

INVENTORY MANAGEMENT THROUGH ONE STEP AHEAD OPTIMAL CONTROL BASED ON LINEAR PROGRAMMING

LUIZ CARLOS ROTH*, AMIT BHAYA AND EUGENIUS KASZKUREWICZ

Abstract. A one step ahead optimal strategy is proposed for the inventory control and management problem, and rewritten as a linear programming problem, permitting practical implementation. Important novel aspects of the proposed solution are that it uses economic value added (EVA), a comprehensive performance index commonly used in business management, instead of regulation to a set point or to a interval of stock values; it does not require knowledge or prediction of the demand distribution; it achieves good efficiency with respect to a globally optimal value, defined in this paper, and no significant bullwhip effect, while being robust to demand and lead time variations. The proposed one step ahead optimal controller is compared with the classical (s, S) controller, as well as with a representative of the inventory and order-based production controller family. In order to make a fair comparison, this paper also proposes a tuning method for the latter two controllers. Numerical experiments based on average performance of the three controllers for a set of normally distributed demands show the superiority of the proposed one step ahead optimal controller, in terms of EVA as well as in terms of other measures proposed in the paper.

Mathematics Subject Classification. 49N05, 90B05, 90C11.

Received April 17, 2022. Accepted November 28, 2022.

1. INTRODUCTION

This paper considers a model of a single divisible good inventory management problem with an economic value added (EVA) performance index, inspired by a similar model presented in Page 179–187 of [14]. A one step ahead optimal (OSAO) strategy, for the EVA index, is proposed for this problem and it is shown that it can be rewritten as a linear programming problem, which permits practical implementation. The interesting novel feature of the proposed OSAO strategy is that it does not need knowledge of the demand distribution and neither does it require demand forecasting, which is usually the case [8, 27, 29]. The related problem of finding the optimal ordering sequence, that maximizes EVA, *under the hypothesis that the entire demand is known over a given simulation horizon*, can also be recast as a linear programming problem. The solution to this hypothetical optimal control problem in which the demand is known for all time is referred to in this paper as the *omniscient solution* and its main purpose is to serve as a benchmark. It should, of course, be kept in mind that the benchmark value of optimal omniscient performance index is unattainable by any implementable controller, since the omniscient solution is computed under the assumption that future demand is known. Thus

Keywords. Inventory management, lead time variation, optimal control, linear programming, robustness.

Electrical Engineering Department, Federal University of Rio de Janeiro, Rio de Janeiro, RJ, Brazil.

*Corresponding author: luiz.roth@coppe.ufrj.br

the value of the optimal omniscient index should be thought of as an upper bound or benchmark for the value of the performance index attainable by any implementable control policy. In addition, the efficiency and robustness of the proposed one step ahead optimal controller will be compared to commonly employed control techniques in supply chain management (SCM), namely the (s, S) policy introduced by Arrow *et al.* [1] and a generalized version of the APIOBPCS controller presented in [29], which is chosen as a representative of the class of Inventory and Order-based Production Control Systems (IOBPCS) introduced by Simon *et al.* [21]. We use a control engineering definition of robustness. “A system is robust when the system has acceptable changes in performance due to model or parameter changes and moderate modeling errors [6]. So, a robust supply chain should be designed to function properly even in the presence of uncertain parameters (for instance, the lead-time)” [22]. There is a vast literature on inventory control and management and a review is given in the following section, focusing only on the papers judged to be most relevant, in the interests of brevity. This paper also presents a novel methodology to make a fair comparison of robustness between the proposed controller, which is parameter-free, with classical controllers, which depend on certain tunable parameters. This methodology is inspired by the training set-testing set paradigm common in machine learning. The training set may be composed of recorded past demands or, if these are unavailable, of randomly generated demands from some hypothesized distribution (usually normal) and serves to optimize the tunable parameters of the classical controllers at some central point of a grid in the mean-standard deviation demand space. Performance of all controllers is then compared at grid points within a neighborhood of the central point, and the resulting surface portrays the robustness of each controller. The parameter tuning problem is nonconvex, thus the optimization involves using black box or pattern search methods. In a different context, the Simultaneous Perturbation Stochastic Algorithm (SPSA) is used to optimize model predictive controllers in [19] and a Particle Swarm Optimization (PSO) algorithm is used to tune the parameters of an inventory controller in [27].

1.1. Literature review

Ivanov *et al.* [9] identify one of the main contributions of control theory as being the application of dynamic feedback control to production-inventory systems and point out that a wide range of control-theoretic tools have been used in this context, ranging from the classical PID control to model predictive control. The paper by Lin *et al.* [12] also surveys control-theoretic approaches to the inventory control problem.

An early idea for inventory control, that is still commonly used, was proposed in [1] and, roughly speaking, consists of monitoring the inventory level and replenishing the stock, whenever it falls below a certain level. Since this control, usually referred to as (s, S) control, uses the stock or inventory level itself, it can be thought of as feedback control, based on the threshold levels s and S , as explained in Section 4.2.

Most control-theoretic methods are based on linear time-invariant mathematical models of a single echelon supply chain, using either transfer function or state space methods. This permits the use of classical ideas such as set point control, in which targets are set for certain variables, such as inventory level and work in progress. Many of these methods can be interpreted as parametrized versions of standard algorithms from the SCM literature, such as order-up-to-inventory control. The acronym IOBPCS, which stands for inventory and order based production control system identifies a popular class of such controllers, with subclasses denoted by prefixes, such as AP (for automatic pipeline) and MP (for matched parameter). A major focus of papers using this paradigm is to study the effect of the controller on the so called bullwhip effect (= amplification of system variables due to demand uncertainties) and on stability with regard to step changes in the demand: see, for example, [8, 11, 13] and the references therein.

Another approach that has been proposed more recently is based on model predictive control (MPC), or, if a performance index based on economic objective is specified, the terminology becomes economic MPC. A model of the plant is used, together with available demand data, to predict future demand and use the predictions to find an ordering sequence that optimizes the specified performance index. Finally, one or more of the computed optimal controls is applied, before a new predict-optimize-control sequence is embarked upon. The reader is referred to [17, 26, 31] and the references therein, for examples of MPC and economic MPC.

This paper uses an approach that is related to the MPC approach; however, it uses only past values of the demand and system state, does not use a predictor, and uses a stage cost that is economic, which is the so called Economic Value Added (EVA) performance index, commonly used in the business and financial world [14]. A slight extension of the greedy control idea [10, 18] is necessary for the inventory control problem on account of the lead time or delay present in the system. The proposed OSAOC (One Step Ahead Optimal Control) approach is to optimize EVA using past demands, for just one step ahead of the current time, apply this optimal control to the system, update its state and repeat the cycle of one step ahead optimization of EVA, thus generating a real-time feedback control. The main differences between the approach of this paper and existing results are as follows: (i) an economic stage cost function is used, similar to the one in [26], but also including discounting of the profit to its present value and not including any target or tracking costs, (ii) using only past demands to compute the present value of the control, in contrast with [17, 31] and other papers using the MPC approach which use more elaborate demand predictors, (iii) a study of the robustness of the proposed controller, with respect to uncertainty in demand as well as lead time uncertainty. This is another novel contribution of this paper, since very few papers on this topic discuss the question of robustness of control to uncertainties [28, 32]; in particular, time-varying lead time uncertainty. The result is a control that is implementable in real time, computationally cheap and, as will be shown, efficient when compared with the globally optimal omniscient solution, which will be defined in the sequel. Comparisons are also made with a classical (s, S) controller and with a performance index, different from EVA, that is also used in the control literature, and are favorable to the proposed one step ahead controller.

This paper is organized as follows. Section 2 introduces the basic notation, the discrete-time dynamical system model of the single echelon supply chain used as the main example in the paper and the economic value added as the objective function, to be maximized, showing how to formulate the omniscient optimal control problem as a linear programming problem. Section 3 shows how to transit from the omniscient optimal control problem to an OSAOC problem, also linear. Section 4 makes a brief introduction of the APIOBPCS controller and the (s, S) policy. Section 5 starts with a simple example of a demand with two step changes in order to exhibit the differences between the omniscient and one step ahead controllers, as well as the classical APIOBPCS and (s, S) controllers. A data driven optimization based methodology that uses training and test sets of normally distributed demands in order to tune the APIOPBCS and (s, S) controllers is proposed and is another novel contribution of this paper. The tuned APIOPBCS and (s, S) controllers are then compared with the proposed OSAO controller. Section 6 showcases the greater robustness of the OSAO controller to demand uncertainty while Section 7 expands the single echelon supply chain model to consider variable lead time and also highlights the greater robustness of the OSAO controller to delay uncertainty. Section 8 contains concluding remarks and suggestions for future work.

2. A SINGLE ECHELON SUPPLY CHAIN MODEL

This section presents a single echelon supply chain (SESC) model in order to define the inventory control problem precisely. The model uses the notation in [4] and is based on [14], which refers to the SESC as a store supply chain. However, the model presented in this section is quite general and represents any single echelon supply chain. This model essentially refers to a single source shipping a single divisible good, with a certain delay or lead time, to some location, which shall be referred to generically as a warehouse, in which there is a stock (inventory). Sales of this good, in response to a demand, are assumed to occur at the warehouse.

A single echelon supply chain thus has two main components: a delay between ordering and receiving, corresponding to a block named conveyor or pipeline and an accumulator, which receives the ordered good after the stipulated delay, corresponding to the warehouse (Fig. 1). It is assumed that there are capacity constraints, both on the warehouse and on shipping. Selling of the good occurs in accordance with the demand and it is assumed that all but the quantity w of the good (the amount in the safety stock) can be sold. Note that since it is assumed that the good is divisible, it is described by a real-valued variable, instead of an integer-valued variable. This is a reasonable approximation when the quantity of the good is large. The manager's objective is to choose

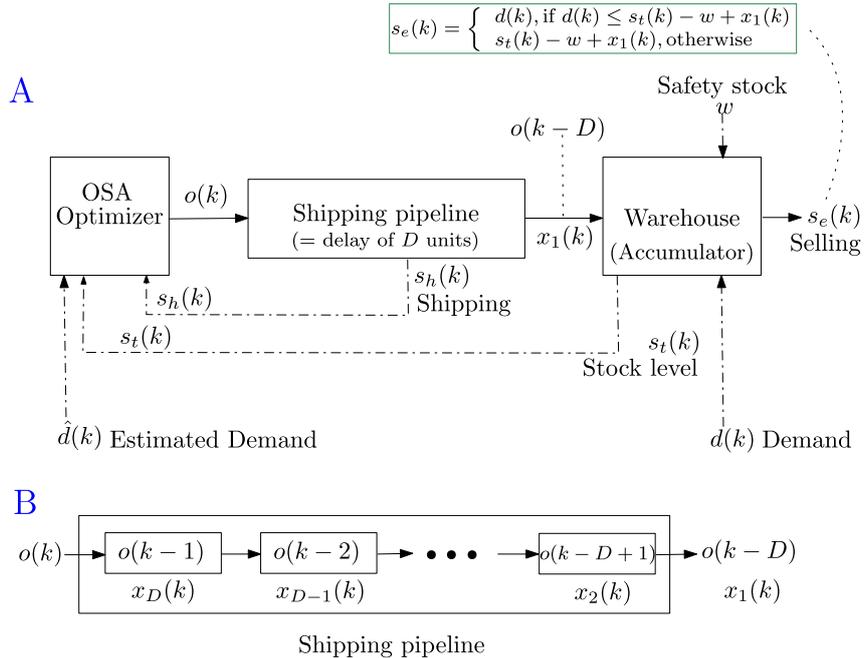


FIGURE 1. A: Block diagram of single echelon supply chain to define the inventory control problem, together with the one step ahead optimizer, showing how the overall scheme functions as a feedback controller. Continuous lines with arrows denote the flow of the good through the system, while dash-dotted lines show the flow of information regarding system variables. The optimizer finds the ordering that maximizes the economic value added, $L(\cdot)$, in accordance with (35) (one step ahead case) or (32) (omniscient case). B: The shipping pipeline in detail: it can be thought of as a conveyor, causing a delay of D units between ordering a certain quantity of the good and receiving it at the warehouse.

an ordering sequence, in response to the demand, that maximizes cumulative profit over a given time horizon, assuming that there are costs and constraints associated to handling, shipping, storage and stockouts. It is also assumed that there is no backordering; thus stockouts imply that customer demand is left unfulfilled, and the negative effect of this on sales is modeled by a high cost for lost sales. Finally, profits (= sales revenues minus costs) are discounted using a fixed interest rate and then added over the given horizon to obtain the cumulative profit, referred to as Economic Value Added (EVA) [24]. The notation used to formulate the mathematical model is summarized in Table 1.

The delay between ordering and receiving is written by introducing a string of D unit delays, which can be thought as a conveyor, as follows:

$$x_i(k+1) = x_{i+1}(k), \quad i = 1, \dots, D-1 \quad (1)$$

$$x_D(k+1) = o(k). \quad (2)$$

Thus the quantity of material in transit (*i.e.*, being shipped) at time k , denoted by $s_h(k)$, can be written as:

$$s_h(k) = \sum_{i=2}^D x_i(k), \quad (3)$$

TABLE 1. Notation adopted to formulate the mathematical model.

Symbol	Meaning
k	discrete time counter.
$x_i(k)$	quantity of the good in the i th stage of shipping, $i = 1, \dots, D$, where $i = 1$ corresponds to receiving at time k .
D	delay between receiving and ordering.
$s_t(k)$	inventory (stock) level of warehouse at discrete time instant k .
$d(k)$	customer demand at warehouse at time k .
$s_e(k)$	sales (selling) at time k .
p	price per unit of material sold.
w	quantity of the good in the safety stock.
$c_{os}(k)$	out-of-stock cost at time k .
$\alpha_{os}(k)$	out-of-stock cost per unit of demand not met at time k .
$c_h(k)$	handling cost at time k .
α_h	handling cost per unit of material handled.
$c_{st}(k)$	cost of maintaining stock level in warehouse at time k .
α_{st}	storage cost per unit of material stored.
$c_{sh}(k)$	shipping cost of material in transit.
α_{sh}	shipping cost per unit of material being shipped.
$c_t(k)$	total cost at time k .
$d_p(k)$	discounted profit at time k .
ρ	discount rate.
$L(k)$	cumulative profit at time k .
K_f	simulation horizon over which cumulative profit is to be maximized.
$s_{h,\max}$	maximum shipping capacity.
$s_{t,\max}$	maximum warehouse capacity.

while the quantity of material being received at the warehouse is $x_1(k)$, which is no longer considered as material being shipped, and corresponds to an order placed D time units in the past, so that ordering, denoted $o(k)$ is just $x_{D+1}(k)$.

Evolution of the stock level in the warehouse at time k is determined by the quantity of material being received ($x_1(k)$) plus the quantity currently in the warehouse $s_t(k)$ (setting aside a safety stock w not to be sold) less the quantity sold ($s_e(k)$) in response to the demand. The corresponding equation is as follows.

$$s_t(k+1) = s_t(k) + x_1(k) - s_e(k). \quad (4)$$

In order to describe the sales $s_e(k)$ at time k , observe that sales can only occur if the inventory level $s_t(k)$ plus the received material $x_1(k)$ less the safety stock w is greater than the demand $d(k)$. Thus sales can be written as follows:

$$s_e(k) = d(k) + \min\{0, s_t(k) - w + x_1(k) - d(k)\}. \quad (5)$$

Thus the quantity of sales matches the demand $d(k)$ if it is less than the effective available inventory level $s_t(k) - w + x_1(k)$, otherwise it is just equal to the latter.

The costs are expressed as follows:

$$c_o(k) = \alpha_{os}(d(k) - s_e(k)) \quad (\text{out of stock cost}) \quad (6)$$

$$c_{st}(k) = \alpha_{st}s_t(k) \quad (\text{storage cost}) \quad (7)$$

$$c_h(k) = \alpha_h(x_1(k) + o(k)) \quad (\text{handling cost}) \quad (8)$$

$$c_{sh}(k) = \alpha_{sh} \sum_{i=2}^D x_i(k) \quad (\text{shipping cost}) \quad (9)$$

$$c_t(k) = c_o(k) + c_{st}(k) + c_h(k) + c_{sh}(k) \quad (\text{total cost}). \quad (10)$$

Discounted profit, also referred to as economic value added (EVA), is the revenue from sales less the total cost (all at time k), appropriately discounted:

$$d_p(k) = (ps_e(k) - c_t(k))e^{-\rho(k-D)}. \quad (11)$$

The cumulative profit $L(k)$, also called cumulative EVA, accumulates the discounted profit:

$$L(k+1) = L(k) + d_p(k). \quad (12)$$

The objective function to be maximized is the cumulative profit at the end of the time horizon and thus, given the demand $d(k)$, $k = D, \dots, K_f - 1$ over the whole simulation horizon, the optimization problem to be solved is in fact an optimal control problem, which will be referred to as the omniscient optimal control problem (since the demand over the entire simulation horizon is assumed to be known), and can be written as follows:

$$\begin{aligned} & \text{maximize} && L(K_f) \\ & \{o(k)\}_{k=D}^{K_f-1}. \end{aligned} \quad (13)$$

In optimal control terminology, this is the Mayer problem [20] of choosing an optimal ordering sequence that maximizes the final cumulative profit $L(K_f)$. More realistically, the following constraints are also needed:

$$o(k) \geq 0, \quad k = 0, \dots, K_f - 1 \quad (14)$$

$$s_h(k) \leq s_{h,\max}, \quad k = 0, \dots, K_f - 1 \quad (15)$$

$$s_t(k) \leq s_{t,\max}, \quad k = 0, \dots, K_f - 1. \quad (16)$$

Observe that equations (1) through (12) are all linear with the exception of (5). Since (5) is piecewise linear, it will now be shown how to write the optimal control problem described in (13), as a linear program. The function \min can be written equivalently as:

$$\min\{0, y\} = \frac{1}{2}(y - |y|). \quad (17)$$

The variable y can be written as the difference of two nonnegative functions y_+ and y_- so that:

$$y = y_+ - y_- \quad (18)$$

$$|y| = y_+ + y_- \quad (19)$$

$$y_+ y_- = 0 \quad (20)$$

$$\min\{0, y\} = \frac{1}{2}(y - |y|) = -y_-. \quad (21)$$

Denoting the effective available inventory level less the demand as $y(k)$:

$$y(k) = s_t(k) - w + x_1(k) - d(k) \quad (22)$$

and using (21), equation (5) is rewritten as follows:

$$s_e(k) = d(k) + \min\{0, y(k)\} \quad (23)$$

$$= d(k) - y_-(k) \quad (24)$$

where $y_-(k)$ is defined from (22) *i.e.*,

$$y_+(k) - y_-(k) = s_t(k) - w + x_1(k) - d(k) \quad (25)$$

which is now recognizably a linear equation involving the additional nonnegative variables:

$$y_+(k), y_-(k) \geq 0. \quad (26)$$

In order to ensure that equation (20) is satisfied, this quadratic constraint is modeled as the two mutually exclusive logical statements:

$$\text{if } y_+(k) > 0, \text{ then } y_-(k) = 0 \quad (27)$$

$$\text{if } y_-(k) > 0, \text{ then } y_+(k) = 0. \quad (28)$$

Let M be any upper bound for y_+ , y_- . Then, it is easy to check that statements (27) and (28) are equivalent to the following linear constraints:

$$y_+(k) \leq M\delta_1(k) \quad (29)$$

$$y_-(k) \leq M\delta_2(k) \quad (30)$$

$$\delta_1(k) + \delta_2(k) = 1 \quad (31)$$

where δ_1 and δ_2 are binary variables.

The omniscient optimal control problem (13) can now be transcribed, *cf.* [5], as the following MILP problem:

$$\begin{aligned} & \text{maximize} && L(K_f) \\ & \{o(k)\}_{k=D}^{K_f-1} \\ & \text{subject to} && (1),(2),(4),(12),(24),(25) \quad \forall k \in [D, K_f - 1], \\ & \text{and } (14),(15),(16),(26) && \quad \forall k \in [D, K_f - 1] \\ & \text{and } (29),(30),(31) && \quad \forall k \in [D, K_f - 1] \end{aligned} \quad (32)$$

where equations (1), (2), (4), (12), (24) and (25) are equality constraints that describe the SESC dynamics. Equations (14) and (26) are nonnegativity constraints. Equations (15) and (16) are capacity constraints on shipping and inventory. Lastly, equations (29), (30) and (31) are constraints that involve the binary variables.

For future reference, the state vector of the single echelon supply chain at instant k is denoted as $\mathbf{z}(k)$ and defined as:

$$\mathbf{z}(k) = [x_1(k), \dots, x_D(k), s_t(k)]. \quad (33)$$

From (1), (2) it follows that the state vector $\mathbf{z}(k)$ can be written equivalently as:

$$\mathbf{z}(k) = [o(k-D), o(k-D+1), \dots, o(k-1), s_t(k)]. \quad (34)$$

Remark 2.1. It should be noted that the omniscient optimal control problem has $(D+3)(K_f-D)$ decision variables *i.e.*, (the $D+1$ variables in the state vector $\mathbf{z}(k)$ together with $y_+(k)$ and $y_-(k)$) times the length (K_f-D) of the simulation horizon. There are $4(K_f-D)$ inequality constraints due to maximum shipping, warehouse capacities and upper bounds of $y_+(k)$, $y_-(k)$. There are $3(K_f-D)$ nonnegativity constraints and $(D+4)(K_f-D)$ equality constraints that describe the SESC dynamics plus the (K_f-D) equality constraints that ensure $\delta_1(k)$ and $\delta_2(k)$ are mutually exclusive.

Remark 2.2. It was noticed for all numerical examples presented in this work that omitting constraint (20) and, thus, transcribing the omniscient optimal control problem into a simpler LP problem yielded exactly the same results obtained from (32). Furthermore, constraint (20) was verified to be true for all k instants despite not being explicitly enforced by the LP problem. This was also the case for the OSAOC problem discussed in Section 3.

Remark 2.3. The omniscient cumulative EVA, even though it is unattainable, serves as a global optimum and as a reference against which the cumulative EVA achieved by causal ordering sequences, obtained by implementable controllers, can be evaluated. The omniscient optimal solution requires knowledge of future peaks and troughs in the demand and uses it, when required, to make anticipative (*i.e.*, noncausal) ordering, ensuring that cumulative profit is maximized. Such noncausal behavior is not possible for any real world controller, which must necessarily be causal or nonanticipative, since it cannot have access to future demands.

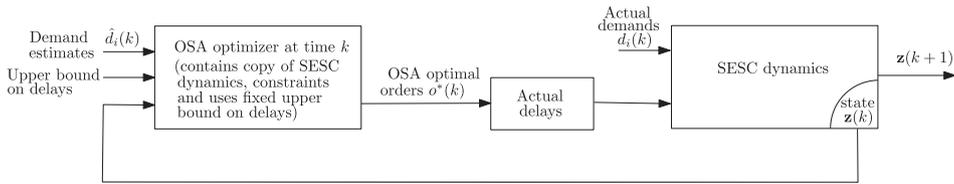


FIGURE 2. The one step ahead optimal control scheme in the case where the actual delays and demands differ from the values used in the one step ahead optimizer. In Section 7, the actual delays are chosen from a random distribution, while the one step ahead optimizer uses upper bounds. Similarly, the actual demands are used to update the supply chain state $\mathbf{z}(k)$, defined in (34), while the one step ahead optimizer uses only past demands and estimates the current demand $\hat{d}(k) = d(k-1)$.

3. ONE STEP AHEAD OPTIMAL CONTROL

In practice, of course, only past demands are known to the supply chain manager, who has to make ordering decisions based on this information and observation of the current variables, such as inventory level, amount of goods in the shipping pipeline, sales levels, etc. This section formulates the so called one step ahead optimal control (OSAOC) problem, that respects practical information constraints, and is shown schematically in Figure 2.

3.1. Linear programming formulation of one step ahead optimal control

Given the delay D between ordering and receiving, it is clear that, at time instant k , the value of the objective function $L(k+1)$ (the cumulative profit) at the next time instant $k+1$ is dictated by the ordering $o(k-D)$. This means that, since the past demands are known and we can estimate the current demand as $\hat{d}(k) = d(k-1)$, one can optimize over the backward horizon $\{k-D, k-D+1, \dots, k\}$ to obtain the optimal sequence $\{o^*(i)\}_{i=k-D}^k$, assuming the state $\mathbf{z}(k) = [x_1(k), \dots, x_D(k), s_t(k)]$ to be given and initializing $\mathbf{z}_0 = \mathbf{z}(k)$. The OSAOC problem just described is denoted $OSA(\mathbf{z}(k))$ and transcribed as the following MILP problem, the structure of which is inherited from the corresponding omniscient problem (32):

$$\begin{aligned}
 & \text{maximize} && L(k+1) \\
 & \text{subject to} && \{o(i)\}_{i=k-D}^k \\
 & \text{and} && (1), (2), (4), (12), (24), (25) \quad \forall i \in [k-D, k], \\
 & \text{and} && (14), (15), (16), (26) \quad \forall i \in [k-D, k], \\
 & \text{and} && (29), (30), (31) \quad \forall i \in [k-D, k] \\
 & \text{and} && \mathbf{z}_0 = \mathbf{z}(k) \text{ (specified initial conditions)}.
 \end{aligned} \tag{35}$$

Now, let the optimal control computed as the solution to (35) be denoted as follows:

$$[o^*(k-D), \dots, o^*(k)] := \operatorname{argmax}_{\{o(i)\}_{i=k-D}^k} L(k+1). \tag{36}$$

Although the entire vector $\{o^*(i)\}_{i=k}^{k+D}$ is determined in (36), examination of equations (1) through (12) and Figure 1-B shows that only the optimal ordering $o^*(k-D)$ affects the value of the objective function (profit) $L(k+1)$. Thus, the computed values $o^*(k-D+1), \dots, o^*(k)$ are discarded and the variables at these instants maintain their pre-optimization values.

As pointed out in (34), the state $\mathbf{z}(k)$ can also be expressed as $\mathbf{z}(k) = [o(k-D), \dots, o(k-1), s_t(k)]$. Thus, after the optimization step is carried out, the state is updated to $\mathbf{z}(k+1) = [o(k-D+1), \dots, o(k-1), o^*(k), s_t(k+1)]$, where $s_t(k+1)$ is obtained by evaluating (4). From this new initial condition, a new optimization step can be performed. With this notation in place, the iterative one step ahead control (OSAOC) scheme over the backward horizon can be written as shown (Algorithm 1).

Algorithm 1 One Step Ahead Optimal Control (OSAOC)

```

1: Initial state  $\mathbf{z}_0 := \mathbf{z}(k_0)$ ,  $k_0 = D$  given
2: while  $D \leq k \leq K_f - 1$  do
3:    $\mathbf{z}(k) := \mathbf{z}_0$ 
4:   solve  $OSA(\mathbf{z}(k))$  defined in (35)
5:   to get  $[o^*(k-D); \dots; o^*(k)]$ 
6:   Discard all computed optimal orderings except  $o^*(k-D)$ 
7:    $o_{\text{osao}}(k) \leftarrow o^*(k-D)$ 
8:   Evaluate RHS of (4) to obtain  $s_t(k+1)$ 
9:    $\mathbf{z}_0 \leftarrow [o(k-D+1), \dots, o(k-1), o_{\text{osao}}(k), s_t(k+1)]$ 
10:   $k \leftarrow k+1$ 
11: end while

```

Remark 3.1. Algorithm 1 relies only on past values of the demand and the current demand estimate $\hat{d}(k) = d(k-1)$: at each instant k , it solves a linear program with $(D+3)(D+1)$ decision variables, $(D+4)(D+1)$ equality constraints describing the SESC dynamics, $3(D+1)$ nonnegativity constraints, $4(D+1)$ upper bound constraints and $(D+1)$ equality constraints that ensure the binary variables are mutually exclusive. This is a linear program that is significantly smaller in terms of the number of decision variables and constraints than the corresponding omniscient optimal control problem (32), whenever the simulation horizon K_f is much larger than the lead time D . For example, assuming $D = 7$ there would be a total of 80 decision variables, 88 equality constraints describing the dynamics, 24 nonnegative constraints, 32 upper bound constraints and 8 equality constraints that ensure the binary variables are mutually exclusive. In contrast, for $K_f = 30$, the omniscient optimal control problem would have 230 decision variables, 253 equality constraints, 69 nonnegative constraints, 92 upper bound constraints and 23 equality constraints involving the binary variables. Thus, even though OSAOC has to be solved at each step, it is a computationally tractable method, which is desirable for real time inventory control. Furthermore, if ordering is made only at the end of the current day, then, of course, current demand $d(k)$ is known and the estimate $\hat{d}(k) = d(k-1)$ does not need to be used.

4. CLASSICAL INVENTORY CONTROL SCHEMES.

This section summarizes the classical control theory based APIOBPCS controller generalized in [29] and the well known (s, S) control policy [1], chosen as candidates for comparison with the proposed OSAOC controller. These two controllers are chosen because the first is a representative of a family of controllers much discussed and cited in the literature [21], while the second is widely used by practitioners, and commonly discussed in textbooks, *e.g.*, [15]. Another reason for the choice is that they are feedback controllers with roughly the same level of simplicity as the proposed controller, although, in contrast to it, the first does require demand forecasting as well as tuning of three controller parameters (T_p, T_i, T_w) and the second requires tuning of the low (s) and high (S) stock level parameters. It is crucial to note, however, that good or optimal parameter tuning for a given demand, will not, in general, be optimal for some other demand. Thus, for fair comparisons, a suitable approach to parameter tuning for the APIOBPCS and (s, S) controllers is required and one is proposed in Section 5.3.

4.1. The APIOBPCS controller

The ordering policy of the APIOBPCS controller Page 2888, Equation (6) of [29] is rewritten as follows (using the notation presented in Section 2):

$$o(k) = \hat{d}(k) + \frac{I_{ref}(k) - s_t(k)}{T_i} + \frac{T_p \hat{d}(k) - s_h(k)}{T_w} \quad (37)$$

where $\hat{d}(k)$ is the estimated demand. $I_{\text{ref}}(k)$ is the inventory level reference, the choice of which is based on some heuristic. Parameter T_i is the time constant associated to the adjustment of inventory level, parameter T_p is the lead time or shipping delay, and parameter T_w is the time constant associated to shipping level adjustment.

Remark 4.1. The APIOBPCS controller was originally designed for WIP (Work-In-Process) instead of pipeline inventory i.e., shipping. Note, however, that both WIP and shipping share similar dynamics, so there is no loss of generality in using shipping instead of WIP.

4.2. The (s, S) inventory controller

The classical (s, S) controller [1] is also chosen for purposes of comparison with the proposed one step ahead optimal control. This is an inventory control policy that is extremely simple to state, and has also proven to be practical and effective, in the seventy years since it was first proposed [2]. It involves the choice of two levels of stock, denoted by s and $S > s$ and proposes ordering to bring the inventory level back in the vicinity of S as soon as it drops below s , and not ordering when it exceeds s . This can be expressed as follows:

$$o(k) = \begin{cases} S - s_t(k) & \text{if } s_t(k) \leq s \\ 0 & \text{otherwise} \end{cases} \quad (38)$$

making it clear that it is a threshold based output feedback control policy (*i.e.*, based only on the component s_t of the entire state feedback vector \mathbf{z}), switching between 0 and the deficit between S (desired stock level) and the actual current stock level s_t , whenever the latter goes below the threshold value s . It is also obvious that, for a given demand, the performance of an (s, S) controller depends on the choice of the parameters s and S . This implies, in turn, that these parameters can be optimized for a given demand, but, of course, there is no guarantee that these values will remain optimal for a different demand. This is discussed further in Section 5.3, where comparisons between the different controllers are made.

5. NUMERICAL EXAMPLES OF OSAOC VERSUS CLASSICAL CONTROLLERS

A demand sequence is generated from a normal distribution, “by far the most popular distribution for making inventory management decisions” [16, 23]. In the first numerical example, this normally distributed demand is used to show the solution of the one step ahead optimal control. For the purpose of a fair comparison, both OSAO and APIOBPCS controllers use the same (and very simple) demand forecast $\hat{d}(k) = d(k-1)$.

Next, the parameters of the APIOBPCS and (s, S) controllers are optimized using a data-driven methodology. The previous numerical example is repeated using the optimized values obtained for these parameters.

In all cases, the cumulative profit L_{osao} obtained for the OSAOC is compared to the final value L_{omni} attained by the omniscient solution for the same demand sequence. Furthermore, these results are also compared to the cumulative profits L_{apiobpcs} and L_{ss} attained respectively by the APIOBPCS controller proposed in [21], derived from classical control theory, and the (s, S) controller proposed in [1].

The ratio between the cumulative profits L_x/L_{omni} is referred to as the *efficiency* η_x , where the subscript x specifies the control applied (*i.e.*, $x \in \{\text{apiobpcs}, \text{osao}, \text{ss}\}$).

The supply chain to be controlled is the single echelon plant described by the dynamics presented in Section 2. Ordering is constrained to be nonnegative, meaning that stock can not be returned to the supplier. The shipping delay D is taken to be 7 time units and K_f is chosen to be equal to 44 time units. All comparisons are carried out over the interval $[D, K_f - 1 - D]$ to exclude the effect of the final transient due to the turnpike effect [30] that occur for the omniscient solution *i.e.*, the comparison interval is [7, 36], which is equivalent to a period of 30 time units.

For all numerical examples, the pipeline was considered initially empty and the other parameters were set as indicated in Table 2. The values of parameters T_p , T_i , T_w (of the APIOBPCS controller), s and S (of the (s, S) controller) are specified in each related subsection since their values change for each numerical example.

TABLE 2. Parameters for the SESC inventory control example problem.

α_{os}	α_{st}	α_h	α_{sh}	p	ρ	$s_{h,max}$	$s_{t,max}$	w	$s_h(0)$	$s_t(0)$
20	5	10	5	100	0.005	100	50	1	0	30

TABLE 3. Cumulative profits and efficiencies.

Controller	Profit (L)	Efficiency (η_x)
APIOBPCS	7306.21	62.80%
OSAO	9968.63	85.69%
(s, S)	6728.56	57.84%
Omniscient solution	11633.35	100%

5.1. Software implementation details

The omniscient solution and the OSAO, APIOBPCS and (s, S) controllers were implemented in the scientific computing language Julia [3] using JuMP [7], an acronym for Julia Mathematical Programming, which is an algebraic modeling language for mathematical optimization embedded in Julia with a wrapper for the CLP (Coin-OR *linear programming*) and CBC (Coin-OR *branch and cut*) solvers. JuMP allows users to express a wide range of optimization problems (linear, mixed-integer, quadratic, conic-quadratic, semidefinite, and nonlinear) since it supports different types of solvers. It also defines variables, constraints and the objective using a simple syntax that closely resembles the mathematical formulation shown in Algorithm 1.

The optimization of the APIOBPCS and (s, S) tuning parameters discussed in Section 5.3 was obtained by MATLAB implementations of these two controllers that uses the Optimization Toolbox, which is set up to carry out black box optimization of a function, using interior-point, sequential quadratic programming (SQP) and active set methods, amongst others. The black box in question is the system model (1)–(5) of the SESC dynamics, and the `fmincon` command was used to optimize the tuning parameters, with the algorithm being chosen as interior point.

5.2. Illustrative example: single normally distributed demand

Experiments in this subsection are carried out with a demand d that has a normal distribution with $\mu = 10$ and $\sigma = 3$. This demand is generated by the Julia command `d = rand(Normal(10,3),K_f)` using the Distributions package. Note that for this distribution there is a 99.7% probability of $1 \leq d(k) \leq 19$ for $k = 0, \dots, K_f$. If, however, a demand with negative values is generated, it is discarded and replaced with a new one until a nonnegative demand sequence is generated. The OSAO controller uses the approach detailed by Algorithm 1. Both OSAO and APIOBPCS controllers use $\hat{d}(k) = d(k-1)$ as an estimate of the unknown current demand. For the APIOBPCS, the gains were set as: $T_i = T_w = T_p = D = 7$, which is based on the setting used by [29]. Lastly, the setpoint I_{ref} was chosen to be equal to 10, which is the average demand over the interval [7, 36]. For the (s, S) controller, the stock levels were set as $s = \mu - \sigma + w = 8$ and $S = \mu + \sigma + w = 14$, which are the deviation from the mean plus the safety stock, noting that this presupposes knowledge of the demand distribution used in the numerical experiment. Such prior knowledge is not required for the proposed OSAO controller.

The attained cumulative profits (12) and the respective efficiencies with regard to the global optimum (omniscient solution) for each controller are presented in Table 3. Figures 3–10 show plots of the important variables for each of the three controllers, in order to make a comparison. Further observations are made in Subsection 5.2.1

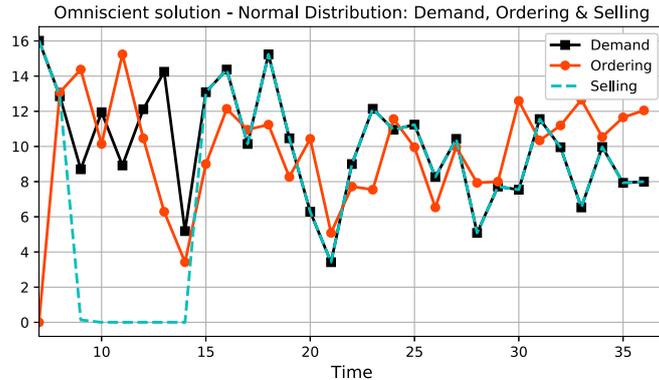


FIGURE 3. Ordering and selling for the omniscient solution with $D = 7$, assuming that demands from $k = 0$ to $k = 6$ are used as the initial past demands, to $k = (K_f - 1 - D) = 36$, in order to remove the turnpike effect, for a normally distributed demand. Thus, the simulated horizon is equivalent to a period of 30 time units.

5.2.1. Discussion of numerical example 1

Figure 3 depicts the ordering and selling for the omniscient solution. Since it assumes knowledge of the demand over the entire simulation horizon, it can anticipate the ordering of stock by D instants and overcome the potentially adverse effects of shipping delay. Note that the demand is not matched at instants $k = 9$ to $k = 14$ due to the initially empty pipeline, but, if the warehouse inventory level is sufficiently large, lost sales (Fig. 9) can be avoided.

Figures 4–6 show the ordering and selling responses for the APIOBPCS, OSAO and (s, S) policies respectively. As expected, all controllers exhibit the same pattern of lost sales due to initialization as the omniscient solution. The OSAO and (s, S) controllers also have lost sales after instant $k = 14$. The APIOBPCS was the only causal (nonanticipative) controller to avoid lost sales after instant $k = 14$, but this was achieved at the cost of a higher usage of the storage space and violation of both maximum warehouse ($s_{t,\max} = 50$) and shipping ($s_{h,\max} = 100$) capacities as can be seen in Figures 7 and 8. Unlike the anticipative noncausal omniscient solution, the causal controllers are not able to keep the inventory level at the minimum, which is the specified safety stock of $w = 1$, after the initial pipeline transient.

Figure 10 depicts the evolution of the cumulative profits. The efficiencies of the causal controllers, at the end of the simulation horizon, with respect to the global optimum (noncausal omniscient solution) are: $\eta_{\text{apiobps}}^L = 62.80\%$, $\eta_{\text{osao}}^L = 85.69\%$ and $\eta_{\text{ss}}^L = 57.84\%$.

5.3. Data driven methodology to optimize the APIOBPCS and (s, S) controllers

The tuning parameters of the APIOBPCS controller and the stock levels of the (s, S) controller chosen heuristically in Section 5.2 yielded results that were considerably inferior to the OSAO controller, raising the question of tuning or optimizing these parameters to improve their performance. Note that, in order to compare the controllers with regard to a specific normally distributed demand d_{spec} , it would not be fair to optimize the APIOBPCS and (s, S) controllers with respect to d_{spec} , because this amounts to giving these two controllers omniscient information (*i.e.*, the demand over the whole period is assumed to be known and is used for the parameter optimization), while the proposed OSAO controller does not use this omniscient information. A reasonable alternative is to optimize the parameters of the APIOBPCS and (s, S) controllers using a set of normally distributed demands similar to the demand d_{spec} . The specific demand d_{spec} , however, can not be included in this set. The following methodology is used. First, a training set, consisting of N randomly generated and normally distributed nonnegative demands with a given mean μ and standard deviation σ is generated.

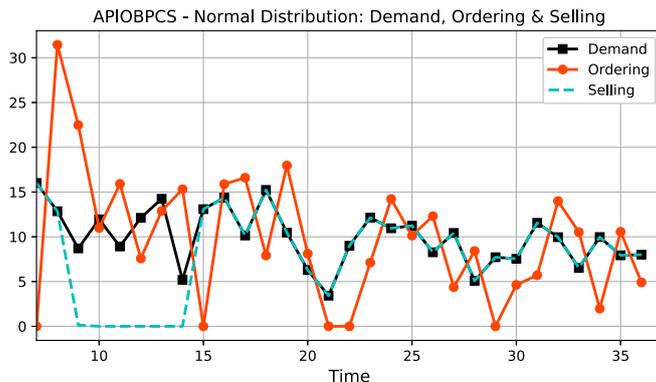


FIGURE 4. Ordering and selling for the APIOBPCS controller with $\hat{d} = d(k - 1)$ and $D = 7$, assuming that demands from $k = 0$ to $k = 6$ are used as the initial past demands, to $k = (K_f - 1 - D) = 36$ for a normally distributed demand. Note that selling tracks demand exactly after the initial transient *i.e.*, for $k \geq 15$.

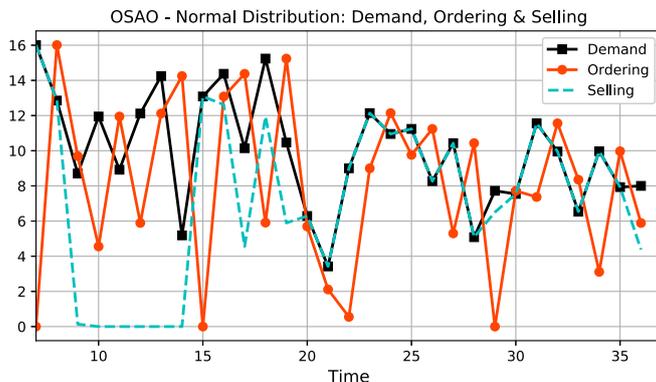


FIGURE 5. Ordering and selling for the OSAO controller with $D = 7$, assuming that demands from $k = 0$ to $k = 6$ are used as the initial past demands, to $k = (K_f - 1 - D) = 36$ for a normally distributed demand.

Evidently, if a dataset of real demands is known it should be used and the optimization is then data driven, with real data, as opposed to synthetic data. With the training set in hand, the parameters of the APIOBPCS and (s, S) controllers are optimized for the demands in the training set using the Matlab Optimization Toolbox and the solver `fmincon`, set to the default interior-point algorithm.

The problem of optimizing the parameter triple (T_p, T_i, T_w) of the APIOBPCS-based controller and the stock levels of the (s, S) controller is now formulated as a black box optimization problem. The black box, which contains the dynamics and constraints, receives either the triple (T_p, T_i, T_w) (when optimizing the APIOBPCS-based controller) or the pair (s, S) (when optimizing the (s, S) controller), in addition to a given demand sequence, and returns the cumulative profit L for this demand. The goal of the optimization is to find the solution (triple (T_p, T_i, T_w) or pair (s, S)) that maximizes the average cumulative profit L of the given demand. For the APIOBPCS controller, the optimization problem to be solved is unconstrained with respect to its parameters. However, given the definition of the (s, S) controller, which requires $S > s$, it is necessary to impose the inequality constraint:

$$s - S < 0. \tag{39}$$

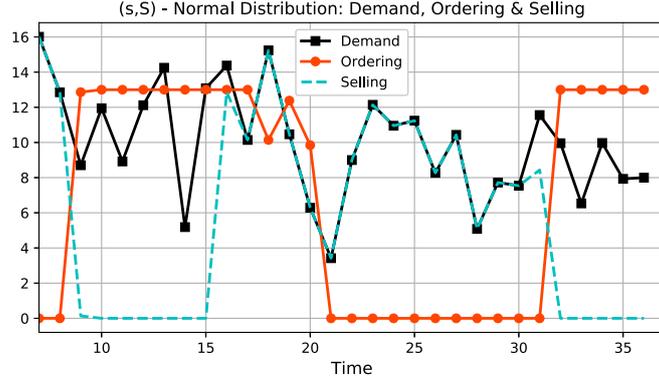


FIGURE 6. Ordering and selling for the (s, S) controller with $D = 7$, assuming that demands from $k = 0$ to $k = 6$ are used as the initial past demands, to $k = (K_f - 1 - D) = 36$ for a normally distributed demand.

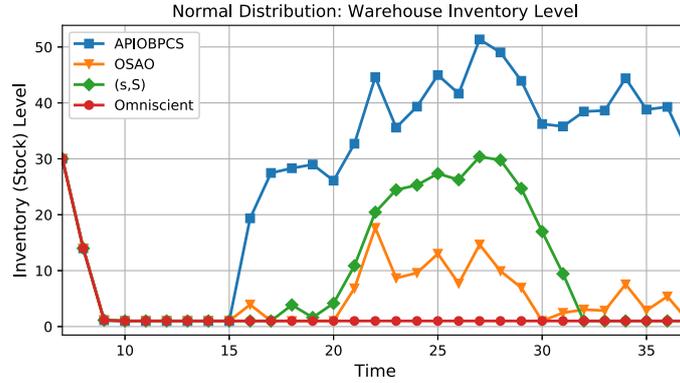


FIGURE 7. Inventory level for all controllers with $D = 7$, assuming that demands from $k = 0$ to $k = 6$ are used as the initial past demands, to $k = (K_f - D) = 37$ for a normally distributed demand. Note that the APIOBPCS controller violates the maximum warehouse capacity ($s_{t,\max} = 50$) at instant $k = 27$.

The capacity constraints on shipping and the warehouse are treated using a penalty function, P , that adds a cost to the cumulative functions L and C whenever $s_t(k) > s_{t,\max}$ or $s_h(k) > s_{h,\max}$ and is defined as follows:

$$P(k) = \omega_1 P_1(k) + \omega_2 P_2(k) \quad (40)$$

where

$$P_1(k) = s_t(k) - s_{t,\max} \quad (\text{store penalty}) \quad (41)$$

$$P_2(k) = s_h(k) - s_{h,\max} \quad (\text{shipping penalty}). \quad (42)$$

Even though it is possible to select the weights ω_1 and ω_2 to be large enough in order to find an optimal solution that generates feasible ordering sequences for all training demands, there are no guarantees that this will remain true for any demand with the same distribution, but outside the training set. Also note that this methodology not only requires the type of distribution to be known in advance, but also its parameters, *i.e.*, the mean (μ) and standard deviation (σ) in the case of a normal distribution, and such information may not always be available

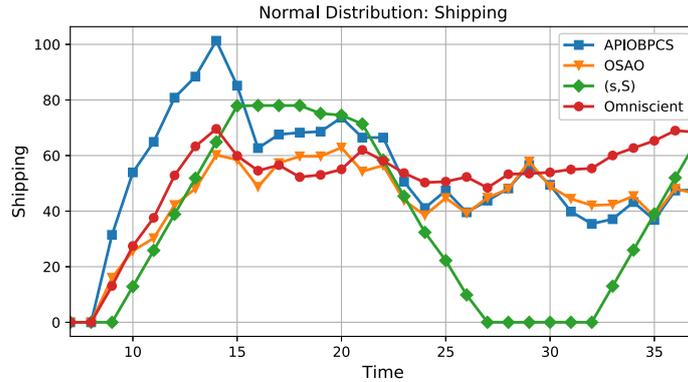


FIGURE 8. Shipping for all controllers with $D = 7$, assuming that demands from $k = 0$ to $k = 6$ are used as the initial past demands, to $k = (K_f - D) = 37$ for a normally distributed demand. Note that the APIOBPCS controller violates the maximum shipping capacity ($s_{h,\max} = 100$) at instant $k = 14$.

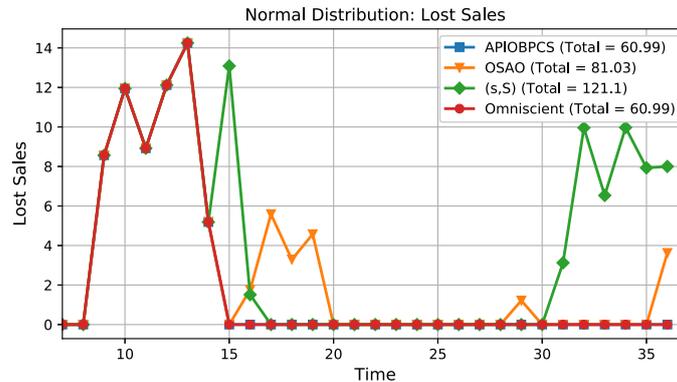


FIGURE 9. Lost sales for all controllers over the simulated horizon, for a normally distributed demand. All controllers lead to lost sales in the initial segment, up to $k = 14$, because the pipeline is initially empty. Both the noncausal omniscient solution and the APIOBPCS controller manage to avoid lost sales after $k = 14$.

in practice. It is important to emphasize that, in contrast, no information about the demand distribution is required for the OSAO controller. Moreover, if the value of the demand distribution parameters expected in the training set differ from the ones in the test set, then the solution found by this method does not guarantee good results for the APIOBPCS and (s, S) controllers, which depend on good tuning of their parameters unlike the parameter-free OSAO controller. See Section 6.

The previous example in Section 5.2 considered a normally distributed demand of $\mu = 10$ and $\sigma = 3$. By randomly generating a training set of $N = 10$ demands with the same $\mu = 10$ and $\sigma = 3$, the fixed gains of the APIOBPCS-based controller that maximized the average profit L of the demands in the training set were found to be: $T_p = 5.03$, $T_i = 16.71$ and $T_w = 15.63$. Analogously, the optimal stock levels for the (s, S) controller were found to be $s = 9.27$ and $S = 10.06$. The weights were set as $\omega_1 = \omega_2 = 20$. All other parameters remained the same as indicated in Table 2.

Figures 11–16 show the results of replacing the parameters used for the APIOBPCS and (s, S) controllers in the example of Section 5.2 with the optimal values obtained in this section. Table 4 summarizes the results

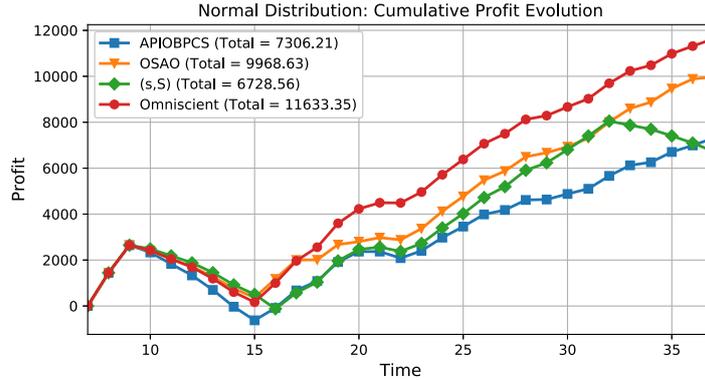


FIGURE 10. Cumulative profit for all controllers over the simulated horizon, for a normally distributed demand. The efficiencies of the causal controllers, at the end of the simulation horizon, with respect to the global optimum (noncausal omniscient solution) are: $\eta_{\text{apiobpcs}} = 62.80\%$, $\eta_{\text{osao}} = 85.69\%$ and $\eta_{\text{ss}} = 57.84\%$.

TABLE 4. Cumulative profits and efficiencies (normal demand of Section 5.2) after optimizing APIOBPCS and (s, S) controllers. Compare with the results of Table 3.

Controller	Profit (L)	Efficiency (η_x)
APIOBPCS	9706.35	83.44%
OSAO	9968.63	85.69%
(s, S)	7684.53	66.06%
Omniscient solution	11633.35	100%

and demonstrates the significant improvement in performance of the APIOBPCS and (s, S) controllers tuned by the proposed method. The result for the OSAO control, which is parameter free, is, of course, the same as in Table 3 and is repeated in Table 4 only for the purposes of convenient comparison. Further comments and observations are made in Subsection 5.3.1 below.

5.3.1. Discussion of numerical results

Figures 11 and 12 show the ordering and selling for the optimized APIOBPCS and (s, S) controllers respectively. There is a significantly lower bullwhip effect in the inventory (stock) level produced by the optimized APIOBPCS controller (compare Figs. 7 and 13). Furthermore, Figures 13 and 14 show that the APIOBPCS controller reaches lower peaks in the warehouse inventory and shipping levels and no longer violates the upper bounds in maximum warehouse ($s_{t,\max} = 50$) and shipping ($s_{h,\max} = 100$) capacities, in contrast to Figures 7 and 8 respectively. As for the optimized (s, S) controller, a reduction in the warehouse inventory level is noticeable in Figure 13, compared to Figure 7. Unlike the anticipative omniscient solution, the implementable controllers are not able to keep the inventory level at the minimum, which is the specified safety stock of $w = 1$, after the initial transient. Figure 15 shows a slight decrease in the lost sales of the optimized (s, S) controller. Lastly, Figure 16 shows that the efficiencies increased from the previous $\eta_{\text{apiobpcs}} = 62.80\%$ to 83.44% for the optimized APIOBPCS and from $\eta_{\text{ss}} = 57.84\%$ to 66.06% for the optimized (s, S) controller, showing the effectiveness of the proposed optimization (tuning) method.

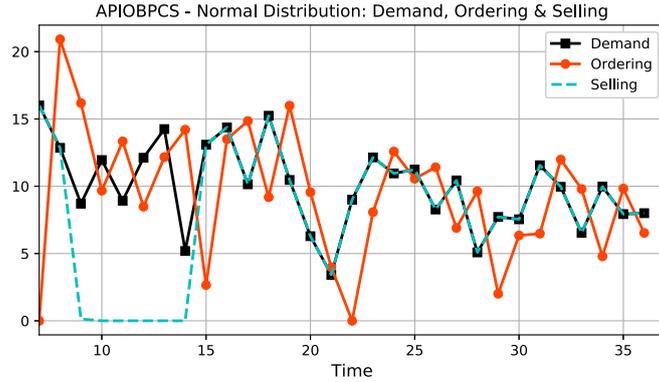


FIGURE 11. Ordering and selling for the APIOBPCS controller with $D = 7$, assuming that demands from $k = 0$ to $k = 6$ are used as the initial past demands, to $k = (K_f - 1 - D) = 36$ for a normally distributed demand. Compare with the results obtained with the unoptimized control in Figure 4.

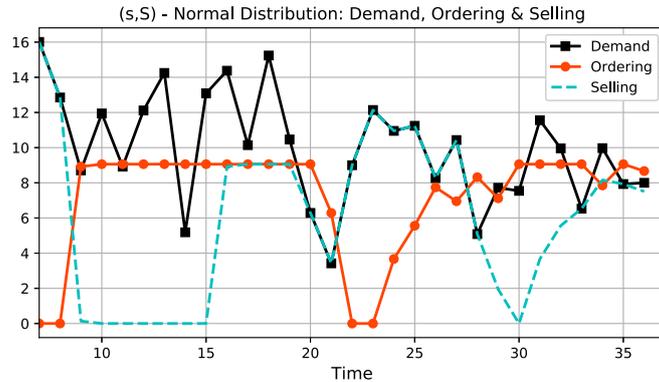


FIGURE 12. Ordering and selling for the (s, S) controller with $D = 7$, assuming that demands from $k = 0$ to $k = 6$ are used as the initial past demands, to $k = (K_f - 1 - D) = 36$ for a normally distributed demand. Compare with the results obtained with the unoptimized control in Figure 6.

5.4. Comparison of average performances for test sets of normally distributed demand

A common problem associated with many inventory control policies is the bullwhip effect of demand uncertainty resulting in larger amplitudes of system variables, such as inventory level. To examine this behavior, two measures are proposed in this paper: the normalized average used capacity A_{st} and the normalized peak stock level M_{st} . The normalized average used capacity A_{st} measures the average level of stock in the warehouse and is defined as the ratio of the mean of the warehouse inventory level s_t over the simulation horizon to the maximum storage capacity $s_{t,max}$:

$$A_{st} = \frac{\text{average use of warehouse capacity}}{\text{maximum warehouse capacity}}. \tag{43}$$

The normalized peak warehouse level M_{st} measures the peak level of stock in the warehouse and is defined as the ratio of the peak value of the warehouse inventory level s_t over the simulation horizon to the maximum

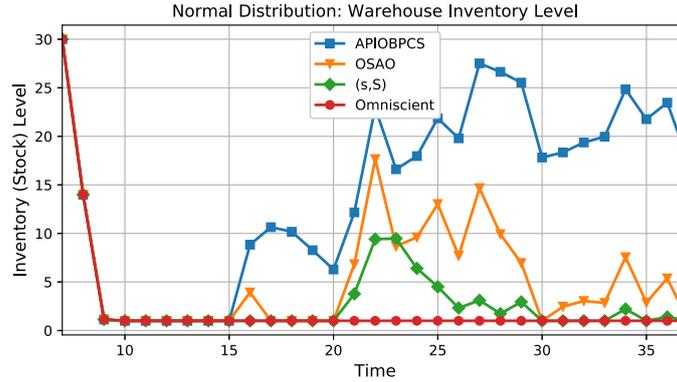


FIGURE 13. Inventory level for all controllers with $D = 7$, assuming that demands from $k = 0$ to $k = 6$ are used as the initial past demands, to $k = (K_f - D) = 37$ for a normally distributed demand. Compare with the results obtained with the unoptimized control in Figure 7.

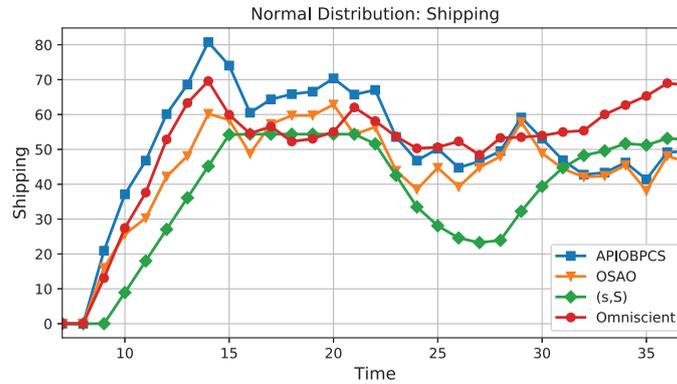


FIGURE 14. Shipping for all controllers with $D = 7$, assuming that demands from $k = 0$ to $k = 6$ are used as the initial past demands, to $k = (K_f - D) = 37$ for a normally distributed demand. Compare with the results obtained with the unoptimized control in Figure 8.

storage capacity $s_{t,\max}$:

$$M_{\text{st}} = \frac{\text{peak stock value}}{\text{maximum warehouse capacity}}. \quad (44)$$

Both these coefficients are calculated over the interval $[2D + 1, K_f - D]$, so as to exclude the initial transient due to pipeline initialization as well as the turnpike effect, and are used to quantify the effect of the variation of demand on the inventory level. Low A_{st} and M_{st} indicate a maximum storage capacity that is larger than required by the current demand level. Low A_{st} and high M_{st} indicate a bullwhip effect in the inventory. Lastly, high A_{st} and M_{st} indicate the inventory is being used close to its full capacity. These inferences are collected in Table 5.

Remark 5.1. The bullwhip effect is commonly defined in inventory control literature as the ratio of the coefficients of variation of the ordering and the demand. Measures greater than the unity are an indication of bullwhip effect due to the amplification of demand variability. However, in our experiments, we found some of these measures to be occasionally misleading, in particular, for the omniscient solution, whose optimal ordering

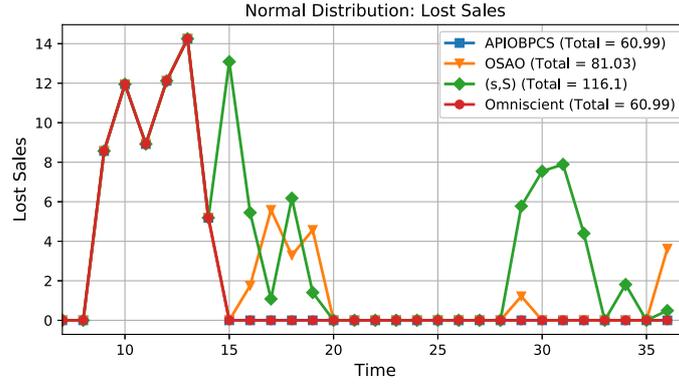


FIGURE 15. Lost sales for all controllers over the simulated horizon, with a normally distributed demand. Compare with the results obtained with the unoptimized control in Figure 9.

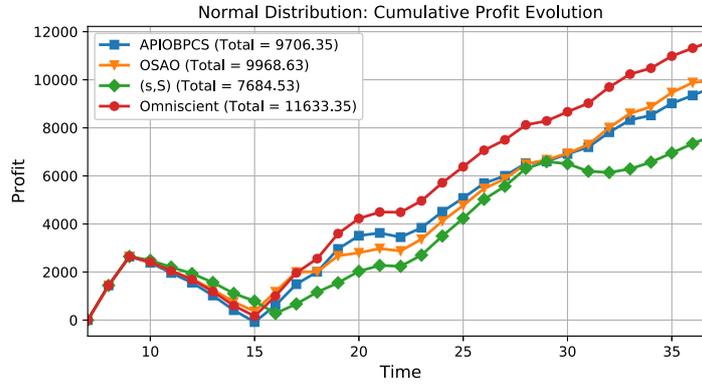


FIGURE 16. Cumulative profit for all controllers over the simulated horizon, with a normally distributed demand. The efficiencies of the causal controllers, at the end of the simulation horizon, with respect to the global optimum (noncausal omniscient solution) are: $\eta_{\text{apiobps}} = 83.44\%$, $\eta_{\text{osao}} = 85.69\%$ and $\eta_{\text{ss}} = 66.06\%$. Compare with the results obtained with the unoptimized control in Figure 10.

TABLE 5. Inferences from combinations of the average used capacity A_{st} and peak warehouse level M_{st} values for a given set of demands and simulation horizon

A_{st}	M_{st}	Inference
Low	Low	Max. storage capacity larger than required for the current demand level.
Low	High	Bullwhip effect in the warehouse inventory.
High	High	Inventory being used close to full capacity and danger of bullwhip induced overflow.

TABLE 6. Average profit efficiencies, average used capacities, peak warehouse levels and respective standard deviations for 10 normal demands ($\mu = 10$, $\sigma = 3$).

Controller	Efficiency (η_x)	Used Capacity (A_{st})	Peak Level (M_{st})
APIOBPCS	81.71 (± 5.33)%	27.69 (± 11.00)%	59.76 (± 13.84)%
OSAO	83.56 (± 5.03)%	10.72 (± 3.70)%	31.99 (± 8.65)%
(s, S)	67.95 (± 5.92)%	4.77 (± 0.93)%	17.08 (± 5.88)%

TABLE 7. Average profit efficiencies, average used capacities, peak warehouse levels and respective standard deviations for 10 normal demands ($\mu = 20$, $\sigma = 3$).

Controller	Efficiency (η_x)	Used capacity (A_{st})	Peak level (M_{st})
APIOBPCS	87.52 (± 4.09)%	7.79 (± 4.17)%	31.38 (± 16.10)%
OSAO	93.87 (± 2.26)%	4.52 (± 1.93)%	17.98 (± 10.82)%
(s, S)	90.41 (± 1.87)%	2.42 (± 0.34)%	7.43 (± 3.70)%

anticipates the demand by D time units (as discussed in Sect. 5.2). For this reason, we decided to characterize the bullwhip effect using indexes (43) and (44), as explained in Table 5.

5.4.1. Training and test set methodology

This section utilizes the approach illustrated in the previous section, for a single test demand, to a set of test demands, in order to compare average performances. Ten samples of normally distributed demands with $\mu = 10$ and $\sigma = 3$ were generated and grouped in a separate set, referred to as test set. The training set, discussed in the previous section (and henceforth denoted as S_1), and the test set (henceforth denoted as S_2) are two distinct sets and it is ensured that $S_1 \cap S_2 = \emptyset$. The parameters for the APIOBPCS controller were optimized using the methodology described in Section 5.3 and the solutions found were $T_p = 5.03$, $T_i = 16.71$, $T_w = 15.63$. Analogously, the optimal stock levels for the (s, S) controller were found to be $s = 9.27$ and $S = 10.06$. The average efficiencies (resp. standard deviation) considering the cumulative profit with respect to the globally optimum (noncausal omniscient solution) are presented in Table 6, along with the average used capacities and peak warehouse values for these ten normally distributed demands. These results indicate that all controllers are operating well below the lower bound specified on the right hand side of (16) and are not being subjected to stressful operating conditions.

Thus, in order to test the controllers under more extreme circumstances, a test set, S_2 , comprised of ten samples of normally distributed demands with $\mu = 20$ and $\sigma = 3$ was generated, thus increasing the mean demand by one hundred percent. Once again, the tuning parameters of the APIOBPCS controller and the stock levels of the (s, S) controller were adjusted using the methodology described in 5.3. Using the same parameters in Table 2 and defining the setpoint $I_{ref} = 20$ *i.e.*, the mean of the normal distribution, led to the stock levels $s = 16.57$ and $S = 17.67$ of the (s, S) controller and the fixed gains $T_p = 2.86$, $T_i = 174.42$, $T_w = 10.63$ of the APIOBPCS controller. Table 7 summarizes the average results (\pm standard deviation) obtained for a test set of 10 normally distributed demands with $\mu = 20$ and $\sigma = 3$. The proposed OS AO controller maintains a higher efficiency than the others (after they have been tuned).

6. ROBUSTNESS TO DEMAND UNCERTAINTY

This section reports on an experimental investigation of the robustness of the APIOBPCS, OS AO and (s, S) controllers with respect to normally distributed demands of uncertain mean and standard deviation. The experiment is designed as follows. The means (resp. standard deviations) of the normally distributed demands

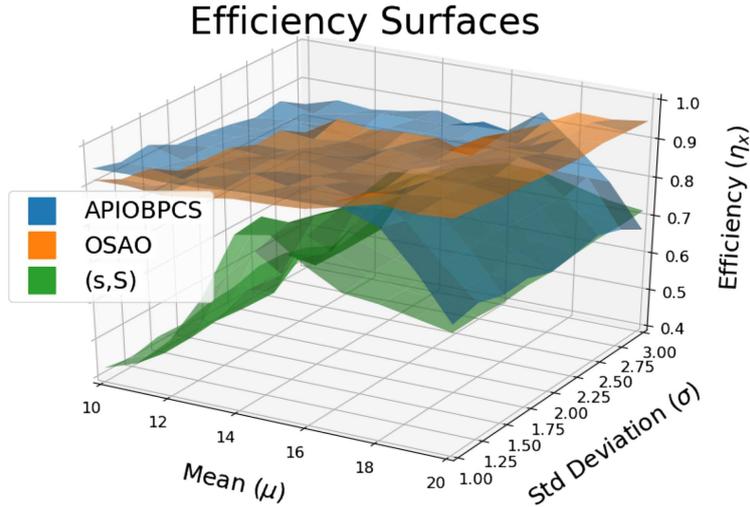


FIGURE 17. Efficiency surfaces for each controller x with respect to the objective function L . Each point on a surface represents the average efficiency η_x^L for 10 normally distributed demands from the test set. Observe that the efficiency of the OSAO controller stay at or above levels of 80% of the (unattainable) omniscient global optimum profit for all combinations of $10 \leq \mu \leq 20$ and $1 \leq \sigma \leq 3$, while the efficiencies of the (s, S) and APIOBPCS controllers fall off considerably in certain areas of the (μ, σ) -space.

are allowed to vary in the intervals $[10, 20]$ (resp. $[1, 3]$), in steps of size 1 (resp. 0.25). At each point on this grid in the (μ, σ) -space, a test set of ten normally distributed demands is generated. The parameters of the APIOBPCS and (s, S) controllers are optimized, using a training set of ten normally distributed demands, at the central point $(15, 2)$ of the (μ, σ) -grid and held fixed for the entire experiment. As before, the test and training sets are disjoint. The average efficiency over the test set of each of the four controllers is computed at each grid point and gives rise to the average efficiency surfaces plotted in Figure 17.

At the central point $\mu = 15, \sigma = 2$ of the (μ, σ) grid, the optimized stock levels of the (s, S) controller were found to be $s = 14.44$ and $S = 15.21$ while, the optimized parameters of the APIOBPCS-based controller were found to be $T_p = 2.52$, $T_i = 172.29$ and $T_w = 80.98$, and these optimized parameters were kept fixed throughout the experiment. The setpoint was defined as $I_{\text{ref}} = 15$, the central value of μ , and all other parameters used in this numerical experiment were the same as indicated in Table 2. As can be seen in Figure 17, the (s, S) controller proved to be less robust as the value of the mean decreased ($\mu < 15$). On the other hand, the APIOBPCS-based controller proved to be less robust as the value of the mean increased ($\mu > 15$). Since the OSAO controller does not require the setting of control parameters (such as stock levels, fixed gains or setpoint), it is seen to be robust over the entire space defined by the intervals of μ and σ , attaining an average profit of at least 80% of the (unattainable) omniscient global maximum profit, and this is one of the significant advantages of the scheme being proposed in this paper.

7. ROBUSTNESS TO LEAD TIME (DELAY) UNCERTAINTY

So far, all experiments in this paper assumed a constant lead time D . However, the time elapsed from the placement of an order up to the instant the good is received at the warehouse can vary in real-life inventory control problems. Unlike the case with demand uncertainty, the introduction of lead time (delay) uncertainty requires a few adjustments to be made in the single echelon supply chain dynamics discussed in Section 2.

First, a sequence of lead times, denoted D_{var} , is defined as follows:

$$D_{\text{var}} = \{D_1, D_2, \dots, D_{K_f}\}, \quad (45)$$

where the k -th element of D_{var} , denoted D_k , corresponds to the actual lead time for an order placed at time instant k , for $k = 1, \dots, K_f$. As with the demand sequence, D_{var} is assumed to be known in advance only while computing the omniscient solution and not known by the implemented controllers OSAO, APIOBPCS and (s, S) .

The length of the conveyor, described by equations (1) and (2), needs to be increased taking into account the maximum lead time, denoted D_{max} and defined as:

$$D_{\text{max}} = \max\{D_1, D_2, \dots, D_{K_f}\}, \quad (46)$$

which can be greater than the expected constant lead time D used previously.

Remark 7.1. As noted in Page 507 of [25], knowledge of lead times and lead time variation is needed for management to be effective. Therefore, the assumption of D_{max} being known beforehand is reasonable. Furthermore, equation (46) provides the minimum required length of the conveyor (*i.e.*, the number of conveyor states) in order for the SESC dynamics to be able to account for all lead times in D_{var} . For real-life applications where lead time variation may not be known, D_{max} only needs to be greater than the maximum expected lead time *i.e.*, $D_{\text{max}} > E[\max(D_{\text{var}})]$. The simple choice of $D_{\text{max}} = \max(D_{\text{var}})$ is made below.

Lastly, the position of the conveyor in which the ordering $o(k)$ is inserted, denoted by the subscript $p(k)$ of x , is no longer fixed and equal to D . Instead, $p(k)$ varies according to the value of $D_k \in [1, D_{\text{max}}]$ at each time instant k for $k = 1, \dots, K_f$. The conveyor state dynamics described in (1) and (2) can, then, be generalized as follows:

$$x_i(k+1) = x_{i+1}(k), \quad i \in \{1, \dots, D_{\text{max}}\} - \{p(k)\} \quad (47)$$

$$x_{p(k)}(k+1) = x_{p(k)+1}(k) + o(k) \quad (48)$$

where $x_{p(k)+1}(k) = 0$ when $p(k) = D_{\text{max}}$.

Figure 18 illustrates how the ordering $o(k)$, placed at time instant k , is inserted at the position $p(k)$ of the conveyor for varying values of D_k .

The number of goods being shipped at a time can be rewritten as:

$$s_h(k) = \sum_{i=2}^{D_{\text{max}}} x_i(k), \quad (49)$$

Hence, the omniscient solution can be obtained by solving the following MILP problem:

$$\begin{aligned} & \text{maximize} && L(K_f) \\ & \{o(k)\}_{k=D}^{K_f-1} \\ & \text{subject to} && (47), (48), (4), (12), (24), (25) \quad \forall k \in [D, K_f - 1], \\ & && \text{and } (14), (15), (16), (26) \quad \forall k \in [D, K_f - 1] \\ & && \text{and } (29), (30), (31) \quad \forall k \in [D, K_f - 1]. \end{aligned} \quad (50)$$

The state vector of the supply chain under lead time uncertainty at instant k is denoted as $\mathbf{z}(k)$ and defined as:

$$\mathbf{z}(k) = [x_1(k), \dots, x_{D_{\text{max}}}(k), s_t(k)]. \quad (51)$$

The OSA optimizer (see Fig. 2), however, does not have any knowledge of the lead time sequence D_{var} in advance other than the highest possible lead time value D_{max} , which sets the length of the new state vector.

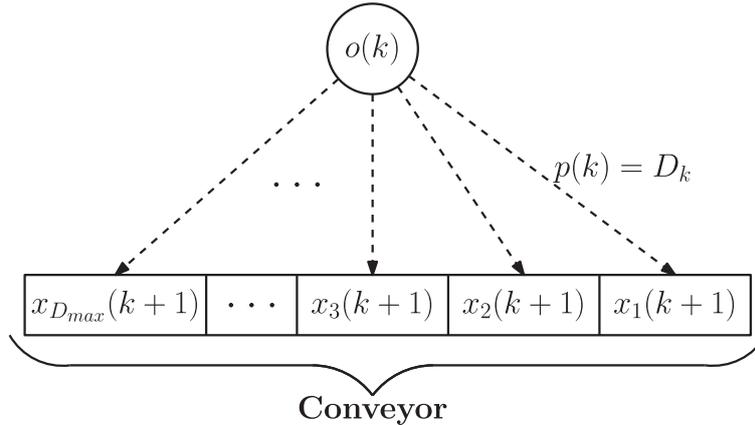


FIGURE 18. Conveyor update. The ordering $o(k)$, placed at time instant k , is inserted at the position $p(k)$ of $x(k+1)$ according to the lead time value $D_k \in [1, D_{\max}]$, yielding equation (48). The first position of the conveyor, x_1 , informs the quantity of goods that arrive at the warehouse while the other positions, x_2 up to $x_{D_{\max}}$ inform the goods currently in transit (being shipped) and their respective stages.

Instead, it assumes a constant lead time of $D \leq D_{\max}$ time units. Thus, the conveyor state dynamics for the OSA optimizer block of the OSAO controller can be rewritten as:

$$x_i(k+1) = x_{i+1}(k), \quad i \in \{1, \dots, D_{\max}\} - \{D\} \quad (52)$$

$$x_D(k+1) = x_{D+1}(k) + o(k). \quad (53)$$

For the particular case where $D = D_{\max}$, equations (52) and (53) become identical to (1) and (2) respectively, since the position $x_{D+1}(k)$ lies outside of the conveyor's boundary and, therefore, is equal to zero.

The OSAOC problem adapted for variable delay, and denoted as $OSA(\mathbf{z}(k), D)$, can be transcribed in the following MILP problem:

$$\begin{aligned} & \text{maximize} && L(k+1) \\ & \text{subject to} && \{o(i)\}_{i=k-D}^k \\ & && \text{subject to (52), (53), (4), (12), (24), (25)} \quad \forall i \in [k-D, k], \\ & && \text{and (14), (15), (16), (26)} \quad \forall i \in [k-D, k] \\ & && \text{and (29), (30), (31)} \quad \forall i \in [k-D, k] \\ & && \text{and } \mathbf{z}_0 = \mathbf{z}(k) \text{ (specified initial conditions)}. \end{aligned} \quad (54)$$

The optimal ordering sequence for the OSAO controller can be obtained by following similar steps as the ones described in Section 3. By solving (54) at a given instant k , we find the optimal ordering $o_{\text{osao}}(k)$. This procedure corresponds to the OSA optimizer block in Figure 2 and assumes no knowledge of D_{var} , using the conveyor state dynamics described by (52) and (53). After the optimal ordering $o_{\text{osao}}(k)$ has been placed, and its value can no longer be changed, the actual conveyor is updated according to equations (47) and (48), using the now known information about the actual lead time $D_k \in [1, D_{\max}]$ at time instant k . Lastly, the SESC dynamics are evaluated using the now known actual demand and the state vector $\mathbf{z}(k+1)$ is updated. From this new initial condition, a new optimization step can be performed. Algorithm 2 summarizes the necessary changes made to the OSAO controller in order to account for lead time uncertainty.

This section reports on an experimental investigation of the robustness of the APIOBPCS, OSAO and (s, S) controllers with respect to ten randomly generated and uniformly distributed delay sequences, with values in the interval $[5, 9]$ *i.e.*, $D_k \in [5, 9]$ for $k = 1, \dots, K_f$. In this experiment, each controller has its cumulative profit

Algorithm 2 OSAOC with delay (lead time) uncertainty

```

1: Initial state  $\mathbf{z}_0 := \mathbf{z}(k_0)$ ,  $k_0 = D_{\max}$  given
2: while  $D \leq k \leq K_f - 1$  do
3:    $\mathbf{z}(k) := \mathbf{z}_0$ 
4:   solve  $OSA(\mathbf{z}(k), D)$  defined in (54)
5:   to get  $[o^*(k - D); \dots; o^*(k)]$ 
6:   Discard all computed optimal orderings except  $o^*(k - D)$ 
7:    $o_{\text{osao}}(k) \leftarrow o^*(k - D)$ 
8:   Evaluate RHS of (4) to obtain  $s_t(k + 1)$ 
9:   Update the conveyor state (47, 48) according to  $D_k$ 
10:   $\mathbf{z}_0 \leftarrow [x_1(k + 1), \dots, x_{D_{\max}}(k + 1), s_t(k + 1)]$ 
11:   $k \leftarrow k + 1$ 
12: end while

```

TABLE 8. Efficiency (\pm standard deviation) of the APIOBPCS, OSAO and (s, S) controllers with respect to constant ($D = 7$) and variable ($D_k \in [5, 9]$) delay sequences.

Controller	Variable delay efficiency	Constant delay efficiency
APIOBPCS	63.06 (± 8.38)%	83.11%
OSAO	71.13 (± 4.12)%	85.69%
(s, S)	49.64 (± 9.51)%	66.06%

L and efficiency η_x evaluated for all ten different delay sequences. The overall performance of each controller is, then, calculated by taking the average (and standard deviation) of the ten cumulative profits obtained for each delay sequence.

Throughout the experiment, the system parameters are set to the values indicated in Table 2 and the same demand profile, used in the first three numerical examples of Section 5, is adopted for all performance evaluations. The tuning parameters of the APIOBPCS and (s, S) controllers are set to the optimized values obtained in Section 5.3 for $D = 7$ time units, which are $T_p = 5.03$, $T_i = 16.71$, $T_w = 15.63$ for the APIOBPCS controller and $s = 9.27$ and $S = 10.06$ for the (s, S) controller.

Table 8 shows the performance (average efficiency and standard deviation) evaluated for each controller with respect to ten uniformly distributed delay sequences. As can be seen, the OSAO controller achieved both the highest average efficiency (above 70%) and lowest standard deviation (below 5%), meaning that its performance suffered less from different delay sequences. For the purpose of convenient comparison, Table 8 also shows the efficiency of the controllers with respect to the constant lead time $D = 7$ (first presented in Tab. 4).

8. CONCLUSIONS

This paper proposed a novel one step ahead optimal control (OSAOC) scheme for a single echelon supply chain model, that does not require prediction. The proposed control scheme requires the solution of a low-dimensional (in terms of the number of decision variables) optimization problem at each time step, thus being perfectly adequate for real time inventory control. The one step ahead control is shown to be efficient, robustly attaining an average profit of at least 80% of the (unattainable) omniscient global optimum for different normally distributed demands. Furthermore, the one step ahead control proved to be robust with respect to delay (lead time) uncertainty, attaining 70% of the globally optimal omniscient solution for different delay sequences. In addition, it should be emphasized that the use of a controller with a fixed structure such as the APIOBPCS controller requires the tuning of parameters (setpoints and gains) associated to its structure. It is difficult to find fixed choices of parameters that work well over a range of demands. In other words, if a set of parameters is chosen to be adequate or even optimal for a given nominal demand d_{nom} , it will, in general, be worse for

some other demand d , even if d is close, in some appropriate metric, to the demand d_{nom} . The degradation of performance will increase as the demand d moves further away from d_{nom} in the given metric. This is to be contrasted with the proposed OSAO controller, which is always locally optimal and does not need any tuning, as changes occur in the lead time or in the demand, for example, in its distribution or in its mean and standard deviation values. This is one of the main advantages of the proposed approach. It is also important to observe that, for the single echelon case, it was shown how to rewrite the OSAOC optimization problem as linear programming problem with a relatively small number of decision variables. From a computational viewpoint, this implies that the simple examples in Section 5 can easily be scaled to much larger dimension. It should also be noted that a large number of supply chain problems, including the multi-echelon case, are in fact described by linear or piecewise linear dynamics. Thus, the proposed scheme is scalable and can: (i) be generalized to large multi-echelon supply chains and (ii) used in a planning mode, to choose inventory size, shipping capacity requirements and prices that ensure profit margins, since these choices can be recast as feasibility problems. It should also be pointed out that, although this paper used the example of a single echelon supply chain, with dynamics such that the resulting OSAO problems were linear programs, the OSAO approach is not limited to dynamics and objective functions that necessarily lead to linear programming problems. The generalization of the model with a single good presented in this paper to a model with multiple goods is straightforward, if it is assumed that some upper bound on the stock of each good is given, together with some lower bounds on their safety stocks. The upper bound would be constrained by the physical capacity of the warehouse. Similarly, if several goods are shipped together, the model could put a common upper bound on the shipping of these goods, and so on. These topics are the subject of ongoing work. The fact that the proposed OSAOC approach does not require demand prediction makes it a strong contender for a cheap and universal control scheme that could be adopted by practitioners who might not be comfortable with the considerably greater sophistication of schemes that require prediction, such as the class of model predictive control schemes.

Acknowledgements. The work of L.C. Roth was supported by a fellowship from the Brazilian Govt agency CAPES, Finance Code 001. The work of A. Bhaya was partially supported by the research grants (BPP/PQ 310646/2016-2) & Universal (406106/2016-9) from the Brazilian Govt agency CNPq. The work of E. Kaszkurewicz supported by a grant from the Rio de Janeiro state agency FAPERJ (Bolsa PVE E-26/201.972/2016). A preliminary version of this paper was presented at the ICORES2020 conference in Malta, in February 2020 [4]. All sections of this paper have been modified and added to substantially, so that the overlap with the conference paper [4] is minimal and occurs only in the basic model description in Section 2. Sections 4 through 7 are completely new.

REFERENCES

- [1] K.J. Arrow, T. Harris and J. Marschak, Optimal inventory policy. *Econometrica* **19** (1951) 250–272.
- [2] L. Benkherouf and B.H. Gilding, Optimal policies for a deterministic continuous-time inventory model with several suppliers. *RAIRO:RO* **55** (2021) S947–S966.
- [3] J. Bezanson, A. Edelman, S. Karpinski and V.B. Shah, Julia: A fresh approach to numerical computing. *SIAM Rev.* **59** (2017) 65–98.
- [4] A. Bhaya, E. Kaszkurewicz and L.C. Roth, One step ahead optimal control of a single echelon supply chain using mathematical programming, in *Proceedings of the 9th International Conference on Operations Research and Enterprise Systems (ICORES)*, Valletta, Malta (2020) 233–240.
- [5] M.D. Canon, C.D. Cullum and E. Polak, Theory of Optimal Control and Mathematical Programming, 1st edition. New York, NY, USA, McGraw-Hill (1970).
- [6] R.C. Dorf and R.H. Bishop, Modern Control Systems, 8th edition. Boston, MA, USA, Addison Wesley Longmann (1998).
- [7] I. Dunning, J. Huchette and M. Lubin, JuMP: A modeling language for mathematical optimization. *SIAM Rev.* **59** (2017) 295–320.
- [8] K. Hoberg, J. Bradley and U. Thonemann, Analyzing the effect of the inventory policy on order and inventory variability with linear control theory. *Eur. J. Oper. Res.* **176** (2007) 1620–1642.
- [9] D. Ivanov, S.P. Sethi, A. Dolgui and B. Sokolov, A survey on control theory applications to operational systems, supply chain management, and industry 4.0. *Annu. Rev. Control* **46** (2018) 134–147.
- [10] E. Lavretsky, Greedy optimal control, in *Proceedings of the American Control Conference (ACC)*, Chicago, IL, USA (2000) 3888–3892.

- [11] J. Lin and M. Naim, Why do nonlinearities matter? The repercussions of linear assumptions on the dynamic behavior of assemble-to-order systems. *Int. J. Prod. Res.* **57** (2019) 6424–6451.
- [12] J. Lin, M. Naim, L. Purvis and J. Gosling, The extension and exploitation of the inventory and order based production control system archetype from 1982 to 2015. *Int. J. Prod. Econ.* **194** (2017) 135–152.
- [13] J. Lin, M. Naim and V. Spiegler, Delivery time dynamics in an assemble-to-order inventory and order based production control system. *Int. J. Prod. Econ.* **223** (2020) 107531.
- [14] B. McGarvey and B. Hannon, *Dynamic Modeling for Business Management: an Introduction*, 1st edition. New York, NY, USA, Springer (2004).
- [15] J.A. Muckstadt and A. Sapra, *Principles of Inventory Management: When you are Down to Four, Order More*, 1st edition. New York, NY, USA, Springer (2010).
- [16] K.G. Murty, Forecasting for supply chain and portfolio management (2006) Accessed on: Feb. 16, 2022, [Online] Available: <http://www-personal.umich.edu/~murty/forecastSCM-PFOLI02.pdf>
- [17] E. Perea-López, B.E. Ydstie and I. Grossmann, A model predictive control strategy for supply chain optimization. *Comput. Chem. Eng.* **27** (2003) 1201–1218.
- [18] J. Qin, Y. Chow, J. Yang and R. Rajagopal, Online modified greedy algorithm for storage control under uncertainty. *IEEE Trans. Power Syst.* **31** (2016) 1729–1743.
- [19] J. Schwartz, W.-L. Wang and D. Rivera, Simulation-based optimization of process control policies for inventory management in supply chains. *Automatica* **42** (2006) 1311–1320.
- [20] S.P. Sethi and G. Thompson, *Optimal Control Theory: Applications to Management Science and Economics*, 2nd edition. New York, NY, USA, Springer (2000).
- [21] J. Simon, M. Naim and D.R. Towill, Dynamic analysis of a WIP compensated decision support system. *Int. J. Manuf. Syst.* **1** (1994) 283–297.
- [22] V. Spiegler, M. Naim and J. Wikner, A control engineering approach to the assessment of supply chain resilience. *Int. J. Prod. Res.* **50** (2012) 6162–6187.
- [23] H. Stadtler, C. Kilger and H. Meyr, *Supply Chain Management and Advanced Planning*, 5th edition. Berlin Heidelberg, Germany, Springer-Verlag (2015).
- [24] J.M. Stern, J.S. Shiele and I. Ross, *The EVA Challenge: Implementing Value Added Change in an Organization*, 1st edition. New York, NY, USA, John Wiley & Sons (2001).
- [25] W.J. Stevenson, *Operations Management*, 14th edition. New York, NY, USA, McGraw-Hill (2021).
- [26] K. Subramanian, J. Rawlings and C. Maravelias, Economic model predictive control for inventory management in supply chains. *Comput. Chem. Eng.* **64** (2014) 71–80.
- [27] H. Tang, H. Zhang, R. Liu and Y. Du, Integrating multi-index materials classification and inventory control in discrete manufacturing industry: Using a hybrid ABC-chaos algorithm. *IEEE Trans. Eng. Manag.* (2020) 1–18.
- [28] F. Tao, T. Fan and K.-K. Lai, Inventory management under power structures with consignment contract subject to inventory inaccuracy. *IEEE Trans. Eng. Manag.* **66** (2019) 763–773.
- [29] T. Tosetti, D. Patiño, F. Capraro and A. Gambier, A new inventory level APIOBPCS-based controller, in *Proceedings of the American Control Conference (ACC)*. Seattle, WA, USA (2008) 2886–2891.
- [30] E. Trélat and E. Zuazua, The turnpike property in finite-dimensional nonlinear optimal control. *J. Differ. Equ.* **258** (2015) 81–114.
- [31] W.-L. Wang, D. Rivera and K. Kempf, Model predictive control strategies for supply chain management in semiconductor manufacturing. *Int. J. Prod. Econ.* **107** (2007) 56–77.
- [32] Z. Wang, T.S. Felix and M. Li, A robust production control policy for hedging against inventory inaccuracy in a multiple-stage production system with time delay. *IEEE Trans. Eng. Manag.* **65** (2018) 474–486.



Please help to maintain this journal in open access!

This journal is currently published in open access under the Subscribe to Open model (S2O). We are thankful to our subscribers and supporters for making it possible to publish this journal in open access in the current year, free of charge for authors and readers.

Check with your library that it subscribes to the journal, or consider making a personal donation to the S2O programme by contacting subscribers@edpsciences.org.

More information, including a list of supporters and financial transparency reports, is available at <https://edpsciences.org/en/subscribe-to-open-s2o>.