



Manufacturing & Service Operations Management

Publication details, including instructions for authors and subscription information:
<http://pubsonline.informs.org>

Production Campaign Planning Under Learning and Decay

Hossein Jahandideh, Kumar Rajaram, Kevin McCardle

To cite this article:

Hossein Jahandideh, Kumar Rajaram, Kevin McCardle (2020) Production Campaign Planning Under Learning and Decay. Manufacturing & Service Operations Management 22(3):615-632. <https://doi.org/10.1287/msom.2018.0766>

Full terms and conditions of use: <https://pubsonline.informs.org/Publications/Librarians-Portal/PubsOnLine-Terms-and-Conditions>

This article may be used only for the purposes of research, teaching, and/or private study. Commercial use or systematic downloading (by robots or other automatic processes) is prohibited without explicit Publisher approval, unless otherwise noted. For more information, contact permissions@informs.org.

The Publisher does not warrant or guarantee the article's accuracy, completeness, merchantability, fitness for a particular purpose, or non-infringement. Descriptions of, or references to, products or publications, or inclusion of an advertisement in this article, neither constitutes nor implies a guarantee, endorsement, or support of claims made of that product, publication, or service.

Copyright © 2019, INFORMS

Please scroll down for article—it is on subsequent pages



With 12,500 members from nearly 90 countries, INFORMS is the largest international association of operations research (O.R.) and analytics professionals and students. INFORMS provides unique networking and learning opportunities for individual professionals, and organizations of all types and sizes, to better understand and use O.R. and analytics tools and methods to transform strategic visions and achieve better outcomes.

For more information on INFORMS, its publications, membership, or meetings visit <http://www.informs.org>

Production Campaign Planning Under Learning and Decay

 Hossein Jahandideh,^a Kumar Rajaram,^a Kevin McCardle^a
^a Anderson School of Management, University of California, Los Angeles, California 90095-1481

Contact: hs.jahan@gmail.com,  <https://orcid.org/0000-0002-9239-5673> (HJ); kumar.rajaram@anderson.ucla.edu,

 <https://orcid.org/0000-0002-6939-8219> (KR); kmccardl@anderson.ucla.edu,  <http://orcid.org/0000-0001-6141-5689> (KM)

Received: April 3, 2018

Revised: September 22, 2018

Accepted: November 9, 2018

Published Online in Articles in Advance:
August 7, 2019

<https://doi.org/10.1287/msom.2018.0766>
Copyright: © 2019 INFORMS

Abstract. Problem definition: We analyze a catalyst-activated batch-production process with uncertainty in production times, learning about catalyst-productivity characteristics and decay of catalyst performance across batches. The goal is to determine the quality level of batches and to decide when to replenish a catalyst so as to minimize average costs, consisting of inventory-holding, backlogging, and catalyst-switching costs. **Academic/practical relevance:** This is an important problem in a variety of process-industry sectors, such as food processing, pharmaceuticals, and specialty chemicals, but has not been adequately studied in the academic literature. This paper also contributes to the stochastic economic lot-sizing literature. **Methodology:** We formulate this problem as a semi-Markov decision process (SMDP) and develop a two-level heuristic to solve this problem. The heuristic consists of a lower-level problem that plans the duration of batches within the current campaign to maximize the efficiency of the catalyst while ensuring that the target attribute level for each batch is set to meet a quality specification represented by an average attribute level across all the batches in a campaign. The higher-level problem determines when to replace the costly catalyst as its productivity decays. To evaluate our heuristic, we present a lower bound on the optimal value of the SMDP. This bound accounts for all costs, as well as the randomness and discreteness in the process. We then extend our methods to multiple-product settings, which results in an advanced stochastic economic lot-sizing problem. **Results:** We test our proposed solution methodology with data from a leading food-processing company and show that our methods outperform current practice with an average improvement of around 22% in costs. In addition, compared with the stochastic lower bounds, our results show that the two-level heuristic attains near-optimal performance for the intractable multidimensional SMDP. **Managerial implications:** Our results generate three important managerial insights. First, our simulation-based lower bound provides a close approximation to the optimal cost of the SMDP, and it is nearly attainable by using a relatively simple two-level heuristic. Second, the reoptimization policy used in the lower-level problem adequately captures the value of information and Bayesian learning. Third, in the higher-level problem of choosing when to replace a catalyst, the intractable multidimensional state of the system is efficiently summarized by a single statistic: the probability of inventory falling below a specific threshold.

Funding: H. Jahandideh thanks the Easton Center for Technology at the UCLA Anderson School of Management for funding support.

Supplemental Material: The online appendix is available at <https://doi.org/10.1287/msom.2018.0766>.

Keywords: [campaign planning](#) • [batch production](#) • [stochastic economic lot sizing problem](#) • [Bayesian updating](#) • [production learning](#) • [semi-Markov processes](#) • [stochastic dynamic programming](#)

1. Introduction

Batch-production processes typically use a reactor to convert a base product into another, more refined product across a wide range of process industries (Korovessi and Linninger 2005). For example, sorghum is refined into sorbitol in a reactor; grain slurry is distilled into spirits in a still; chemical compounds are converted into a pharmaceutical drug such as interferon in an autoclave; and iron ore is processed into steel in a furnace. In batch manufacturing, as opposed to discrete production, batch size is determined by the capacity of the reactor. Because an

individual reactor is used for a variety of products, each with different characteristics, batch size tends not to be optimal for any one product. To mitigate the suboptimality of the reactor size, multiple batches of the same product are run in sequence (in what is called a *campaign*), in essence approximating the optimal batch size. Running a lengthy campaign of an individual product has two material effects: one beneficial and the other costly. The pooling of the batches in a campaign reduces the variation in the quality specification of the individual batches, which is beneficial; the longer the campaign, the lower is the variation. By contrast, the

running of a lengthy campaign of one product requires storage of these batches, larger inventories of the other products (increasing holding costs), or shortages (incurring backorder costs).

In several batch-production processes, catalysts are used to control the characteristics of production. Catalyst-activated batch-production processes are found in food processing, specialty chemicals, oil segregation, pharmaceuticals, and biotech manufacturing processes. Specific examples of different types of catalyst-activated batch processes include (1) purification and segregation, such as degradation of diclofenac in water (Hofmann et al. 2007), (2) multiphase reactions in fine chemicals and pharmaceuticals (Mills and Chaudhury 1997), (3) catalytic cracking processes such as in ethylene production (Jain and Grossman 1998, Keller et al. 2010), (4) homogeneous and heterogeneous alkali-catalyzed batch processes for biodiesel production (Sakai et al. 2009), and (5) reactive batch distillation for fuel components such as oxygenates (Korovessi and Linninger 2005). A more technical classification of reaction processes based on the type of catalysis and the type of phases involved is provided by Mills and Chaudhury (1997).

Across all these industrial applications of catalyst-activated processes, the primary role of the catalyst is to help the reactor modify the product and achieve a target attribute level in a given batch. This target attribute level could represent the concentration of a chemical formed as a by-product during the reaction (Molnar-Peril et al. 1990) or physical characteristics such as density or viscosity. Because of the decay of the catalyst, the producer has to switch to a new catalyst at some point and incur the associated switchover costs. These include the costs of cleaning the reactor and replacing the costly catalyst and other reagents that regulate the reaction. Typically, because of small differences in the composition of the catalyst, the initial productivity of a new catalyst is not known but is a random value drawn from a known distribution estimated from historical data. One needs to observe the time it takes to achieve a target attribute level in a given batch to estimate the productivity of the catalyst. Such observations are noisy because they can be affected by even small unpredictable deviations (or random shocks) in the initial conditions of the reaction such as temperature, pressure, quantity, and quality of the batch. These noisy observations can be used to update the estimate of the catalyst productivity through Bayesian updating. Although the type of reactor, nature of the catalyst, and composition of the batch will vary by industry, the decisions from a planning perspective in catalyst-activated batch processing are the same. They consist of determining the duration of the batches and deciding when to replenish a catalyst so as to minimize the expected average costs (consisting of inventory-holding, backlogging, and catalyst-

switching costs). This is done while making sure that the target attribute level for each batch is set to meet a quality specification represented by an average attribute level across all the batches in a campaign.

Our work was motivated by the batch-production processes at a multinational food-processing corporation and, in particular, the company's production of sorbitol and modified starches. These products, once produced, become raw material inputs into other downstream production processes. These products are commodities—their price and quality specifications are set by the market. The reactor needs to be operated precisely to attain the required quality specification, and production planning needs to be managed carefully to reduce costs. As noted earlier, managing the costs usually entails running a sequence of batches of a single product in a campaign. Although lengthy campaigns increase holding costs and the likelihood of backorder costs, they reduce the switchover costs incurred between campaigns. Because prices are set by the market, profitability results from minimizing the sum of these costs.

The mathematical models presented in this paper are drawn from two examples of catalyst-activated batch-production processes at the company mentioned previously. We chose these examples because these processes were fairly prevalent across this industry, and the products manufactured at these processes contributed to a large fraction of the firm's profits. In the first example, we consider the production of sorbitol manufactured to a specific attribute level in a catalyst-activated reactor. The sorbitol is then used as the raw material in a capital-intensive downstream refining process that purifies the sorbitol to meet the quality specifications of customers from cosmetics and pharmaceutical industries. Because the downstream refinery is operated at a known production rate, the demand for the upstream sorbitol process (which is our focus) can be taken to be equal to this known downstream production rate. Additionally, the backlogging cost for the sorbitol process is specified by the refinery to reflect the cost of not adequately meeting their customer demand. However, these backlogging costs need not capture the very expensive costs of refinery shutdowns. This is because the refinery is designed to "idle" with the work in process if there are temporary shortages in inputs. The single-product version of the model is based on the sorbitol production process. The second example considers the manufacturing of modified starches. Here different types of modified starches are produced one at a time in a single reactor to meet the quality specifications by customers from other food-processing industries, who use these starches as inputs in their manufacturing processes. Here, again, demand for the modified-starch production process is

known by detailed downstream customer contracts, which also specify the per-unit penalty or backlogging costs of unmet demand. The multiproduct version of the model is motivated by the modified-starches production process.

Although sorbitol and modified starch are different products manufactured in two dissimilar catalyst-activated batch-production processes, the goal of the planning model at each process is to determine the duration of the batches and decide when to replenish a catalyst so as to minimize expected costs and meet the quality specification. As we learn about the decay of the productivity of the catalyst and use this in planning, we effectively consider a production campaign planning model under learning and decay. Furthermore, in both these settings, because the downstream processes often have stages with biological or chemical reactions, there is variability in the production rate of these processes (Rajaram et al. 1999). To manage such variability, buffers are installed between the upstream and downstream processes (Rajaram and Tian 2009). When demand in the upstream process is set assuming a constant downstream production rate and when in reality there is variability in this production rate, buffers store the inventory that would exist between the processes in the interim. Thus, in effect, the buffers decouple the impact of production variability in the downstream process from the output of the upstream processes. In light of this, and because demand is known in both the sorbitol and modified-starch processes, we will assume that demand is deterministic in the corresponding models. This assumption could also be reasonable in other process-industry settings, where the outputs of these catalyst-activated batch processes are intermediary products used as inputs to downstream processes, and there are buffers decoupling these processes.

We formulate the single-product batch-production planning problem with learning and decay as a semi-Markov average-cost model. To make this tractable, we develop a two-level heuristic that decomposes and solves this problem in two levels. The lower-level problem plans the duration of batches within the current campaign to maximize the efficiency of the catalyst while satisfying the target attribute level for each batch. The higher-level problem is a binary decision after each batch: whether to end this campaign and switch to a new catalyst. The lower level is formulated as a stochastic dynamic programming problem, similar to the Bayesian decision model by Mazzola and McCardle (1996), which had a production learning curve (as opposed to our decay curve). Despite the similarities of the models, our problem has an additional constraint (the average attribute-level constraint), which adds one more

dimension to the state space. Therefore, we adopt a reoptimization policy that relies on learning to take care of itself and show that it has near-optimal performance. The higher-level problem is to design a control policy mapping the state space to a binary control variable—changing the catalyst or continuing. The objective is to minimize the average costs (switchover, backlogging, and inventory). The state space consists of the current inventory level, current consumption of the catalyst (measured by the total time the current catalyst has actively been used to produce batches), and the current belief regarding the catalyst productivity parameter. Solving for an optimal mapping from the state space to the decision variable is intractable. We propose a policy to approximately solve this problem. To evaluate the performance of the two-level heuristic, we obtain a lower bound on the optimal performance of the original integrated decision process. This bound simultaneously accounts for costs, randomness, and the discrete nature of the process. We also compare the performance of our heuristic with the fixed-cycle policy that is currently practiced by this company.

We then extend our model to consider the multiproduct case with uncertainty in production times. This can be regarded as a stochastic economic lot scheduling problem (SELSP) with fixed batch sizes, learning, and decay. Because of the complexity of this problem, we solve this using a dynamic two-level heuristic. Vaughan (2007) compares dynamic versus cyclic policies for SELSP problems and shows that, in many cases, cyclic policies perform better than dynamic policies. However, in our context, cyclic policies suffer from delayed reaction toward backlogged demand. Therefore, a dynamic policy is needed to recognize and attend to the critical product that would otherwise cause large backlogging costs. The multiproduct version of the two-level heuristic is benchmarked with a practitioner's heuristic used by this company. We also develop a lower bound on this problem to evaluate the performance of these heuristics.

The remainder of this paper is organized as follows. In the next section, we review the relevant literature and state our contributions. We then formulate the single-product problem as a semi-Markov decision process (SMDP) in Section 3, followed by lower bounds on the optimal cost of the SMDP in Section 4. In Section 5, we present our solution methodology based on a two-level heuristic that involves decomposing the problem into two levels of decision making. To benchmark the two-level heuristic, we also provide a practitioner's heuristic that is currently employed at this company. We extend our methodology to the multiple-product case in Section 6. In Section 7, we compare the performance of the heuristics and lower bound on real factory data and

present managerial insights. Conclusions and future research directions are provided in Section 8.

2. Literature Review

The literature related to our work can be classified by problem type: single product and multiproduct. The single-product problem is related to the single-item SELSP. The literature considers uncertainty in this problem owing to demand (Federgruen and Katalan 1996), yield (Grasman et al. 2008), and quality (Papachristos and Konstantaras 2006). Levi and Shi (2013) review a wide selection of the single-item SELSP literature, many of which show that (s, S) policies are optimal under specific problem configurations. In our problem, (s, S) policies are not feasible because, depending on the current catalyst productivity and the decaying nature of the catalyst, it may not be possible to produce up to S . Another important feature of our problem is that we learn about the current productivity of the catalyst by observing the performance in previous batches. This learning is used to predict catalyst performance and production times of future batches.

The relevant literature for multiproduct models includes work on the SELSP (Winands et al. 2011) and economic lot-sizing models under production decay in the chemical engineering literature (Casas-Liza et al. 2005, Liu et al. 2014, Vieira et al. 2017). An important dimension in this paper is that it incorporates production decay into the SELSP in both the single and multiproduct problems, an aspect not considered in the operations management literature. In addition, this literature has not considered production decay, even in the deterministic economic lot-sizing problem (ELSP). Although the chemical engineering literature has considered the deterministic ELSP with production decay, this stream does not consider uncertainty in the decay process and Bayesian learning to use this aspect in production campaign planning. Furthermore, this work does not address the structural properties of this problem, and the attempted problems are tractable enough to be solved with off-the-shelf mixed-integer programming methods. To the best of our knowledge, ours is the first paper to consider all of this together in a SELSP with uncertain production decay.

In terms of solution methods for the SELSP, dynamic approaches to solve the SELSP include Rajaram and Karmarkar (2002), Dusonchet and Hongler (2003), and Wang et al. (2012). In terms of application, Rajaram and Karmarkar (2004) also consider a campaign-planning problem applied to the food-processing industry. None of the methods used in these papers can be used to solve the problem considered in this paper because of batching, learning across batches, and decay in performance of the catalyst.

In this context, our paper makes the following contributions. First, to the best of our knowledge, this is the first paper to consider the campaign-planning problem under production time uncertainty, learning, and decay. As discussed previously, this is an important problem that has not been adequately studied in the academic literature. Second, we formulate this problem as a semi-Markov decision process, incorporating key aspects of this problem that include uncertainty in production time, learning about productivity characteristics, and decay in catalyst performance. Third, we develop efficient, near-optimal solution methods to solve this problem. In addition, our approach to find lower bounds by associating a continuous-state-space dynamic programming problem with a similar but regenerative process can be applied to other stochastic dynamic programming problems. Fourth, we validate our model and solution methods with real data from the process industry. Fifth, we provide several insights that could be useful for practitioners in other industries with a similar production setting.

3. Model Formulation

Consider a batch-production process in which the present state of the production system is defined by the current inventory level and the state of the catalyst currently in use (to be explained in further detail). Based on this information, the firm must decide whether to replace the current catalyst and start a new campaign or to produce another batch with the current catalyst and determine the attribute level of the next batch. We denote the current inventory level as I and treat it as a continuous variable. For ease of exposition and without loss of generality, we measure inventory as batches and assume that the batch size is equal to 1. That is, inventory replenishes by discrete counts but depletes continuously with a constant per-unit-time demand. We define the following parameters and variables:

Parameters:

d : constant demand rate (batches/unit time).

C_B : backlog cost (\$/batch/unit time).

C_I : finished product inventory holding cost (\$/batch/unit time).

C_S : cost of changing a catalyst (\$).

C_∞ : cost of ending a campaign prematurely (\$).

t_s : switchover time required to change a catalyst (unit time).

q_i^0 : initial attribute level of batch i (normalized, dimensionless stochastic parameter).

\mathbf{q}^0 : vector of initial attribute levels of all batches produced by a catalyst.

Stochastic variables:

b : inverse productivity parameter of a catalyst.

z_i : random shock observed by the catalyst while producing batch i .

$\mathbf{z} = [z_1, z_2, \dots, z_N]$: combining the z_i values of a campaign into one vector.

Decision variables:

N : number of batches produced by a catalyst—that is, the length of a campaign.

q_i : the target attribute level of the i th batch in a campaign (normalized, dimensionless).

The decision variables imply the following intermediary variables:

$\mathbf{q} = [q_1, q_2, \dots, q_N]$: vector of target attribute levels of all batches produced by a catalyst.

$Q_i = \sum_{j=1}^{i-1} q_j$: sum of all target attribute levels up to the beginning of batch i .

t_i : time spent by batch i in the reactor (unit time/batch).

$T_i = \sum_{j=1}^{i-1} t_j$: total consumption of the catalyst up to batch i (unit time).

We define the following functions:

$\gamma(b)$: a density function representing the current belief distribution over possible values of the parameter b . The initial prior belief is $\gamma_0(b)$.

$\tau(\mathbf{q}, \mathbf{q}^0, b, \mathbf{z})$: time spent on a catalyst with production schedule \mathbf{q} , initial attribute levels \mathbf{q}^0 , and inverse productivity b . Takes random values based on the realizations of the random shocks z_i (unit time).

$\tau^*(N)$: expected optimal time it would take to produce N batches in a campaign, given full information on b and \mathbf{z} (unit time).

$g(I, \tau)$: total inventory-holding and backlogging cost during time length τ , where the inventory level starts at I and ends at $I - \tau d$ (unit cost).

In catalyst-activated batch processes, t_i , the processing time for batch i , is determined by the consumption of the catalyst, the productivity of the catalyst, and the target attribute level (Steinfeld et al. 1989). To find the exact form of t_i , one needs to solve the differential equation defining the chemical reaction. Employing the formulation by Steinfeld et al. (1989), we get

$$t_i = k(T_i)(b + z_i)f(q_i). \quad (1)$$

Here $k(T_i)$ is a monotone increasing function of the total consumption of the catalyst T_i . The inverse productivity $b + z_i$ consists of two components: The first component b is the catalyst-specific inverse productivity, which, for each catalyst, is drawn from a known distribution with mean μ_b and variance σ_b^2 ; it only takes positive values. The second component z_i is the batch-specific random shock, representing the randomness in the initial conditions of the reaction; these are independent and identically distributed variables with mean zero and a known distribution. We assume that $b + z_i$ is always positive. Finally, $f(q_i)$ is a convex decreasing function capturing the structure of the chemical reaction. Given $f(\cdot)$, we can use (1)

to express the target attribute level of a batch i as a function of the time t_i it spends in the reactor as

$$q_i = f^{-1}\left(\frac{t_i}{k(T_i)(b + z_i)}\right). \quad (2)$$

For example, if $k(T_i) = T_i + 1$ and $f(q_i) = -\ln(q_i/q_i^0)$ (Weekman 1968), the relation between q and t would take the form

$$q_i = q_i^0 \exp\left(-\frac{t_i}{(T_i + 1)(b + z_i)}\right). \quad (3)$$

The decision to be made prior to placing the next batch inside the reactor is to either choose a target attribute level q_{n+1} for the next batch or to end the current campaign and replace the catalyst. We do not allow the option of removing a batch before meeting the target attribute level.

The belief distribution $\gamma(b)$ is updated by observing the pair $(t_i, q_i)|T_i$ after each batch. At the end of the campaign and after replacing the catalyst, the new inventory level will be

$$I' = I - (\tau(\mathbf{q}, \mathbf{q}^0, b, \mathbf{z}) + t_s)d + N. \quad (4)$$

Because of the randomness of b and z_i , the new state (I') will be a random function of the old state (I) and the vector of actions q_i summarized in \mathbf{q} . The campaign time $\tau(\mathbf{q}, \mathbf{q}^0, b, \mathbf{z})$ is also a random function of \mathbf{q} . The cost of this transition is the inventory and backlogging costs during the time $\tau(\mathbf{q}, \mathbf{q}^0, b, \mathbf{z}) + t_s$ plus the cost of changing the catalyst. We assume that backlogging costs occur if the total demand during the catalyst lifespan (i.e., $(\tau(\mathbf{q}, \mathbf{q}^0, b, \mathbf{z}) + t_s)d$) exceeds the initial inventory level I . Furthermore, in these processes, the convention is to assume that the work-in-process inventory holding costs are constant across all process stages, and they are regarded as sunk costs. However, once the batches are pooled and converted to finished products, the holding costs for the product increases, and this is assessed on a per-batch basis. We adopt this convention and define C_I as the holding costs per batch of finished goods inventory. That is, we only consider the inventory holding cost for the pooled batches constituting the finished goods inventory.

Let the function $g(I, \tau)$ represent the total inventory and backlogging costs during a time span of τ when the starting inventory is I and no batches are added to the inventory during τ . To derive $g(I, \tau)$, we need to consider three cases. First, if we do not run out of inventory during time τ , we incur only inventory-holding costs, which are proportional to the average inventory $(I - \tau d/2)$ multiplied by the duration of the time horizon (τ). Second, if we start from a positive inventory level but run out during time τ , we incur

both inventory and backlogging costs. The length of time with positive inventory is I/d , and the average inventory level during this time is $I/2$. The final inventory is $I - \tau d$; hence, the length of time with backlogging is $(I - \tau d)/d$, and average backlogging during this time is $(I - \tau d)/2$. Finally, if we start from a negative inventory, the only cost during τ will come from backlogging and is computed similarly to the case where we only incur inventory holding cost. Thus, $g(I, \tau)$ is computed as

$$g(I, \tau) \triangleq \begin{cases} \tau(I - \tau d/2)C_I & \text{if } I - \tau d \geq 0 \\ I^2/2d \times C_I - C_B & \text{if } I \geq 0, I - \tau d < 0 \\ (I - \tau d)^2/2d & \\ -\tau(I - \tau d/2)C_B & \text{if } I < 0. \end{cases} \quad (5)$$

In order to define the objective function, we first define the term *cycle*. A cycle refers to the length of time between the ending of two subsequent campaigns. A cycle of duration τ_j consists of a campaign with length $\tau_j^c = \tau(\mathbf{q}_j, \mathbf{q}_j^0, b_j, \mathbf{z}_j) + t_s$ and an idle time τ_j^0 before setting up the campaign (idle times are allowed to balance inventory with demand and avoid accumulation of inventory). During a cycle, no batches are added to the inventory; hence, the realized cost during cycle j is $g(I_j, \tau_j)$, where the cycle starts at inventory I_j . This problem can be formulated as a semi-Markov average-cost problem with transition cost $g(I_j, \tau_j)$. The cost function for the average-cost problem is

$$\lim_{R \rightarrow \infty} E \left\{ \frac{\sum_{j=1}^R [C_s + g(I_j, \tau_j)]}{\sum_{j=1}^R \tau_j} \right\}. \quad (6)$$

We formulate the problem of minimizing (6) as a Bayesian stochastic dynamic program. A control policy maps the state space to a decision of either choosing a target attribute level q_{n+1} for the next batch or ending the current campaign. The state space consists of the current inventory level (I), number of batches produced so far in the current campaign (n), current belief distribution $\gamma(b)$, the total consumption of the current catalyst (T), and the cumulative attribute level of the n batches produced in the current campaign (Q). For simplicity and conformance to practice, we allow idle time periods after ending campaigns and immediately prior to setting up campaigns. We consider two types of states. The first is when a campaign is in process, whereas the second is when a campaign has finished and the next campaign has not yet been set up. Denote the differential costs of the first state by $h(I, n, Q, T, \gamma(b))$ and the differential costs of the second state by $w(I)$. The optimal average cost of the problem is denoted by λ^* ,

which is treated as a variable in the Bellman equation. The SMDP is formalized as follows:

$$\begin{aligned} (\text{SMDP}) \quad & h(I, n, Q, T, \gamma(b)) \\ & = \min\{C_s + C_\infty I(Q > n) + \min_{t' \geq 0} [w(I + n - t'd) \\ & \quad + g(I, t') - \lambda^* t']\}, \min_{q_{n+1}} E_{t_{n+1}|q_{n+1}} [h(I - t_{n+1}d, n + 1, \\ & \quad Q + q_{n+1}, T + t_{n+1}, \gamma'(b)) + g(I, t_{n+1}) - \lambda^* t_{n+1}]; \\ w(I) = & \min_{t \geq t_s} [h(I - td, 0, 0, 0, \gamma_0(b)) + g(I, t) - \lambda^* t]. \quad (7) \end{aligned}$$

Here $\gamma_0(b)$ is the prior belief distribution on b (the distribution from which b is drawn), and $\gamma'(b)$ is the belief distribution over b after observing the next batch. The duration of the next batch, denoted by t_{n+1} , is random and depends on the target attribute level q_{n+1} . The first term in the minimization represents the decision to switch to a new catalyst with the option to allow an idle time of t' . Here the term $C_\infty I(Q > n)$ disallows switching if the attribute-level constraint is not met (by penalizing it with a sufficiently large C_∞). The second term represents the decision to produce another batch with the current catalyst, in which case the next target attribute level q_{n+1} must also be chosen.

As a consequence of the curse of dimensionality, this problem is too complex to solve analytically. Therefore, in Propositions 1–4, we derive properties of the SMDP. Specifically, Propositions 1 and 2 will be used in Section 4 to construct a lower bound for the SMDP, whereas Propositions 3 and 4 are used in developing the two-level heuristic described in Section 5.1.

Proposition 1. *The differential cost function $w(I)$ has a global minimizer \bar{I}^* .*

All proofs are provided in Section A of the online appendix. According to Proposition 1, the level \bar{I}^* is the ideal inventory to have at the beginning of a cycle. However, it is not necessarily optimal to immediately start the next campaign at \bar{I}^* , and we might allow some idle time. This is shown in Proposition 2.

Proposition 2. *There exists an inventory level $I_0^* \leq \bar{I}^*$ such that if the inventory level at the beginning of a cycle is \bar{I}^* , it is optimal to delay the campaign setup until inventory falls to I_0^* .*

For the purpose of the next two propositions, we define four new parameters $I_\lambda, \underline{I}^*, \bar{N}$, and N_0 , which are calculated using the main parameters of the model. The significance of these parameters will become clear in the following propositions and the discussion following Proposition 4. Define I_λ as the nonpositive solution to the equation $dg/dt(I_\lambda, 0) = \lambda^*$. Next, define \bar{N} and N_0 as the smallest integers satisfying $\bar{I}^* - \bar{N} < I_\lambda$ and

$I_0^* - N_0 < I_\lambda$, respectively. Finally, define $0 \leq \underline{I}^* \leq I_\lambda$ as the solution to the equation $dg/dt(\underline{I}^* + N_0, 0) = dg/dt(\underline{I}^*, 0)$.

Proposition 3. *If in the current state $Q < n = \bar{N}$, then $I \geq \bar{I} - \bar{N}$ is a sufficient optimality condition to end the campaign, irrespective of all other state variables.*

Corollary. *If in the current state $Q < n = \bar{N}$, then both $I \geq I_\lambda$ and $I \geq \underline{I}^*$ are sufficient optimality conditions to end the campaign. This results from the fact that $I \geq I_\lambda$ and $I \geq \underline{I}^*$ both imply $I \geq \bar{I} - \bar{N}$.*

Proposition 4. *Given $I \geq \underline{I}^*$ and $n \geq N_0$, if all but one of the state variables I , Q , T , and $\gamma(\cdot)$ are fixed, there exists a probability threshold Ψ such that it is optimal to end the current campaign if and only if (iff) $P(I' < \underline{I}^*) \geq \Psi$.*

Proposition 4 shows that when $I \geq \underline{I}^*$ and enough batches have been produced ($n \geq N_0$), the variable $P(I' < \underline{I}^*)$ contains sufficient information to replace any one of the variables I , Q , T , or $\gamma(\cdot)$ individually in the Bellman equation. We will use Propositions 3 and 4 in developing the two-level heuristic described in Section 5. In the next section, we present a procedure to compute lower bounds on the optimal average cost of the SMDP, which will be used to evaluate the performance of our two-level heuristic. Some results from the lower-bound analysis are also used to develop this heuristic.

4. Lower Bounds

To compute a lower bound on the optimal average cost of the SMDP, we first consider a deterministic version of the problem and compute an associated lower bound. This lower bound does not consider the cost of randomness and discrete production and is usually a loose bound. However, we consider this bound for two reasons: (1) the resulting (loose) lower bound is used to construct a tighter stochastic lower bound, and (2) this tractable model structure and the respective insights are used to construct our heuristic solution in Section 5. We then present a stochastic lower bound that accounts for discrete production and the randomness of the process. This bound resolves the inadequacies of the deterministic bound, which result from ignoring discreteness and randomness, but still ignores the uncertainty on production parameters and assumes perfect knowledge of the (randomly) realized catalyst productivity of each batch; in other words, it assumes that we can optimally exploit the productivity of a catalyst as if we had full information. Such clairvoyant bounds have been used in the stochastic programming literature (Ciocan and Farias 2012). Clairvoyant bounds underestimate optimal costs because they assume more accurate learning than is actually possible

(Brown and Smith 2013). However, we found in our computational analysis that this bound performed quite well in our problem context because we have considered randomness and discrete production of the process in computing this bound.

4.1. Deterministic Lower Bound

The deterministic version of the problem is formed by assuming that a campaign with N batches always takes a deterministic amount of time equal to $\tau^*(N)$, where $\tau^*(N)$ is the expected optimal time it would take to produce N batches in a campaign, given full information on b and \mathbf{z} or, formally,

$$\tau^*(N) = E_{\mathbf{q}^0, b, \mathbf{z}} \min[\tau(\mathbf{q}, \mathbf{q}^0, b, \mathbf{z})]. \quad (8)$$

To relax the integer constraint on N and allow continuous production, we define $\tau^*(N)$ for noninteger values of N by a weighted average of the production time of $\lceil N \rceil$ and $\lfloor N \rfloor$, the two closest integers to N :

$$\tau^*(N) \triangleq (\lceil N \rceil - N)\tau^*(\lfloor N \rfloor) + (N - \lfloor N \rfloor)\tau^*(\lceil N \rceil) \\ N \text{ non integer.} \quad (9)$$

To see the reasoning behind Equation (9), note that one way to produce N batches per campaign on average is to produce $\lfloor N \rfloor$ batches in $(\lceil N \rceil - N)$ fraction of the campaigns and $\lceil N \rceil$ batches in $(N - \lfloor N \rfloor)$ fraction of the campaigns, leading to an average time of $(\lceil N \rceil - N)\tau^*(\lfloor N \rfloor) + (N - \lfloor N \rfloor)\tau^*(\lceil N \rceil)$ per campaign. The function $\tau^*(N)$ is piecewise linear and convex. It is convex because for an integer N , the following inequality holds as a result of decaying productivity:

$$\tau^*(N+1) - \tau^*(N) \geq \tau^*(N) - \tau^*(N-1). \quad (10)$$

Let T_{cyc} be the total length of a cycle, including the idle time and the campaign time.

Proposition 5. *The following economic production quantity (EPQ) problem provides a lower bound on λ^* , the optimal average cost of the SMDP:*

$$[\text{EPQ}] \quad \lambda_{\text{EPQ}} = \min_{T_{\text{cyc}}, \bar{I}} \frac{C_s + g(\bar{I}, T_{\text{cyc}})}{T_{\text{cyc}}} \\ \text{s.t. } \tau^*(T_{\text{cyc}}d) + t_s \leq T_{\text{cyc}}. \quad (11)$$

Here the objective of the EPQ is to minimize the average cost during a fixed cycle. The decision variables T_{cyc} and \bar{I} denote the length and the starting inventory of the cycle, respectively. The total cost during a cycle is equal to a one-time switching cost plus inventory holding and backlogging costs during the cycle ($g(\bar{I}, T_{\text{cyc}})$). A total of $T_{\text{cyc}}d$ batches are produced in a campaign to match the total demand during the length of the cycle. The constraint ensures

that the total time required to produce $T_{cyc}d$ batches (i.e., $\tau^*(T_{cyc}d)$) plus the setup time is less than or equal to the length of the cycle. We define the following parameters based on the solution to (11):

T_{cyc}^* : optimal cycle length in (11).

T^M : largest cycle length satisfying the production constraint in (11).

T^m : smallest cycle length satisfying the production constraint in (11).

$N^* \triangleq T_{cyc}^*d$: total demand during T_{cyc}^* . It is similar to the optimal order quantity Q^* in an unconstrained economic order quantity model.

\bar{I} : optimal inventory at the beginning of each cycle (i.e., after adding to inventory the batches produced in the previous campaign).

$\underline{I} \triangleq \bar{I} - N^*$: lowest inventory reached at the end of a cycle, just before the newly produced batches are added.

$C_{IB} \triangleq C_I C_B / (C_I + C_B)$: “balanced” inventory holding and backlogging costs per unit time in an EPQ with backlogging (will be discussed shortly).

C_s^0 : the supremum value of all C_s such that the constraint in (11) is not binding at the optimal solution (will be discussed shortly).

Given a cycle length T_{cyc} , inventory will be positive for a fraction of the cycle, and for the remaining time, inventory will be negative. We show that the optimal fraction of time where inventory is positive and where it is negative are proportional to C_B and C_I , respectively.

Proposition 6. *The optimal \bar{I} can be derived as a function of T_{cyc} and replaced in the objective function. The resulting objective function is convex in T_{cyc} .*

Corollary 6.1. *Problem EPQ becomes a convex optimization problem in the single variable T_{cyc} .*

In the proof of Proposition 6 (in the online appendix), we see that the objective over T_{cyc} becomes $C_s/T_{cyc} + C_I C_B / (C_I + C_B) T_{cyc} d / 2$, which is similar to an EPQ without backlogging in which $C_{IB} \triangleq C_I C_B / (C_I + C_B)$ has replaced the inventory holding cost. The parameter C_{IB} is interpreted as the balanced inventory cost (holding and backlogging) per unit time when the cycle length is optimally allocated between positive and negative inventory. The optimal production quantity in the unconstrained EPQ problem is $N^u = \sqrt{2dC_s}$.

If $T_{cyc}^u = N^u/d$ satisfies the production constraint, then the constraint is not binding, and T_{cyc}^u is optimal for problem (11). In this case, the following relations would hold:

$$N^* = N^u = \sqrt{\frac{2C_s d}{C_{IB}}}, \quad \bar{I} = \frac{C_I}{C_I + C_B} N^*,$$

$$\lambda_{EPQ} = \sqrt{2C_{IB} C_s d} = \underline{I} C_B = N^* C_{IB}. \quad (12)$$

In order for T_{cyc}^u to be feasible (hence, optimal), we must have $T^m \leq T_{cyc}^u \leq T^M$. If T_{cyc}^u does not satisfy the constraint, then the constraint is binding at the optimal T_{cyc}^* because, by Proposition 6, the objective function is convex in T_{cyc} . If $T_{cyc}^u < T^m$, then $T_{cyc}^* = T^m$, and if $T_{cyc}^u > T^M$, then $T_{cyc}^* = T^M$.

4.2. Stochastic Lower Bound

The solution to the EPQ problem gives a lower bound for the original problem, but it does not account for the cost of randomness and discrete production. To obtain a tighter bound, we use the EPQ problem and define a stochastic process that is similar to the actual production process but is regenerative and, hence, more tractable. The optimal cost of this regenerative process is a tighter bound than the EPQ bound for the original problem.

Recall from Propositions 1 and 2 that an ideal process is one that begins every cycle at \bar{I}^* and sets up every campaign at $I_0^* \leq \bar{I}^*$. Based on this, we define a regenerative process that always starts a campaign at I_0^* and allows free disposal so that inventory can instantly drop to the ideal level \bar{I}^* if it exceeds it. We then use Propositions 1 and 2 to establish (in Proposition 7) that the optimal cost of this regenerative process provides a lower bound on the optimal cost of the original process.

We define the regenerative process by modifying the original process as follows:

1. Production always starts at I_0^* .
2. If the inventory level I at the end of a campaign is less than I_0^* , the inventory level is instantly raised to I_0^* at a cost of $\lambda_{EPQ} t_{|I|} - g(I_0^*, t_{|I|})$, where λ_{EPQ} is the optimal cost of problem (11) and $t_{|I|} = (I_0^* - I)/d$.
3. If the inventory level I at the end of a campaign is greater than I_0^* , the inventory level is instantly dropped to a level I' of choice ($I \geq I' \geq I_0^*$), and then production is idled until inventory reaches I_0^* .

Proposition 7. *There exists a regenerative process satisfying 1–3 that has a lower average cost than the optimal cost of the original process. As a result, the optimal average cost of the regenerative process is less than the optimal average cost of the original process.*

In the regenerative process, all campaigns start at I_0^* ; hence, it is more tractable than the original process. However, computing the optimal cost of the regenerative process is still not straightforward. In Section B of the online appendix, we present an algorithm to compute a lower bound on the optimal performance of the regenerative process given I_0^* . A lower bound on the optimal cost of the original process then can be obtained by a line search over values of I_0^* .

5. Heuristics and Upper Bounds

In this section, we present a two-level heuristic to solve the SMDP. This provides an upper bound on the

value of the SMDP. To benchmark the performance of this heuristic, we also describe a practitioner’s heuristic currently employed at a large food-processing company.

5.1. Two-Level Heuristic

In the two-level heuristic, we decompose the campaign-planning problem with learning and decay represented by the SMDP into two levels: a lower-level problem at the batch level, determining the duration of batches in a campaign, and a higher-level problem at the campaign level, determining when to end a campaign. These levels are defined as

- Level 1: Batch planning—Choose the attribute level q_{n+1} of the next batch.
- Level 2: Catalyst switching—While batch n is inside the reactor, use a control policy to decide whether to change the catalyst after this batch or move on to batch $n + 1$.

We next describe the solution method for each level.

Level 1: Batch Planning. In batch planning, we focus on minimizing the campaign duration for a given number of batches N . We minimize the campaign duration because it offers more buffer time and, hence, more flexibility. The objective is to minimize $T_N + t_N$ such that the average attribute-level constraint is satisfied. The number of batches N is tentatively chosen as the expected number of batches in the campaign, which depends on the higher-level policy discussed in the next subsection. For ease of exposition and without loss of generality, the attribute level is normalized such that the required average attribute level is less than or equal to 1. Hence, the average attribute-level constraint becomes

$$\sum_{i=1}^N q_i \leq N. \quad (13)$$

With this constraint, the state space should include the cumulative attribute level. After completion of batch $i - 1$, a decision q_i is made for the i th batch, given the current state of the campaign $[Q_i, T_i, \gamma(b)]$. The resulting stochastic dynamic programming (DP) is represented by the following Bellman equation:

$$\begin{aligned} v_i(Q_i, T_i, \gamma(b)) &= \min_{q_i} [E_{z_i, b} \{ (b + z_i)k(T_i)f(q_i) \\ &+ v_{i+1}(Q_i + q_i, T_i + (b + z_i)k(T_i)f(q_i), \gamma'(b)) \}], \end{aligned} \quad (14)$$

where $\gamma(b)$ is the current belief distribution on the parameter b , and $\gamma'(b)$ is the updated belief after observing the random outcome of $b + z_i$. We use a

standard Bayesian updating procedure to get $\gamma'(b)$: each period based on our observation of the pair (q_i, t_i) , we observe the implied productivity $b_i \triangleq b + z_i$ through $t_i = b_i k(T_i)f(q_i)$ or equivalently $b_i = t_i / k(T_i)f(q_i)$.

We use this observation along with the known density of z_i (i.e., $\kappa(z_i)$) to update $\gamma(b)$ according to $\gamma'(b) \propto \gamma(b)\kappa(b_i - b)$. Here the current belief $\gamma(b)$ acts as the prior belief, and the new probability density for b , according to Bayes’ rule, is proportional to the previous density multiplied by the likelihood of observing b_i conditional on b (i.e., $\kappa(b_i - b)$).

An analytical closed-form solution to this problem is not available in general, and the continuous multidimensional state space of this stochastic Bayesian DP problem makes it intractable to numerically find an optimal policy (Mazzola and McCardle 1996). Hence, we approach this problem with the following reoptimization heuristic: first, we find the vector \mathbf{q} that minimizes the expected campaign length $E_{q^0, b, z} \{ \tau(\mathbf{q}, \mathbf{q}^0, b, z) \}$, ignoring the fact that \mathbf{q} can be adjusted in the future. After the first batch q_1 is completed, we incorporate learning by updating $\gamma(b)$ and reoptimizing the remaining batches 2 through N of \mathbf{q} . We then implement the revised second batch and repeat.

The reoptimization policy does not directly anticipate the value of information in its decision process. In Section C of the online appendix, we set up a tractable three-batch example and compare the reoptimization policy with the optimal policy. We make three important observations. First, in all problem settings considered, the choice of q_1 under the reoptimization policy is very close to the optimal choice of q_1 . Second, the resulting average campaign time is almost identical under the two policies (they are within 1% of each other in all problem settings considered). Third, both policies attain near-optimal exploitation of the catalyst, as if full information on b were available a priori (in every problem setting, the expected campaign duration under the two policies is within 2% of the minimum campaign duration). In addition, our computational results in Section 7 show that our two-level heuristic achieves near-optimal costs, implying that the reoptimization policy adequately captures the dynamic value of information in this problem.

With the reoptimization policy in place, we are interested in understanding how the chosen \mathbf{q} will differ from the myopic policy of having every batch meet the target attribute level (i.e., $\mathbf{q} = [1, 1, \dots, 1]$). Note that a smaller q_i implies that batch i has a bigger contribution to satisfying the average attribute-level constraint (13). Therefore, we interpret assigning a smaller q_i to batch i as placing a higher “load” on batch i . We are interested in whether it is better to place higher load (i.e., smaller q_i) on the production of the

first batches (while the productivity of the catalyst is still high) and leave a lower load on the concluding batches or, conversely, place a lighter load on the initial batches to avoid an overly decayed catalyst when it reaches the final batches. As shown in Proposition 8, this depends on the form of $k(T_i)$, the function that defines the dependency of the productivity decay rate (equivalently, the rate of increase in processing time) on the total consumption. Denote $\mathbf{q}^* = [q_1^*, \dots, q_N^*]$ as the solution to $\arg \min_{\mathbf{q}} E_{\mathbf{q}^0, b, \mathbf{z}} \{\tau(\mathbf{q}, \mathbf{q}^0, b, \mathbf{z})\}$.

- Proposition 8.** (i) If $k(T)$ is convex in T , then there exists at least one optimal solution where $q_i^* \geq q_{i+1}^*$. Further, if $k(T)$ is strictly convex, there are no optimal solutions where $q_i^* \leq q_{i+1}^*$.
 (ii) If $k(T)$ is concave in T , then there exists at least one optimal solution where $q_i^* \leq q_{i+1}^*$. Further, if $k(T)$ is strictly concave, there are no optimal solutions where $q_i^* \geq q_{i+1}^*$.
 (iii) If $k(T)$ is an affine function, then $q_i^* = q_{i+1}^*$ for all i .

Proposition 8 establishes that a higher load should be allocated to the finishing batches if $k(T)$ is a convex function and to the beginning batches if $k(T)$ is a concave function. To see the intuition behind this finding, consider a two-batch example ($N = 2$). When $k(T)$ is convex, the decay rate is increasing in total consumption; thus, if a high load is placed on the first batch, the second batch will face an overly decayed catalyst, increasing the total time required to process the two batches. The reverse is true for a concave $k(T)$.

Figure 1 compares $\mathbf{q}^* = \arg \min_{\mathbf{q}} \tau(\mathbf{q}, \mathbf{q}^0, b, \mathbf{z})$ with $\mathbf{q} = [1, \dots, 1]$ for a convex $k(T)$ and fixed \mathbf{q}^0, b , and \mathbf{z} .

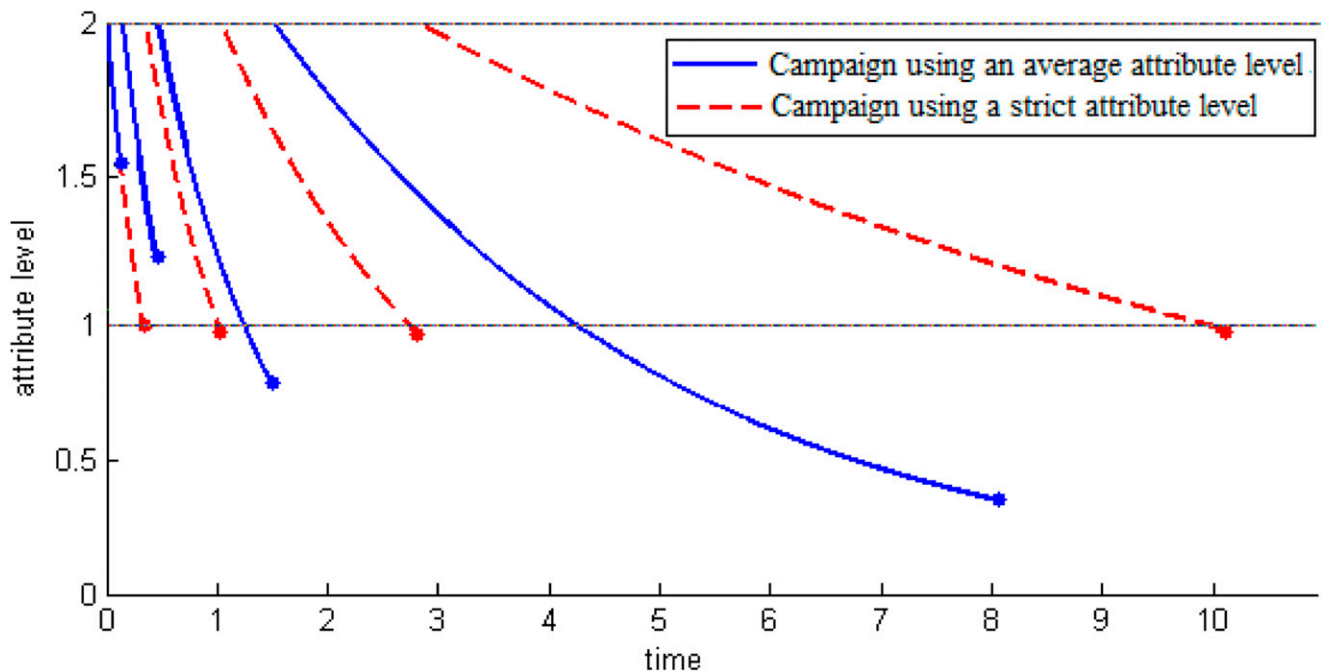
Each downward slope represents one batch, where the attribute level is reduced from an initial level. Each upward jump represents removing a batch and placing another batch inside the reactor. In this example, the optimal campaign that meets the quality specification on average across the batches in a campaign is 20% faster than the campaign that strictly attains the quality specification in all batches.

Proposition 8 states that for convex (concave) $k(T)$, the relation $q_1^* \geq q_2^*$ ($q_1^* \leq q_2^*$) holds. We were also interested in evaluating the dependence of this ordering on the level of uncertainty in b . For two distributions $\gamma(\cdot)$ and $\tilde{\gamma}(\cdot)$ with equal mean μ , $\tilde{\gamma}(\cdot)$ is considered stochastically more variable than $\gamma(\cdot)$ if the inequality $E_{b \sim \tilde{\gamma}(\cdot)}[y(b)] \geq E_{b \sim \gamma(\cdot)}[y(b)]$ holds for any convex function $y(b)$. The following proposition shows that for $N = 2$, a stochastically more variable prior belief on b results in a greater difference between q_1 and q_2 .

Proposition 9. Let $\gamma(\cdot)$ and $\tilde{\gamma}(\cdot)$ be two prior beliefs on b with equal mean μ , where $\tilde{\gamma}(\cdot)$ is stochastically more variable than $\gamma(\cdot)$. Let $\mathbf{q} = [q_1, q_2]$ and $\tilde{\mathbf{q}} = [\tilde{q}_1, \tilde{q}_2]$ be the respective two-batch optimal attribute-level vectors. The following inequality holds: $|\tilde{q}_1 - \tilde{q}_2| \geq |q_1 - q_2|$.

Proposition 9 informs us that the optimal spread between the attribute levels in a campaign increases when there is more uncertainty in the prior distribution of catalyst productivity. The intuition behind this finding complements that of Proposition 8. Here, for a convex catalyst decay function $k(T)$, we would

Figure 1. (Color online) Duration of Batches in a Reactor for Campaigns with Strict and Average Attribute Levels in a Catalytic Reaction Characterized by $q_i = 2 \exp(-2(T + 1)^{-2.3} t_i / b)$



place a lower load (or higher q_1) on the beginning batch so that the final batch starts at a lower catalyst consumption. However, with increasing uncertainty in b , the risk of starting the final batch at a high catalyst consumption increases. Thus, it is optimal to reduce the risk by further decreasing the target load (or increasing q_1) on the first batch. This increases the spread. With similar reasoning, we can show that the optimal spread also increases in the stochastic variability of the prior for concave $k(T)$.

Level 2: Catalyst Switching. In catalyst switching, we develop a policy by which to decide when to switch the catalyst. A control policy maps the state space to a binary decision: switch or don't switch. The state space consists of the current inventory level (I), number of batches produced so far in the current campaign (n), current belief distribution of b ($\gamma(b)$), and the total consumption of the current catalyst (T). The Bellman equation for this average-cost problem is an approximation to the SMDP where the Q dimension is removed from the state space, and the decision variable is binary (instead of being a continuous choice of q_n , which is now handled by the lower-level problem):

$$\begin{aligned} h(I, n, T, \gamma(b)) &= \min\{C_s + C_\infty I(Q < n) \\ &\quad + \min_{t' \geq 0} [w(I + n - t'd) + g(I, t') - \lambda^* t'], \\ &\quad E_{t_{n+1}} [h(I - t_{n+1}d, n + 1, T + t_{n+1}, \gamma'(b)) \\ &\quad + g(I, t_{n+1}) - \lambda^* t_{n+1}]\}; \\ w(I) &= \min_{t \geq t_s} \{h(I - td, 0, 0, \gamma_0(b)) + g(I, t) - \lambda^* t\}. \end{aligned} \quad (15)$$

The state space is multidimensional and has continuous elements. Even if the elements of learning and decay are removed, the state space is still too large to obtain exact solutions (Loehndorf and Minner 2013), even numerically. Hence, we propose an approximate policy to efficiently summarize the large state of the system and apply it in a simple decision rule. For this purpose, we resort to the structural properties of the optimal solution, described in Propositions 3 and 4.

Proposition 3 establishes that if a sufficient number of batches \bar{N} are successfully produced before inventory drops too low, it is optimal to end the campaign irrespective of other state variables. By contrast, Proposition 4 shows that the probability $P(I' \leq \underline{I}^*)$ can individually replace any of the state variables I , Q , T , or $\gamma(\cdot)$ by means of a probability threshold policy. These propositions motivate a policy that targets \bar{N} batches in a campaign and ends a campaign either if this target is reached or if the probability of inventory dropping below \underline{I}^* exceeds a threshold Ψ . Because the exact values of the parameters \bar{N} and \underline{I}^* are not known,

we use corresponding values from the deterministic approximation (EPQ). This heuristic effectively follows as closely as possible the optimal deterministic process suggested by the solution to the EPQ problem. Let \bar{I} and T_{cyc}^* be the optimal solution to (11). Define $\underline{I} \triangleq \bar{I} - T_{\text{cyc}}^* d$ and $I_0 \triangleq \underline{I} + [\tau(T_{\text{cyc}}^* d) + t_s]d$. In the absence of randomness and discreteness, we would like the inventory level at the beginning of a cycle to be \bar{I} , idle the process until inventory reaches I_0 , at which time we set up a campaign and produce batches until inventory is at \underline{I} . The batches produced return inventory to \bar{I} . However, in the discrete-production stochastic process, this does not necessarily happen. At some point, we expect that if we produce another batch, the inventory level at the end of that batch will be lower than \underline{I} , which would result in excess backlogging costs. By contrast, if we do not produce another batch and switch the catalyst before reaching the optimal level of \underline{I} , it would result in higher average switching costs and possibly higher inventory costs in the next cycle.

To choose between these options, we propose to switch before inventory gets to \underline{I} only if the probability of falling below inventory \underline{I} after the next batch is greater than some threshold Ψ . The probability $P[I_{n+1} < \underline{I}]$ is calculated using the distribution of z_n and the current belief over b :

$$\begin{aligned} P[I_{n+1} < \underline{I}] &= P[I - t_{n+1}d < \underline{I}] = P[t_{n+1}d > I - \underline{I}] \\ &= P[(b + z_{n+1})k(T_{n+1})f(q_{n+1})d > I - \underline{I}] \\ &= P\left[b + z_{n+1} > \frac{I - \underline{I}}{k(T_{n+1})f(q_{n+1})d}\right]. \end{aligned} \quad (16)$$

To illustrate the implications of this policy, note that the boundary case of $\Psi = 1$ translates to a case where the catalyst is switched only after inventory falls below \underline{I} . Our numerical results show that if the threshold Ψ is correctly specified, this policy is nearly optimal for many problem instances. The optimal threshold Ψ depends on the cost parameters (C_I , C_B , C_S) and is not analytically computable because it would require solving a Bellman equation almost as big as the original problem (15). Therefore, we choose Ψ by simulating the process and doing a line search over Ψ .

Denote I_n for the inventory level after the production of batch n . Given the threshold, our heuristic separates into the following two cases based on the initial inventory level after the previous campaign (I).

Case 1. $I \geq I_0$. Idle the process until inventory reaches I_0 ; then set up and start the next campaign. Produce batches $1, \dots, n$ until $P[I_{n+1} \leq \underline{I}] \geq \Psi$. For $i \geq n$, end the campaign if and only if (i) $I_i + i \geq I_0$ or (ii) $I_i + i \geq E[I_{i+1}] + i + 1$.

Case 2. $I < I_0$. Set up and start the next campaign with zero idle time. Find the smallest n such that $P[I_{n+1} \leq \underline{I}] \geq \Psi$ and either $E[I_n] + n \geq I_0$ or $E[I_n] + n \geq E[I_{n+1}] + n + 1$. For this n , if the inequality $E[I_{n+1}] + n \geq I_0$ holds,

run the campaign as in Case 1. Otherwise, produce N batches such that N maximizes $N/(\tau^*(N) + t_s)$; this N maximizes the production rate.

The additional conditions in Case 1 enhance the threshold policy by ensuring that the process remains stable after the switch. Condition (i) implies that the next campaign will start above inventory I_0 . If condition (ii) holds, we expect that if we switch now, we start the next campaign with a higher inventory compared with switching after the next batch. In Case 2, we use the same policy as in Case 1 only if we expect that we will be able to start the next campaign from inventory above I_0 . Otherwise, we produce at the maximum production rate to bring the process back up to stable conditions.

5.2. Practitioner's Heuristic

To benchmark the two-level heuristic, we compare it with a practitioner's heuristic that was implemented as part of a broader project described in Rajaram et al. (1999). The decision variables are t^* , how long to leave each batch in the reactor, and N , the maximum number of batches in a campaign. The optimal t^* and N (not necessarily unique) solve the following optimization problem:

$$\min_{N, t^*} \left[C_I \frac{N}{2} + C_S \frac{d}{N} \right] \quad (17a)$$

$$\text{s.t. } \frac{N}{Nt^* + t_s} > d, \quad (17b)$$

$$\sum_{i=1}^N f^{-1} \left(\frac{t^*}{k(it^*)\mu_b} \right) \leq N. \quad (17c)$$

In (17a), the average inventory during a cycle is $N/2$; hence, the average inventory holding cost is equal to $C_I N/2$. The average length of a cycle is N/d ; hence, the average switching cost per unit time is equal to $C_S d/N$. The constraint (17b) ensures that the average production rate exceeds the demand rate. This follows because the choice of $\{N, t^*\}$ implies a production rate of N batches per $Nt^* + t_s$ units of time; hence, for feasibility, $N/(Nt^* + t_s)$ has to be greater than the demand rate d . Constraint (17c) is the attribute-level constraint, approximating $b + z_n$ by the expected value μ_b .

The optimal N and t are found by discretizing t and performing a grid search. However, t^* will not be unique because the objective function of (17) depends only on the variable N . We choose the smallest feasible t^* to increase the production rate, which increases the idle time between campaigns and allows a larger buffer in case of a bad catalyst outcome.

Once the optimal t^* and N are found, the practitioner's heuristic is implemented as follows: set $t_i = t^*$ for all i , and observe the resulting q_i values. From the observed q_i values, update the belief distribution $\gamma(b)$. After batch n , decide whether the catalyst has the

potential to produce another batch with $t_{n+1} = t^*$ while preserving the average attribute-level constraint: $\sum_{i=1}^{n+1} f^{-1}(t^*/(k(it^*)(b + z_i))) \leq n + 1$.

Given the current information, use the expected value of $b + z_n$ (i.e., $E[b|\gamma(b)]$) to evaluate the constraint; produce another batch if and only if $\sum_{i=1}^{n+1} f^{-1}(t^*/(k(it^*)E[b|\gamma(b)])) \leq n + 1$ and $n + 1 \leq N$. If the prediction is wrong and the next batch violates the constraint, stop the current campaign and incur a rework cost. Once the campaign is ended, the catalyst is replaced, and the new batches are released to inventory.

Observe that this heuristic does not allow for deliberate backlogging and is designed for the settings where $C_B \gg C_I$; it does not provide a fair benchmark when C_B and C_I are comparable. To enhance this heuristic and provide a fair benchmark in all settings, we replace the cost term C_I in (17) by $C_{IB} = C_I C_B / (C_I + C_B)$, which represents the optimal balance between inventory holding and backlogging costs (see Section 4.1). To optimally balance these costs, instead of starting and ending the cycles at $I = N$ and $I = 0$, we start and end the cycles at $I = NC_I / (C_I + C_B)$ and $I = -NC_I / (C_I + C_B)$, respectively. Finally, because t^* is chosen as the smallest feasible value for the chosen N , the production rate will be higher than the demand rate. Therefore, idle times are chosen such that the cycles start and finish at these inventory levels.

6. Multiple Products

We now consider a setting where multiple products are produced in a single reactor. Each product has its own fixed demand rate, backlogging costs, and inventory-holding costs. This problem is now a stochastic economic lot-sizing problem with switching costs, batch production, learning, and decay.

During the production of a given campaign, after each batch is produced, a decision must be made: continue this campaign and produce another batch of the current product or finish this campaign, add the produced batches to inventory, and start a new campaign. Note that only one type of product can be produced in a campaign. After a campaign is finished and the produced batches are added to inventory, the next decision is which product to produce next and how much idle time, if any, to allow. These decisions are based not only on the number of batches produced n , the total consumption of the catalyst T , and the current belief distribution $\gamma(b)$ over the inverse productivity parameter b for the current product but also on the inventory level I of all the products. We extend the ideas discussed in the single-product setting to obtain a lower bound and a heuristic for the multiple-product setting.

6.1. Lower Bound

Let R be the total number of products. We modify the previously introduced parameters and variables by

adding indices r (or superscripts, for inventory variables) to denote the product type. A deterministic lower bound on the optimal performance of the stochastic system is to assume that each product undergoes an EPQ process independent of the others, except that the sum of fractions of time that the machine is busy cannot be greater than 1:

$$[\text{MEPQ}] \quad \lambda_{\text{ELSP}} \triangleq \min_{\{N_r, \bar{I}^r\}} \sum_{r=1}^R \frac{C_s + g(\bar{I}^r, N_r/d_r)}{N_r/d_r} \quad (18)$$

$$\text{s.t.} \quad \sum_{r=1}^R \frac{\tau_r^*(N_r) + t_s}{N_r/d_r} \leq 1.$$

We can relax the constraint in (18) using a Lagrangian multiplier δ . This leads to

$$F(\delta) \triangleq \min_{\{N_r, \bar{I}^r\}} \sum_{r=1}^R \frac{C_s + g(\bar{I}^r, N_r/d_r) + (\tau_r^*(N_r) + t_s)\delta}{N_r/d_r} - \delta. \quad (19)$$

The minimization problem decomposes into R separate problems

$$\min_{N_r, \bar{I}^r} \frac{C_s + g(\bar{I}^r, N_r/d_r) + (\tau_r^*(N_r) + t_s)\delta}{N_r/d_r}. \quad (20)$$

Each of these R problems can be transformed into a single-variable problem over N_r . This is done by noting that the optimal fractional allocation of a cycle between positive and negative inventory is fixed, and hence, the cycle length N_r/d uniquely determines \bar{I} . We solve these R single-variable problems separately:

$$\min_{N_r} Z(N_r) = \frac{C_s + C_{IBr}N_r^2/2d_r + (\tau_r^*(N_r) + t_s)\delta}{N_r/d_r}, \quad (21)$$

where $C_{IBr} \triangleq C_{Ir}C_{Br}/(C_{Ir} + C_{Br})$ (this can be shown using a similar logic to the proof of Proposition 6). Because $\tau_r^*(N_r)$ is convex, the numerator is convex. Further, it is well known that if $f(x)$ is convex, then $f(x)/x$ is quasi-convex; thus, $Z(N_r)$ is quasi-convex in N_r . We optimize over N_r by using simple numerical methods, so $F(\delta)$ is easy to evaluate. We maximize the concave function $F(\delta)$ using a golden section search, obtaining a lower bound on the average cost of the deterministic relaxation of the original stochastic problem.

Proposition 10. *Strong duality holds in problem MEPQ.*

Based on Proposition 10, the N_r values obtained by the above procedure are feasible for problem MEPQ. If N_r/d_r is the same for all products, the R products could be functioning as if they were independent EPQ systems. The sum of the average costs of these R systems forms a lower bound on the optimal average cost of the original problem. To improve this lower bound, we use the same procedure presented in Section 4.2

by constructing a separate regenerative process to individually improve the lower bound for each product and compute the sum of the improved costs.

6.2. Heuristics

6.2.1. Multiproduct Two-Level Heuristic. Similar to the two-level heuristic for the single-product case, this heuristic uses the \bar{I}^r and N_r values that solve the lower-bound problem MEPQ. Define $I^r = \bar{I}^r - N_r$ and $I_0^r = I^r + (\tau_r^*(N_r) + t_s)d_r$. Ideally, we would like to start a campaign of product r when its inventory level reaches I_0^r and produce N_r batches until its inventory level reaches I^r . During this time, we do not want the inventory level of any other product r' to go below its respective $I_0^{r'}$. This may not be possible because in the midst of a campaign of product r , the inventory level of some other product r' would (in expectation) go below its $I_0^{r'}$ if another batch of product r is produced. We must trade off producing fewer batches of product r with starting the next campaign with less inventory of product r' .

Similar to the single-product case, we use a probability threshold Ψ , chosen by a line search over choices of $\Psi \in [0, 1]$. Our heuristic dynamically makes decisions by monitoring the inventory level of all products and the changes in belief over catalyst productivity $\gamma(b)$. We end the campaign if for some r' the probability of dropping below inventory $I^{r'}$ during the next batch is greater than the threshold. We add a few conditions to ensure that the process is stable (i.e., inventory does not arbitrarily increase or decrease).

At the end of a campaign, we need to choose the next product r^c to produce. For this purpose, we try to choose the r^c that would otherwise induce the largest backlogging cost. Note that if r^c is not produced in the next campaign, it will not be replenished for at least the duration of the next two campaigns. We approximate the duration of the next two campaigns by $\hat{\tau} \triangleq \min_r \tau^*(N^r) + \max_r \tau^*(N^r)$ and choose r^c as the product with the largest backlogging cost during this time, starting at its current inventory I^r and ending at $I^r - \hat{\tau}d^r$. With this choice of r^c , we describe the proposed policy for the next campaign.

Let I^r be the inventory level of product r at the end of a campaign and I_n^r be the inventory level of product r after producing batch n in the current campaign. For each product r , define $N_m^r \triangleq \arg \max_N \{N/(\tau_r^*(N) + t_s)\}$. We split the proposed policy into the following three scenarios depending on the inventory vector \mathbf{I} after the end of the previous campaign.

Case 1. ($I^r \leq I_0^r$ for More Than One r) This implies that the current inventory of more than one product is below its optimal starting value, indicating that a shortage might occur by the end of the next campaign. Set up a campaign for product r^c , and produce exactly $N_m^{r^c}$ batches.

Case 2 ($I^c \leq I_0^c$ but $I^r > I_0^r$ for All $r \neq r^c$). Unlike Case 1, the only imminent shortage is r^c . Produce n batches of product r^c until either (i) $I_n^c + n \geq \bar{I}^c$, (ii) producing batch $n + 1$ would exceed the time allocated to this campaign ($T_{n+1} + E[t_{n+1}] \geq \tau_{r^c}^*(N_{r^c})$), or (iii) $n \geq N_m^{r^c}$ and for some product r' we have $P[I_{n+1}^{r'} < \bar{I}^{r'}] \geq \Psi$.

Case 3 ($I^r > I_0^r$ for All r). Let r be the product with the lowest $(I^r - I_0^r)/d_r$ (excluding r^c). Compute $J \triangleq I_0^r + \tau_{r^c}^*(N_r)d_r$. If $J \geq I^r$, set up a campaign of product r^c without any idle time. If $J < I^r$, allow enough idle time such that either I^r drops to J or I^c drops to I_0^c (whichever happens first) and then set up a campaign of product r^c and follow the procedure in Case 2.

6.2.2. Practitioner’s Heuristic. Similar to the single-product practitioner’s heuristic, the multiple-product practitioner’s heuristic is part of the implementation described in Rajaram et al. (1999). In their heuristic, they use a fixed cycle length with one campaign of each product. Because switchover costs are not product dependent, the sequence of products inside the cycle is not considered. Similar to the single-product setting, a fixed batch-operation time t_r^* and a target number of batches per campaign N_r are chosen for each product r . The policy determining when to end a campaign of product r is the same as in the single-product case. The initial problem solved to determine the cycle length L , batch durations t_r^* , and target number of batches N_r is

$$(PH) \min_{\{N_r, t_r^*\}, L} \frac{\sum_{r=1}^R \left(\frac{N_r}{2} C_{I_r} + C_S \right)}{L}$$

s.t. $N_r \geq Ld_r \quad \forall r,$ (22a)

$$\sum_{r=1}^R (N_r t_r^* + t_s) \leq L, \quad (22b)$$

$$\sum_{n=1}^{N_r} f^{-1} \left(\frac{t_r^*}{k(nt_r^*)\mu_{br}} \right) \leq N_r \quad \forall r, N_r \in \{1, 2, \dots\}. \quad (22c)$$

The objective is to minimize the total average inventory-holding and switching costs during the cycle of length L . Constraint (22a) ensures that the number of batches that are planned to be produced should not be lower than the demand of product r during the cycle. Constraint (22b) enforces that the sum of campaign times does not exceed the cycle length. Constraint (22c) ensures that the mixture of batches for each product meets the attribute-level

constraint. The algorithm used to approximately solve the practitioner’s heuristic (PH) is provided in Section D of the online appendix.

As noted for the single-product case, the practitioner’s heuristic requires enhancements to provide a fair benchmark in all problem settings. These include enhancing the solution by replacing all C_{I_r} in (22) by $C_{IB_r} = C_{I_r} C_{B_r} / (C_{I_r} + C_{B_r})$. In addition, after a campaign of product r is completed, we delay releasing the prepared batches such that the replenished inventory begins at $N^r C_{B_r} / (C_{I_r} + C_{B_r})$.

7. Computational Results

To evaluate our method for the single- and multi-product problems, we compare it with the practitioner’s heuristic and with the appropriate lower bound. We first consider the single-product case. We were provided data from a sorbitol-production process at the company described in the Introduction. Salient details are summarized in Table 1.

We observe that for the actual problem parameters, the simulated average cost of our two-level heuristic is 12% lower than the practitioner’s heuristic, and the average cost of our heuristic is only 1.5% above the computed lower bound. This result shows significant potential for saving costs and attaining near-optimal costs if our approach is implemented. To test our heuristic under a wide range of parameter settings and to capture settings for other industries, we varied the actual parameters to obtain new problem instances by considering (1) concave, convex, and exponential $k(T)$, (2) high traffic versus low traffic, (3) low C_S , medium C_S , and high C_S , and (4) $C_B = \rho C_{I_r}$, $\rho \in [0.2, 0.5, 1, 2, 5]$.

We consider two classes of decay functions, each specified by three parameters, η , α , and β . The function $k^{(1)}(T_i) = \eta(1 + \beta T_i)^\alpha$ best resembles the decay function observed in our application, and the function $k^{(2)}(T_i) = (\eta + \alpha \exp(-\beta T_i))^{-1}$ conforms with the exponential-decay function used by Casas-Liza et al. (2005) and other papers on performance decay. We performed a sensitivity analysis by varying the parameters η , α , and β for both $k^{(1)}(T_i)$ and $k^{(2)}(T_i)$. Here we observe that as long as the resulting problem is feasible or when average production rate can meet demand, the performance of our two-level heuristic is robust to changes in these parameters. Therefore, for brevity, we restrict ourselves to three representative decay

Table 1. Problem Settings at the Sorbitol-Production Process

Parameter	d	C_I	C_B	C_S	t_s
Value	0.13	1	7	125	15
Variable or function	b	z_i	q_i^0	$k(T_i)$	$f(q_i)$
Distribution or form	$N(1.2, 0.2)$	$N(0, 0.15)$	$N(2, 0.2)$	$\frac{1}{2}(1 + T_i)^{1.2}$	$-\log\left(\frac{q_i}{q_i^0}\right)$

functions: a convex decay function $k_{conv}(T_i) = \frac{1}{2}(1 + T_i)^{1.2}$, a concave decay function $k_{conc}(T_i) = (1 + T_i)^{0.7}$, and an exponential decay function $k_{exp}(T_i) = (0.25 + 0.75 \{\exp(-0.2T_i)\})^{-1}$. In all our experiments, we used $f(q) = -\ln(q/q^0)$ in accordance with the literature (Steinfeld et al. 1989). The initial attribute level q^0 varied from batch to batch, and the data appeared approximately normal. Therefore, in our simulations, q_i^0 was drawn from a normal distribution with mean 2 and standard deviation 0.2.

Define the *capacity utilization* as the minimum fraction of time that the machine would be busy (i.e., not idle) to meet demand. In Table 2, “Low traffic” refers to a capacity utilization of 30%, whereas “High traffic” refers to a capacity utilization of 75%. Note that the process can operate at very high levels of utilization (i.e., $\geq 90\%$) only if we are able to exploit catalyst productivity and increase the production rate.

The results for the single-product case are shown in Table 2. The results are expressed as a *percentage gap*, defined as the difference between the value of the appropriate heuristic and the lower-bound solution as a percentage of the lower bound for a particular setting defined by the first entry in the row. The results shown for the row are from the problem instances achieved by varying the remaining parameters one at a time. For example, consider the first row in Table 2 with the problem setting convex $k(T)$. Varying the remaining parameters one at a time results in $1 \times 2 \times 3 \times 5 = 30$ problem instances. Similarly, consider the fourth row with problem setting high traffic. Here there are $3 \times 1 \times 3 \times 5 = 45$ problem instances. In this manner, we compile all the results in Table 2. The final row in the table considers problem instances with 90% utilization. Because this utilization level is not feasible for the practitioner’s heuristic, we do not have results in those columns.

Based on our computational analysis, we make the following observations:

- The maximum utilization levels at which the practitioner’s heuristic could operate were between 75% and 85%, and this was not significantly related to the problem parameters. After this level, the heuristic was unable to develop a feasible production plan that would ultimately satisfy backlog in future campaigns. This is because, unlike the two-level heuristic, it does not exploit the catalyst productivity and increase the production rate while developing a production plan. In contrast, the two-level heuristic considers this aspect and is able to meet demand even at 98% utilization. However, at utilization levels above 90%, the costs of the two-level heuristic also grow exponentially. This is because congestion effects owing to the randomness in catalyst productivity become more pronounced. This increases backlogging and overall costs.

- In general, the practitioner’s heuristic performs well when the utilization is low and when backlogging costs C_B are similar to inventory costs C_I . When utilization is high, there is more backlogging owing to randomness. However, the practitioner’s heuristic does not explicitly consider randomness in catalyst productivity while making decisions, which leads to unforeseen levels of backlogging. Thus, under these circumstances, its performance worsens, and this is more pronounced when $C_B \gg C_I$.

- In all problem instances, the two-level heuristic significantly outperforms the practitioner’s heuristic. In particular, the average cost of the two-level heuristic is 11% lower than the practitioner’s heuristic, and this cost improvement ranges from 4% to 25%. In addition, the two-level heuristic achieves very low gaps with the lower bound despite the stochastic nature of the problem. This is because reoptimization with learning captures most of the value of information, and

Table 2. Percentage Gaps of Heuristics for the Single-Product Problem

Problem setting	Percent gap from lower bound					
	Practitioner’s heuristic			Two-level heuristic		
	Minimum	Mean	Maximum	Minimum	Mean	Maximum
Convex $k(T)$	10.0	17.8	29.4	0.83	1.71	2.82
Concave $k(T)$	4.25	7.91	16.4	0.73	1.58	2.92
Exponential $k(T)$	5.16	12.4	23.9	0.72	1.68	2.87
High traffic	6.37	16.8	29.4	1.00	2.03	2.92
Low traffic	4.25	8.60	12.68	0.72	1.29	1.90
Low C_S	4.25	9.80	13.7	0.72	1.53	2.66
Medium C_S	4.62	13.8	25.9	0.79	1.59	2.79
High C_S	5.25	14.5	29.4	0.81	1.85	2.92
$C_B = C_I$	4.59	11.7	25.1	0.81	1.55	2.79
$C_B = 2C_I$	6.14	13.7	26.1	0.76	1.82	2.85
$C_B = 5C_I$	7.35	16.5	29.4	0.72	1.85	2.92
$C_B = 0.5C_I$	4.25	10.7	23.6	0.73	1.54	2.25
$C_B = 0.2C_I$	4.63	10.8	22.7	0.79	1.52	2.43
90% utilization	—	—	—	3.24	5.10	7.61

the probability threshold policy for catalyst switching provides a near-optimal decision rule.

- The superior performance of the two-level heuristic over the practitioner’s heuristic occurs for two reasons: First, the two-level heuristic exploits the productivity of the catalyst and is able to produce a fixed number of batches in a smaller time span, thus increasing the production rate of the process. Consequently, the two-level heuristic can produce more batches in a cycle and meet demand at a faster rate. This advantage is more pronounced when there is greater uncertainty in catalyst performance and the constraint in the EPQ problem is binding. Second, the practitioner’s heuristic uses expected parameter values in its decision process. Thus, it does not make efficient use of the available information (i.e., probability distributions on parameters, etc.) and is more prone to making suboptimal decisions.

- We observe that the optimal probability threshold Ψ that is used to determine catalyst switching in the two-level heuristic decreases in C_B and increases in C_S . A lower Ψ results in a lower risk of backlogging, compensating for the higher C_B . By contrast, a low Ψ may result in switching more frequently than desired. Therefore, Ψ is increasing in C_S .

- The heuristic is robust to the choice of Ψ . A deviation of ± 0.2 from the optimal Ψ increases the average cost by less than 5% in all problem instances. A possible explanation for this finding is the well-known insensitivity of the basic EPQ model to small deviations from the optimal cycle length (Schwarz 2008).

- When the utilization levels are between the levels observed in practice (i.e., between 30% and 75%), the gaps for the two-level heuristic are not sensitive to changes in the parameters. By contrast, the gap gets systematically higher for all parameter choices when the utilization is increased to 90%. This is because now significant backlogging costs are incurred, and this increases the cost of the two-level heuristic. However, the lower bound is unchanged because it is calculated by assuming a regenerative process and does not carry backlog over from one cycle to the next. This leads to higher gaps. Even under this extreme case of

high utilization, we see in Table 1 that the average gaps of the two-level heuristic are 5.10%, which is reasonable for a complicated stochastic planning problem.

The main managerial insight obtained from the simulations is that when we use the two-level heuristic for the single-product problem, near-optimal costs can be achieved by using a relatively simple policy. This policy only requires computing the probability of inventory falling below L , as defined in (16). Such a policy effectively makes use of all the information available about cost, current efficacy of the catalyst, and inventory level.

For the multiple-product setting, we were provided data on real parameter settings for a modified starch process with five products, summarized in Table 3. Here, again, we varied these parameters to obtain new problem instances and capture settings for other industries. The approach was similar to that in the single-product problem, except when analyzing the relation between C_B and C_I . Here $C_B \gg C_I$ is defined as $C_B \geq 5C_I$ for all products. In addition, $C_B \sim C_I$ is defined as $0.2C_I < C_B < 5C_I$ for all products and $0.5C_I \leq C_B \leq 2C_I$ for at least two products. Finally, $C_B \ll C_I$ is defined as $C_B \leq 0.2C_I$ for all products.

We also consider an additional comparison in the multiproduct case. Let $\hat{T}_r \triangleq \sqrt{2C_{S_r}/C_{I_{B_r}}d_r}$ be the optimal cycle length for product r from an EPQ perspective (Schwarz 2008), where $C_{I_{B_r}}$ is the balanced inventory holding and backlogging cost for product r , as defined in Section 6.1. When \hat{T}_r is similar across products, a simple rotation production cycle as employed in the practitioner’s heuristic can be near optimal. When \hat{T}_r across products is highly variable, the rotation cycles do not perform well. The corresponding gaps under these two scenarios are included in Table 4.

It is evident in Table 4 that the proposed two-level heuristic can significantly improve on the practitioner’s heuristic in all problem settings. In particular, for the actual problem parameters, the simulated average cost of our two-level heuristic was 22.07% lower than the practitioner’s heuristic. Thus, this approach, if used, has the potential to significantly reduce operational costs. Also note that the percentage gap of the

Table 3. Problem Settings Measured at the Modified Starch Process

Parameter/variable	Values, distributions, or functional forms for each product number				
	1	2	3	4	5
$d(\times 10^3)$	12	1	25	18	7
C_I	1	0.5	7	9	10
C_B	20	0.05	35	50	100
C_S	125	125	125	125	125
t_s	15	15	15	15	15
b	$N(1, 0.15)$	$N(0.2, 0.03)$	$N(1.2, 0.2)$	$N(0.9, 0.15)$	$N(0.7, 0.12)$
z_i	$N(0, 0.15)$	$N(0, 0.03)$	$N(0, 0.15)$	$N(0, 0.1)$	$N(0, 0.1)$

Note. For all products, the decay characteristic is $k(T_i) = (1 + T_i)^{0.7}$, the normalized initial attribute level q_i^0 is drawn from $N(2, 0.2)$, and the function $f(q_i)$ has the standard form q_i/q_i^0 .

Table 4. Percentage Gaps of Heuristics for the Multiple-Product Problem

Problem setting	Percent gap from lower bound					
	Practitioner’s heuristic			Two-level heuristic		
	Minimum	Mean	Maximum	Minimum	Mean	Maximum
Convex $k(T)$	19.60	36.9	53.2	0.56	4.19	7.88
Concave $k(T)$	6.77	19.3	39.6	0.67	3.76	6.21
Exponential $k(T)$	7.03	26.6	48.4	0.79	3.99	7.34
High traffic	9.88	31.1	53.2	1.28	4.62	7.88
Low traffic	6.77	24.1	41.9	0.56	3.34	5.71
Low C_S	6.77	21.4	34.6	0.56	3.55	6.21
Medium C_S	9.46	25.3	41.2	0.85	3.75	7.13
High C_S	10.5	36.1	53.2	1.27	4.64	7.88
$C_B \gg C_I$	14.1	33.1	52.6	1.71	4.47	7.88
$C_B \sim C_I$	7.14	26.5	53.2	0.67	4.25	7.80
$C_B \ll C_I$	6.77	23.3	39.7	0.56	3.22	5.51
Zero \hat{T}_r variability	6.77	19.2	34.4	0.56	2.88	4.92
Low \hat{T}_r variability	9.15	24.5	39.75	1.19	3.44	5.56
High \hat{T}_r variability	24.0	39.2	53.2	3.37	5.62	7.88
90% utilization	—	—	—	4.56	8.57	13.9

two-level heuristic with the lower bound is less than 10% in all cases and equal to 3.58% on average across multiple-product settings. These results are encouraging, given the complexity of the problem. The following additional observations can be drawn from the multiproduct case:

- Under the practitioner’s heuristic, the average cost of operation is significantly higher when the optimal cycle length \hat{T}_r is varied across products compared with when products have a similar \hat{T}_r . This is because the practitioner’s heuristic uses a rotation cycle in which the cycle length for all the products is the same. As \hat{T}_r across the products becomes more variable, it becomes less sensible to have the same cycle length for all products. In contrast, the two-level heuristic is a dynamic policy designed specifically to take such difference in product parameters into account. However, even with the two-level heuristic, costs are slightly higher when \hat{T}_r across products is more variable because it becomes harder to reach a stable production pattern. Nevertheless, the gaps of the two-level heuristic with the lower bound are quite low for all problem settings.

- Similar to the single-product case, the percentage gaps of the practitioner’s heuristic significantly increase when C_S is high. However, the two-level heuristic performs well because it dynamically controls all products to follow their optimal EPQ cycles as closely as possible. The EPQ cycle trades off C_S with C_I and C_B and is robust to changes in the production quantity. This explains why slight deviations do not significantly increase the average costs as long as a good policy is in place to make the switching decisions.

The following managerial insights can be drawn from the computational analysis. These could also be useful for practitioners in similar industries:

1. Our relatively simple two-level heuristic nearly attains the optimal cost of the intractable stochastic decision process. The optimal cost is closely approximated by our simulation-based stochastic lower bound.

2. For minimizing the duration of a campaign of a fixed number of batches, the value of information, Bayesian learning, and dynamic decision making can be adequately captured by employing a reoptimization policy in conjunction with observing and learning.

3. When deciding whether to switch, the probability that the inventory of each product r falls below its respective threshold I^r provides an efficient summary of the intractable multidimensional state of the system. This can be used to make the important decision of when to change the catalyst and switch to the next campaign.

4. When deciding on the next product to produce, we can choose the product with the greatest expected backlogging cost during the next two campaigns. This is somewhat similar to employing a one-step look-ahead policy, and our numerical results support its effectiveness.

Note that our methods still require repeated Bayesian updating of the belief distribution on the catalyst parameter (this is needed for more accurate computations of $E[t_{n+1}]$ and $P[I_{n+1} < I]$). However, such mathematical procedures seem amenable to implementation, given the ready availability of data from the process-control system and tools from standard commercially available statistical software.

8. Conclusions

The problem of production-campaign planning with uncertainty in production times, learning about production characteristics, and decay in catalyst performance is a challenging but important problem in a variety of process-industry sectors such as food

processing, pharmaceuticals, and specialty chemicals. We first considered the single-product case and formulated it as an SMDP. To solve this problem, we developed a two-level heuristic. We then considered the multiple-product case. We modeled the relaxed deterministic approximation as a constrained economic lot-sizing model and used a Lagrangian relaxation to solve it. We were able to extend all results and techniques of the single-product to the multiple-product case.

The computational results for the single- and multiple-product problems show that the associated two-level heuristic achieves low percentage gaps with the lower bound on the optimal average costs. Our approach significantly outperforms the practitioner's heuristic currently employed by a leading food-processing company. Furthermore, the two-level heuristics are robust to changes in the cost parameters. This allows us to further simplify the proposed dynamic policy and present general and easy-to-implement operational guidelines for practitioners.

This paper opens up several avenues for future research. First, based on our application context, we assumed that the demand rate was known and constant. However, there could be substantial seasonality in downstream demand in other settings, and it may be more appropriate to consider a time-varying demand rate for these situations. Second, our model could be extended to the case with multiple reactors. Third, one could consider alternate quality models that may be required for meeting attribute quality levels. Fourth, these techniques could be adapted for production-campaign planning at catalyst-activated batch-production processes in other process-industry settings. This would undoubtedly require incorporating different types of production constraints. All these extensions might require significant modifications to the methods presented in this paper and could be fruitful areas for future work.

References

- Brown DB, Smith JE (2013) Optimal sequential exploration: Bandits, clairvoyants, and wildcats. *Oper. Res.* 61(3):644–665.
- Casas-Liza J, Pinto JM, Papageorgiou LG (2005) Mixed integer optimization for cyclic scheduling of multiproduct plants under exponential performance decay. *Chemical Engrg. Res. Design* 83(10):1208–1217.
- Ciocan DF, Farias V (2012) Model predictive control for dynamic resource allocation. *Math. Oper. Res.* 37(3):501–525.
- Dusonchet F, Hongler MO (2003) Continuous-time restless bandit and dynamic scheduling for make-to-stock production. *IEEE Trans. Robotics Automation* 19(6):977–990.
- Federgruen A, Katalan Z (1996) The stochastic economic lot scheduling problem: Cyclical base-stock policies with idle times. *Management Sci.* 42(6):783–796.
- Grasman SE, Olsen TL, Birge JR (2008) Setting basestock levels in multi-product systems with setups and random yield. *IIE Trans.* 40(12):1158–1170.
- Hofmann J, Freier U, Wecks M, Hohmann S (2007) Degradation of diclofenac in water by heterogeneous catalytic oxidation with H_2O_2 . *Appl. Catalysis B Environ.* 70(1–4):447–451.
- Jain V, Grossmann IE (1998) Cyclic scheduling of continuous parallel process units with decaying performance. *AIChE J.* 44(7):1623–1636.
- Keller N, Rebmann G, Keller V (2010) Catalysts, mechanisms and industrial processes for the dimethylcarbonate synthesis. *J. Molecular Catalysis A: Chemical* 317(1–2):1–18.
- Korovessi E, Linnings AA (2005) *Batch Processes* (CRC Press, Boca Raton, FL).
- Levi R, Shi C (2013) Approximation algorithms for the stochastic lot-sizing problem with order lead times. *Oper. Res.* 61(3):593–602.
- Liu S, Yahia A, Papageorgiou LG (2014) Optimal production and maintenance planning of biopharmaceutical manufacturing under performance decay. *Indust. Engrg. Chemical Res.* 53(44):17075–17091.
- Loehndorf N, Minner S (2013) Simulation optimization for the stochastic economic lot scheduling problem. *IIE Trans.* 45(7):796–810.
- Mazzola JB, McCardle KF (1996) A Bayesian approach to managing learning curve uncertainty. *Management Sci.* 42(5):680–692.
- Mills PL, Chaudhari RV (1997) Multiphase catalytic reactor engineering and design for pharmaceuticals and fine chemicals. *Catalysis Today* 37(4):367–404.
- Molnar-Perl I, Morvai M, Pinter-Szakacs M, Kovago A, Petroczy J (1990) Gas-liquid chromatographic assay of the components of electrochemically reduced glucose solutions: Impurities in sorbitol samples. *J. Chromatography A* 520:185–191.
- Papachristos S, Konstantaras I (2006) Economic ordering quantity models for items with imperfect quality. *Internat. J. Prod. Econom.* 100(1):148–154.
- Rajaram K, Karmarkar US (2002) Product cycling with uncertain yields: Analysis and application to the process industry. *Oper. Res.* 50(4):680–691.
- Rajaram K, Karmarkar US (2004) Campaign planning and scheduling for multiproduct batch operations with applications to the food-processing industry. *Manufacturing Service Oper. Management* 6(3):253–269.
- Rajaram K, Tian Z (2009) Buffer location and sizing to optimize cost and quality in semi-continuous manufacturing processes: Methodology and application. *IIE Trans.* 41(12):1035–1048.
- Rajaram K, Jaikumar R (posthumously), Behlau F, van Esch F, Heynen C, Kaiser R, Kuttner A, van de Wege I (1999) Robust process control at Cerestar's refineries. *Interfaces* 29(1):30–48.
- Sakai T, Kawashima A, Koshikawa T (2009) Economic assessment of batch biodiesel production processes using homogeneous and heterogeneous alkali catalysts. *Bioresource Tech.* 100(13):3268–3276.
- Schwarz LB (2008) The economic order-quantity (EOQ) model. Chhaged D, Lowe TJ, eds. *Building Intuition* (Springer, New York), 135–154.
- Steinfeld JI, Francisco JS, Hase WL (1989) *Chemical Kinetics and Dynamics*, vol. 3 (Prentice Hall, Englewood Cliffs, NJ).
- Vaughan TS (2007) Cyclical schedules vs. dynamic sequencing: Replenishment dynamics and inventory efficiency. *Internat. J. Production Econom.* 107(2):518–527.
- Vieira M, Pinto-Varela T, Barbosa-Póvoa AP (2017) Production and maintenance planning optimisation in biopharmaceutical processes under performance decay using a continuous-time formulation: A multi-objective approach. *Comput. Chemical Engrg.* 107:111–139.
- Wang J, Li X, Zhu X (2012) Intelligent dynamic control of stochastic economic lot scheduling by agent-based reinforcement learning. *Internat. J. Prod. Res.* 50(16):4381–4395.
- Weekman VW Jr (1968) Model of catalytic cracking conversion in fixed, moving, and fluid-bed reactors. *Indust. Engrg Chemical Process Design Dev.* 7(1):90–95.
- Winands EM, Adan IJ, Van Houtum GJ (2011) The stochastic economic lot scheduling problem: A survey. *Eur. J. Oper. Res.* 210(1):1–9.