_t, S_t) denote the policy parameters for period t. Function $G_t(y) + cy$ can be used to identify optimal policy parameters (s_t, S_t). In particular, the order-up-to-level S_t is defined as the value minimising $G_t(y) + cy$; whereas the parameter s_t is given by the value $s_t < S_t$ such that $G_t(s_t) + cs_t = G_t(S_t) + cS_t + K$ (see Scarf, 1960, Eq. (5)). K-convexity of the function $G_t(y) + cy$ ensures the uniqueness of s_t and S_t (Scarf, 1960).

Example. We illustrate the concepts introduced on a 4-period example. Demand d_t is normally distributed in each period t with mean $\mu_t = \{20, 40, 60, 40\}$, for $t = 1, \ldots, 4$ respectively. Standard deviation σ_t of demand in period t is equal to $0.25\mu_t$. Other parameters are $K = 100$, $h = 1$, $b = 10$, and $c = 0$. We plot $G_1(y)$ in Fig. 1 for initial inventory levels $y \in (0, 200)$. The order-up-to-level is $S_1 = 70$, $G_1(S_1) = 263$, the reorder point is $s_1 = 14$, and $G_1(s_1) = 363$. Note that $G_1(s_1) + cs_1 = G_1(S_1) + cS_1 + K$. The optimal policy is to order up to 70 if the initial inventory drops below 14.

4. MINLP approximation of $G_t(y)$ function

In this section, we exploit an MINLP model to approximate the function $G_t(y)$ in Eq. (3). Our model follows the control policy known as "static-dynamic uncertainty" strategy, known as (R, S) policy, originally introduced in Bookbinder and Tan (1988).
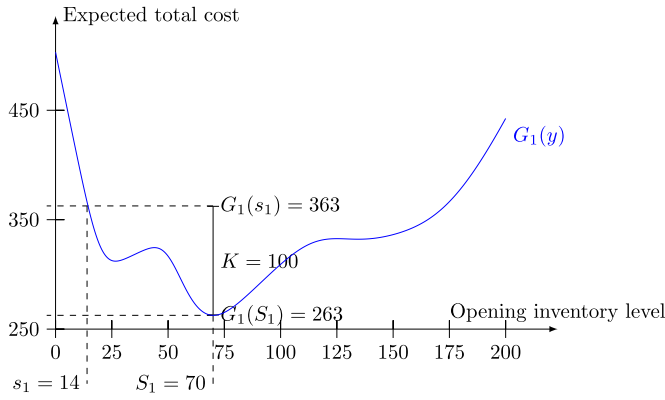
Fig. 1. Plot of $G_1(y)$.

Under this strategy, the timing of orders and order-up-to-levels are expected to be determined at the beginning of the planning horizon, while associated order quantities are decided upon only when orders are issued. As illustrated in Rossi et al. (2015), this strategy provides a cost performance which is close to the optimal "dynamic uncertainty" strategy. However, optimal (s, S) parameters cannot be immediately derived from existing mathematical programming models operating under a static-dynamic uncertainty strategy, such as (Rossi et al., 2015; Tarim & Kingsman, 2006). We next illustrate how a model operating under a static-dynamic uncertainty strategy can be used to approximate the function $G_t(y)$ in Eq. (3). In the rest of this section, without loss of generality, we focus on the case $G_1(y)$.

Consider a random variable ω and a scalar variable x. The first order loss function is defined as $L(x, \omega) = E[\max(\omega - x, 0)]$, where E denotes the expected value with respect to the random variable ω. The complementary first order loss function is defined as $\hat{L}(x, \omega) = E[\max(x - \omega, 0)]$. Like Rossi et al. (2015), we will model non-linear holding and penalty costs by means of this function.

Let $t = 1, \ldots, T$ and consider three sets of decision variables: $\tilde{I}_t$, the expected closing inventory level at the end of period t, with I_0 denoting the initial inventory level; δ_t, a binary variable which is set to one if an order is placed in period t; P_{jt}, a binary variable which is set to one if the most recent replenishment up to period t was issued in period j, where $j \leq t$ — if no replenishment occurs before or at period t, then we let $P_{1t} = 1$, this allows us to properly account for demand variance from the beginning of the planning horizon in Constraints (9) and (10). Let $\tilde{d}_{jt}$ denote the expected value of the demand over periods $j, \ldots, t$, i.e. $\tilde{d}_{jt} = \tilde{d}_j + \cdots + \tilde{d}_t$. Decision variables $H_t \geq 0$ and $B_t \geq 0$ represent end of period t expected excess inventory and back-orders, respectively. An MINLP formulation for the non-stationary stochastic lot-sizing problem under the "static-dynamic" uncertainty strategy, obtained following the modelling strategy in Rossi et al. (2015), is shown in Fig. 2.

The objective function (4) minimizes the expected total cost over the planning horizon. In the objective function, expected variable ordering costs are reformulated via $c \sum_{t=1}^{T} Q_t = c\tilde{I}_T + c \sum_{t=1}^{T} \tilde{d}_t - cI_0$ by using the reformulation strategy originally introduced in Tarim and Kingsman (2004) at p. 112 – note that term $c \sum_{t=1}^{T} \tilde{d}_t - cI_0$ is a constant. Constraints (5) is an indicator constraint (Belotti et al., 2016) capturing the reorder condition. Constraints (6) are the inventory balance equations. Constraints (7) indicate the most recent replenishment before period t was issued in period j. Constraints (8) identify uniquely the period in which the most recent replenishment prior to t took place. Constraints (9) and (10) are indicator constraints modelling end of period t expected excess inventory and back-orders by means of the first order loss function.

Fig. 2. The formulation of the non-stationary stochastic lot-sizing problem.

We now discuss how to adapt the model in Fig. 2 in order to compute, for a given y, an approximate value of $G_1(y)$; see Eq. (3). We call this modified model MINLP-s, and use superscript s to label decision variables in this model.

In addition to constraints in Fig. 2, MINLP-s features constraint

$$\delta_1^s = 0, \tag{13}$$

which forces the model not to issue an order in period 1. When $\delta_1^s = 0$, the objective function (4) becomes

$$G_1^s(I_0^s) = \underbrace{hH_1^s + bB_1^s}_{f_1(I_0^s)}$$

$$+ \underbrace{\sum_{t=2}^{T}(K\delta_t^s + hH_t^s + bB_t^s)}_{\text{fixed ordering, holding, and penalty cost for } t=\{2,\ldots,T\}} + \underbrace{c\tilde{I}_T^s + c\sum_{t=2}^{T}\tilde{d}_t - c\tilde{I}_1^s}_{\text{proportional ordering cost for } t=\{2,\ldots,T\}} \tag{14}$$

$$\underbrace{}_{\approx E[C_2(I_0^s - d_1)]}$$

which denotes the expected total cost of controlling the system optimally over the planning horizon $1, \ldots, T$ when the initial inventory level is I_0^s and no order is issued in period 1; hence $c(\tilde{I}_1^s + \tilde{d}_1 - I_0^s) = 0$.

MINLP-s can be reformulated into an MILP model by using the approach discussed in Rossi et al. (2015) to piecewise linearize loss functions in constraints (9) and (10). For further details please refer to Appendix A.

Example. In Fig. 3, we plot $G_1^s(y)$ obtained via the MILP-s for the same 4-period numerical example in Fig. 1 with initial inventory level $I_0^s \in (0, 200)$.

Since $G_1^s(y)$ approximates $G_1(y)$, we can now use $G_1^s(y) + cy$ to find approximate values $\hat{S}_1$ and $\hat{s}_1$ for S_1 and s_1.

5. An MINLP-based model to approximate (s, S) policy parameters

In this section, we exploit the results presented in the previous section to introduce an MINLP-based heuristic for approximating optimal (s, S) policies. To the best of our knowledge, this is the first MINLP model in the literature for computing near-optimal (s, S) policy parameters.
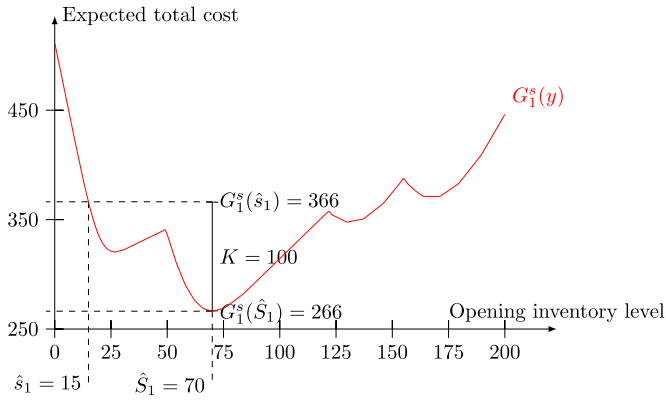
Fig. 3. Plot of $G_1^s(y)$.

In a similar fashion to "MINLP-s", we introduce "MINLP-S" to be the approximation of $C_t(I_{t-1})$ in Eq. (2). Similarly to Eq. (14), let the objective function $C_1^S(I_0)$ of MINLP-S denote the expected total cost of controlling the system optimally over the planning horizon $1, \ldots, T$ given the initial inventory level I_0. We use the superscript S to represent decision variables in MINLP-S,

$$C_1^S(I_0) = \sum_{t=1}^{T}(K\delta_t^S + hH_t^S + bB_t^S) + cI_T^S + c\sum_{t=1}^{T}\tilde{d}_t - cI_0. \quad (15)$$

MINLP-S imposes the constraint

$$\delta_1^S = 1, \quad (16)$$

which forces the model to place a replenishment in period 1.

In the MINLP-S model, $\hat{S}_1$ denotes an approximation of the optimal order-up-to-level S_1. Since $G_1^s(I_0^s)$ is an approximation of $G_1(I_0^s)$, by leveraging Scarf's result (see Scarf, 1960, Eq. (4)) on the study of $G(y) + cy$, we can identify $\hat{s}_1 = I_0^s$ such that $G_1^s(I_0^s) + cI_0^s = G_1^s(\hat{S}_1) + c\hat{S}_1 + K$. Therefore, we can approximate s_1 by imposing the constraint

$$G_1^s(I_0^s) + cI_0^s = C_1^S(I_0^S) + cI_0^S, \quad (17)$$

in which I_0^S represents an approximation $\hat{S}_1$ of the optimal order-up-to-level S_1[1]. Note that $C_1^S(I_0^S)$ includes the fixed ordering cost K because of Constraint (16); variable ordering cost in $C_1^S(I_0^S)$ is zero because I_0^S is its global minimizer. Therefore Eq. (17) is equivalent to $G_1^s(I_0^s) + cI_0^s = G_1^s(\hat{S}_1) + c\hat{S}_1 + K$.

Finally, since $s_1 \leq S_1$, we introduce an additional constraint to ensure that the reorder point is not greater than the order-up-to-level,

$$I_0^s \leq I_0^S. \quad (18)$$

Note that, in contrast to the true value $G_1(y) + cy$, there is no guarantee that K-convexity holds for its approximation $G_1^s(y) + cy$. For some instances we may therefore have multiple values $\hat{s}_1$ such that (17) holds. As we will demonstrate in our computational study, leaving to the solver the freedom to choose one of such values in a non-deterministic fashion leads to competitive results.

MINLP-S and MINLP-s are connected by Eq. (17), in such a way the order-up-to-level S_1 and the reorder point s_1 are approximated simultaneously. For the joint MINLP model, in addition to decision variables in MINLP-S and MINLP-s, we consider I_0^S, a dummy variable representing the approximate order-up-to-level $\hat{S}_1$; and I_0^s, which captures the approximate reorder point $\hat{s}_1$.

---

[1] I_0^s, which is a dummy variable, should not be confused with the actual initial inventory level I_0, which is needed to account for variable ordering costs.

Table 1
Optimal (s, S) policy parameters obtained via the joint MINLP heuristic and the stochastic dynamic programming.

	MP				SDP			
t	1	2	3	4	1	2	3	4
s_t	15.0	29.0	58.1	29.0	14.0	29.0	58.0	28.0
S_t	70.3	54.0	117	54.0	70.0	141	114	53.0
$G_t^s(s_t)$	366	311	193	118	363	303	190	118

Our holistic MINLP model objective features two parts: the first part, $C_1^S(I_0)$, comes from MINLP-S; the second part, $G_1^s(I_0^s) + cI_0^s - f_1(I_0^s) \approx \mathrm{E}[C_2(I_0^s - d_1)]$, from MINLP-$s$. Note that the term $f_1(I_0^s)$, which enhances computational performance of the model, can be introduced because holding and penalty costs in period 1 for model MINLP-s are already uniquely determined by Eq. (17). After dropping the constant term $c\sum_{t=1}^{1} d_t - cI_0$ in the first part and the constant term $c\sum_{t=1}^{T}\tilde{d}_t$ in the second part, we minimize the following holistic objective function

$$\min\left(\sum_{t=1}^{T}(K\delta_t^S + hH_t^S + bB_t^S) + cI_T^S + \sum_{t=2}^{T}(K\delta_t^s + hH_t^s + bB_t^s) + cI_T^s\right); \quad (19)$$

Constraints of the joint MINLP model are those of both MINLP-S and MINLP-s in addition to the linking constraints (13), (16)–(18). After solving the joint MINLP model over planning horizon $k, \ldots, T$, the estimated order-up-to-level $\hat{S}_k$ is equal to I_{k-1}^S, and the estimated reorder point $\hat{s}_k$ is equal to I_{k-1}^s. As previously discussed, the joint MINLP model can be linearized via the piecewise-linear approximation proposed in Rossi et al. (2015). In our MILP model, (9) and (10) are modelled via the piecewise OPL expression (IBM, 2011). For a complete overview of the MILP model refer to Appendix B.

Example. We now use the same 4-period numerical example in Fig. 3 to demonstrate the modelling strategy behind the joint MINLP heuristic (MP). We observe that, for period 1, the approximated order-up-to-level is $S_1 = 70.3$, the reorder point is $s_1 = 15.0$, and $G_1^s(s_1) = 366$ (363, after simulation) as shown in Fig. 1. By solving the joint MINLP model repeatedly, s_t, S_t, and $G_t^s(s_t)$, for $t = 1, \ldots, 4$, are estimated as shown in Table 1. We also compare our results against the optimal solutions obtained via SDP in Table 1; note that although different order-up-to-levels, e.g. S_2, are obtained, the optimal expected total costs are similar.

6. A binary search approach to approximate (s, S) policy parameters

The joint MINLP heuristic presented in the last section is valuable, since it can be easily linearized into an MILP model that can be solved by off-the-shelf solvers. However, according to our experience it can only effectively tackle small-size instances. To preserve the advantage of relying on an MILP model, one may investigate efficient reformulations, valid inequalities, or may explore cut generation techniques that enhance computational performances; we however choose to leave this investigation as future work.

In order to tackle larger-size problems, in this section we introduce an efficient approach that combines the model MINLP-s discussed in Section 5 and a binary search strategy. This approach relies on the MINLP models previously introduced, but it has the disadvantage of requiring dedicated code for the search procedure.

Our binary search strategy (Algorithm 1) is structured as follows.

Computation of S (lines 2–3). We first let I_0^s to be a decision variable in MINLP-s and minimize $G_1^s(I_0^s) + cI_0^s$ to estimate the order-up-to-level S_1.

Data: costs (ordering cost, holding cost, penalty cost), mean
demand and standard deviation of each period, stepsize
Result: pairs of s and S for each time period

1 **for** $k = 1$ **to** T **do**
2   Minimize MINLP-s in Section 5 in OPL;
3   Obtain $G_k^s(\hat{S}_k)$ and $\hat{S}_k$;
4     $low =$ a large negative integer; $high = \hat{S}_k$;
5   **while** $low < high$ **do**
6        $mid = low + \text{round}((high - low)/2)$;
7   Run the MINLP-s with $I_{k-1}^s = mid$ in OPL;
8   Obtain current cost $G_k^s(I_{k-1}^s)$;
9   **if** $G_k^s(I_{k-1}^s) - G_k^s(\hat{S}_k) - K - c(\hat{S}_k - I_{k-1}^s) < 0$ **then**
10           $high = mid - $ stepsize;
11   **else if** $G_k^s(I_{k-1}^s) - G_k^s(\hat{S}_k) - K - c(\hat{S}_k - I_{k-1}^s) > 0$ **then**
12           $low = mid + $ stepsize;
13   **else**
14           $\hat{s}_k = I_{k-1}^s$;
15           $low = high$;
16   **end**
17   **end**
18 **end**

Algorithm 1: The binary search algorithm.

Table 2
Near-optimal (s, S) policy parameters obtained via the binary search approach.

t	1	2	3	4
s_t	15	29.0	58.1	29.0
S_t	70.3	54.0	116	54.0
$G_t^s(s_t)$	366	311	193	118

Computation of s (lines 5–17). Since $G_1^s(I_0^s)$ is an approximation of $G_1(y)$, we can conduct a binary search to approximate the reorder point $\hat{s}_1$ by $I_0^s \leq \hat{S}_1$ at which $G_1^s(I_0^s) + cI_0^s = G_1^s(\hat{S}_1) + c\hat{S}_1 + K$. When the binary search terminates, the estimated reorder point $\hat{s}_k$ is equal to I_{k-1}^s.

By repeating this procedure (line 1) over the planning horizon $k, \ldots, T$, we find pairs of $\hat{S}_k$ and $\hat{s}_k$, where $k = 1, \ldots, T$.

Example. We illustrate the solution method discussed via the same 4-period numerical example presented in Fig. 1. We assume the step size of the binary search is 0.01. The order-up-to-level $\hat{S}_1 = 70.3$ and $G_1^s(70.3) = 266$. We then set $low = -200$, $high = 70.3$. While $low < high$, the mid is updated via the comparison of $G_1^s(I_0^s) + K$ and $G_1^s(\hat{S}_1) + K$. Eventually, we obtain the reorder point $\hat{s}_1 = 15$ at which $G_1^s(\hat{s}_1) + c\hat{s}_1 = G_1^s(\hat{S}_1) + c\hat{S}_1 + K$. By repeating this procedure we obtain $\hat{S}_t$, $\hat{s}_t$, and $G_t^s(s_t)$, for each period $t = 1, \ldots, 4$ as displayed in Table 2. After simulation, we obtain the expected total cost 363.

7. Computational experiments

In this section, we present an extensive analysis of the heuristics discussed in Sections 5 (MP) and 6 (BS). We first design a test bed featuring instances defined over an 8-period planning horizon (Section 7.1). On this test bed, we assess the behaviour of the optimality gap and the computational efficiency of both the MP and BS heuristics. Then we assess the computational performance of the BS heuristics on a test bed featuring larger instances on a 25-period planning horizon (Section 7.2). For all cases, MINLP models are solved by employing the piecewise

linearization strategy discussed in Rossi et al. (2015), which can be easily implemented in OPL by means of the piecewise syntax. Numerical experiments are conducted by using IBM ILOG CPLEX Optimization Studio 12.7 and MATLAB R2014a on a 3.2 gigahertz Intel(R) Core(TM) with 8 gigabytes of RAM.

7.1. An 8-period test bed

We consider a test bed which includes 540 instances. Specifically, we incorporate ten demand patterns displayed in Fig. 4. These patterns comprising two life cycle patterns (LCY1 and LCY2), two sinusoidal patterns (SIN1 and SIN2), a stationary pattern (STA), a random pattern (RAND), and four empirical patterns (EMP1,..., EMP4). Full details on the experimental set-up are given in Appendix C. Fixed ordering cost K ranges in {200, 300, 400}, proportional ordering cost c ranges in {0, 1}, and the penalty cost b takes values {5, 10, 20}. We assume that demand d_t in each period t is independent and normally distributed with mean $\tilde{d}_t$ and coefficient of variation $c_v \in \{0.1, 0.2, 0.3\}$; note that $\sigma_t = c_v \tilde{d}_t$. Since we operate under the assumption of normality, our models can be readily linearized by using the piecewise linearization parameters available in Rossi, Tarim, Prestwich, and Hnich (2014). However, the reader should note that our proposed modelling strategy is distribution independent, see Rossi et al. (2015).

We set the SDP model discussed in Section 2 as a benchmark. We compare against this benchmark in terms of optimality gap and computational time. First of all, we obtain optimal parameters for each test instance by implementing an SDP algorithm in MATLAB. Then, we solve each instance by adopting both modelling heuristics presented in Sections 5 and 6. Specifically, for the MP heuristic we employ seven segments in the piecewise-linear approximations of B_t and H_t (for $t = 1, \ldots, T$) in order to guarantee reasonable computational performances; for the BS heuristic, whose computational performance is only marginally affected by an increased number of segments in the linearization, we employ eleven segments and a step size 0.1. To estimate the cost of the policies obtained via our heuristics, we simulate all policies via Monte Carlo Simulation (10,000 replications).

Table 3 gives an overview of optimality gaps (%) of methods discussed in this study for different pivoting parameters. It is difficult to make a general remark with respect to demand pattern, and fixed ordering cost; while the proportional ordering cost has a negative correlation with the optimality gap. An increase in proportional ordering cost slightly reduces the optimality gap. While an increase in penalty cost increases the optimality gap. Specifically, when penalty cost increases from 10 to 20, the optimal gap rises from 0.25 to 0.42% and from 0.27 to 0.35%, respectively. Similarly, an increase in coefficient of variation increases the optimality gap. For example, the optimality gap of the BS heuristic increases significantly from 0.16 to 0.40% as the coefficient of variation increases from 0.1 to 0.3. Overall, the average optimality gap of the MP heuristic is 0.29%, and that of the BS heuristic is 0.26%. This discrepancy ought to be expected, since in the case of the BS method a higher number of segments has been employed.

Existing heuristics Askin (1981) and Bollapragada and Morton (1999) were reimplemented by Dural-Selcuk et al. (2016) and assessed on a test bed that neatly resembles the one adopted in this work. As shown in Dural-Selcuk et al. (2016), Askin's optimality gap is 3.9%, and Bollapragada and Morton's is 4.9%. The optimality gap of our heuristic is 0.29% when seven segments are employed in the piecewise linearization, and it drops to 0.26% when eleven segments are employed. Our models outperform both Askin (1981) and Bollapragada and Morton (1999) in terms of optimality gap on the test bed here considered.

We also assess the accuracy of our models by comparing the costs predicted by our models against the costs obtained via
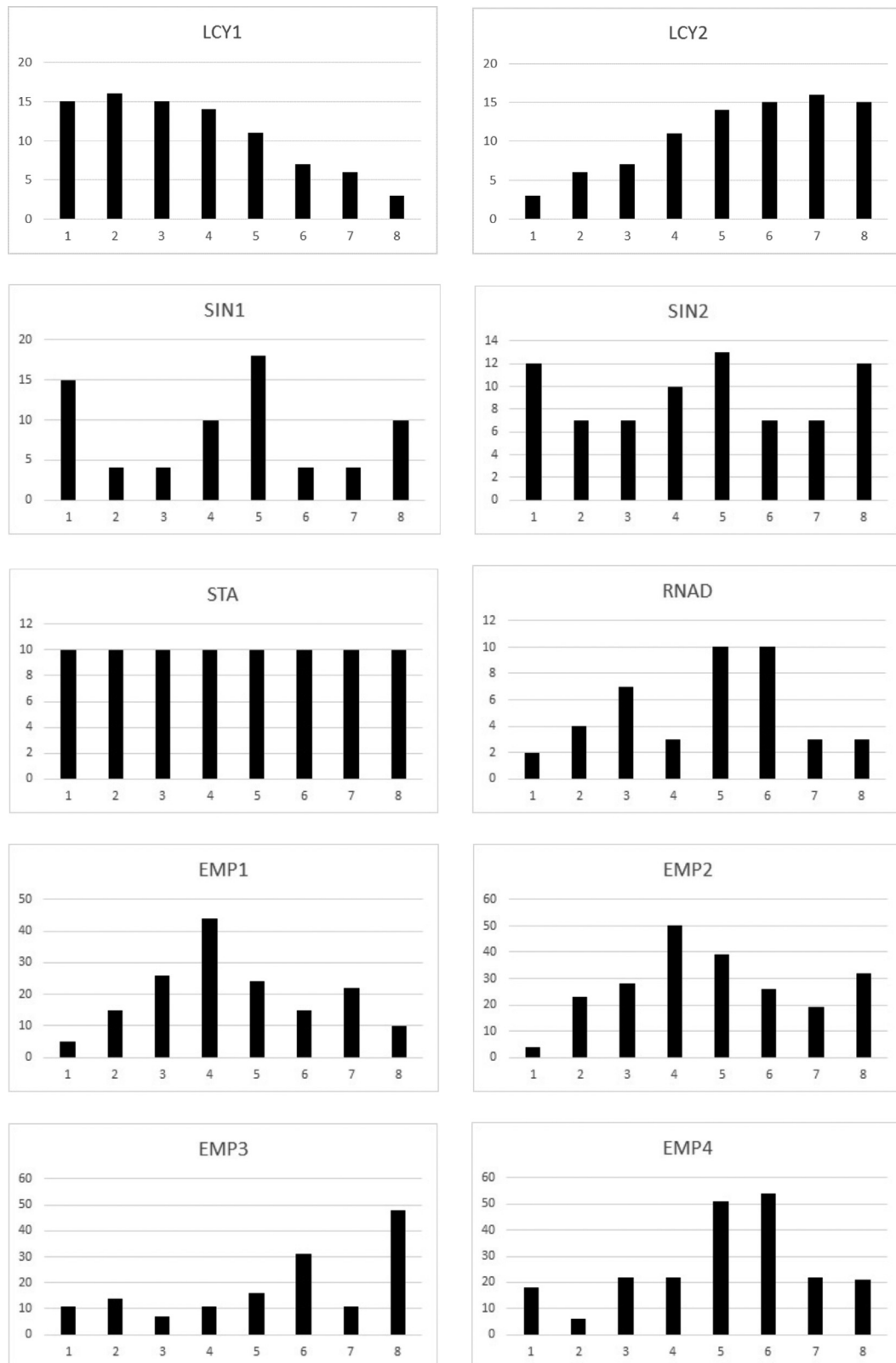
Fig. 4. Demand patterns in our computational analysis.

simulation. We note that both MP and BS heuristics have high accuracy for the 8-period numerical experiments. For further details please refer to Table D.8 in Appendix D.

Table 4 shows computational times of our models for different pivoting parameters. Note "STDEV" in Table 4 represents the standard deviation. Overall, the computational time of BS method remains stable for different set-up parameters; while that of MP and SDP algorithms fluctuate. We observe that the fixed ordering cost, proportional ordering cost, penalty cost, and coefficient of variation do not have significant effect on the computational

Table 3
Average optimality gaps % of the 8-period numerical experiment for different pivoting parameters.

Modelling methods	MP	BS
Demand pattern		
LCY1	0.25	0.33
LCY2	0.11	0.18
SIN1	0.13	0.20
SIN2	0.10	0.19
STA	0.50	0.14
RAND	0.16	0.22
EMP1	0.41	0.35
EMP2	0.86	0.52
EMP3	0.15	0.19
EMP4	0.28	0.28
Fixed ordering cost		
200	0.31	0.29
300	0.24	0.22
400	0.34	0.27
Proportional ordering cost		
0	0.33	0.29
1	0.26	0.23
Penalty cost		
5	0.21	0.16
10	0.25	0.27
20	0.42	0.35
Coefficient of variation		
0.1	0.22	0.16
0.2	0.26	0.22
0.3	0.40	0.40
Average	0.29	0.26

Table 4
Average computational times (seconds) of the 8-period numerical for different pivoting parameters.

Settings	MP		BS		SDP	
	Mean	STDEV	Mean	STDEV	Mean	STDEV
Demand pattern						
LCY1	3.54	0.98	7.17	1.21	13.58	0.86
LCY2	6.26	4.52	7.29	1.06	13.61	0.81
SIN1	4.67	3.20	6.48	0.69	13.31	1.11
SIN2	4.15	1.89	6.41	0.63	13.60	0.82
STA	5.52	3.68	6.48	0.72	9.95	2.29
RAND	3.60	0.87	7.11	1.32	710.12	2.95
EMP1	7.65	6.21	7.32	0.96	121.81	28.60
EMP2	14.03	13.60	7.28	1.19	107.20	7.37
EMP3	14.32	11.81	7.02	0.83	104.41	10.17
EMP4	15.12	15.35	7.52	1.20	122.71	27.79
Fixed ordering cost						
200	10.29	11.18	7.11	1.00	53.03	51.94
300	7.17	6.94	6.99	1.00	53.07	51.98
400	6.19	5.40	6.93	1.08	52.99	51.91
Proportional ordering cost						
0	8.49	9.06	7.64	0.99	60.21	60.12
1	7.28	7.57	6.38	0.59	45.85	40.85
Penalty cost						
5	8.05	7.92	6.96	0.90	53.08	52.03
10	8.72	10.60	6.86	1.02	52.74	51.94
20	6.84	8.84	7.17	1.14	52.97	51.85
Coefficient of variation						
0.1	6.42	6.16	7.00	1.08	53.06	51.96
0.2	7.98	8.92	7.02	0.99	53.01	51.92
0.3	9.26	9.43	7.01	1.03	53.02	51.95
Average	7.89	8.39	7.01	1.03	53.03	51.85

Table 5
BS heuristics on a 25-period test bed, average computational times (seconds) with different setting parameters.

Settings	Mean	Standard deviation
Demand pattern		
LCY1	531.66	204.45
LCY2	740.73	322.92
SIN1	500.44	177.17
SIN2	1622.92	624.58
STA	1709.00	706.67
RAND	407.08	131.11
EMP1	633.09	126.63
EMP2	188.19	37.45
EMP3	974.93	305.16
EMP4	173.95	44.87
Fixed ordering cost		
500	934.92	811.90
1000	762.96	540.73
1500	546.75	341.41
Proportional ordering cost		
0	827.15	680.28
1	669.25	534.88
Penalty cost		
5	713.45	564.80
10	782.53	669.09
20	744.28	612.21
Coefficient of variation		
0.1	679.34	567.29
0.2	755.92	619.07
0.3	809.34	656.18
Average	748.20	616.43

$$d_t = \text{round}\left(\frac{190 \times e^{-(t-13)^2}}{2 \times 5^2}\right), \qquad t = 1, 2, \ldots, T \qquad (20)$$

$$d_t = \text{round}\left(\frac{170 \times e^{-(t-13)^2}}{2 \times 6^2}\right), \qquad t = 1, 2, \ldots, T \qquad (21)$$

$$d_t = \text{round}\left(70 \times \sin(0.8t) + 80\right), \qquad t = 1, 2, \ldots, T \qquad (22)$$

$$d_t = \text{round}\left(30 \times \sin(0.8t) + 100\right), \qquad t = 1, 2, \ldots, T \qquad (23)$$

$$d_t = 100, \qquad t = 1, 2, \ldots, T \qquad (24)$$

$$d_t = \text{round}(\text{random}(0, 250)), \qquad t = 1, 2, \ldots, T \qquad (25)$$

Fig. 5. Expressions for generating demand data.

efficiency of BS and SDP algorithms. However, the computational time of MP heuristic drops significantly with the increase of fixed ordering cost, and proportional ordering cost; while grows greatly with the increase of the coefficient of variation. On average, the computational time of MP, BS, and SDP are 7.89, 7.01, and 53.03 seconds.

7.2. A 25-period test bed

As shown in Section 7.1 for the 8-period test bed, both the MP and the BS methods provide tight optimality gaps and acceptable computational efficiency. We now extend the 8-period test bed to 25 periods with larger instances. Demands of LCY1, LCY2, SIN1, SIN2, STA, and RAND are generated with expressions (20), (21), (22), (23), (24), and (25) in Fig. 5. Demands of EMP1, EMP2, EMP3 and EMP4 are derived from Strijbosch, Syntetos, Boylan, and Janssen (2011). Full details are given in Appendix C. Assume that fixed ordering cost ranges in {500, 1000, 1500}, proportional ordering cost ranges in {0, 1}, penalty cost takes values {5, 10, 20}, and the coefficients of standard deviations are {0.1, 0.2, 0.3}.

We obtain optimal (s, S) parameters and record computational times obtained via the BS method. For the first 15 periods we perform binary search with step size 1 in order to ensure fast convergence; for the last 10 periods, we adopt a step size 0.1 to enhance accuracy. The number of segments used in the piecewise linearization is eleven. To estimate the cost of the policy obtained via our approximation, we simulate each instance ten thousand times in MATLAB.

We observe that the BS model has high accuracy even for the large-size numerical experiments. We report detailed model accuracy in Table D.9 in Appendix D.

In Table 5, we summarise computational times of the BS model for different pivoting parameters. It is difficult to make a general remark with respect to demand patterns. An increase in fixed ordering cost significantly decreases the computational time. For instance, the computational time drops from 934.92 to 546.75 seconds as the fixed ordering cost increases from 500 to 1500. An increase in proportional ordering cost decreases the computational time. In contrast, an increase in coefficient of variation increases the computational time. For instance, when the coefficient of variation rises from 0.1 to 0.2, the computational time increases from 679.34 to 809.34 seconds. On average, the computational time is 748.20 seconds and the standard deviation is 616.43 seconds.

8. Conclusion

In this paper we discussed two MINLP-based heuristics for tackling non-stationary stochastic lot-sizing problems under (s, S) policy.

Our first heuristic – the first MINLP heuristic for computing near-optimal non-stationary (s, S) policy parameters – is based on mathematical programming models that can be solved by using off-the-shelf optimization packages. These MINLP models can be linearized via the approach discussed in Rossi et al. (2015) and can be implemented in OPL by adopting the `piecewise` expression.

Our second heuristic is a binary search strategy that leverages the aforementioned MINLP models and can tackle larger-size problems. However, this latter heuristic requires dedicated code.

We conducted an extensive computational study comprising 540 instances. We considered ten demand patterns, three fixed ordering costs, two proportional ordering cost, three penalty costs and three coefficients of variation.

We first conducted a numerical study on small instances (8-period). We investigated the performance of both models by contrasting costs of the policy obtained with our models against costs of the optimal policy obtained via the stochastic dynamic programming. Optimality gaps observed are generally below 0.3%. Our sensitivity analysis showed that the optimality gap is tighter when the demand keeps stable, and performance deteriorate with the increase of the penalty cost and the coefficient of variation; both heuristics provide tighter gaps than those reported in the literature (Askin, 1981; Bollapragada & Morton, 1999).

The computational study carried out on larger instances (25-period) showed that the computational efficiency of the binary search approach is reasonable: around 748.20 seconds on average.

Appendix A. The `piecewise` OPL constraint

Rossi et al. (2015) piecewise linearized loss functions in constraints (9) and (10) by employing piecewise linear approximations based on Jesen's and Edmundson–Madanski inequalities. An alternative strategy is to model these non-linear functions by exploring the `piecewise` syntax in OPL. By using this syntax, a piecewise function is specified by giving a set of slopes which represent the

```
piecewise(i in 1..W){
slope[i] -> breakpoint[i];
 slope[W+1]
}(<knownpoint>,<valuepoint>)<value>;
```

Fig. A.6. The syntax of the `piecewise` command in OPL.

$$P_{jt} = 1 \rightarrow H_t = \texttt{piecewise}\{l_i \rightarrow X_i; 1\}(0, \hat{L}_{up}(0, d_{jt}))\tilde{I}_t,$$
$$i = 1, \ldots, W; \; j = 1, \ldots, t. \quad \text{(A.1)}$$

$$P_{jt} = 1 \rightarrow B_t = \texttt{piecewise}\{-1 + l_i \rightarrow X_i; 0\}(0, \hat{L}_{up}(0, d_{jt}))\tilde{I}_t,$$
$$i = 1, \ldots, W; \; j = 1, \ldots, t. \quad \text{(A.2)}$$

Fig. A.7. Rewriting holding and penalty costs by adopting `piecewise` syntax.

linear variation for each linear segment; a set of breakpoints at which slopes change; and the function value at a known point.

The `piecewise` syntax in OPL is given in Fig. A.6. `W` is the number of breakpoints of the piecewise function. `slope[i]` and `breakpoint[i]` denote slope and breakpoint of segment i. Segment i goes from breakpoint $(i-1)$ to breakpoint (i). `<valuepoint>` is the function value at a known point `<knownpoint>`. Finally, `<value>` represents the value at which we evaluate the function.

For the OPL `piecewise` syntax, there are three key components: slope, breakpoint, and function value at a known point. The following lemmas will demonstrate how to deduce their values. Let Ω be the support of ω. Let $(\Omega_i)_{i=1,\ldots,W+1}$ be a partition of Ω in $W + 1$ segments.

Lemma 1. *The slope of ith segment is written as*

$$l_i = \sum_{k=1}^{i-1} p_k, i \in \{1, 2, \ldots, W + 1\},$$

where $p_i = Pr\{\omega \in \Omega_i\} = \int_{\Omega_i} g_\omega(t)dt$, $g_\omega(\cdot)$ denotes the probability density function of ω.

Proof. Observation from Rossi et al. (2014), Lemma 11. □

Lemma 2. *The ith breakpoint can be written as*

$$X_i = E[\omega|\Omega_i], i \in \{1, 2, \ldots, W\}.$$

Proof. Observation from Rossi et al. (2014), Lemma 11. □

Note that when ω follows a normal distribution with mean μ and standard deviation σ, then $\hat{L}_{up}(x, \omega) = \sigma \hat{L}_{up}(\frac{x-\mu}{\sigma}, Z)$, where Z follows a standard normal distribution, see Lemma 7 in Rossi et al. (2014).

Lemma 3. *Assume that the partition of Ω is symmetric with respect to 0, then the function value $\hat{L}_{up}(x, \omega)$ at point 0 can be written as follows.*

$$\hat{L}_{up}(0, \omega) = \begin{cases} -\sum_{k=\frac{W+1}{2}}^{W+1} p_k E[\omega|\Omega_k] + e_W, & W \text{ is odd} \\ -\frac{1}{2}(\sum_{k=1}^{\frac{W}{2}} p_k E[\omega|\Omega_k] + \sum_{k=\frac{W}{2}+1}^{W+1} p_k E[\omega|\Omega_k]) + e_W, & W \text{ is even} \end{cases}$$

where e_W represents the approximation error.

Proof. Since the partition of Ω is symmetric when W is odd, $x = 0$ is the central breakpoint. Hence, the function value at this breakpoint can be calculated directly. However, when W is even,

the function value at point $x = 0$ is the average of nearest two symmetric breakpoints $X_{\frac{W}{2}}$ and $X_{\frac{W}{2}+1}$. $\square$

Following Lemmas 1–3, constraints (9) and (10) in Fig. 2 can be expressed as Eqs. (A.1) and (A.2) in Fig. A.7, for $t = 1, \ldots, T$.

Appendix B. The MILP model

The joint MILP model to calculate near-optimal (s, S) policy parameters for the non-stationary stochastic lot-sizing problem is presented below.[2] In the joint MP model, constraints (B.3) represent the costs of controlling the system optimally when the initial inventory level is I_0^s; constraints (B.14) denote the costs of controlling the system optimally when the initial inventory level is I_0^s, and no order is placed in period 1. These two constraints are connected via constraints (B.27) such that the order-up-to-level S_1 and reorder point s_1 are approximated by I_0^S and I_0^s respectively.

$$\min\left(\sum_{t=1}^{T}(K\delta_t^S + hH_t^S + bB_t^S) + c\tilde{I}_T^S + \sum_{t=2}^{T}(K\delta_t^s + hH_t^s + bB_t^s) + c\tilde{I}_T^s \right) \tag{B.1}$$

Subject to, $t = 1, \ldots, T$ $\tag{B.2}$

$$C_1^S(I_0^S) = \sum_{t=1}^{T}(K\delta_t^S + hH_t^S + bB_t^S) + c\tilde{I}_T^S + c\sum_{t=1}^{T}\tilde{d}_t - cI_0^S \tag{B.3}$$

$$\tilde{I}_t^S + \tilde{d}_t - \tilde{I}_{t-1}^S \geq 0 \tag{B.4}$$

$$\delta_t^S = 0 \to \tilde{I}_t^S + \tilde{d}_t - \tilde{I}_{t-1}^S = 0 \tag{B.5}$$

$$\sum_{j=1}^{t}P_{jt}^S = 1 \tag{B.6}$$

$$P_{jt}^S \geq \delta_j^S - \sum_{k=j+1}^{t}\delta_k, j = 1, \ldots, t \tag{B.7}$$

$$\delta_1^S = 1 \tag{B.8}$$

$$I_0^S = \tilde{I}_1^S + \tilde{d}_1 \tag{B.9}$$

$$H_t^S \geq (I_t^S + \sum_{j=1}^{t}d_{jt}P_{jt}^S)\sum_{k=1}^{i}p_k - \sum_{j=1}^{t}(\sum_{k=1}^{i}p_kE[d_{jt}|\Omega_i] - e_W)P_{jt}^S,$$
$$i = 1, \ldots, W \tag{B.10}$$

$$B_t^S \geq -I_t^S + (I_t^S + \sum_{j=1}^{t}d_{jt}P_{jt}^S)\sum_{k=1}^{i}p_k - \sum_{j=1}^{t}(\sum_{k=1}^{i}p_kE[d_{jt}|\Omega_i] - e_W)P_{jt}^S,$$
$$i = 1, \ldots, W \tag{B.11}$$

$$P_{jt}^S \in \{0, 1\}, j = 1, \ldots, t \tag{B.12}$$

---
[2] The loss function is piecewise linearized via constraints (B.10), (B.11), (B.22), and (B.23).

$$\delta_t^S \in \{0, 1\} \tag{B.13}$$

$$G_1^s(I_0^s) = (hH_1^s + bB_1^s) + \sum_{t=2}^{T}(K\delta_t^s + hH_t^s + bB_t^s) + c\tilde{I}_T^s + c\sum_{t=2}^{T}\tilde{d}_t - cI_1^s \tag{B.14}$$

$$\tilde{I}_t^s + \tilde{d}_t - \tilde{I}_{t-1}^s \geq 0 \tag{B.15}$$

$$\delta_t^s = 0 \to \tilde{I}_t^s + \tilde{d}_t - \tilde{I}_{t-1}^s = 0 \tag{B.16}$$

$$\sum_{j=1}^{t}P_{jt}^s = 1 \tag{B.17}$$

$$P_{jt}^s \geq \delta_j - \sum_{k=j+1}^{t}\delta_k^s, j = 1, \ldots, t \tag{B.18}$$

$$\delta_1^s = 0 \tag{B.19}$$

$$P_{jt}^s = 1 \to H_t^s = \texttt{piecewise}\{l_i \to X_i; 1\}(0, \hat{L}_{up}(0, d_{jt}))\tilde{I}_t^s,$$
$$i = 1, \ldots, W$$
$$j = 1, \ldots, t \tag{B.20}$$

$$P_{jt}^s = 1 \to B_t^s = \texttt{piecewise}\{-1 + l_i \to X_i; 0\}(0, \hat{L}_{up}(0, d_{jt}))\tilde{I}_t^s$$
$$i = 1, \ldots, W$$
$$j = 1, \ldots, t \tag{B.21}$$

$$H_t^s \geq (I_t^s + \sum_{j=1}^{t}d_{jt}P_{jt}^s)\sum_{k=1}^{i}p_k - \sum_{j=1}^{t}(\sum_{k=1}^{i}p_kE[d_{jt}|\Omega_i] - e_W)P_{jt}^s, \quad i = 1, \ldots, W \tag{B.22}$$

$$B_t^s \geq -I_t^s + (I_t^s + \sum_{j=1}^{t}d_{jt}P_{jt}^s)\sum_{k=1}^{i}p_k - \sum_{j=1}^{t}(\sum_{k=1}^{i}p_kE[d_{jt}|\Omega_i] - e_W)P_{jt}^s,$$
$$i = 1, \ldots, W \tag{B.23}$$

$$P_{jt}^s \in \{0, 1\}, j = 1, \ldots, t \tag{B.24}$$

$$\delta_t^s \in \{0, 1\} \tag{B.25}$$

$$I_0^s \leq I_0^S \tag{B.26}$$

$$G_1^s(I_0^s) = C_1^S(I_0^s) + c \cdot (I_0^S - I_0^s) \tag{B.27}$$

Appendix C. Test bed

Periodic demands with different demand patterns under the eight period computational study are displayed in Table C.6. The demand of each period under the twenty-five periods numerical example is shown in Table C.7. The first column represents period indexes; the rest columns denote various demands.

Table C.6
Demand data of the 8-period computational analysis.

Period	LCY1	LCY2	SIN1	SIN2	STA	RAND	EMP1	EMP2	EMP3	EMP4
1	15	3	15	12	10	2	5	4	11	18
2	16	6	4	7	10	4	15	23	14	6
3	15	7	4	7	10	7	26	28	7	22
4	14	11	10	10	10	3	44	50	11	22
5	11	14	18	13	10	10	24	39	16	51
6	7	15	4	7	10	10	15	26	31	54
7	6	16	4	7	10	3	22	19	11	22
8	3	15	10	12	10	3	10	32	48	21

Table C.7
Demand data of the 25-period computational analysis.

Period	LCY1	LCY2	SIN1	SIN2	STA	RAND	EMP1	EMP2	EMP3	EMP4
1	11	23	130	122	100	178	2	47	44	49
2	17	32	150	130	100	178	51	81	116	188
3	26	42	127	120	100	136	152	236	264	64
4	38	55	76	98	100	211	467	394	144	279
5	53	70	27	77	100	119	268	164	146	453
6	71	86	10	70	100	165	489	287	198	224
7	92	103	36	81	100	47	446	508	74	223
8	115	120	88	103	100	100	248	391	183	517
9	138	136	136	124	100	62	281	754	204	291
10	159	150	149	130	100	31	363	694	114	547
11	175	161	121	118	100	43	155	261	165	646
12	186	168	68	95	100	199	293	195	318	224
13	190	170	22	75	100	172	220	320	119	215
14	186	168	11	71	100	96	93	111	482	440
15	175	161	42	84	100	69	107	191	534	116
16	159	150	96	107	100	8	234	160	136	185
17	138	136	140	126	100	29	124	55	260	211
18	115	120	148	129	100	135	184	84	299	26
19	92	103	114	115	100	97	223	58	76	55
20	71	86	60	91	100	70	101	0	218	0
21	53	70	18	73	100	248	123	0	323	0
22	38	55	14	72	100	57	99	0	102	0
23	26	42	50	87	100	11	31	0	174	0
24	17	32	104	110	100	94	82	0	284	0
25	11	23	144	127	100	13	0	0	0	0

Table D.8
Model accuracy of the 8-period numerical experiments.

Modelling methods	MP	BS
Demand pattern		
LCY1	1.52	0.66
LCY2	7.47	3.42
SIN1	0.99	0.37
SIN2	0.84	0.30
STA	1.25	0.66
RAND	4.57	2.10
EMP1	8.75	4.50
EMP2	6.82	3.05
EMP3	1.83	0.81
EMP4	2.59	0.73
Fixed ordering cost		
200	3.14	1.36
300	3.71	1.66
400	4.15	1.96
Proportional ordering cost		
0	4.00	0.58
1	3.33	4.72
Penalty cost		
5	5.29	2.47
10	3.27	1.33
20	2.42	1.18
Coefficient of variation		
0.1	2.94	1.33
0.2	3.74	1.60
0.3	4.31	2.05
Average gap	3.66	1.66

Table D.9
Accuracy of the 25-period numerical experiments.

Modelling method	BS
Demand pattern	
LCY1	2.32
LCY2	2.97
SIN1	2.65
SIN2	2.50
STA	1.90
RAND	2.81
EMP1	4.15
EMP2	5.19
EMP3	3.79
EMP4	5.55
Fixed ordering cost	
500	3.27
1000	3.46
1500	3.42
Proportional ordering cost	
0	3.52
1	3.24
Penalty cost	
5	2.56
10	3.23
20	4.34
Coefficient of variation	
0.1	1.68
0.2	3.13
0.3	5.34
Average gap	3.38

Appendix D. Model accuracy

We employ the index of model accuracy ($= \frac{|\text{model result} - \text{simulation result}|}{\text{simulation result}} \times 100$) to evaluate the cost measure. We report the model accuracy of the 8-period numerical experiment in Table. D.8, and the 25-period numerical experiment in Table. D.9.

References

Archibald, B., & Silver, E. (1978). (s, S) policies under continuous review and discrete compound poisson demand. *Management Science, 24*(9), 899–909. doi:10.1287/mnsc.24.9.899.

Askin, R. (1981). A procedure for production lot sizing with probabilistic dynamic demand. *AIIE Transactions, 13*(2), 132–137. doi:10.1080/05695558108974545.

Bellman, R. (1957). *Dynamic programming* p. 92. Princeton University Press.

Belotti, P., Bonami, P., Fischetti, M., Lodi, A., Monaci, M., Nogales-Gómez, A., & Salvagnin, D. (2016). On handling indicator constraints in mixed integer programming. *Computational Optimization and Applications, 65*(3), 545–566. doi:10.1007/s10589-016-9847-8.

Bollapragada, S., & Morton, T. (1999). A simple heuristic for computing nonstationary (s, S) policies. *Operations Research, 47*(4), 576–584. doi:10.1287/opre.47.4.576.

Bookbinder, J., & Tan, J. (1988). Strategies for the probabilistic lot-sizing problem with service-level constraints. *Management Science, 34*(9), 1096–1108. doi:10.1287/mnsc.34.9.1096.

Chen, F., & Song, J. (2001). Optimal policies for multiechelon inventory problems with Markov-modulated demand. *Operations Research, 49*(2), 226–234. doi:10.1287/opre.49.2.226.13528.

Dural-Selcuk, G., Kilic, O., Tarim, S., & Rossi, R. (2016). A comparison of non-stationary stochastic lot-sizing strategies. arXiv:1607.08896.

Federgruen, A., & Zipkin, P. (1984). An efficient algorithm for computing optimal (s, S) policies. *Operations research, 32*(6), 1268–1285. doi:10.1287/opre.32.6.1268.

Feng, Y., & Xiao, B. (2000). A new algorithm for computing optimal (s, S) policies in a stochastic single item/location inventory system. *IIE Transactions, 32*(11), 1081–1090. doi:10.1080/07408170008967463.

Graves, S. (1999). A single-item inventory model for a nonstationary demand process. *Manufacturing & Service Operations Management, 1*(1), 50–61. doi:10.1287/msom.1.1.50.

Hu, J., Zhang, C., & Zhu, C. (2016). (s, S) inventory systems with correlated demands. *INFORMS Journal on Computing, 28*(4), 603–611. doi:10.1287/ijoc.2016.0698.

IBM (2011). IBM ILOG CPLEX optimization studio OPL language reference manual, IBM.

Iglehart, D. (1963). Optimality of (s, S) policies in the infinite horizon dynamic inventory problem. *Management Science, 9*(2), 259–267. doi:10.1287/mnsc.9.2.259.

Johnson, G., & Thompson, H. (1975). Optimality of myopic inventory policies for certain dependent demand processes. *Management Science, 21*(11), 1303–1307. doi:10.1287/mnsc.21.11.1303.

Lim, Y. F., & Wang, C. (2017). Inventory management based on target-oriented robust optimization. *Management Science, 63*(12), 4409–4427. doi:10.1287/mnsc.2016.2565.

Qiu, R., Sun, M., & Lim, Y. F. (2017). Optimizing (s, S) policies for multi-period inventory models with demand distribution uncertainty: Robust dynamic programing approaches. *European Journal of Operational Research, 261*(3), 880–892. doi:10.1016/j.ejor.2017.02.027.

Rossi, R., Kilic, O., & Tarim, S. (2015). Piecewise linear approximations for the static–dynamic uncertainty strategy in stochastic lot-sizing. *Omega, 50*, 126–140. doi:10.1016/j.omega.2014.08.003.

Rossi, R., Tarim, S., Prestwich, S., & Hnich, B. (2014). Piecewise linear lower and upper bounds for the standard normal first order loss function. *Applied Mathematics and Computation, 231*, 489–502. doi:10.1016/j.amc.2014.01.019.

Sahin, I. (1982). On the objective function behavior in (s, S) inventory models. *Operations Research, 30*(4), 709–724. doi:10.1287/opre.30.4.709.

Scarf, H. (1960). Optimality of (s, S) policies in the dynamic inventory problem. In *Mathematical methods in the social science* (pp. 196–202). Stanford University Press.

Sethi, S., & Cheng, F. (1997). Optimality of (s, S) policies in inventory models with Markovian demand. *Operations Research, 45*(6), 931–939. doi:10.1287/opre.45.6.931.

Stidham, S. (1977). Cost models for stochastic clearing systems. *Operations Research, 25*(1), 100–127. doi:10.1287/opre.25.1.100.

Strijbosch, L., Syntetos, A., Boylan, J., & Janssen, E. (2011). On the interaction between forecasting and stock control: The case of non-stationary demand. *International Journal of Production Economics, 133*(1), 470–480. doi:10.1016/j.ijpe.2009.10.032.

Tarim, S., & Kingsman, B. (2004). The stochastic dynamic production/inventory lot-sizing problem with service-level constraints. *International Journal of Production Economics, 88*(1), 105–119. doi:10.1016/s0925-5273(03)00182-8.

Tarim, S., & Kingsman, B. (2006). Modelling and computing (R^n, S^n) policies for inventory systems with non-stationary stochastic demand. *European Journal of Operational Research, 174*(1), 581–599. doi:10.1016/j.ejor.2005.01.053.

Tunc, H., Kilic, O., Tarim, S., & Eksioglu, B. (2013). A simple approach for assessing the cost of system nervousness. *International Journal of Production Economics, 141*(2), 619–625. doi:10.1016/j.ijpe.2012.09.022.

Veinott, J., Arthur, F., & Wagner, H. (1965). Computing optimal (s, S) inventory policies. *Management Science, 11*(5), 525–552. doi:10.1287/mnsc.11.5.525.

Zheng, Y., & Federgruen, A. (1991). Finding optimal (s, S) policies is about as simple as evaluating a single policy. *Operations Research, 39*(4), 654–665. doi:10.1287/opre.39.4.654.