WORKING PAPER (SUBMITTED)

A unifying approximate dynamic programming model for the economic lot scheduling problem

Dan Adelman, Christiane Barz
University of Chicago, Booth School of Business, Chicago, IL, 60637, dan.adelman@chicagoGSB.edu, christiane.barz@chicagoGSB.edu

We formulate the well-known economic lot scheduling problem (ELSP) with sequence dependent setup times and costs as a semi-Markov decision process. Using an affine approximation of the bias function, we obtain a semi-infinite linear program determining a lower bound for the minimum average cost rate. Under a very mild condition, we can reduce this problem to a relatively small convex quadratically constrained linear problem by exploiting the structure of the objective function and the state space. This problem is equivalent to the lower bound problem derived by Dobson (1992) and reduces to the well-known lower bound problem introduced in Bomberger (1966) for sequence-dependent setups. We thus provide a framework that unifies previous work, and opens new paths for future research on tighter lower bounds and dynamic heuristics.

Key words: economic lot scheduling problem; sequence-dependent setups; semi-Markov decision process; approximate dynamic programming;
History: Submitted December 2008. Revised January 2009.

1. Introduction

The classical economic lot scheduling problem (ELSP) schedules production runs for several products on a single machine. Typically, it is assumed that a setup (of a given cost and time) must be performed every time production is changed from one product to another, and that demand and production rates are constant. The goal is to minimize the long-run average total cost, composed of holding and setup cost, over an infinite horizon without incurring any stock-outs.

Ever since the ELSP was first introduced by Rogers (1958) half a century ago, researchers have proposed an abundance of different solution heuristics. The vast majority of the work concentrates on the special case of sequence-independent setup times and costs, which was shown to be NP hard under cyclic schedules by Hsu (1983) and Gallego and Shaw (1997). The most common classification of heuristics distinguishes between basic period and varying lot size approaches as suggested by Bomberger (1966), Elmaghraby (1978), Dobson (1987), and Zipkin (1991). Both approaches are based on production frequencies that are usually estimated by either the independent (economic production quantity) solution or the tighter lower bound proposed by Bomberger (1966). Overviews can be found e.g. in Elmaghraby (1978) and Carstensen (1999).

Although sequence-dependent setup times and costs are reported to be prevalent in most practical applications, the literature on this general case is relatively small. Early exceptions like Maxwell (1964), Sing and Foster (1987), and Inman and Jones (1993) consider only very restrictive forms of setup times. Taylor et al. (1997) suggest a simple heuristic based on economic production quantities and a ordering of products based on traveling salesman solutions. Wagner and Davis (2002) propose a search heuristic over the set of cyclic production schedules. The most sophisticated approach was pursued by Dobson (1992). Applying a series of different relaxations, he derives a new lower bound problem to obtain production frequencies that are used in a power-of-two heuristic.

Aside from pure search heuristics, almost all the research on the ELSP broadly follows either the basic period or Dobson's varying lot size approach. The goal of this paper is to provide a coherent unifying framework, based on approximate dynamic programming, that opens new paths

for future research on tighter lower bounds and dynamic heuristics. This potential for impact exists because the approximation framework is based on formulations that have the power, at least theoretically, to solve the ELSP exactly, and it is easily extensible with stronger approximation forms. In particular, we

- formulate the sequence-dependent ELSP as a semi-Markov decision process and discuss new structural properties of the state space as well as communication properties;
- introduce an affine approximation to the value function, in which the approximation parameters can be interpreted from a dual formulation as marginal values for machine time, products, and machine states;
- exploit the structure of a semi-infinite linear program to reduce it to a mixed integer problem with linear objective and a modest (finite) number of binary variables and convex constraints;
- show the equivalence of this mixed integer problem to a relatively small convex quadratically constrained linear problem under a very mild condition (if the condition is not fulfilled, the convex quadratically constrained linear problem relaxes the original problem);
- present a new flow-based lower bound problem that can be viewed as a intuitive extension of Bomberger's (1966) lower bound problem to sequence-dependent setups;
- show dual equivalence of the convex quadratically constrained linear problem to the flow-based problem;
- prove the equivalence of the convex quadratically constrained linear problem to the lower bound problem by Dobson (1992). As a side result, we obtain that some of the variables introduced in Dobson's formulation are superfluous.

The connections we make with previous work are surprising and nontrivial. In particular, it is completely unexpected that the flow-based and the convex quadratically constrained linear lower bound problems are equivalent to Dobson's model and that this is equivalent to making an affine value function approximation in a semi-Markov decision process. This is a powerful connection based on first principles that provides a path forward for future research on the ELSP. Furthermore, the ideas we use to compress the resulting semi-infinite program into a compact convex quadratic optimization problem may be helpful in applying approximate dynamic programming in other settings.

Our focus in this paper is to make these connections with previous models. Other authors have already used these models as part of heuristics for constructing cyclic schedules. Furthermore, implementing the approximate dynamic control policy that emerges from this work involves many technical issues that require an in-depth analysis to resolve. Therefore, we leave consideration of ADP-based dynamic control policies to a companion paper, Adelman and Barz (2009).

1.1. Outline

After a short survey of related literature on the lot scheduling feasibility problem and approximate dynamic programming, we will formulate the sequence-dependent ELSP as a semi-Markov decision process in Section 2. In Section 3, we discuss some structural properties of our model leading to an infinite dimensional linear programming model. Introducing an affine approximation to the value function, we analyze the resulting optimization problem in Section 4. In Section 5, we introduce a new flow-based lower bound problem and compare our lower bound problems to the lower bound problems derived by Bomberger (1966) and Dobson (1992). Section 6 summarizes and highlights directions for future research.

1.2. Other related literature

A related problem to the ELSP is the question if given inventory levels are sufficient to serve demand over an infinite horizon without stock-outs. This lot scheduling feasibility problem is connected to the state space of our Semi-MDP. Anderson (1990) showed that this problem NP-hard.

Although semi-Markov formulations of the ELSP were suggested by Aragone and Gonzalez (2000) in a sequence-independent setting with no setup times, and e.g. by Qui and Loulou (1995) and Hodgson et al. (1997) in stochastic versions of the problem, the curse of dimensionality prevents their application to problems of realistic size. To our knowledge, approximate dynamic programming techniques as described here have never been applied to the ELSP before.

Overviews on approximate dynamic programming are given by Bertsekas and Tsitsiklis (1996), Bertsekas (2005), and Powell (2007). Like most of the literature on approximate dynamic programming, their focus lies on simulation-based methods for adaptively computing value function approximations. Our approximations, however, will lead to an optimization problem that can be solved directly with no need for simulation.

The variables of our lower bound problem can be interpreted as prices of machine time, state, and products. In this sense, our contribution is connected to the work by Adelman (2004a) and Adelman (2004b), who derives heuristics from a dynamic programming formulation of a joint replenishment and inventory/routing problem. While he relies on column generation to calculate the prices, however, we can determine our prices by the solution of a relatively small convex optimization problem.

2. The economic lot scheduling problem

There is one machine that can be used to produce different products $i = 1, \ldots, I$, where product i is produced at a rate $0 < p_i < \infty$. We denote the set of products as $\mathcal{I} = \{1, \ldots, I\}$. The machine can only produce one product at a time. When switching from one product i_1 to another product i_2, a setup must be performed that causes both costs of $0 \le c_{i_1 i_2} < \infty$ and a setup time of $0 \le \tau_{i_1 i_2} < \infty$ during which production is paused. There is a constant demand of $0 < \lambda_i < p_i$ for each product $i \in \mathcal{I}$. Denoting the utilization for product i by $\rho_i = \lambda_i / p_i$ we assume $\sum_{i=1}^{I} \rho_i < 1$.

Finished products of type i that were not yet consumed by demand cause holding costs of $\phi_i > 0$ per time unit and item. We do not allow for the event that demand cannot be served immediately (from inventory or production), which we will call a *stock-out*.

2.1. The control problem

The controller wants to minimize long-term average costs from a given initial system state composed of a machine state m (the product the machine is setup for) and initial inventory levels for all products $\vec{s} = (s_1, \ldots, s_I)$.

Let $(\vec{s^n}, m^n)$ denote the state in decision epoch n. The controller then chooses an action composed of an idle time $u^n \ge 0$, the index of the next product the machine should produce $q^n \in \mathcal{I}$, and the corresponding production time $t^n \ge 0$.

So, starting in system state $(\vec{s}, m) = (\vec{s^1}, m^1)$, the controller chooses an action (u^1, q^1, t^1) such that no stock-outs occur. Then, the new machine state is $m^2 = q^1$ and the new inventory levels after idling, setup for product q^1, and production (i.e. after a time $u^1 + \tau_{m^1 q^1} + t^1$) are $\vec{s^2}$ with $s^2_{q^1} = s^1_{q^1} - \lambda_{q^1} \cdot (u^1 + \tau_{m^1 q^1} + t^1) + p_{q^1} t^1$ and $s^2_i = s^1_i - \lambda_i \cdot (u^1 + \tau_{m^1 q^1} + t^1)$ for all $i \ne q^1$. Being in state $(\vec{s^2}, m^2)$, she chooses (u^2, q^2, t^2) such that stock-outs are avoided and so on. In general, the state transitions follow

$$s_i^{n+1} = s_i^n - \lambda_i \cdot (u^n + \tau_{m^n q^n} + t^n) + p_i t^n \delta_{q^n i} \qquad \text{for all } i \in \mathcal{I},\ n \in \mathbb{N}, \tag{1}$$

$$m^{n+1} = q^n \qquad \text{for all } n \in \mathbb{N}, \tag{2}$$

$$m^1 = m, \quad \text{and} \quad s_i^1 = s_i \qquad \text{for all } i \in \mathcal{I}, \tag{3}$$

with $\delta_{q^n i} = 1$ if $q^n = i$ and 0 else.

For accounting purposes, all holding costs for items produced during time t^n (which we will call the nth production run) are incurred at the moment the production is started. Assume that the

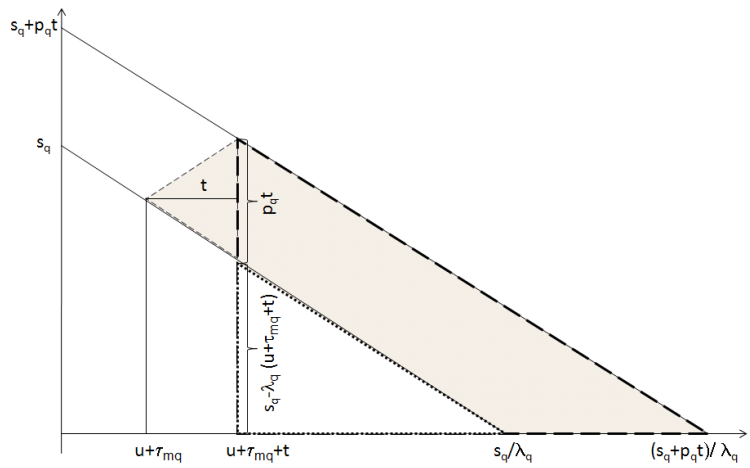

Figure 1 Holding costs

system is in state $(\vec{s}, m)$ at time 0. If there were no more production of product q in the future, then inventory of product q would decline at a rate of λ_q until time s_q/λ_q. To determine incremental holding costs, we can assume that it remains at 0 thereafter and compare it with the inventory costs when we produce. After the decision (u, q, t) is made, inventory starts increasing at rate $p_q - \lambda_q$ after the initial idle and setup time. Another t time units later, it decreases again at rate λ_q. If there were no more production of product q, the inventory level would be 0 at time $(s_q + p_q t)/\lambda_q$. This behavior is illustrated in Figure 1.

In this case, the controller faces not only the setup costs c_{mq} but also holding costs of ϕ_q times the gray shaded area (representing the difference between the two inventory trajectories) indicated in Figure 1. Adding the triangular gray shaded area between times $u + \tau_{mq}$ and $u + \tau_{mq} + t$ and the following trapezoid (the difference between the big dashed right-angled triangle and the small dotted right-angled triangle) gives costs of

$$\begin{aligned}
\kappa(s_q, m, u, q, t) &= c_{mq} + \phi_q \cdot \left[\frac{p_q t^2}{2} + \frac{\left(s_q - \lambda_q(u + \tau_{mq} + t) + p_q t\right)^2}{2\lambda_q} - \frac{\left(s_q - \lambda_q(u + \tau_{mq} + t)\right)^2}{2\lambda_q} \right] \\
&= c_{mq} + \phi_q \cdot \left[\frac{p_q t^2}{2} + \frac{(p_q t)^2}{2\lambda_q} + \frac{2\left(s_q - \lambda_q(u + \tau_{mq} + t)\right) p_q t}{2\lambda_q} \right] \\
&= c_{mq} + \frac{\phi_q p_q t}{2\lambda_q} \cdot \left(2s_q - 2\lambda_q(u + \tau_{mq}) + t(p_q - \lambda_q)\right). \qquad (4)
\end{aligned}$$

Although these costs do not depend on the inventory levels of other products, we will frequently write $\kappa(\vec{s}, m, u, q, t) = \kappa(s_q, m, u, q, t)$ when we speak of the direct costs from action (u, q, t) in state $(\vec{s}, m)$.

Using this notation, we can formulate the decision maker's problem to find an optimal sequence of actions $\{(u^n, q^n, t^n)\}_{n=1,2,\ldots}$ starting with inventories of $\vec{s}$ in machine state m, as

$$\inf_{\{(u^n, q^n, t^n)\}_{n=1,2,\ldots}} \limsup_{N \to \infty} \frac{\sum_{n=1}^{N} \kappa(\vec{s^n}, m^n, u^n, q^n, t^n)}{\sum_{n=1}^{N} (u^n + \tau_{m^n q^n} + t^n)}$$

under (1), (2), (3), $u^n, t^n \geq 0$, and $q^n \in \mathcal{I}$ for all $n \in \mathbb{N}$.

2.2. The state space

As the state space $\mathcal{S}$ we choose the set of all *admissible* combinations of machine states $m \in \mathcal{I}$ and inventory levels $\vec{s} \in \mathbb{R}_+^I$. By admissible we mean that given $m = m^1$ and $\vec{s} = \vec{s^1}$ there exists an

infinite sequence of state-action pairs $\{(\vec{s^n}, m^n), (u^n, q^n, t^n)\}_{n=1,2,\ldots}$, that avoids stock-outs, i.e. for which all products $i = 1, \ldots, I$ and decision epochs $n \in \mathbb{N}$ the inventory level after idling and setup (when it is lowest) is non-negative,

$$s_i^n - \lambda_i \cdot (u^n + \tau_{m^n q^n}) \geq 0 \text{ for all } i \in \mathcal{I},\ n \in \mathbb{N}. \tag{5}$$

LEMMA 1. *$\mathcal{S}$ is non-empty if and only if $\sum_{i=1}^{I} \rho_i < 1$.*

Proof. If $\sum_{i=1}^{I} \rho_i \geq 1$, it is impossible to satisfy demand in the long run, independent of the starting inventory, see e.g. Lemma 1 in Anderson (1990), so $\mathcal{S}$ is empty. To show the other direction, we demonstrate states that are in $\mathcal{S}$ if $\sum_{i=1}^{I} \rho_i < 1$.

If $\sum_{i=1}^{I} \rho_i < 1$, we can always construct an infinite sequence of actions $\{(u^n, q^n, t^n)\}_{n=1,2,\ldots}$ without idling, $u^n = 0$ for all n, cycling through all the products 1 through I, i.e. with $q^{i+kI} = i$ and production times of $t^{i+k\cdot I} = \rho_i \sum_{j=1}^{I} \tau_{q^j q^{j+1}} / (1 - \sum_{j=1}^{I} \rho_j)$ for all $k \in \mathbb{N}$, $i \in \mathfrak{I}$. Since these production times are the solution to

$$\lambda_i \cdot \left(\sum_{j=1}^{I} \tau_{q^j q^{j+1}} + \sum_{i=1}^{I} t^{i+k\cdot I} \right) = p_i t^{i+k\cdot I} \quad \text{for all } k \in \mathbb{N}, i \in \mathfrak{I},$$

the total production of product i within I production epochs equals the demand for that product in that time. So by ensuring that there is enough inventory to survive I production epochs (which have a total length of $T = \sum_{j=1}^{I} \tau_{q^j q^{j+1}} / (1 - \sum_{j=1}^{I} \rho_j)$), i.e. having $\vec{s}$ with $s_i = \lambda_i \sum_{j=1}^{I} \tau_{q^j q^{j+1}} / (1 - \sum_{j=1}^{I} \rho_j)$, we can enure that no stock-outs will ever occur and there are states in $\mathcal{S}$. □

Given a certain machine state m, we denote the set of all inventory levels that form an admissible state by

$$\mathcal{S}(m) = \{\vec{s} \mid (\vec{s}, m) \in \mathcal{S}\}.$$

We call a state $(\vec{s}, m) \in \mathcal{S}$ *strongly admissible* if there exists some $\epsilon > 0$ such that the state $(\vec{s} - \epsilon\vec{1}, m)$ is still in $\mathcal{S}$, where $\vec{1}$ denotes an I-dimensional vector with all elements equal to 1. The set of all strongly admissible states is the interior of $\mathcal{S}$, which we denote by $\mathcal{S}^S \subset \mathcal{S}$. Let $\mathcal{S}^S(m)$ denote the set of all inventory levels that form a strongly admissible state given machine state m.

The structure of $\mathcal{S}$ (and $\mathcal{S}^S$) is not trivial. Anderson (1990) showed in his Theorem 2 that even in the case of sequence-independent setup times, it is NP-hard to decide if a given state $(\vec{s}, m)$ lies in $\mathcal{S}$. Furthermore, the following examples show that the sets $\mathcal{S}(m)$ need in general be neither convex nor closed.

Example (non-convexity of $\mathcal{S}$). Think of three identical products with $\lambda_i = 1$, $p_i = 9$ for all $i = 1, 2, 3$, $\tau_{i_1 i_2} = \tau = 1$ for all $i_1, i_2 = 1, 2, 3$ and an initial machine state of $m = 3$.

The inventory levels of $\vec{s} = (1.1, 4.1, 2.6)$ and $\vec{s'} = (4.1, 1.1, 2.6)$ are both in $\mathcal{S}(3)$ and $\mathcal{S}^S(3)$. To see this for $\vec{s}$, consider actions with $u^n = 0$ for all n, $t^n = 0.5$ for all n and $q^{1+3k} = 1$, $q^{2+3k} = 3$ and $q^{3+3k} = 2$ for all $k \in \mathbb{N}_0$; to see this for $\vec{s'}$ alter the production sequence to $q^{1+3k} = 2$, $q^{2+3k} = 3$ and $q^{3+3k} = 1$ for all $k \in \mathbb{N}_0$ all other things being equal. In both cases, the inventory levels of all three products loop in the range between 0.1 and 4.1, so both $\vec{s}$ and $\vec{s'}$ lie in $\mathcal{S}(3)$. Clearly these states would still be admissible if we had $0 < \epsilon \leq 0.1$ items less of each product. Hence, both states are strongly admissible, i.e. $\vec{s}$ and $\vec{s'}$ are in $\mathcal{S}^S(3)$.

$\vec{s''} = (2.6, 2.6, 2.6) = 0.5\vec{s} + 0.5\vec{s'}$, however, is not in $\mathcal{S}(3)$! No matter how idle times, sequences, or production times are chosen, stock-outs cannot be avoided. This is because in order not to stock-out every product must be set up sooner or later. Here, this means that the initial inventory of the first product produced must suffice to cover the demand that occurs during one setup at least (the time before production of this product can be started). The initial inventory of another product must suffice to cover at least two setups. The initial inventory of the last product in the production sequence must suffice to cover at least three setups. Since all $s_i = 2.6 < 3 = 3\lambda_i\tau$, however, we conclude that we cannot avoid stock-outs of the product that is produced at last. □

Example (non-closedness of $\mathcal{S}$). To see that the set $\mathcal{S}(m)$ need not be closed, take any number of products, setup costs, demand and production rates and let $\tau_{ij}=0$ for all $i,j\in\mathcal{I}$. Then, for all m, $\mathcal{S}(m)$ contains all inventory vectors $\vec{s}$ with $s_i>0$ for all $i\in\mathcal{I}$ except for at most one $i\in\mathcal{I}$ with $s_i=0$. This set is obviously not closed. □

2.3. The action space

Let $\vec{s'}(\vec{s},m,u,q,t)$ represent the inventory vector after the action (u,q,t) was executed starting from state $(\vec{s},m)$, i.e. $s'_q(\vec{s},m,u,q,t)=s_q-\lambda_q\cdot(u+\tau_{mq}+t)+p_qt$ and $s'_i=s_i-\lambda_i\cdot(u+\tau_{mq}+t)$ for all $i\neq q$. We define the action space $\mathcal{A}(\vec{s},m)$ when starting in $(\vec{s},m)$ as the set of all actions (u,q,t) that prevent a direct stock-out by fulfilling (5) and that also prevent long-term stock-outs by leading to a state in $\mathcal{S}$:

$$\mathcal{A}(\vec{s},m)=\{(u,q,t):t,u\geq 0,q\in\mathcal{I},\vec{s'}(\vec{s},m,u,q,t)\in\mathcal{S}(q),s_q-\lambda_q\cdot(u+\tau_{mq})\geq 0\}.$$

Since $\vec{s'}(\vec{s},m,u,q,t)\in\mathcal{S}(m)$ ensures the non-negativity of inventory before production for all $i\neq q$, we need to postulate this property only for product q.

We denote the set of all feasible idling and production times (u,t) given the next product q by

$$\mathcal{A}(\vec{s},m,q)=\{(u,t)\mid t,u\geq 0:(u,q,t)\in\mathcal{A}(\vec{s},m)\}.$$

2.4. The semi-MDP formulation

Finally, let us formulate the semi-MDP that underlies the ELSP. Since our state and action space have a rather non-trivial structure, we cannot exclude that the decision-problem is multichain. This is why the average cost rate might depend on the system state, which we denote by $g(\vec{s},m)$. Taking into account that given action (u,q,t), the decision period lasts $u+\tau_{mq}+t$ time units, the optimality equations read

$$g(\vec{s},m)=\inf_{(u,q,t)\in\mathcal{A}(\vec{s},m)}g(\vec{s'}(\vec{s},m,u,q,t),q) \tag{6}$$

$$h(\vec{s},m)=\inf_{(u,q,t)\in\mathcal{A}_g(\vec{s},m)}\{\kappa(\vec{s},m,u,q,t)-g(\vec{s'}(\vec{s},m,u,q,t),q)\cdot(u+\tau_{mq}+t)+h(\vec{s'}(\vec{s},m,u,q,t),q)\} \tag{7}$$

for all $(\vec{s},m)\in\mathcal{S}$, where $A_g(\vec{s},m)$ is the set of actions $a\in A(\vec{s},m)$ that maximizes the right-hand side of (6). In the discrete case, $A_g(\vec{s},m)$ can be replaced by $A(\vec{s},m)$ in (7), see e.g. Schäl (1992). To the authors' knowledge there is no literature in dynamic programming ensuring the existence of solutions and the validity of replacing $A_g(\vec{s},m)$ by $A(\vec{s},m)$ that would apply in our setting.

3. Properties of the semi-MDP

In this section, we discuss the structure of our semi-MDP formulation to formulate an infinite dimensional linear program which determines a lower bound to the actual minimum average cost rate.

3.1. Communication properties

The following lemma is a straight-forward extension of Anderson's Lemma 2 to incorporate sequence-dependent setups (a proof following the lines of Anderson (1990) is given in the appendix).

Lemma 2. *Given a machine state m, starting inventory levels $\vec{s}\in\mathcal{S}^S(m)$, a target machine state m' and target inventory levels of $s'_i\geq s_i$ for all $i\in\mathcal{I}$, there exists a feasible sequence of actions $\{(0,q^n,t^n)\}_{n=1,2,\ldots}$ with the property that for some $n=N$, $\infty>N\geq 1$, the machine state is $q^N=m'$ and the target inventory levels are reached, i.e. $s_i^N\geq s'_i$ for all $i\in\mathcal{I}$.*

We extend this lemma further by claiming

LEMMA 3. *For every state $(\vec{s}, m) \in \mathcal{S}^S$, there exists a feasible production sequence and production times that move the system from $(\vec{s}, m)$ to any $(\vec{s'}, m') \in \mathcal{S}$ without stock-outs and without leaving $\mathcal{S}$. This sequence has a finite number of steps, finite time duration, and finite cost.*

Proof. Let $\vec{s''} = \max(\vec{s}, \vec{s'}) + \epsilon\vec{1}$ be the component-wise maximum of the inventory vectors increased by some $\epsilon > 0$. Note that $\vec{s''}$ is both in $\mathcal{S}^S(m)$ and in $\mathcal{S}^S(m')$.

We demonstrate a finite production sequence and production times that lead from $(\vec{s}, m)$ to a state $(\vec{s''}, m')$, and then from $(\vec{s''}, m')$ to $(\vec{s'}, m')$.

Note that in Lemma 2, we can judiciously replace production time with idle time in order to achieve the target inventory $\vec{s''}$ exactly. Hence, there exist a finite production sequence and production times that lead from $(\vec{s}, m)$ to $(\vec{s''}, m')$.

So let $\{(0, q^n, t^n)\}_{n=1,2,\ldots}$ be a sequence of actions that avoid stock-outs starting in state $(\vec{s'}, m') \in \mathcal{S}$. Such a sequence exists due to the definition of $\mathcal{S}$ and clearly also avoids stock-outs when starting in $(\vec{s''}, m')$. Using this sequence with additional production times as introduced in Lemma 2 and replacing judiciously production time with idle time, we can reach $(\vec{s'}, m')$ (with $s'_i \leq s''_i$ for all $i \in \mathcal{I}$) in a finite number of steps, finite time duration and hence, finite cost. □

Lemma 3 only shows that there exists a sequence from every state in $\mathcal{S}^S$ to every state in $\mathcal{S}$ with finite time duration and cost. The time and cost to reach this other state in $\mathcal{S}$, however, need not be bounded as the following example shows.

Example (unbounded time and cost). Suppose there are three identical products with $\lambda_i = 1$, $p_i = 6$ for all $i = 1, 2, 3$ and $\tau_{12} = \tau_{23} = \tau_{31} = 1$ and $\tau_{i_1 i_2} = 20$ for all other $i_1, i_2 = 1, 2, 3$, $c_{i_1 i_2} = 1$ for all $i_1, i_2 = 1, 2, 3$.

The state $((1, 3, 5), 3)$ is in $\mathcal{S}$ but not in $\mathcal{S}^S$ because no setup for the first product is possible with $s_1 = 1 - \epsilon$. Once state $((1, 3, 5), 3)$ is reached, the only way to avoid stock-outs is to keep cycling without idling and with production times of 1.

This is true because given inventory levels of (1,3,5), a production sequence that does not lead to stock-outs must start with $q^1 = 1$, $q^2 = 2$, $q^3 = 3$. If the total inventory $s_1 + s_2 + s_3$ is not increased (to a level of 20 or more) after the first 3 decision epochs, the production sequence will continue with $q^4 = 1$, $q^5 = 2$, $q^6 = 3$ and so on. Comparing the total demand during the setups of the first three decision epochs (which is $3 \cdot \sum_{i=1}^{3} \lambda_i = 9$) with the total net production per time unit ($p_i - \sum_{i=1}^{3} \lambda_i = 3$) shows that a total production time $t^1 + t^2 + t^3$ of 3 is needed to keep total inventory at the same level. $t^1 + t^2 + t^3 > 3$ would be needed to actually increase total inventory. In order to fulfill (5), however, we obtain the following restrictions on production times:

$$s_2 - u^1 - \tau_{31} - t^1 - u^2 - \tau_{12} \geq 0 \;\Rightarrow t^1 \leq 1 - u^1$$
$$s_3 - u^1 - \tau_{31} - t^1 - u^2 - \tau_{12} - t^2 - u^3 - \tau_{23} \geq 0 \;\Rightarrow t^2 \leq 2 - u^1 - u^2 - t^1.$$

As a consequence, $s_1^4 \leq 1 \cdot p_1 - (\tau_{12} + \tau_{23} + 1) = 3$ and $q^4 = 1$. To fulfill (5) for $n = 4$, $t^1 + t^2 + t^3 \leq 3 - u^1 - u^2 - u^3 \leq 3$. If the initial inventory level of s_1 is lower than 1, no setup can ever be performed without a stock-out. If s_2 is lower than 3, $t^1 < 1$ and $s_1^4 < 3$, which leads to $t^1 + t^2 + t^3 < 3$ and means that the total inventory will drop after each cycle through production of products 1, 2 and 3 causing stock-outs eventually. The same effect can be shown if $s_3 < 5$. Hence, the only actions avoiding stock-outs are $t^n = 1$ and $u^n = 0$ for all $n \geq 1$.

Now, consider the states $(\vec{s'}, m') = ((1 + \epsilon, 3 + \epsilon, 5 + \epsilon), 3)$ and $(\vec{s''}, m'') = ((2 + \epsilon, 4 + \epsilon, 6 + \epsilon), 3)$ for some $\epsilon > 0$. Starting in $(\vec{s'}, m')$ the maximum time production can be extended in each cycle through the products 1, 2, and 3, is ϵ, following the lines of the proof of Lemma 2. So it is feasible to cycle through the products with no idling $u^n = 0$ and production times $t^n = 1 + \epsilon/3$. Doing this, the inventory of each product i is increased by

$$p_i t^n - \lambda_i \cdot (3t^n + \tau_{12} + \tau_{23} + \tau_{31}) = \epsilon$$

in each cycle. Since there is no way to increase production times even further without provoking stock-outs and $s''_i - s'_i = 1$ for all i, the costs to get from state $(\vec{s'}, m,')$ to $(\vec{s''}, m'')$ are larger than $1/\epsilon$ (constituting the setup costs) and the time is longer than $1/\epsilon$ (the time spent on setups). Because for $\epsilon \to 0$, these lower bounds go to infinity, neither the costs nor the time to reach $(\vec{s''}, m'')$ from $(\vec{s'}, m')$ can be bounded.

3.2. Unichain vs. multichain

The lemmas in the previous section show that all states in $\mathcal{S}$ can be reached from states within $\mathcal{S}^S$. This means that the minimum average costs $g(\vec{s}, m)$ are equal for all states within $\mathcal{S}^S$. The results cannot be generalized to the whole set of $\mathcal{S}$, however. As the following example illustrates, our semi-MDP is multichain.

Example (multichain). Consider a four product-setting with $\lambda_i = 1$, $p_i = 64$, and $\phi_i = 25/504$ for all $i = 1, 2, 3, 4$. The setup times $\tau = (\tau_{i_1 i_2})$ and the setup costs $c = (c_{i_1 i_2})$ are given by

$$\tau = \begin{pmatrix} 20 & 1 & 20 & 1 \\ 2 & 20 & 2 & 20 \\ 20 & 1 & 20 & 1 \\ 2 & 20 & 2 & 20 \end{pmatrix}, \qquad c = \begin{pmatrix} 0 & 1 & 0 & 5 \\ 5 & 0 & 1 & 0 \\ 0 & 5 & 0 & 1 \\ 1 & 0 & 5 & 0 \end{pmatrix}.$$

Now consider the state $(\vec{s}, m) = ((2, 3.1, 5.2, 6.3), 4)$. The total inventory of that state is 16.6. In addition, note that in order to execute any sequence of actions $\{u^n, q^n, t^n\}_{\{n=1,2\ldots\}}$ starting in machine state $m = q^0$, a minimum total inventory of more than

$$\sum_{i=1}^{I} \sum_{n=1}^{\min\{N:q^n=i\}} \tau_{q^{n-1}q^n},$$

is needed independently of production times. This is because the initial inventory of product 1 must suffice to cover the demand until it is produced the first time, which is the demand during the setup times $\sum_{n=1}^{\min\{N:q^n=1\}} \tau_{q^{n-1}q^n}$ plus the production and idling time until 1 is produced the first time; the initial inventory of product 2 must suffice to cover demand for a time of at least $\sum_{n=1}^{\min\{N:q^n=2\}} \tau_{q^{n-1}q^n}$, the initial inventory of product 3 must suffice to cover demand for a time of at least $\sum_{n=1}^{\min\{N:q^n=3\}} \tau_{q^{n-1}q^n}$, and so on.

Starting from $m = 4$ in this example, a total inventory of 16.6 or less implies that the production sequence must start with either $1, 2, 3, 4$, or $1, 4, 3, 2$, or $3, 4, 1, 2$, or $3, 2, 1, 4$. Due to the particular allocation of inventory in state $(\vec{s}, m) = ((2, 3.1, 5.2, 6.3), 4)$, there is actually no other feasible production sequence than 1,2,3,4 to start with. The total time spent on setups over these first 4 production epochs is 6. In order to at least recover a vector with a total inventory of 16.6, we have to ensure that

$$16.6 \le 16.6 - \sum_{i=1}^{4} u^i \lambda_i - 6 \cdot \sum_{i=1}^{4} \lambda_i + \sum_{i=1}^{4} t^i (p_i - \lambda_i) \le 16.6 - 6 \cdot 4 + (t^1 + t^2 + t^3 + t^4)(64 - 4).$$

Hence, a total time of $t^1 + t^2 + t^3 + t^4 \ge 0.4$ must be spent on production. The only way to do this without stock-outs is to choose $t^1 = t^2 = t^3 = 0.1$ implying $t^4 = 0.1$. After these 4 decision epochs, the state is again $(\vec{s}, m) = ((2, 3.1, 5.2, 6.3), 4)$. Lower production times will lead to lower inventory levels after the first 4 decision epochs, which will circumvent a total production equal to demand in the future and makes stock-outs unavoidable.

Summarizing the above, starting in state $(\vec{s}, m)$ actions of $u^n = 0$, $t^n = 0.1$ for all $n \in \mathbb{N}$, and $q^{1+4k} = 1$, $q^{2+4k} = 2$, $q^{3+4k} = 3$, and $q^{4+4k} = 4$ for all $k \in \mathbb{N}_0$ never lead to stock-outs. Therefore, the state is admissible, $(\vec{s}, m) \in \mathcal{S}$. As we have demonstrated, this initial state leaves no other choice

but to follow these actions. So the average costs starting in ((2,3.1,5.2,6.3),4) equals the sum over the four one stage setup and holding costs, $\kappa(\vec{s},m,u,q,t)$, divided by the sum over the setup and production time over the four stages, $g((2,3.1,5.2,6.3),4) = 4(1+1)/6.4 = 1.25$.

Note, however, that $(\vec{s},m)$ is not strongly admissible, $(\vec{s},m) \notin \mathcal{S}^S$. Starting from $(\vec{s}-\vec{1}\epsilon,m) = ((2-\epsilon,3.1-\epsilon,5.2-\epsilon,6.3-\epsilon),4)$ we still need $t^1+t^2+t^3+t^4 \geq 0.4$ in order not to decrease total inventory. This is impossible because $t^i \leq 0.1-\epsilon < 0.1$ must hold for all i to prevent stock-outs.

Similarly, the state $(\vec{s},m) = ((5.2,3.1,2,6.3),4) \in \mathcal{S}$ (with a total inventory of 16.6) is not strongly admissible and leaves no other choice but actions of $u^n = 0$, $t^n = 0.1$ for all $n \in \mathbb{N}$, and $q^{1+4k} = 3$, $q^{2+4k} = 2$, $q^{3+4k} = 1$, and $q^{4+4k} = 4$ for all $k \in \mathbb{N}_0$ with an average cost of $g((5.2,3.1,2,6.3),4) = 4(5+1)/6.4 = 3.75$.

3.3. An infinite dimensional linear program

From the discussion above, we know that all states in $\mathcal{S}$ can be reached from states within the interior of $\mathcal{S}$, i.e. from states in $\mathcal{S}^S$, in a finite number of steps (and at finite costs). Hence, the minimum average costs $g(\vec{s},m)$ are equal for all states within $\mathcal{S}^S$ and equal the minimum average costs over all initial states in $\mathcal{S}$.

The standard approach to obtain the linear programming formulation is to write the optimality equations (6) and (7) as inequalities. Further postulating that the inequality corresponding to (7) holds for all $(u,q,t) \in \mathcal{A}(\vec{s},m) \setminus \mathcal{A}_g(\vec{s},m)$ and that the minimum average cost rate is a constant g gives the the infinite dimensional linear program

$$\sup g \tag{8}$$

$$h(\vec{s},m) - h(\vec{s'}(\vec{s},m,u,q,t),q) \leq \kappa(\vec{s},m,u,q,t) - g\cdot(u+\tau_{mq}+t) \;\; \forall(\vec{s},m) \in \mathcal{S}, (u,q,t) \in \mathcal{A}(\vec{s},m) \tag{9}$$

with unknown $h : \mathcal{S} \to \mathbb{R}$ and $g \in \mathbb{R}$. Since we included additional constraints, this program will return an average cost rate g that is at most as large as the minimum average cost rate that can be obtained irrespective of the starting state within $\mathcal{S}$.

If the replacement of $\mathcal{A}_g$ by $\mathcal{A}$ is feasible in (7), the program (8)-(9) will return a value g equal to this minimum average cost rate. Since it is not ensured that starting from states in $\mathcal{S}\backslash\mathcal{S}^S$ all other states in $\mathcal{S}$ can be reached, however, the minimum average costs $g(\vec{s},m)$ for states in $\mathcal{S}\backslash\mathcal{S}^S$ might still be larger than g.

4. An Affine Approximation

Unfortunately, it is impossible to solve (8)-(9), even for small problem instances. So we approximate the gain and bias function as follows

$$g = \sum_{i=1}^{I} \lambda_i V_i - \xi \tag{10}$$

$$h(\vec{s},m) \approx \theta_m - \sum_{i=1}^{I} s_i V_i, \qquad \forall(\vec{s},m) \in \mathcal{S}. \tag{11}$$

The approximation of the gain function allows to interpret it as the difference of the marginal value gained from serving demand per time unit (at rate λ_i demand is satisfied with products of marginal value V_i per unit) and the marginal value of machine time, ξ. The bias function is composed of a marginal value of being in machine state m, θ_m, and the total value of all products on inventory (all s_i items on inventory are weighted by their marginal values V_i). We subtract the total value of all products on inventory because high inventories of products allow to delay the costs of production

and holding the produced items. Hence, the expected total difference between the cost starting in $(\vec{s}, m)$ and the stationary cost should be decreasing in inventory.

Apply the approximation in (8)-(9). Using (10), (11) as well as $s'_q(\vec{s}, m, u, q, t) = s_q - \lambda_q \cdot (u + \tau_{mq} + t) + p_q t$ and $s'_i(\vec{s}, m, u, q, t) = s_i - \lambda_i \cdot (u + \tau_{mq} + t)$ for all $i \neq q$, some rearranging of terms gives

$$\begin{aligned}
&h(\vec{s}, m) - h(\vec{s'}(\vec{s}, m, u, q, t), q) + g \cdot (u + \tau_{mq} + t) \\
&= \theta_m - \sum_{i=1}^{I} s_i V_i - \theta_q + \sum_{i=1}^{I} s'_i V_i + \sum_{i=1}^{I} \lambda_i V_i \cdot (u + \tau_{mq} + t) - \xi \cdot (u + \tau_{mq} + t) \\
&= \theta_m - \theta_q + p_q t V_q - \xi \cdot (u + \tau_{mq} + t).
\end{aligned}$$

Then, (8)-(9) becomes the semi-infinite program

$$\max_{\xi, \theta_1, V_1, \ldots, \theta_I, V_I} \sum_{i=1}^{I} \lambda_i V_i - \xi \tag{12}$$

$$\theta_m - \theta_q + t p_q V_q - \xi \cdot (u + \tau_{mq} + t) \leq \kappa(\vec{s}, m, u, q, t) \quad \forall (\vec{s}, m) \in \mathcal{S}, (u, q, t) \in \mathcal{A}(\vec{s}, m) \tag{13}$$

$$\xi \in \mathbb{R}, \ V_i, \theta_i \in \mathbb{R} \qquad \forall i \in \mathcal{I}. \tag{14}$$

Expressing the set of all $(\vec{s}, m) \in \mathcal{S}, (u, q, t) \in \mathcal{A}(\vec{s}, m)$ in terms of $\mathcal{S}(m)$ and $\mathcal{A}(\vec{s}, m, q)$, (13) reads

$$\theta_m - \theta_q + t p_q V_q - \xi \cdot (u + \tau_{mq} + t) \leq \kappa(\vec{s}, m, u, q, t) \quad \forall m, q \in \mathcal{I}, \vec{s} \in \mathcal{S}(m), (u, t) \in \mathcal{A}(\vec{s}, m, q).$$

Note that we can set one value of the θ_i's arbitrarily. In the following we set $\theta_I = 0$. Because we restrict the bias function to this particular form, (12)-(14) gives a lower bound to the sequence-dependent ELSP problem (8)-(9).

Consider a dual formulation of (12)-(14),

$$\inf_{J, (\vec{s}_j, m_j), (u_j, q_j, t_j), x_j, j=1,\ldots,J} \sum_{j=1}^{J} \kappa(\vec{s}_j, m_j, u_j, q_j, t_j) \cdot x_j \tag{15}$$

$$\sum_{j=1}^{J} x_j t_j p_{q_j} 1\{q_j = i\} = \lambda_i, \quad i = 1, \ldots, I \tag{16}$$

$$\sum_{j=1}^{J} x_j \cdot (1\{m_j = i\} - 1\{q_j = i\}) = 0, \quad i = 1, \ldots, I-1 \tag{17}$$

$$\sum_{j=1}^{J} x_j \cdot (u_j + \tau_{m_j q_j} + t_j) = 1 \tag{18}$$

$$J \in \mathbb{N}_0, \ x_j \geq 0, \ (\vec{s}_j, m_j) \in \mathcal{S}, \ (u_j, q_j, t_j) \in \mathcal{A}(\vec{s}_j, m_j) \quad j = 1, \ldots, J. \tag{19}$$

This dual has a very intuitive appeal. Interpreting x_j as the frequency of being in state $(\vec{s}_j, m_j)$ and choosing action (u_j, q_j, t_j), the objective function (15) minimizes the average changeover and holding costs. The first constraint (16) states that for each product the demand rate must equal its time-averaged production rate. The second constraint (17) says that the rate at which changeovers to product i occur must equal the rate at which changeovers from product i are made. Since the action (u_j, q_j, t_j) in state $(\vec{s}_j, m_j)$ consumes idling, setup, and production time $u_j + \tau_{m_j q_j} + t_j$, the third constraint (18) represents the capacity restriction given a single machine.

Although the following theorem does not imply the solvability of problem (15)-(19), we can show that the infimum equals the solution of the solvable problem (12)-(14). We will use the intuition of (15)-(19) and the proof showing that its infimum equals the solution of (12)-(14) to compare our approach to an extension of Bomberger's (1966) lower bound problem to sequence-dependent setups in Section 5.

THEOREM 1. *The semi-infinite program (12)-(14) is solvable. Its value is equal to the value of problem (15)-(19).*

Proof. Let $\vec{r}$ denote the right-hand sides of (16)-(18) and define the $2I$-rowed column vectors $\vec{a}(s_j, m_j, u_j, q_j, t_j)$ so that (16)-(18) can be represented as

$$\sum_{j=1}^{J} \vec{a}(s_j, m_j, u_j, q_j, t_j) \cdot x_j = \vec{r}.$$

We first show that (12)-(14) is superconsistent by demonstrating that $\vec{r}$ has a maximal representation, i.e. $\sum_{j=1}^{2I} \vec{a}(\vec{s}_j, m_j, u_j, q_j, t_j) x_j = \vec{r}$ with $x_j > 0$ for all $j = 1, \ldots, 2I$ and $\vec{a}(\vec{s}_1, m_1, u_1, q_1, t_1), \ldots, \vec{a}(\vec{s}_{2I}, m_{2I}, u_{2I}, q_{2I}, t_{2I})$ linear independent, cf. Glashoff and Gustafson (1983), p.89, Lemma 18. Combining this with the fact that (15) is bounded by 0 (both $\kappa(\vec{s}_j, m_j, u_j, q_j, t_j)$ and x_j are non-negative for all j), gives the proposition, cf. Glashoff and Gustafson (1983), p.84, Theorem 7.

To constitute a maximal representation of $\vec{r}$, consider I solution pairs $x_i, \vec{s}_i, m_i, u_i, q_i, t_i$ and $x_{I+i}, \vec{s}_{I+i}, m_{I+i}, u_{I+i}, q_{I+i}, t_{I+i}$ with $m_i = i = q_{I+i}$ and $q_i = i+1 = m_{I+i}$ for all $i < I$, $m_I = I$, $q_I = 1$, $q_{2I} = I$, $m_{2I} = 1$, $u_i = u_{I+i} = 0$, $t_i = \rho_{i+1} T/2$ for $i < I$, $t_I = \rho_1 T/2$, and $t_{I+i} = \rho_i T/2$ with

$$T = \frac{\sum_{i=1}^{I-1} (\tau_{i,i+1} + \tau_{i+1,i}) + \tau_{1,I} + \tau_{I,1}}{1 - \sum_{i=1}^{I} \rho_i}$$

for all $i \in \mathcal{I}$ and set $\vec{s}_i$, $\vec{s}_{I+i}$ sufficiently large, e.g. to $2\vec{\lambda}T$. All actions (u_j, q_j, t_j) are in $\mathcal{A}(\vec{s}_j, m_j)$ for this choice because all inventories are large enough to execute (u_j, q_j, t_j) and then cycle through $1, \ldots, I$ without ever stocking out. This is true because $u_j + \tau_{m_j q_j} + t_j < T$ (by construction) leading to inventory that is at least $\vec{\lambda}T > \vec{\lambda}(\sum_{i=1}^{I-1}(\tau_{i,i+1}) + \tau_{I,1})/(1 - \sum_{i=1}^{I} \rho_i)$, the demand during the time needed to cycle through all products $1, \ldots, I$ once without idling and production times that restore the initial inventories.

Choosing $x_j = 1/T$ for all $j = 1, \ldots, 2I$, it is easy to verify that $\sum_{j=1}^{2I} \vec{a}(\vec{s}_j, m_j, u_j, q_j, t_j) x_j = \vec{r}$.

We show that the vectors $\vec{a}(\vec{s}_j, m_j, u_j, q_j, t_j)$ are linear independent by contradiction. Suppose a vector $\vec{a}(\vec{s}_{j'}, m_{j'}, u_{j'}, q_{j'}, t_{j'})$ were a linear combination of the other vectors $\vec{a}(\vec{s}_j, m_j, u_j, q_j, t_j)$, $j = 1, \ldots, 2I$, $j \neq j'$. To simplify notation, we assume $j' \leq I$ but the same argument can be made if $j' > I$. Note that the rows $I+1$ to $2I-1$ consist of at most one element +1 and one -1, all others being 0. They can be interpreted as arcs from one product j' to product $j'+1$ if $j' < I$ and to 1 if $j' = I$ (by construction). The only linear combination that gives the same values in these rows (i.e. represents the same path between the two products), is

$$\sum_{j=I+1}^{I+j'-1} \vec{a}(\vec{s}_j, m_j, u_j, q_j, t_j) + \vec{a}(\vec{s}_{2I}, m_{2I}, u_{2I}, q_{2I}, t_{2I}) + \sum_{j=j'+1}^{I} \vec{a}(\vec{s}_j, m_j, u_j, q_j, t_j).$$

(This corresponds to a path from j' to product 1, from 1 to I and from I to $j'+1$). Looking at the first I rows of the vector, however, this linear combination will lead to non-zero values in all rows but j', whereas $\vec{a}(\vec{s}_{j'}, m_{j'}, u_{j'}, q_{j'}, t_{j'})$ has a non-zero value only in row $j'+1$. Hence, the contradiction is established. □

The dual suggests that ξ, the Lagrangian multiplier of the capacity constraint, can be interpreted as cost of machine time. The following lemma shows that these costs are non-negative.

LEMMA 4. *For every feasible solution $\xi, V_1, \ldots, V_I, \theta_1, \theta_{I-1}$ to (12)-(14), the value of ξ is non-negative, $\xi \geq 0$.*

The intuition behind this lemma is the following: Suppose $\xi < 0$ in some feasible solution $\xi, V_1, \ldots, V_I, \theta_1, \ldots, \theta_{I-1}$. In this case, the left-hand side of (13) is increasing in u and the right-hand side of (13) is not changed if $\lambda_q u$ and s_q are increased at the same time by the same amount (see Figure 1 and equation (4)). So for given $s_q - \lambda_q u$, the left-hand side of (13) can be made arbitrarily large without changing the right-hand side, i.e. the value of $\kappa(\vec{s}, m, u, q, t)$. Hence, (13) is impossible to satisfy for all feasible combinations of $\vec{s}$, m, u, q, and t and the solution $\xi, V_1, \ldots, V_I, \theta_1, \ldots, \theta_{I-1}$ is not feasible. The proof follows this intuition, showing that there exist $(\vec{s}, m) \in \mathcal{S}$ and $(u, q, t) \in \mathcal{A}(\vec{s}, m)$ that lead to the contradiction.

Before we can go to the details of the proof, note that for any given production sequence $\vec{q} = (q^{[1]}, \ldots, q^{[N]})$ of length N, it is easy to show that there always exist initial inventories $\hat{s}(\vec{q})$ that are componentwise sufficiently large enough to repeatedly follow a non-idling cyclic schedule $\{(\vec{s}^{[n]}, m^{[n]}), (0, q^{[n]}, t^{[n]})\}_{n=1,\ldots,N}$ where production of a product is only started when its inventories are zero (zero-switch) and no stock-outs are allowed. This fact will play a role in the proof of this lemma and in Lemma 5. The calculation of $\hat{s}(\vec{q})$ is outlined in the appendix.

Proof. We prove the lemma by contradiction. Assuming $\xi < 0$, we will present values $m, q \in \mathcal{I}, \vec{s} \in \mathcal{S}(m), (u, t) \in \mathcal{A}(\vec{s}, m, q)$ for which (13) is infeasible.

Consider e.g. $m = I - 1$, $q = I$ and a production sequence $\vec{q} = (1, 2, \ldots, I)$. For any fixed values of $V_I, \theta_{I-1} \in \mathbb{R}$ and $\xi < 0$, choose

$$u = \max\left\{0, \frac{1}{\xi} \cdot \left[-\theta_{I-1} - \frac{\hat{s}_I(\vec{q})(p_I V_I - \xi)}{p_I - \lambda_I} + \xi\tau_{I-1,I} + c_{I-1,I} + \frac{\phi_I p_I \hat{s}_I^2(\vec{q})}{(p_I - \lambda_I)2\lambda_I} + 1\right]\right\}$$

with

$$\begin{aligned} t &= \hat{s}_I(\vec{q})/(p_I - \lambda_I) \\ s_I &= \lambda \cdot (u + \tau_{I-1,I}) \\ s_i &= \hat{s}_i(\vec{q}) + \lambda \cdot (u + \tau_{I-1,I} + t) \text{ for all } i \neq I. \end{aligned}$$

This choice ensures that there is enough starting inventory to switch to product I and produce for a time t leading to inventories of $\vec{\hat{s}}(\vec{q}) = (\hat{s}_1(\vec{q}), \ldots, \hat{s}_I(\vec{q}))$. By definition, $\vec{\hat{s}}(\vec{q})$ is in $\mathcal{S}(I) = \mathcal{S}(q)$. Hence, there is a feasible sequence of actions starting with (u, q, t) and then following a cyclic non-idling zero-switch schedule with sequence $\vec{q}$ that avoids stock-outs. Furthermore, the values of t and u are non-negative and lead to a state in $\mathcal{S}$. So, $(\vec{s}, I-1) \in \mathcal{S}$ and $(u, I, t) \in \mathcal{A}(\vec{s}, I-1)$.

Using $\theta_I = 0$, the left-hand side of (13) reads

$$\begin{aligned} &\theta_{I-1} + t \cdot (p_I V_I - \xi) - \xi\tau_{I-1,I} - \kappa(\vec{s}, m, u, q, t) - \xi u \\ &= \theta_{I-1} + \frac{\hat{s}_I(\vec{q})(p_I V_I - \xi)}{p_I - \lambda_I} - \xi\tau_{I-1,I} - c_{I-1,I} - \frac{\phi_I p_I \hat{s}_I^2(\vec{q})}{(p_I - \lambda_I)2\lambda_I} - \xi u \\ &\geq \theta_{I-1} + \frac{\hat{s}_I(\vec{q})(p_I V_I - \xi)}{p_I - \lambda_I} - \xi\tau_{I-1,I} - c_{I-1,I} - \frac{\phi_I p_I \hat{s}_I^2(\vec{q})}{(p_I - \lambda_I)2\lambda_I} \\ &\quad - \theta_{I-1} - \frac{\hat{s}_I(\vec{q})(p_I V_I - \xi)}{p_I - \lambda_I} + \xi\tau_{I-1,I} + c_{I-1,I} + \frac{\phi_I p_I \hat{s}_I^2(\vec{q})}{(p_I - \lambda_I)2\lambda_I} + 1 \\ &= 1 > 0 \end{aligned}$$

Hence, for any values of $V_1, \ldots, V_I$, $\theta_1, \ldots, \theta_{I-1}$ and $\xi < 0$ we can always find a violated constraint (13) and there will be no feasible solution with $\xi < 0$. $\square$

4.1. The set of constraints

We have shown that the semi-infinite linear optimization problem (12)-(14) is solvable. In this section we will simplify the representation of (13), the set of an uncountable number of linear constraints.

Suppose that constraint (13) is violated for given values of $\theta_1, \ldots, \theta_{I-1}$, $V_1, \ldots, V_I$, and ξ. Then,

$$\sup_{\vec{s},m,u,q,t} \theta_m - \theta_q + tp_qV_q - \xi \cdot (u + \tau_{mq} + t) - c_{mq} - \frac{\phi_q p_q t}{2\lambda_q} \cdot (2s_q - 2\lambda_q \cdot (u + \tau_{mq}) + t(p_q - \lambda_q)) \quad (20)$$

$$\vec{s} \in \mathcal{S}(m) \quad (21)$$

$$s_q - \lambda_q \cdot (u + \tau_{mq}) \geq 0 \quad (22)$$

$$\vec{s} - \vec{\lambda} \cdot (u + \tau_{mq} + t) + \vec{p}\, \vec{e}_q t \in \mathcal{S}(q) \quad (23)$$

$$m, q \in \mathcal{I}, \quad u, t \geq 0 \quad (24)$$

has a positive value (where $\vec{e}_q$ denotes a unit vector with element 1 in row q and 0 else).

As mentioned before, the set of admissible inventory positions given the machine is setup for state m, $\mathcal{S}(m)$, might be neither convex nor closed and it is NP hard to determine if a given inventory position even is in $\mathcal{S}(m)$ or not. Hence, this problem seems complicated at first sight. It can, however, be simplified substantially because the state $(\vec{s}, m)$ is a variable.

Let τ^{sp}_{mq} be the length of the shortest setup time path from product m to product q and $\vec{\eta}_{mq}$ represent the sequence of products visited on this path including the end product q but not the starting product m. So if the shortest setup time path is to go directly from m to q, we have $\tau^{sp}_{mq} = \tau_{mq}$ and $\vec{\eta}_{mq} = (q)$. If the triangle-inequality $\tau_{mq} \leq \tau_{mi} + \tau_{iq}$ is satisfied for all $m, i, q \in \mathcal{I}$, this direct setup is always shortest, but it need not be the shortest setup path in general. Similarly, in the case of $m = q$ the shortest setup path from product m to itself can be to go to m directly, with $\tau^{sp}_{mm} = \tau_{mm}$ and $\eta_{mm} = (m)$ but we do not restrict our analysis to this case. Further, let $\vec{q}_{\eta_{mq}}$ be a sequence of products that starts with the sequence given in $\vec{\eta}_{mq}$ followed by any (arbitrary) permutation of the products $1, 2, \ldots, I$.

Given this notation, we can simplify the set of constraints as follows.

Lemma 5. *In problem (20)-(24), constraints (21), (22), and (23) can be replaced by*

$$s_q \geq \max\{\lambda_q \cdot (u + \tau_{mq}), \lambda_q \cdot (u + \tau_{mq}) - (p_q - \lambda_q)t + \lambda_q \tau^{sp}_{qq}\} \quad (25)$$

$$s_i \geq \max\{\hat{s}_i(\vec{q}_{\eta_{mq}}), \hat{s}_i(\vec{q}_{\eta_{qq}}) + \lambda_i \cdot (u + \tau_{mq} + t)\} \qquad \textit{for all } i \neq q \quad (26)$$

without changing the optimal objective value in (20).

Proof. As a first step, note that the inventory levels of products i with $i \neq q$ have no impact on the objective function (20) and that the structure of $\mathcal{S}(m)$ (and $\mathcal{S}(q)$) is such that given inventory levels $s_1, \ldots, s_I$ in $\mathcal{S}(m)$ ($\mathcal{S}(q)$), all inventory levels with components larger than s_i for all i are also in $\mathcal{S}(m)$ ($\mathcal{S}(q)$). So actually, we can choose all the s_i's for $i \neq q$ to be arbitrarily large to ensure feasibility without affecting (20).

As a second step, consider a state with current machine state m and $s_q < \lambda_q \tau^{sp}_{mq}$. In this case stock-outs are unavoidable because there is not enough inventory to cover the demand that is observed before product q can be set up. Hence condition (21) is violated. Choosing $s_q \geq \lambda_q \tau^{sp}_{mq}$ and setting the other inventory levels to be large enough to allow for a cyclic schedule starting with the shortest setup path to q and visiting each product at least once, e.g.

$$s_i \geq \hat{s}_i(\vec{q}_{\eta_{mq}}) \qquad \text{for all } i \neq q,$$

however, ensures $\vec{s} \in \mathcal{S}(m)$. (Remember that $\hat{s}_i(\vec{q}_{\eta_{mq}})$ represents the initial inventory of product i that suffices to avoid stock outs executing a non-idling cyclic zero-switch schedule with a production

sequence that starts with the shortest setup path from m to q followed by an arbitrary but given permutation of all products.)

Similarly, if the inventory of product q after action (u,q,t) in state $(\vec{s},m)$, $s_q-\lambda_q\cdot(u+\tau_{mq}+t)+p_qt$, does not suffice to cover the demand that is incurred before a second production run of this product can be started, condition (23) will not be met. Since the shortest time to set up product q given the machine is set up for q is τ_{qq}^{sp}, this situation occurs if $s_q-\lambda_q\cdot(u+\tau_{mq}+t)+p_qt<\lambda_q\tau_{qq}^{sp}$. Choosing $s_q-\lambda_q\cdot(u+\tau_{mq}+t)+p_qt\geq\lambda_q\tau_{qq}^{sp}$ and again setting the other inventory levels large enough to allow for this second setup of q followed by a rotation through all products in $\mathcal{I}$, e.g.

$$s_i-\lambda_i\cdot(u+\tau_{mq}+t)\geq\hat{s}_i(\vec{q}_{\eta_{qq}})\qquad\text{for all } i\neq q,$$

ensures $\vec{s}\in\mathcal{S}(q)$.

Summarizing the above, adding constraint (22), and using that $\tau_{mq}^{sp}\leq\tau_{mq}$ and $u\geq 0$ gives

$$\begin{aligned}
s_q &\geq \max\{\lambda_q\tau_{mq}^{sp},\lambda_q\cdot(u+\tau_{mq}),\lambda_q\cdot(u+\tau_{mq}+t)-p_qt+\lambda_q\tau_{qq}^{sp}\}\\
&=\max\{\lambda_q\cdot(u+\tau_{mq}),\lambda_q\cdot(u+\tau_{mq})-(p_q-\lambda_q)t+\lambda_q\tau_{qq}^{sp}\},\\
s_i &\geq \max\{\hat{s}_i(\vec{q}_{\eta_{mq}}),\hat{s}_i(\vec{q}_{\eta_{qq}})+\lambda_i\cdot(u+\tau_{mq}+t)\}\qquad\text{for all } i\neq q.\quad\square
\end{aligned}$$

Using this lemma, we can simplify the set of constraints even further as the following proposition shows.

Proposition 1. *Problem (20)-(24) has the same optimal solution and objective value as*

$$\max_{m,q\in\mathcal{I}}\ \theta_m-\theta_q-\xi\tau_{mq}-c_{mq}+\nu_q(V_q,\xi)\tag{27}$$

with

$$\nu_q(V_q,\xi)=\begin{cases}\frac{(p_qV_q-\xi)^2\lambda_q}{2\phi_qp_q\cdot(p_q-\lambda_q)} & \textit{if } p_qV_q-\xi\geq\tau_{qq}^{sp}\phi_qp_q\\ \frac{\lambda_q\tau_{qq}^{sp}}{p_q-\lambda_q}\cdot\left(p_qV_q-\xi-\frac{1}{2}\tau_{qq}^{sp}\phi_qp_q\right) & \textit{if } 2(p_qV_q-\xi)\geq\tau_{qq}^{sp}\phi_qp_q>p_qV_q-\xi\\ 0 & \textit{if } 2(p_qV_q-\xi)<\tau_{qq}^{sp}\phi_qp_q.\end{cases}$$

Proof. From Lemma 5, we know that we can replace (21)-(23) by (25)-(26) and consider the problem (20), (24),(25), and (26).

If there exists a feasible solution $\vec{s}$, m, u, q, t, we can always set the inventory levels s_i, $i\neq q$, according to (26) without changing the objective value. So in our optimization problem, we can neglect all variables s_i with $i\neq q$ and the corresponding condition (26). Further, observe that the objective function is decreasing in s_q. So we want to make s_q as low as possible. The lowest value is obtained when there is equality in (25), i.e.

$$s_q=\max\{\lambda_q\cdot(u+\tau_{mq}),\lambda_q\cdot(u+\tau_{mq})-(p_q-\lambda_q)\cdot t+\lambda_q\tau_{qq}^{sp}\}.$$

Plugging the value of s_q into (20), (24), yields

$$\begin{aligned}
&\max_{u,t\geq0,m,q\in\mathcal{I}}\theta_m-\theta_q+tp_qV_q-\xi\cdot(u+\tau_{mq}+t)-c_{mq}\\
&\quad-\frac{\phi_qp_qt}{2\lambda_q}\cdot\Big(2\max\{\lambda_q\cdot(u+\tau_{mq}),\lambda_q\cdot(u+\tau_{mq})-(p_q-\lambda_q)t+\lambda_q\tau_{qq}^{sp}\}-2\lambda_q\cdot(u+\tau_{mq})+t\cdot(p_q-\lambda_q)\Big)\\
&=\max_{u,t\geq0,m,q\in\mathcal{I}}\theta_m-\theta_q+tp_qV_q-\xi\cdot(u+\tau_{mq}+t)-c_{mq}\\
&\quad-\frac{\phi_qp_qt}{2\lambda_q}\cdot\Big(2\max\{\lambda_q\tau_{qq}^{sp}-(p_q-\lambda_q)\cdot t,0\}+t\cdot(p_q-\lambda_q)\Big).
\end{aligned}$$

From Lemma 4, we know $\xi \geq 0$. Therefore, the objective is decreasing linearly in u and the maximum is achieved for $u = 0$. The problem then simplifies to

$$\max_{t\geq 0, m,q\in\mathcal{I}} \theta_m - \theta_q + tp_qV_q - \xi\cdot(\tau_{mq}+t) - c_{mq} - \frac{\phi_q p_q t}{2\lambda_q}\cdot\Big(2\max\{0, \lambda_q\tau_{qq}^{sp} - (p_q-\lambda_q)t\} + t\cdot(p_q-\lambda_q)\Big).$$

To find the maximum over t given m and q, we analyze the two cases $t \leq (\lambda_q\tau_{qq}^{sp})/(p_q-\lambda_q)$ and $t \geq (\lambda_q\tau_{qq}^{sp})/(p_q-\lambda_q)$ separately. In the first case, $\lambda_q\tau_{qq}^{sp} - (p_q-\lambda_q)t \geq 0$, in the second case, the inequality is reversed. So we consider

$$\max_{m,q\in\mathcal{I}} \theta_m - \theta_q - \xi\tau_{mq} - c_{mq} + \max\Bigg\{\max_{0\leq t\leq(\lambda_q\tau_{qq}^{sp})/(p_q-\lambda_q)}\Big\{t\cdot\big(p_qV_q - \xi - \phi_q p_q\tau_{qq}^{sp}\big)$$
$$+ t^2\cdot\frac{\phi_q p_q\cdot(p_q-\lambda_q)}{2\lambda_q}\Big\}, \max_{t\geq(\lambda_q\tau_{qq}^{sp})/(p_q-\lambda_q)}\Big\{t\cdot(p_qV_q-\xi) - t^2\cdot\frac{\phi_q p_q\cdot(p_q-\lambda_q)}{2\lambda_q}\Big\}\Bigg\}.$$

Now let us write $t_q = (\lambda_q\tau_{qq}^{sp})/(p_q-\lambda_q) < \infty$ for short. The first argument of the (inner) maximum represents the maximum over a function that is convex in t and restricted to the interval $t \in [0, t_q]$. Hence, the maximum is either at $t = t_q$ with value

$$\frac{\lambda_q\tau_{qq}^{sp}}{p_q-\lambda_q}\cdot\left(p_qV_q - \xi - \frac{1}{2}\tau_{qq}^{sp}\phi_q p_q\right)$$

if $2(p_qV_q - \xi) \geq \tau_{qq}^{sp}\phi_q p_q$ or at $t = 0$ with a value of 0 otherwise. The second argument of the (inner) maximum represents the maximum over a function that is concave in t restricted to $t \in [t_q, \infty)$. Therefore, the maximum is at the extremum of the unrestricted function, at $t = t^* = (p_qV_q - \xi)\lambda_q/(\phi_q p_q(p_q-\lambda_q)) < \infty$, if this value is in the feasible area $[t_q, \infty)$, i.e. if

$$t^* = \frac{(p_qV_q-\xi)\lambda_q}{\phi_q p_q(p_q-\lambda_q)} \geq t_q = \frac{\lambda_q\tau_{qq}^{sp}}{p_q-\lambda_q} \Leftrightarrow p_qV_q - \xi \geq \tau_{qq}^{sp}\phi_q p_q.$$

The maximum then has a value of

$$\frac{(p_qV_q-\xi)^2\lambda_q}{2\phi_q p_q(p_q-\lambda_q)}.$$

If $t^* \notin [t_q, \infty)$, we know that the function is decreasing within $[t_q, \infty)$. In this case the maximum is again at the boundary $t = t_q$. To sum things up, solving (20)-(24) is equivalent to finding (27). □

4.2. Reduction to a convex problem

In light of Proposition 1, two approaches for solving (12)-(14) seem straightforward. First, one might approach the problem by solving (15)-(19) using column generation. A starting set of feasible $(\vec{s}, m, u, q, t)$ combinations can be derived from simple cyclic schedules. The subproblems reduce to the calculation of (27).

Second, one could exploit the structure of the conditions given in (27) directly. In this case, we look for values of ξ, $V_1, \ldots, V_I$ and $\theta_1, \ldots, \theta_{I-1}$ that solve

$$\max_{\xi\geq 0, V_1,\ldots,V_I,\theta_1,\ldots,\theta_{I-1}} \sum_{i=1}^{I}\lambda_i V_i - \xi \tag{28}$$

$$\text{s.t. } \theta_m - \theta_q - \xi\tau_{mq} - c_{mq} + \nu_q(V_q, \xi) \leq 0 \qquad \forall\ m, q \in \mathcal{I} \tag{29}$$

using $\theta_I = 0$. This problem could be solved directly (as a mixed integer convex quadratically constrained linear program). A convex quadratically constrained program that gives a lower bound to (28)-(29) is given in the following theorem. If $\tau_{qq}^{sp} \leq \tau_{mq}(1-\rho_q)/(1-\sum_{i=1}^{I}\rho_i)$ for all $m,q \in \mathcal{I}$, this program gives the same solution and objective value as (28)-(29). Since $(1-\rho_q)/(1-\sum_{i=1}^{I}\rho_i) > 1$, this condition is weaker than a requirement of the form $\tau_{qq}^{sp} \leq \tau_{mq}$ for all $m,q \in \mathcal{I}$ and is fulfilled in many practical applications.

Theorem 2. *The program*

$$\max_{\xi \geq 0, V_1,\ldots,V_I,\theta_1,\ldots,\theta_{I-1}} \sum_{i=1}^{I} \lambda_i V_i - \xi \tag{30}$$

$$s.t.\ \theta_m - \theta_q - \xi\tau_{mq} - c_{mq} + \frac{(p_q V_q - \xi)^2 \lambda_q}{2\phi_q p_q (p_q - \lambda_q)} \leq 0 \qquad \forall m,q \in \mathcal{I} \tag{31}$$

$(\theta_I = 0)$ *is convex quadratically constrained and gives a lower bound to problem (28)-(29). If*

$$\tau_{qq}^{sp} \leq \tau_{mq}(1-\rho_q)/(1-\sum_{i=1}^{I}\rho_i) \quad \forall m,q \in \mathcal{I}, \tag{32}$$

then (28)-(29) and (30)-(31) are equivalent.

Proof. In our proof, a different formulation of (28)-(29) will be more convenient to work with. To obtain it, substitute the net marginal value rates from production $y_q := p_q V_q - \xi$ and write

$$b_q(\vec{\theta},\xi) := \min_{m \in \mathcal{I}} \{c_{mq} + \xi\tau_{mq} - \theta_m + \theta_q\} \tag{33}$$

with $\vec{\theta} = (\theta_1,\ldots,\theta_I)$ and $\theta_I = 0$ to simplify notation. Because $\nu_q(V_q,\xi) \geq 0$ for all q, we can conclude from (29) that for all feasible solutions, $b_q(\vec{\theta},\xi) \geq 0$ for all $q \in \mathcal{I}$. Then, (28)-(29) can be written as

$$\max_{\xi \geq 0, y_1,\ldots,y_I,\theta_1,\ldots,\theta_{I-1}} \sum_{i=1}^{I} \rho_i y_i - (1-\sum_{i=1}^{I}\rho_i)\xi \tag{34}$$

$$y_q \leq \sqrt{\frac{2\phi_q p_q (p_q - \lambda_q)}{\lambda_q} \cdot b_q(\vec{\theta},\xi)} \qquad \forall q : y_q \geq \tau_{qq}^{sp}\phi_q p_q \tag{35}$$

$$y_q \leq \frac{1}{2}\tau_{qq}^{sp}\phi_q p_q + \frac{p_q - \lambda_q}{\lambda_q \tau_{qq}^{sp}} \cdot b_q(\vec{\theta},\xi) \quad \forall q : \tau_{qq}^{sp} > 0,\ \frac{\tau_{qq}^{sp}\phi_q p_q}{2} \leq y_q \leq \tau_{qq}^{sp}\phi_q p_q \tag{36}$$

$$b_q(\vec{\theta},\xi) \geq 0 \quad \forall q \in \mathcal{I} \tag{37}$$

For $y_q = \tau_{qq}^{sp}\phi_q p_q$ conditions (35) and (36) coincide, and for $y_q = \tau_{qq}^{sp}\phi_q p_q/2$ conditions (36) and (37) coincide. Therefore, given ξ either (35) or (36) will be tight for each q - otherwise (34) could be improved.

In a first step, we show that the problem is never relaxed by dropping (36) and postulating (35) for all q (instead of only imposing this restriction for all q with $y_q \geq \tau_{qq}^{sp}\phi_q p_q$). We show the convexity of our problem in the second step. In a third step, we show that if (32) holds, any solution $\xi, y_1,\ldots,y_I,\theta_1,\ldots,\theta_{I-1}$ to (34)-(37), with objective value g and $y_q < \tau_{qq}^{sp}\phi_q p_q$ for some q, can be transformed into a solution with $y'_q \geq \tau_{qq}^{sp}\phi_q p_q$ for all $q \in \mathcal{I}$ and an objective value $g' \geq q$. Hence, the tightening of the conditions has no impact on the objective value in this case.

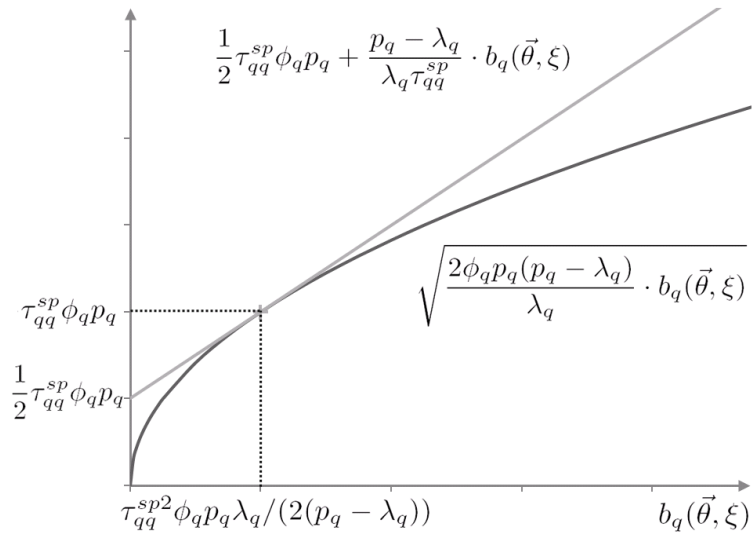

Figure 2 The region of feasible y_q's.

First step. We first show that the problem is never relaxed by postulating (35) for all y_q because the right-hand side of (35) is always lower than the right-hand-side of (36). To see this, view both as a function of $b_q(\vec{\theta},\xi)$ (as defined in (33)). Then, the right-hand side of (35) is concave in $b_q(\vec{\theta},\xi)$ and (36) represents a tangent through $(b_q(\vec{\theta},\xi)=\tau_{qq}^{sp2}\phi_q p_q \lambda_q/(2(p_q-\lambda_q)), \tau_{qq}^{sp}\phi_q p_q)$, see Figure 2. So the problem

$$\max_{\xi\geq 0, y_1,\ldots,y_I,\theta_1,\ldots,\theta_{I-1}} \sum_{q=1}^{I} \rho_q y_q - (1-\sum_{q=1}^{I}\rho_q)\xi \tag{38}$$

$$y_q \leq \sqrt{\frac{2\phi_q p_q(p_q-\lambda_q)}{\lambda_q} \cdot b_q(\vec{\theta},\xi)} \quad \forall q \in \mathcal{I} \tag{39}$$

$$b_q(\vec{\theta},\xi) \geq 0 \quad \forall q \in \mathcal{I} \tag{40}$$

has tighter constraints than (34)-(37) and hence an objective value that is no larger than the value of the original problem. Condition (40) is superfluous, because (39) implies $b_q(\vec{\theta},\xi) \geq 0.5\lambda_q y_q^2/(\phi_q p_q(p_q-\lambda_q)) \geq 0$. Program (30)-(31) is obtained from (38)-(39) by resubstituting $y_q = p_q V_q - \xi$ and $b_q(\vec{\theta},\xi)$.

Second step. To see that the quadratic problem (30)-(31) is convex, note that the objective (30) is linear. Therefore, we only need to show that (31) is convex for all $m,q \in \mathcal{I}$. The Hessian matrix of the left-hand side of (31) for fixed m and q has non-zero values only in the rows and columns that correspond to the variables ξ and V_q. Hence, to prove that the Hessian is positive semi-definite and the left-hand side of (31) convex for all $m,q \in \mathcal{I}$, it suffices to ensure that $\vec{z}^T H_{mq}\vec{z} \geq 0$ with $\vec{z}=(z_1,z_2)$ and H_{mq} representing the 2×2 matrix consisting of these values. We obtain

$$\begin{aligned}\vec{z}^T H_{mq}\vec{z} &= \vec{z}^T \begin{pmatrix} \frac{\lambda_q}{p_q\phi_q(p_q-\lambda_q)} & \frac{-\lambda_q}{\phi_q(p_q-\lambda_q)} \\ \frac{-\lambda_q}{\phi_q(p_q-\lambda_q)} & \frac{\lambda_q p_q}{\phi_q(p_q-\lambda_q)} \end{pmatrix} \vec{z} = \frac{\lambda_q p_q}{\phi_q(p_q-\lambda_q)} \vec{z}^T \begin{pmatrix} \frac{1}{p_q^2} & \frac{-1}{p_q} \\ \frac{-1}{p_q} & 1 \end{pmatrix} \vec{z} \\ &= \frac{\lambda_q p_q}{\phi_q(p_q-\lambda_q)} \cdot \left(z_1 - \frac{z_2}{p_q}\right)^2 \geq 0.\end{aligned}$$

Third step. First, we show by contradiction that there is no optimal solution with $y_q^* < \tau_{qq}^{sp}\phi_q p_q/2$. For fixed ξ, an optimal solution to (34)-(37) would make each y_q as large as possible. So assume there is an optimal ξ^* and a q with $y_q^* < \tau_{qq}^{sp}\phi_q p_q/2$. Then, a solution with $y_q^* = \tau_{qq}^{sp}\phi_q p_q/2$ would still be feasible and would have a larger objective value. Hence, we have a contradiction.

Now assume (32) and that there is an optimal solution with ξ^* and $\tau_{qq}^{sp}\phi_q p_q/2 \le y_q^* < \tau_{qq}^{sp}\phi_q p_q$ for some $q \in \mathcal{I}$. For this q, let m' be the minimizer of (33). Then, an increase of ξ^* to $\xi^* + \epsilon$ (with $\epsilon > 0$ sufficiently small) allows y_q^* to increase by $\epsilon\tau_{m'q}(p_q - \lambda_q)/(\lambda\tau_{qq}^{sp})$. Since all other constraints on the y_q's are only relaxed by this increase, this increase has no impact on the feasibility of the y_q's. The objective, however, is changed by

$$\epsilon\left(\frac{\rho_q\tau_{m'q}(p_q-\lambda_q)}{\lambda\tau_{qq}^{sp}} - (1-\sum_{i=1}^{I}\rho_i)\right) = \epsilon\left(\frac{\tau_{m'q}(1-\rho_q)}{\tau_{qq}^{sp}} - (1-\sum_{i=1}^{I}\rho_i)\right),$$

which is nonnegative by assumption. Hence, if (32) holds, it suffices to consider solutions with $y_q \ge \tau_{qq}^{sp}\phi_q p_q$ for all $q \in \mathcal{I}$ without loss of optimality. □

Another, more intuitive way to view problem (30)-(31) is to see it as a relaxation of the set of feasible actions. The left-hand side of (31) is the solution to (20)-(24) when constraint (23) is dropped.

Lemma 6. *The convex program (30)-(31) corresponds to a relaxation of the action space $\mathcal{A}(\vec{s}, m)$ to*

$$\tilde{\mathcal{A}}(\vec{s}, m) = \{(u, q, t) : t, u \ge 0, q \in \mathcal{I}, s_q - \lambda_q \cdot (u + \tau_{mq}) \ge 0\} \supseteq \mathcal{A}(\vec{s}, m)$$

when solving (12)-(14).

Proof. Starting again from (12)-(14), relaxing $\mathcal{A}(\vec{s}, m)$ to $\tilde{\mathcal{A}}(\vec{s}, m)$ translates into dropping condition (23) in (20)-(24). Hence, conditions (25) and (26) simplify to

$$\begin{aligned} &s_q \ge \lambda_q \cdot (u + \tau_{mq}) \\ &s_i \ge \hat{s}_i(\vec{q}_{\eta_{mq}}) \qquad \text{for all } i \ne q. \end{aligned}$$

Using this simplification in the proof of Proposition 1, yields

$$\begin{aligned} &\max_{u,t\ge 0, m,q\in\mathcal{I}} \theta_m - \theta_q + tp_qV_q - \xi\cdot(u + \tau_{mq} + t) - c_{mq} - \frac{\phi_q p_q t^2(p_q - \lambda_q)}{2\lambda_q} \\ = &\max_{t\ge 0, m,q\in\mathcal{I}} \theta_m - \theta_q + tp_qV_q - \xi\cdot(\tau_{mq} + t) - c_{mq} - \frac{\phi_q p_q t^2(p_q - \lambda_q)}{2\lambda_q} \end{aligned}$$

since the objective is again decreasing linearly in u. So for fixed m and q, the maximum of this function (that is concave in t) is either at

$$t^* = \frac{(p_qV_q - \xi)\lambda_q}{\phi_q p_q(p_q - \lambda_q)} \ge t_q = \frac{\lambda_q\tau_{qq}^{sp}}{p_q - \lambda_q} \Leftrightarrow p_qV_q - \xi \ge \tau_{qq}^{sp}\phi_q p_q$$

with value

$$\theta_m - \theta_q - \xi\tau_{mq} - c_{mq} + \frac{(p_qV_q - \xi)^2\lambda_q}{2\phi_q p_q(p_q - \lambda_q)}$$

if $t^* \ge 0$, it is at $t^* = 0$ with value $\theta_m - \theta_q - \xi\tau_{mq} - c_{mq}$ otherwise. Hence, we obtain the result of Proposition 1 with

$$\nu_q(V_q, \xi) = \begin{cases} \frac{(p_qV_q-\xi)^2\lambda_q}{2\phi_q p_q\cdot(p_q-\lambda_q)} & \text{if } p_qV_q - \xi \ge 0 \\ 0 & \text{if } p_qV_q - \xi < 0 \end{cases}$$

in (27). As explained in the third step of the proof to Theorem 2, an optimal solution has the largest possible value of $y_q = p_q V_q - \xi$ for each q. In the solution of (28)-(29) with this $\nu_q(V_q, \xi)$, we will therefore have $y_q = p_q V_q - \xi \geq 0$ establishing the direct connection of (30)-(31) and (12)-(14). □

4.3. Properties of the parameters

From Lemma 4 we know that the marginal value of machine-time is always non-negative. Since we showed in the third step of the proof to Theorem 2 that irrespective of condition (32) an optimal solution fulfills

$$p_q V_q - \xi \geq \tau_{qq}^{sp} \phi_q p_q / 2 \qquad \Leftrightarrow \qquad V_q \geq \xi / p_q + \tau_{qq}^{sp} \phi_q / 2 \quad (\geq 0)$$

for all q, we additionally know that the product values are always non-negative. If condition (32) holds, we can further conclude from the third step of the proof of Theorem 2:

Corollary 1. *Given (32), the marginal value of one unit of product q, V_q, fulfills $V_q \geq \tau_{qq}^{sp} \phi_q + \xi / p_q$.*

Intuitively, this result means that the value V_q of one marginal unit of product q should be at least as large as the value of the machine time to produce it, ξ / p, plus the marginal inventory costs for that unit while inventory is depleted over the smallest possible time until q is produced again, $\tau_{qq}^{sp} \phi_q$.

In addition, it makes sense that machine-time has no value if all setup times are 0. Intuitively this follows from the assumption that the total utilization is smaller than 1.

Lemma 7. *In a problem with $\tau_{ij} = 0$ for all $i, j \in \mathcal{I}$, every optimal solution to (28)-(29) gives $\xi = 0$.*

Proof. Using $\tau_{ij} = 0$ for all $i, j \in \mathcal{I}$ and $b_q(\vec{\theta}) = \min_{m \in \mathcal{I}} \{\theta_q - \theta_m + c_{mq}\}$, (34)-(37) simplifies to

$$\max_{\xi \geq 0, y_1, \ldots, y_I, \theta_1, \ldots, \theta_{I-1}} \sum_{i=1}^{I} \rho_i y_i - (1 - \sum_{i=1}^{I} \rho_i) \xi$$

$$y_q \leq \sqrt{\frac{2 \phi_q p_q (p_q - \lambda_q)}{\lambda_q} \cdot b_q(\vec{\theta})} \qquad \forall q : y_q \geq 0$$

since $y_q \geq 0$ for all $q \in \mathcal{I}$. Because $(1 - \sum_{i=1}^{I} \rho_i) > 0$, it is obvious that every solution $\xi, y_1, \ldots, y_I, \theta_1, \theta_{I-1}$ with $\xi > 0$ to this problem can be improved by setting $\xi = 0$ without violating any condition. Hence, no solution with $\xi = 0$ can be optimal. □

Loosely speaking, the values θ_q can be interpreted as the value of having product q set up. A setup from product m to q hence creates costs of $\theta_m - \theta_q$. The following lemma says that in a sequence-independent setting, there is no gain or loss from setting up another product (in addition to the cost of the machine time used).

Lemma 8. *If both the setup times and the setup costs are sequence-independent, there exists an optimal solution to (28)-(29) with $\theta_q = 0$ for all $q \in \mathcal{I}$.*

Proof. In the sequence-independent setting with setup costs $c_1, \ldots, c_I$ and setup times $\tau_1, \ldots, \tau_I$, we can reformulate problem (28)-(29) to read

$$\max_{\xi \geq 0, V_1, \ldots, V_I, \theta_1, \ldots, \theta_{I-1}} \sum_{i=1}^{I} \lambda_i V_i - \xi$$

$$\text{s.t.} \; -\xi \tau_q - c_q + \nu_q(V_q, \xi) \leq \min_{m \in \mathcal{I}} \{\theta_q - \theta_m\} \qquad \forall \; q \in \mathcal{I}$$

with $\theta_I = 0$. Since $\min_{m \in \mathcal{I}} \{\theta_q - \theta_m\} \leq \theta_q - \theta_q = 0$ for all values of $\theta_1, \ldots, \theta_{I-1}$ and q, the feasible area is largest if all right-hand sides of the constraints equal 0. This is obtained for $\theta_1 = \cdots = \theta_{I-1} = \theta_I = 0$. □

5. Comparison to other lower bounds

In this section, we introduce a flow-based nonlinear problem that captures the intuition behind the dual formulation (15)-(19). We then compare the lower bound problems (28)-(29) and (30)-(31) to this nonlinear problem and to the lower bound problems introduced by Dobson (1992) and Bomberger (1966).

5.1. A flow-based lower bound problem

Let us simplify the problem in the following way: every time the machine state is changed from product m to product q, product q is produced for a time t_{mq} and production is only started when there is zero inventory of this product, i.e. $s_q = \lambda_q(u + \tau_{mq})$ holds. We denote the rate of these changeovers from m to q by x_{mq}. Average costs then equal the sum over all $\kappa(\lambda_q(u+\tau_{mq}), m, u, q, t) \cdot x_{mq}$, i.e. the objective can be written as

$$\min \sum_{m=1}^{I} \sum_{q=1}^{I} \Big(c_{mq} + \frac{\phi_q t_{mq}^2 p_q (p_q - \lambda_q)}{2\lambda_q} \Big) x_{mq}. \tag{41}$$

Consider the following constraints:

$$\sum_{m=1}^{I} p_q t_{mq} x_{mq} = \lambda_q \qquad \text{for all } q \in \mathcal{I} \tag{42}$$

$$\sum_{i=1}^{I} x_{mi} = \sum_{i=1}^{I} x_{im} \qquad \text{for all } m \in \mathcal{I} \tag{43}$$

$$\sum_{m=1}^{I} \sum_{q=1}^{I} (\tau_{mq} + t_{mq}) x_{mq} \leq 1 \tag{44}$$

$$x_{mq}, t_{mq} \geq 0 \qquad \text{for all } m, q \in \mathcal{I}. \tag{45}$$

Condition (42) says that the average demand should equal average production of each product. Further, we postulate that the average rate of changeovers into machine state m should equal the average rate of changeovers leaving machine state m, (43). Because the time needed for a changeover from m to q is at least $\tau_{mq} + t_{mq}$ (and higher if there is idling), the fact that there is only one machine translates into (44). This condition can also be stated as $\sum_{m=1}^{I} \sum_{q=1}^{I} \tau_{mq} x_{mq} \leq 1 - \sum_{q=1}^{I} \rho_q$.

The following theorem demonstrates that (30)-(31) can be viewed as the dual of (41)-(45). Furthermore, a direct connection between our approximated semi-MDP (15)-(19) and (41)-(45) is established.

Theorem 3. *The nonlinear problem (41)-(45) gives a lower bound to (15)-(19) and is equivalent to (30)-(31).*

Proof. Using the relaxed action space introduced in Lemma 6, introduce the following relaxation of (19),

$$J \in \mathbb{N}_0,\ x_j \geq 0,\ (\vec{s}_j, m_j) \in \mathcal{S},\ (u_j, q_j, t_j) \in \tilde{\mathcal{A}}(\vec{s}_j, m_j) \quad j = 1, \ldots, J. \tag{46}$$

We will use three steps to proof the proposition. First, we show that there is a mapping of every feasible solution to (15)-(18), (46) to a feasible solution to (41)-(45) with an objective value of equal or lower value. We can hence conclude that (41)-(45) gives a lower bound to the problem (15)-(18), (46). Since (46) relaxes (19), this shows the first part of the theorem.

In a second step, we then show that every feasible solution to (41)-(45) can be transformed into a feasible solution of (15)-(18),(46) that gives the same objective value. This second step establishes

that (15)-(18),(46) gives a lower bound to (41)-(45). Combining the two steps gives that the values of the linear program (15)-(18),(46) and the nonlinear program (41)-(45) are the same.

In the third step, we show that the linear program (15)-(18),(46) has the same value as the convex program (30)-(31).

First step. Let $J,(\vec{s}_j,m_j),(u_j,q_j,t_j),x_j,j=1,\ldots,J$ be a feasible solution to (15)-(18),(46). The variable t_{mq} represents the average production time when product q is produced after changing from product m, i.e.

$$t_{mq}=\frac{\sum_{j=1}^{J}x_jt_j1\{m_j=m,q_j=q\}}{\sum_{j=1}^{J}x_j1\{m_j=m,q_j=q\}}$$

with $t_{mq}=0$ if $\sum_{j=1}^{J}x_j1\{m_j=m,q_j=q\}=0$, and x_{mq} is the average rate changeovers from m to q occur, i.e.

$$x_{mq}=\sum_{j=1}^{J}x_j1\{m_j=m,q_j=q\},$$

where $1\{\cdot\}$ is the indicator function being equal to 1 if the argument is true and 0 else.

We will show that if $J,(\vec{s}_j,m_j),(u_j,q_j,t_j),x_j,j=1,\ldots,J$ is feasible in (15)-(18),(46), then $t_{mq},x_{mq},m,q\in\mathcal{I}$ is feasible in problem (41)-(45) and has an objective value that is equal or lower.

The feasible solution $J,(\vec{s}_j,m_j),(u_j,q_j,t_j),x_j,j=1,\ldots,J$ obviously fulfills (16). Hence,

$$\begin{aligned}\lambda_q&=\sum_{j=1}^{J}x_jt_jp_q1\{q_j=q\}=\sum_{j=1}^{J}x_jt_jp_q\sum_{m=1}^{I}1\{m_j=m,q_j=q\}=\sum_{m=1}^{I}p_q\sum_{j=1}^{J}x_jt_j1\{m_j=m,q_j=q\}\\&=\sum_{m=1}^{I}p_qt_{mq}x_{mq},\end{aligned}$$

and (42) holds.

In addition, it follows that

$$\begin{aligned}0&=\sum_{j=1}^{J}x_j1\{m_j=m\}-\sum_{j=1}^{J}x_j1\{q_j=m\}\\&=\sum_{j=1}^{J}x_j\sum_{i=1}^{I}1\{m_j=m,q_j=i\}-\sum_{j=1}^{J}x_j\sum_{i=1}^{I}1\{m_j=i,q_j=m\}\\&=\sum_{i=1}^{I}\sum_{j=1}^{J}x_j1\{m_j=m,q_j=i\}-\sum_{i=1}^{I}\sum_{j=1}^{J}x_j1\{m_j=i,q_j=m\}=\sum_{i=1}^{I}x_{mi}-\sum_{i=1}^{I}x_{im},\end{aligned}$$

meaning that (17) implies (43). Moreover, a solution that fulfills (18) translates into a solution fulfilling (44) because

$$\begin{aligned}1&=\sum_{j=1}^{J}x_j(\tau_{m_jq_j}+t_j+u_j)\geq\sum_{j=1}^{J}x_j(\tau_{m_jq_j}+t_j)\\&=\sum_{j=1}^{J}x_j\tau_{m_jq_j}\sum_{m=1}^{I}\sum_{q=1}^{I}1\{m_j=m,q_j=q\}+\sum_{j=1}^{J}x_jt_j\sum_{m=1}^{I}\sum_{q=1}^{I}1\{m_j=m,q_j=q\}\\&=\sum_{m=1}^{I}\sum_{q=1}^{I}\tau_{mq}\sum_{j=1}^{J}x_j1\{m_j=m,q_j=q\}+\sum_{m=1}^{I}\sum_{q=1}^{I}\sum_{j=1}^{J}x_jt_j1\{m_j=m,q_j=q\}\end{aligned}$$

$$= \sum_{m=1}^{I}\sum_{q=1}^{I} \tau_{mq} x_{mq} + \sum_{m=1}^{I}\sum_{q=1}^{I} t_{mq} x_{mq} = \sum_{m=1}^{I}\sum_{q=1}^{I} (\tau_{mq} + t_{mq}) x_{mq}.$$

Any solution obeying (46) corresponds to non-negative x_{mq} and t_{mq} values. Hence, (45) holds.

To see that for these solutions, the value of (41) is not larger than (15), note that for all m_j, q_j, and t_j

$$\begin{aligned}
&\sum_{m=1}^{I}\sum_{q=1}^{I} \Big(c_{mq} + \frac{\phi_q t_{mq}^2 p_q (p_q - \lambda_q)}{2\lambda_q} \Big) x_{mq} \\
&= \sum_{m=1}^{I}\sum_{q=1}^{I} c_{mq} x_{mq} + \sum_{m=1}^{I}\sum_{q=1}^{I} \frac{\phi_q p_q (p_q - \lambda_q)}{2\lambda_q} t_{mq}^2 x_{mq} \\
&= \sum_{m=1}^{I}\sum_{q=1}^{I} c_{mq} \sum_{j=1}^{J} x_j 1\{m_j = m, q_j = q\} \\
&\quad + \sum_{m=1}^{I}\sum_{q=1}^{I} \frac{\phi_q p_q (p_q - \lambda_q)}{2\lambda_q} \left(\frac{\sum_{j=1}^{J} x_j t_j 1\{m_j = m, q_j = q\}}{\sum_{j=1}^{J} x_j 1\{m_j = m, q_j = q\}} \right)^2 \cdot \left(\sum_{j=1}^{J} x_j 1\{m_j = m, q_j = q\} \right) \\
&\le \sum_{j=1}^{J} c_{m_j q_j} x_j + \sum_{m=1}^{I}\sum_{q=1}^{I} \frac{\phi_q p_q (p_q - \lambda_q)}{2\lambda_q} \left(\sum_{j=1}^{J} x_j t_j^2 1\{m_j = m, q_j = q\} \right) \\
&= \sum_{j=1}^{J} \left(c_{m_j q_j} + \frac{\phi_{q_j} p_{q_j} (p_{q_j} - \lambda_{q_j}) t_j^2}{2\lambda_{q_j}} \right) x_j \\
&\le \sum_{j=1}^{J} \kappa(\vec{s}_j, m_j, u_j, q_j, t_j) x_j,
\end{aligned}$$

using Jensen's inequality for the first inequality and the fact that (46) implies $s_q - \lambda_q(u + \tau_{mq}) \ge 0$ for the second.

Second step. Now assume that a feasible solution $(t_{mq}, x_{mq})_{m,q \in \mathcal{I}}$ to (41)-(45) is given. We will show that this solution can be transformed into a feasible solution to (15)-(18),(46) with the same objective value.

Let $J = I^2$, and for all $m', q' = 1, \ldots, I$, frequencies $x_{I(m'-1)+q'} = x_{m'q'}$, production times $t_{(I-1)m'+q'} = t_{m'q'}$, machine states $m_{I(m'-1)+1} = \cdots = m_{I(m'-1)+I} = m'$, and next products $q_{I(1-1)+q'} = \cdots = q_{I(I-1)+q'} = q'$. Idling times are set to $u_j = 0$ except for one (arbitrary) $j = j'$ with $x_{j'} > 0$, where we choose $u_{j'} = (1 - \sum_{j=1}^{J} x_j(\tau_{m_j q_j} + t_j))/x_j$. (Such an index exists since a solution with $x_{m'q'} = 0$ for all m', q' would be violating (16) due to $\lambda_q > 0$ for all q.) Inventory levels $\vec{s}_j$ are set to $s_q = \lambda_q \cdot (u + \tau_{mq})$ and $s_i = \hat{s}_i(\vec{q}_{\eta_{mq}})$. This solution represents a feasible solution to (15)-(18), (46) because we

(a) demonstrated above that (42) and (43) correspond to their counterparts (16) and (17),

(b) chose the idling times to fulfill (18) exactly and not only with inequality as ensured by (44),

(c) chose the inventory vector such that (46) is fulfilled. (Part (c) can be seen following the lines of the proof of Lemma 6.)

Using $s_q = \lambda_q \cdot (u_{mq} + \tau_{mq})$ with $u_{mq} = 0$ for all m, q but the one corresponding to index j', the objective value (15) of this solution is

$$\sum_{j=1}^{J} \kappa(\vec{s}_j, m_j, u_j, q_j, t_j) \cdot x_j = \sum_{m'=1}^{I}\sum_{q'=1}^{I} \Big(c_{mq} + \frac{\phi_q t_{mq} p_q}{\lambda_q} \cdot (0) + \frac{\phi_q t_{mq}^2 p_q (p_q - \lambda_q)}{2\lambda_q} \Big) x_{mq},$$

which equals (41).

Third step. As we have shown in Lemma 6, the convex program (30) - (31) is equivalent to solving (12)-(14) with relaxed action space $\tilde{\mathcal{A}}(\vec{s},m)$. In the corresponding dual, (15)-(19) this translates into replacing $(u_j,q_j,t_j)\in\mathcal{A}(\vec{s},m)$ by $(u_j,q_j,t_j)\in\tilde{\mathcal{A}}(\vec{s},m)$. So the dual is (15)-(18),(46). The proof of Theorem 1 can be used without changes to show that there is no duality gap between (30)-(31) and (15)-(18),(46). Hence, the values of these two problems are the same. □

5.2. Bomberger's lower bound problem

The above problem is a natural extension of the well-known lower-bound problem suggested by Bomberger (1966) to sequence-dependent setup times and costs. The following lemma shows that it reduces to Bomberger's lower bound in sequence-independent settings.

PROPOSITION 2. *In the sequence-independent setting with setup costs $c_1,\ldots,c_I$ and setup times $\tau_1,\ldots,\tau_I$, problem (41)-(45) gives the same optimal solution as Bomberger's well known lower bound problem*

$$\min_{T_1,\ldots,T_I\geq 0}\sum_{q=1}^{I}\left(\frac{c_q}{T_q}+\frac{1}{2}\phi_q\rho_q(p_q-\lambda_q)T_q\right) \tag{47}$$

$$s.t.\sum_{q=1}^{I}\tau_q/T_q\leq 1-\sum_{q=1}^{I}\rho_q. \tag{48}$$

Proof. We show that in the case of sequence-independent setup costs and times, problem (41)-(45) reduces to (47)-(48).

Denoting the Lagrangian multipliers of (43) as $V_1',\ldots,V_I'$, and the multiplier of (44) as ξ', a necessary condition for optimality in (41) - (45) is that the derivative of the Lagrangian with respect to t equals 0,

$$\frac{\phi_q p_q(p_q-\lambda_q)}{\lambda_q}t_{mq}x_{mq}+V_q'p_qx_{mq}+\xi'x_{mq}=0$$

for all $m,q\in\mathcal{I}$. Hence, it must hold that for all m with $x_{mq}\neq 0$

$$t_{mq}=\frac{-\lambda_q(p_qV_q'-\xi')}{\phi_q p_q(p_q-\lambda_q)},$$

which does not depend on m. Plugging this into (43) and using (42) gives

$$\sum_{i=1}^{I}x_{qi}=\sum_{i=1}^{I}x_{iq}=-\frac{\rho_q\phi_q p_q(p_q-\lambda_q)}{\lambda_q(p_qV_q'-\xi')},$$

which we denote by $1/T_q$, the rate at which changes into/leaving machine state i occur. It is clear from the problem formulation that t_{mq} can be chosen arbitrarily for $x_{mq}=0$. So to simplify the representation, let

$$t_{mq}=\frac{-\lambda_q(p_qV_q'-\xi')}{\phi_q p_q(p_q-\lambda_q)}=-\rho_q\frac{\lambda_q(p_qV_q'-\xi')}{\rho_q\phi_q p_q(p_q-\lambda_q)}=\rho_qT_q$$

for all $m,q\in\mathcal{I}$.

Knowing that in an optimal solution $t_{mq}=\rho_qT_q$ and $\sum_{i=1}^{I}x_{qi}=1/T_q$, we can replace the variables t_{mq} and x_{mq} by these terms (with variables T_q) in (41) - (45). The objective (41) then simplifies to

$$\sum_{m=1}^{I}\sum_{q=1}^{I}\left(c_q+\frac{\phi_q t_{mq}^2p_q(p_q-\lambda_q)}{2\lambda_q}\right)x_{mq}=\sum_{q=1}^{I}\left(c_q+\frac{\phi_q(\rho_qT_q)^2p_q(p_q-\lambda_q)}{2\lambda_q}\right)\sum_{m=1}^{I}x_{mq}$$

$$=\sum_{q=1}^{I}\left(c_q+\frac{\phi_q(\rho_qT_q)^2p_q(p_q-\lambda_q)}{2\lambda_q}\right)/T_q=\sum_{q=1}^{I}\frac{c_q}{T_q}+\frac{1}{2}\phi_q\rho_q(p_q-\lambda_q)T_q,$$

constraints (42) and (43) are fulfilled automatically by this replacement, (44) is changed to

$$\sum_{m=1}^{I}\sum_{q=1}^{I}(\tau_q+t_{mq})x_{mq}=\sum_{q=1}^{I}(\tau_q+\rho_q T_q)\sum_{m=1}^{I}x_{mq}=\sum_{q=1}^{I}\tau_q/T_q+\rho_q\leq 1,$$

which is equivalent to (48). Finally, (45) translates into $T_q\geq 0$. □

Even for sequence-dependent problems, we can conclude from this proof that the optimal values t^*_{mq} in the nonlinear problem (41)-(45), would actually never depend on the machine state m. We allowed the production times of product q, t_{mq}, to depend on m only to emphasize the parallels to the dual formulation (15)-(19).

5.3. Dobson's lower bound problem

In his paper, Dobson derives the optimization problem P5

$$\max_{\xi'>0,\vec{\theta}^1,\vec{\theta}^2}\left\{\min_{f>0,T>0}\frac{1}{T}\left\{\sum_{q=1}^{I}\bar{c}_{e_q}(f_q-1)+v(G)-\sum_{q=1}^{I}(\theta_q^1+\theta_q^2)f_q+T\sum_{q=1}^{I}\frac{\bar{h}_q}{f_q}\right\}-\xi'(1-\rho)\right\}$$

where f_q represents the frequency of producing product q in one cycle of a cyclic schedule (integrality was relaxed in a previous step), T is the cycle length of this cycle (the sum over all setup, idling and production times), $\rho=\sum_{q=1}^{I}\rho_q$, $\bar{h}_q=0.5\phi_q(p_q-\lambda_q)\rho_q$, and $v(G)$ is the value of a 1-tree problem on a graph with edges $\bar{c}_{mq}(\xi',\theta^1,\theta^2)=\min\{c_{mq}+\xi'\tau_{mq}+\theta_m^1+\theta_q^2,c_{qm}+\xi'\tau_{qm}+\theta_q^1+\theta_m^2\}$. The least weighted edge incident on q (in terms of $\bar{c}(\cdot)$) is denoted by e_q and $\vec{\theta}^1=(\theta_1^1,\ldots,\theta_I^1)$, $\vec{\theta}^2=(\theta_1^2,\ldots,\theta_I^2)$. Dobson then shows that the inner optimization problem has a lower bound of $2\sum_q(b'_q(\xi',\theta^1,\theta^2)\bar{h}_q)^{0.5}$, with $b'_q(\xi',\theta^1,\theta^2)=\bar{c}_{e_q}(\xi',\theta^1,\theta^2)-\theta_q^1-\theta_q^2$ or

$$b'_q(\xi',\vec{\theta}^1,\vec{\theta}^2)=\min_{m\in\mathcal{I}}\left\{\min\{c_{mq}+\xi'\tau_{mq}+\theta_m^1-\theta_q^1,c_{qm}+\xi'\tau_{qm}+\theta_m^2-\theta_q^2\}\right\}. \tag{49}$$

Replacing the inner optimization problem with this bound gives his lower bound problem P6

$$\begin{aligned}\max_{\xi'>0,\vec{\theta}^1,\vec{\theta}^2}\quad & L(\xi',\vec{\theta}^1,\vec{\theta}^2)=2\left(\sum_q\sqrt{b'_q(\xi',\vec{\theta}^1,\vec{\theta}^2)\bar{h}_q}\right)-\xi'(1-\sum_q\rho_q)\\ =\max_{\xi'>0,\vec{\theta}^1,\vec{\theta}^2}\quad & \sum_{q=1}^{I}\sqrt{2\phi_q\rho_q(p_q-\lambda_q)\cdot b'_q(\xi',\vec{\theta}^1,\vec{\theta}^2)}-\xi'(1-\sum_{q=1}^{I}\rho_q).\end{aligned} \tag{50}$$

Although this problem looks very different at first sight, we can show that it gives the same solution as our lower bound problem (30)-(31).

THEOREM 4. *Problems (30)-(31) and (50) are equivalent.*

Proof. We first reformulate Dobson's problem (50) to facilitate the comparison with (30)-(31). In a second step, we then give a mapping of every feasible solution to (50) to a feasible solution of (30)-(31) that does not change the objective value and vice versa. Hence, the two problems are equivalent.

Introduce decision variables y'_q (representing $\sqrt{2\phi_q(p_q-\lambda_q)b'_q(\xi',\vec{\theta}^1,\vec{\theta}^2)/\rho_q}$) and the program

$$\begin{aligned}\max_{\xi'\geq 0,y'_1,\ldots,y'_I,\vec{\theta}^1,\vec{\theta}^2}\quad & \sum_{i=1}^{I}\rho_i y'_i-(1-\sum_{i=1}^{I}\rho_i)\xi'\\ \text{s.t.}\quad & -b'_q(\xi',\vec{\theta}^1,\vec{\theta}^2)+\frac{{y'_q}^2\lambda_q}{2\phi_q p_q(p_q-\lambda_q)}\leq 0 \qquad \forall q\in\mathcal{I}.\end{aligned}$$

Since the objective function is increasing in each y'_q, each condition would be tight in an optimal solution, giving $y'_q = \sqrt{2\phi_q(p_q-\lambda_q)b'_q(\xi',\vec{\theta}^1,\vec{\theta}^2)/\rho_q}$ and simplifying back to (50). Replacing $b'_q(\xi',\vec{\theta}^1,\vec{\theta}^2)$ by (49) and substituting $p_qV'_q-\xi'=y'_q$, we can write this program as

$$\max_{\xi'\geq 0, V'_1,\ldots,V'_I,\theta^1_1,\ldots,\theta^1_I,\theta^2_1,\ldots,\theta^2_I} \sum_{i=1}^{I} \lambda_i V'_i - \xi' \tag{51}$$

$$\text{s.t. } \theta^1_q - \theta^1_m - c_{mq} - \xi'\tau_{mq} + \frac{(p_qV'_q-\xi')^2\lambda_q}{2\phi_q p_q(p_q-\lambda_q)} \leq 0 \qquad \forall m,q\in\mathcal{I} \tag{52}$$

$$\theta^2_q - \theta^2_m - c_{qm} - \xi'\tau_{qm} + \frac{(p_qV'_q-\xi')^2\lambda_q}{2\phi_q p_q(p_q-\lambda_q)} \leq 0 \qquad \forall m,q\in\mathcal{I}. \tag{53}$$

Note the similarity of (51)-(53) to (30)-(31). In (30)-(31), the values of $\vec{\theta}$ only play a role in terms of their differences. So we can we relax the constraint $\theta_I=0$ in (30)-(31) to $\theta_I\in\mathbb{R}$ without changing the objective value.

Suppose now a feasible solution to (51)-(53) is given by $\xi', V'_i, \theta^1_i, \theta^2_i$, $i\in\mathcal{I}$. Then, choosing $\xi=\xi'$, $V_i=V'_i$, and $\theta_i=-\theta^1_i$ for all $i\in\mathcal{I}$ is a feasible solution to (30)-(31) and has the same objective value. This direction is straightforward, since using $\theta_i=-\theta^1_i$, program (30)-(31) is identical to (51)-(53) when all constraints (53) are dropped.

Now, suppose a feasible solution to (30)-(31) is given by ξ, V_i, θ_i, $i\in\mathcal{I}$. Then, choosing $\xi'=\xi'$, $V'_i=V_i$, $\theta^1_i=-\theta_i$, and $\theta^2_i=\theta_i-(p_iV_i-\xi)^2\lambda_i/(2\phi_ip_i(p_i-\lambda_i))$ for all $i\in\mathcal{I}$ is a feasible solution to (51)-(53) and has the same objective value. The objective values are the same since the values of the V_i's and ξ are the same as the values of the V'_i's and ξ'. To see that the proposed solution is feasible to (51)-(53), remember that ξ, V_i, θ_i, $i\in\mathcal{I}$, is feasible in (30)-(31). To illustrate the idea of the mapping, write all constraints of (30)-(31) twice, namely as

$$\theta_m-\theta_q-c_{mq}-\xi\tau_{mq}+\frac{(p_qV_q-\xi)^2\lambda_q}{2\phi_qp_q(p_q-\lambda_q)} = \theta^1_q-\theta^1_m-c_{mq}-\xi'\tau_{mq}+\frac{(p_qV'_q-\xi')^2\lambda_q}{2\phi_qp_q(p_q-\lambda_q)} \leq 0 \quad \forall m,q\in\mathcal{I},$$

$$\text{and} \qquad \theta^1_m-\theta^1_q-c_{qm}-\xi'\tau_{qm}+\frac{(p_mV'_m-\xi')^2\lambda_m}{2\phi_mp_m(p_m-\lambda_m)} \leq 0 \quad \forall m,q\in\mathcal{I}.$$

These constraints are satisfied for all feasible ξ, V_i, θ_i, $i\in\mathcal{I}$, in (30)-(31). Hence, we can immediately conclude that (52) is satisfied for $\xi'=\xi'$, $V'_i=V_i$, $\theta^1_i=-\theta_i$. Plugging $\theta^2_i=\theta_i-(p_iV_i-\xi)^2\lambda_i/(2\phi_ip_i(p_i-\lambda_i)) = -\theta^1_i-(p_iV'_i-\xi')^2\lambda_i/(2\phi_ip_i(p_i-\lambda_i))$ into (53) yields

$$-\theta^1_q-\frac{(p_qV'_q-\xi')^2\lambda_q}{2\phi_qp_q(p_q-\lambda_q)}+\theta^1_m+\frac{(p_mV'_m-\xi')^2\lambda_m}{2\phi_mp_m(p_m-\lambda_i)}-c_{qm}-\xi'\tau_{qm}+\frac{(p_qV'_q-\xi')^2\lambda_q}{2\phi_qp_q(p_q-\lambda_q)}$$
$$=\theta^1_m-\theta^1_q-c_{qm}-\xi'\tau_{qm}+\frac{(p_mV'_m-\xi')^2\lambda_m}{2\phi_mp_m(p_m-\lambda_m)} \leq 0 \quad \forall m,q\in\mathcal{I}.$$

Therefore, the given solution is a feasible solution to Dobson's problem. □

It follows immediately that the problem derived in Dobson (1992) over a number of relaxations is equivalent to the lower bound problem that originates from a simple affine approximation of the value function of the dynamic programming formulation. In addition, we can conclude that the variables $\vec{\theta}_2$ are superfluous.

6. Summary of Results

We provided a new perspective on the well-known sequence-dependent ELSP and analyzed the corresponding infinite dimensional linear program to determine the minimum average cost rate.

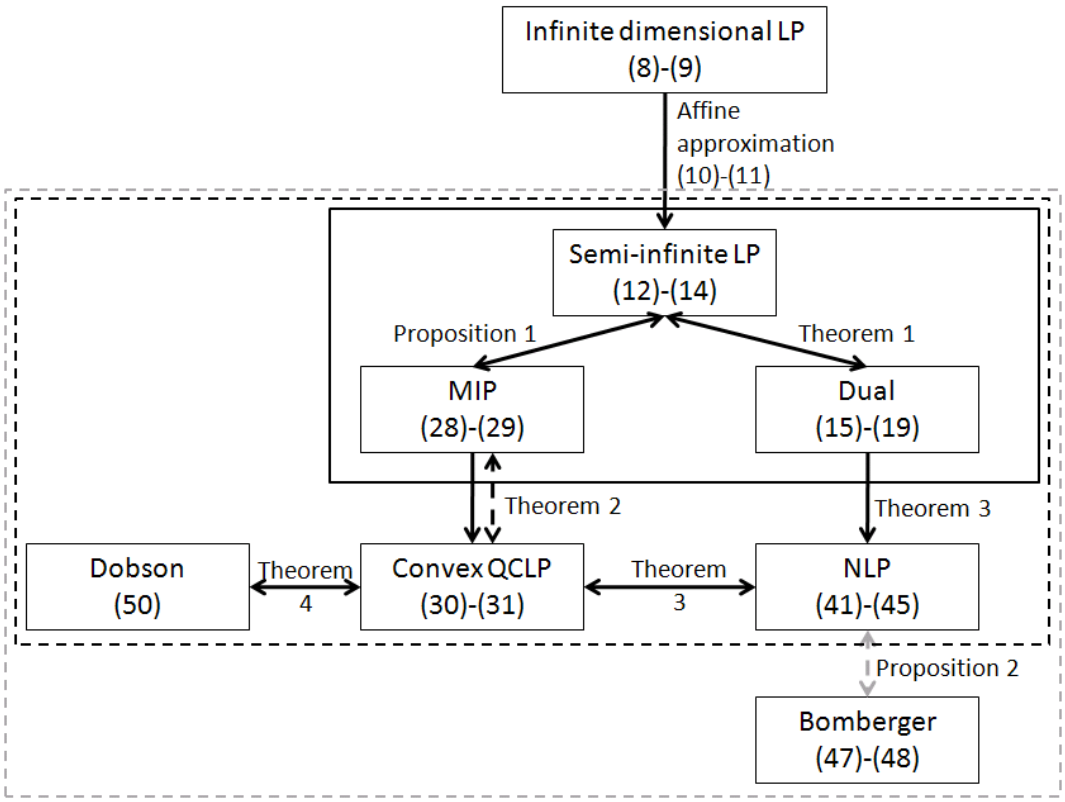

Figure 3 Summary of all results concerning lower bounds. Black arrows and boxes represent general relations between models, black dashed style is used for interrelations in problems satisfying (32), relations valid only in sequence-independent problems are displayed in grey dashed style.

We used an affine approximation of the value function to derive tractable programs providing a lower bound to the minimum average cost rate with parameters that can be interpreted as marginal values of products, machine time, and machine state.

Figure 3 summarizes the comparative results concerning the lower bound problems presented in this paper and in the previous literature. In this figure, a black arrow from problem 1 to problem 2 means that problem 2 gives a weaker lower bound. If the arrow is in both directions, the two problems yield the same lower bound. Dashed black arrows have the same interpretation under condition (32), dashed gray arrows represent relationships in sequence-independent settings. Boxes frame problems that all yield the same lower bound under the assumption of the corresponding style.

From the infinite dimensional linear problem (8)-(9), we obtained the semi-infinite linear program (12)-(14) by an affine approximation. Both its dual (15)-(19) and a finite dimensional formulation (28)-(29) that can be solved as a mixed integer problem have the same value. In addition, we were able to show that the simple convex quadratically constrained linear program (30)-(31) gives a weaker lower bound than these three problems. If the relatively mild assumption (32) is satisfied, however, the value is the same. Showing that problem (30)-(31) is equivalent to the lower bound introduced in Dobson (1992), we connected Dobson's contribution to our dynamic programming approach. Using the intuition of the dual formulation (15)-(19), we further suggested a weaker lower bound problem (41)-(45) that simplifies to the well-known lower bound problem introduced by Bomberger (1966) for sequence-independent problems. Showing a dual relationship between (41)-(45) and (30)-(31), we can conclude that in sequence-independent problems all our finite and semi-definite programs yield the same value because (32) is by definition fulfilled in these settings.

This new perspective on the ELSP also pointed at many directions for future research: Since previously published lower bounds for this problem can be captured by an affine approximation of the value function, it seems consequential that more general approximations should yield even tighter bounds. From a theoretical standpoint, the state space $\mathcal{S}$ (and the optimality equations (6)-(7)) should be analyzed more rigourously and tractable approximations of the state space should be evaluated. From a more practical point of view, it seems straightforward to use the prices obtained from solving (28)-(29) in a price-directed, dynamic heuristic. Following this approach, however, even the determination of feasible actions (u, q, t) given state $(\vec{s}, m)$ is difficult since there is no known closed-form representation of $\mathcal{S}$. Adelman and Barz (2009) discuss these problems and introduce a dynamic price-directed heuristic.

Appendix A: Determination of $\vec{\hat{s}}(\vec{q})$

Given a production sequence $\vec{q} = (q^{[1]}, \ldots, q^{[N]})$, let $\mathcal{Z}(\vec{q}) = \{z_i^1, \ldots, z_i^{Z_i(\vec{q})}\}$ be the set of $Z_i(\vec{q})$ indices in the sequence $\vec{q}$ where product i is produced, i.e. the set of all z with $q^{[z]} = i$.

Using this, the production times necessary to execute a zero-switch policy, i.e. that start production of a product when the inventory level of this product equals 0, given $\vec{q} = (q^{[1]}, \ldots, q^{[N]})$ and idling times $\vec{u} = (u^{[1]}, \ldots, u^{[N]})$, are the solution of

$$p_i t^{[z_i^j]} = \lambda_i \sum_{n=z_i^j}^{z_i^{j+1}-1} (t^{[n]} + \tau_{q^{[n]}q^{[n+1]}} + u^{[n+1]}),$$

see e.g. Dobson (1992).

Let $t^{0[1]}, \ldots, t^{0[N]}$ denote the solution of this linear system of equations given $u^{[1]} = \cdots = u^{[N]} = 0$. Because in every production run of a given product enough is produced to satisfy all demand until it is produced the next time, ensuring that initial inventories cover the demand until the first production run avoids stock-outs long-term. Hence,

$$\hat{s}_i(\vec{q}) = \lambda_i \tau_{q^{[N]}q^{([1])}} + \lambda_i \sum_{n=1}^{z_i^1-1} (t^{0[n]} + \tau_{q^{[n]}q^{[n+1]}}) \quad \forall i \in \mathcal{I}.$$

Appendix B: Proof of Lemma 2

Let $\{(0, q^n, t^n)\}_{n=1,2,\ldots}$ be a sequence of actions that avoid stock-outs starting in state $(\vec{s} - \epsilon\vec{1}, m)$. Then, starting in $(\vec{s}, m)$, the sequence $\{(0, q^n, t^n)\}_{n=1,2,\ldots}$ clearly avoids stock-outs, too. In the following we will use the fact that we could actually even increase some of the production times a little because we have stocks that are an amount ϵ larger than they need to be.

We define values y_k that have the properties that $y_0 < y_1 < y_2, \ldots,$ that $q^{y_k} = q^{y_{k+1}} = m'$, and that each of the products in $\mathcal{I} \backslash \{m\}$ is produced between the y_kth and the y_{k+1}th production run. More formally, we let

$$\begin{aligned} y_0 &= \min_{n \geq 0} \{n : q^n = m'\} \\ y_k &= \min_{n > \max_{i \in \mathcal{I} \backslash \{m\}} \min_{n > y_{k-1}} \{n : q^n = i\}} \{n : q^n = m'\}, \text{ for all } k \in \mathbb{N}. \end{aligned}$$

We show how we can increase the inventory of each product within decision periods $y_k + 1, \ldots, y_{k+1}$ for $k \in \mathbb{N}_0$ by

$$\beta = \frac{\epsilon}{\alpha \max_{i \in \mathcal{I}} \lambda_i} \quad \text{with} \quad \alpha = \frac{\sum_{i=1}^I 1/p_i}{1 - \sum_{i=1}^I \rho_i}.$$

Doing this, we will choose one single production period with $q^n = i$ and $y_k < n \leq y_{k+1}$ for each product. We increase the production time of this period from t_n to $t_n + \beta/p_i + \lambda_i \beta\alpha/p_i$. This increases the length of decision periods $y_k + 1$ to y_{k+1} in total by

$$\sum_{i=1}^I \frac{\beta}{p_i} + \sum_{i=1}^I \frac{\alpha\lambda_i}{p_i} = \beta\alpha \left(\frac{1}{\alpha} \sum_{i=1}^I \frac{1}{p_i} + \sum_{i=1}^I \frac{\lambda_i}{p_i} \right) = \beta\alpha.$$

Note that we chose β such that the total demand for product i within this time, $\lambda_i\beta\alpha$, is lower than ϵ. Now consider e.g. the periods y_0+1 to y_1. If we write s_i^n for the original inventory level of product i after decision period n and $s_i''^n$ for the corresponding new inventory level, we have

$$\begin{aligned} s_i''^{n-1} - \lambda_i\tau_{q^{n-1}q^n} &\geq s_i^{n-1} - \lambda_i\beta\alpha - \lambda_i\tau_{q^{n-1}q^n} \\ &\geq s_i^{n-1} - \epsilon - \lambda_i\tau_{q^{n-1}q^n} \\ &\geq 0, \text{ for } n = y_0+1,\ldots,y_1. \end{aligned}$$

Hence, the changed production times do not lead to stock-outs. The effect of the $\beta/p_i + \lambda_i\beta\alpha/p_i$ time units extra production time for each product on the inventory is

$$\begin{aligned} s_i''^{y_1} &= s_i^{y_1} - \lambda_i\beta\alpha + p_i(\beta/p_i + \lambda_i\beta\alpha/p_i) \\ &= s_i^{y_1} - \lambda_i\beta\alpha + \beta + \lambda_i\beta\alpha \\ &= s_i^{y_1} + \beta. \end{aligned}$$

Repeating this argumentation for production periods y_1+1 to y_2, y_2+1 to y_3, and so on, one can easily see that $s_i''^{n-1} - \lambda_i\tau_{q^{n-1}q^n} \geq 0$ for all n and that

$$s_i''^{y_k} = s_i^{y_k} + k\beta.$$

Because our original sequence of actions was admissible, we know that $s_i^{y_k} \geq 0$ for all k. Therefore, we can ensure machine state m' and $s_i''^{y_k} \geq s_i'$ for all i by letting $N = y_{k'}$ with

$$k' \geq \frac{1}{\beta}\max_{i\in\mathcal{I}} s_i'.$$

References

Adelman, D. (2004a). Price-directed replenishment of subsets: Methodology and its application to inventory routing. *Manufacturing & Service Operations Management*, Vol. 5, 348371.

Adelman, D. (2004b). A price-directed approach to stochastic inventory/routing. *Operations Research*, Vol. 52, 499514.

Adelman, D., and Barz, C. (2009). A price-directed dynamic heuristic for the economic lot scheduling model. *Working Paper*, Graduate School of Business, University of Chicago.

Anderson, E. J. (1990). Testing feasibility in a lot scheduling problem. *Operations Research*, Vol. 38, 1079-1088.

Aragone, L.S., and Gonzalez, R.L.V. (2000). A fast computational procedure to solve the multi-item single machine lot scheduling optimization problem: the average cost case. *Mathamatics of Operations Research*, Vol. 25, 455-475.

Bertsekas, D. P. (2005). Dynamic programming and suboptimal control: A survey from ADP tp MPC. *European Journal on Control*, Vol. 11, 310334.

Bertsekas, D. P., and Tsitsiklis, J. N. (1996). *Neuro-Dynamic Programming.* Athena Scientific.

Bomberger, E. E. (1966). A dynamic programming approach to a lot size scheduling problem. *Management Science*, Vol. 12, 778-784.

Carstensen, P. (1999). The economic lot scheduling problem - survey and LP-based method. *OR Spektrum*, Vol. 21, 429460.

Dobson, G. (1987). The economic lot-scheduling problem: achieving feasibility using time-varying lot sizes. *Operations Research*, Vol. 35, 764-771.

Dobson, G. (1992). The cyclic lot scheduling problem with sequence-dependent setups. *Operations Research*, Vol. 40, 736-749.

Elmaghraby, S.E. (1978). The economic lot scheduling problem (ELSP): Review and extensions. *Management Science*, Vol. 24, 587-598.

Gallego, G. and Shaw, D.X. (1997). Complexity of the ELSP with general cyclic schedules. *IIE Transactions*, Vol. 29, 109-113.

Glashoff, K., and Gustafson, S.-A. (1983). *Linear optimization and approximation.* Springer, New York-Heidelberg-Berlin.

Hodgson, T.J, Ge, G. King, R.E., Said, H. (1997). Dynamic lot size/sequencing policies in a multi-product single-machine system. *IIE Transactions*, Vol. 29, 127-137.

Hsu, W.L. (1983). On the general feasibility test of scheduling lot sizes for several products on one machine. *Management Science*, Vol. 29, 93-105.

Inman, R.R., and Jones, P.C. (1993). Decomposition for scheduling flexible manufacturing systems. *Operations Research*, Vol. 41, 608-617.

Maxwell, W.L. (1964). The scheduling of economic lot sizes. *Naval Research Logistics Quarterly*, Vol. 11, 89-124.

Powell, W.B. (2007). *Approximate Dynamic Programming.* John Wiley & Sons, Inc.

Qui, J., and Loulou, R. (1995). Multiproduct production/inventory control under random demands. *IEEE Transactions on automatic control*, Vol. 40, 350-356.

Rogers, J. (1958). A computational approach to the economic lot scheduling problem. *Management Science*, Vol. 4, 264-291.

Schäl, M. (1992). On the Second Optimality Equation for Semi-Markov Decision Models. *Mathamatics of Operations Research*, Vol. 17, 470-486.

Sing, H., and Foster, J.B. (1987). Production Scheduling with sequence dependent setup costs. *IIE Transactions*, Vol. 1, 43-49.

Taylor, G.D., Taha, H.A., Chowning, K.M. (1997). A heuristic model for the sequence-dependent lot scheduling problem. *Production Planning& Control*, Vol. 8, 213-225.

Wagner, B.J., and Davis, D.J. (2002).A search heuristic for the sequence-dependent economic lot scheduling problem. *European Journal of Operational Research*, Vol.141, 133-146.

Zipkin, P.H. (1991). Computing optimal lot sizes in the economic lot scheduling problem. *Operations Research*, Vol. 39, 56-63.