Online Supplement for “A Price-Directed Heuristic for the Economic Lot Scheduling Problem”

Daniel Adelman

University of Chicago, Booth School of Business, Chicago, IL, 60637, dan.adelman@chicagobooth.edu

Christiane Barz

University of California, Anderson School of Management, Los Angeles, CA, 90095, christiane.barz@anderson.ucla.edu

This online supplement includes the appendix of the paper “A Price-Directed Heuristic for the Economic Lot Scheduling Problem”.

Appendix A: List of Symbols

The following represents a list of all symbols used in the paper excluding the appendix.

a	action (u, q, t)
$\mathcal{A}(\vec{s}, m)$	set of feasible actions in state $(\vec{s}, m)$
$\mathcal{A}(\vec{s}, m, q)$	$\{(u, t) \mid t, u \geq 0 : (u, q, t) \in \mathcal{A}(\vec{s}, m)\}$
$\mathcal{A}_g(\vec{s}, m)$	set of actions $a \in \mathcal{A}(\vec{s}, m)$ that minimize the right-hand side of (5)
$\tilde{\mathcal{A}}_{\geq}^w(m)$	action space approximation given $\tilde{S}_{\geq}^w(m)$
$c_{i_1 i_2}$	setup cost when switching from product i_1 to i_2
$\delta(i = j)$	equals 1 for $i = j$, 0 otherwise
ϕ_i	holding costs (per unit and time) for product i
g	average costs
$g(\vec{s}, m)$	average costs when starting in state $(\vec{s}, m)$
$h(\vec{s}, m)$	the value of the bias function in state $(\vec{s}, m)$
i	index used for products
I	number of products
$\mathcal{I}$	set of products $\{1, \dots, I\}$
$J(\pi_{(\vec{s}, m)})$	average costs of trajectory $\pi_{(\vec{s}, m)}$
$\kappa(s_q, m, u, q, t)$	one-stage costs in state $(\vec{s}, m)$ for action (u, q, t)
K	Number of steps executed in the lookahead heuristic
K'	Length of the cyclic production sequence if the lookahead heuristic did
λ_i	demand rate for product i
$\vec{\lambda}$	$(\lambda_1, \dots, \lambda_I)$
m	machine state, the product that the machine is set up for
m^n	machine state in decision epoch n
n	number of decision epoch
N	length of a given cyclic schedule
$\mathbb{N}$	the set of natural numbers
p_i	production rate when producing product i
$\vec{p}_i$	vector with value p_q in row q and 0 else
$\pi_{(\vec{s}, m)}$	a trajectory, a sequence of state action pairs fulfilling (2)

q	next product that should be produced
q^n	next product that should be produced in decision epoch n
$q^{[n]}$	the product that should be produced in decision epochs $n + kN$, $k \in \mathbb{N}_0$
$\vec{q}$	a production sequence $(q^{[1]}, \dots, q^{[N]})$
ρ_i	λ_i/p_i
$\mathbb{R}$	the set of real numbers
s_i	inventory level of product i
s_i^n	inventory level of product i in decision epoch n
$\vec{s}$	$(s_1, \dots, s_I)$
$\vec{s}^n$	$(s_1^n, \dots, s_I^n)$
$\mathcal{S}$	the state space, the set of admissible states
$\mathcal{S}(m)$	$\{\vec{s} \mid (\vec{s}, m) \in \mathcal{S}\}$
$\vec{s}(m)$	an inventory vector in $\mathcal{S}(m)$
$\tilde{s}_i(m)$	inventory of product i for a given inventory vector in $\mathcal{S}(m)$
$\vec{s}^w(m)$	the w th inventory vector in $\mathcal{S}(m)$
$\tilde{s}_i^w(m)$	inventory of product i for the w th inventory vector in $\mathcal{S}(m)$
$\tilde{\mathcal{S}}^{W_m}(m)$	a set of inventory vectors in $\mathcal{S}(m)$, $\{\vec{s}^1(m), \dots, \vec{s}^{W_m}(m)\}$
$\tilde{\mathcal{S}}_{\geq}^W(m)$	the approximation of $\mathcal{S}(m)$ given by $\{\vec{s} \mid \exists w \in 1 \dots W : \forall i \in \mathcal{I} : s_i \geq \tilde{s}_i^w(m)\}$
t	production time
t^n	production time in epoch n
$t^{[n]}$	the production time in decision epochs $n + kN$, $k \in \mathbb{N}_0$
$\tau_{i_1 i_2}$	setup time when switching from product i_1 to i_2
τ_{qq}^{sp}	the length of the shortest setup time path from product q back to product q
θ_i	marginal value of machine state i
u	idle time
u^n	idling time in epoch n
$u^{[n]}$	the idling time in decision epochs $n + kN$, $k \in \mathbb{N}_0$
V_i	marginal product value
W_m	size of the set $\tilde{\mathcal{S}}^{W_m}(m)$
ξ	marginal machine time value

Appendix B: Solving for the Best Schedule on a Given Sequence

To find the best static production schedule for a given sequence $\vec{q} = (q^{[1]}, \dots, q^{[N]})$, we solve the following nonlinear problem in MINOS:

$$\begin{aligned}
\min_{T, t^{[n]}, u^{[n]}, s_i^{[n]'}, i \in \mathcal{I}, n=1, \dots, N} \quad & \frac{1}{T} \left(c_{q^{[N]} q^{[1]}} + \sum_{n=1}^{N-1} c_{q^{[n]} q^{[n+1]}} + \sum_{i=1}^I \frac{\phi_i (p_i - \lambda_i) p_i \sum_{n=1}^N \delta\{i = q^{[n]}\} (t^{[n]})^2}{2\lambda_i} \right. \\
& \left. + \sum_{i=1}^I \frac{\phi_i p_i \sum_{n=1}^N \delta\{i = q^{[n]}\} t^{[n]} s_i^{[n]'}}{\lambda_i} \right) \\
s.t. \quad & \tau_{q^{[N]} q^{[1]}} + \sum_{n=1}^{N-1} \tau_{q^{[n]} q^{[n+1]}} + \sum_{n=1}^N (t^{[n]} + u^{[n]}) = T \\
& \sum_{n=1}^N \delta\{i = q^{[n]}\} t^{[n]} p_i \geq \lambda_i T \quad \forall i \in \mathcal{I} \\
& s_i^{[n+1]'} = s_i^{[n]'} + \delta\{i = q^{[n]}\} p_i t^{[n]} - \lambda_i (t^{[n]} + \tau_{q^{[n]} q^{[n+1]}} + u^{[n+1]}) \quad \forall i \in \mathcal{I}, n = 1, \dots, N \\
& s_i^{[1]'} = s_i^{[N]'} + \delta\{i = q^{[N]}\} p_i t^{[N]} - \lambda_i (t^{[N]} + \tau_{q^{[N]} q^{[1]}} + u^{[1]}) \quad \forall i \in \mathcal{I}
\end{aligned}$$

$$T, t^{[n]}, u^{[n]}, s_i^{[n]'} \geq 0 \quad \forall i \in \mathcal{I}, n = 1, \dots, N$$

In this problem, T is the total time it takes to execute the cycle once, $t^{[n]}$ and $u^{[n]}$ represent the production time of product $q^{[n]}$ and the idling time before production starts. The value $s_i^{[n]}'$ represents the inventory of item $q^{[n]}$ right before its production starts, so $s_i^{[n]}' = s_i^{[n]} - \lambda_{q^{[n]}}(u^{[n]} + \tau_{q^{[n-1]}q^{[n]}})$. The goal is to minimize the average total cost, which is composed of setup and inventory holding cost. The constraints ensure that the inventory levels remain non-negative. When implementing the above problem, the equality signs can be relaxed to inequalities for numerical stability.

Appendix C: The One-Product Case

To illustrate our approach, consider the case of only one product. This product can be produced at a rate of p , demand is λ , holding costs are ϕ per item per time unit. Every time production is started, a setup must be performed. This setup takes τ time units and costs c . It is well-known (see e.g. Zipkin (2000), p.53) that the optimal policy is to start the setup every time the inventory is at $\lambda\tau$ and then produce a quantity of $\max\{\tau\lambda p/(p-\lambda), \sqrt{2c\lambda p/(\phi(p-\lambda))}\}$.

We will show that our approach prescribes exactly this optimal policy. First, we find the dual prices associated with the product, V^* , and machine time, ξ^* , by solving (9)-(10). If $\xi = \tau\lambda p\phi/(2(p-\lambda)) - c/\tau \geq 0$, then, $\xi^* = \xi$ and $V^* = \xi/p + \phi\tau$ are an optimal solution to (9)-(10). Otherwise, $\xi^* = 0$ and $V^* = \sqrt{2\phi(p-\lambda)c/(p\lambda)}$.

This optimal solution makes sense intuitively. If the performance of the system is limited by the capacity constraint; i.e. if machine time has a value $\xi^* > 0$, the value of the product, V^* , equals the value of the machine time needed to produce one (marginal) item plus the inventory costs for holding the item during the setup. If (marginal) machine time is considered worthless ($\xi^* = 0$), the value gained from the demand at rate λ (i.e., λV^*) is increasing in the setup and holding costs, increasing in the demand rate and decreasing in the production rate. The optimal solution represents the well-known optimal average total cost for the one-product case. So the lower bound obtained by (9)-(10) is indeed tight in this example.

In the one-product setting, the state space $\mathcal{S}$ is easy to describe. Letting s denote the inventory of the product, stock-outs can be avoided for all inventory positions with $s \geq \lambda\tau$. Hence, $\mathcal{S} = \{(s, 1) : s \geq \lambda\tau\}$ and (11)-(13) reads

$$\begin{aligned} \max_{u \geq 0, t \geq 0} \quad & tpV^* - \xi^*(u + \tau + t) - c - \frac{\phi pt}{2\lambda}(2s - 2\lambda(u + \tau) + t(p - \lambda)) \\ & s - \lambda(u + \tau) \geq 0 \\ & s - \lambda(\tau + t + u) + p t \geq \lambda\tau. \end{aligned}$$

In state $(s, 1)$, this simple greedy policy yields the optimal action of

$$(u^*, q^*, t^*) = \left(\frac{s}{\lambda} - \tau, 1, \max \left\{ \frac{\tau\lambda}{p - \lambda}, \sqrt{\frac{2c\lambda}{\phi p(p - \lambda)}} \right\} \right),$$

as can be verified by checking the KKT conditions.

Appendix D: Executing the Three-Step Heuristic

To illustrate our procedure, we discuss a small three-product problem in detail. In particular, take $\lambda_1 = \lambda_2 = \lambda_3 = 1$, $p_1 = p_2 = p_3 = 8$, $\phi_1 = 100, \phi_2 = \phi_3 = 10$, $c_{ij} = 1$ for all products i and j . We assume that setups to and from product 1 are faster than setups between products 2 and 3, so $\tau_{11} = \tau_{12} = \tau_{13} = \tau_{21} = \tau_{31} = 1$ and $\tau_{22} = \tau_{23} = \tau_{32} = \tau_{33} = 5$.

First, we solve the lower bound problem (9)-(10) to obtain $V_1 = 265.728$, $V_2 = V_3 = 109.456$, $\xi = 297.47$ and $\theta_1 = \theta_2 = \theta_3 = 0$, which yields the lower bound of 187.169.

To construct the initial state space approximation, note that states $((1.0, 3.4, 9.8), 3)$, $((9.8, 1.0, 7.4), 1)$, $((7.4, 9.8, 5.0), 2)$ are visited on a non-idling cyclic schedules with production sequence $\vec{q} = (1, 2, 3)$ and states $((1.0, 3.4, 9.8), 2)$, $((7.4, 9.8, 5.0), 1)$, $((9.8, 1.0, 7.4), 3)$ are visited on such a schedule with production sequence $\vec{q} = (1, 3, 2)$. This gives an initial state space approximation of $\hat{\mathcal{S}}^2(1) = \{(9.8, 1.0, 7.4), (7.4, 9.8, 5.0)\}$, $\hat{\mathcal{S}}^2(2) = \{(1.0, 3.4, 9.8), (7.4, 9.8, 5.0)\}$ and $\hat{\mathcal{S}}^2(3) = \{(1.0, 3.4, 9.8), (9.8, 1.0, 7.4)\}$. In this example, the average cost minimizing rotation schedule also has zero idling. Following such a rotation schedule would give average total cost of 588.268.

Figure A1 The set $\tilde{S}_{\geq}^{18}(1)$.

We start the lookahead heuristic in machine state $m = 1$ with inventories $\bar{s}^1 = (9.8, 1.0, 7.4)$. In part (i), we solve the 2-step lookahead with $m = 1$ and $\bar{s}^1 = (9.8, 1.0, 7.4)$, which yields $u^1 = 0$, $q^1 = 2$, $t^1 = 1.4$, $\bar{s}^2 = (7.4, 9.8, 5)$, $u^2 = 0$, $q^2 = 3$, $t^2 = 1.4$, and $\bar{s}^3 = (1.0, 3.4, 9.8)$ with $w = 1$. Since $\bar{s}^2$ and $\bar{s}^3$ are in the current state space approximation, there is no need to update the approximation; nothing is done in (ii). In (iii), we implement action $(u^1, q^1, t^1) = (0, 2, 1.4)$.

We repeat (i) to (iii) starting from the new state; i.e., (i) we solve the 2-step lookahead with initial $m = 2$ and inventories $\bar{s}^1 = (7.4, 9.8, 5.0)$. This gives $u^1 = 0$, $q^1 = 1$, $t^1 = 0$, $\bar{s}^2 = (6.4, 8.8, 4.0)$, $u^2 = 0$, $q^2 = 3$, $t^2 = 0.9714$, and $\bar{s}^3 = (4.4286, 6.8286, 9.8)$ with $w = 1$. So, before implementing any action, we update the approximation to get $\tilde{S}^3(1) = \{(9.8, 1.0, 7.4), (7.4, 9.8, 5.0), (6.4, 8.8, 4.0)\}$. We then follow action $q^1 = 1, u^1 = 0, t^1 = 0$ leading to inventory levels of $\bar{s}^2 = (6.4, 8.8, 4)$ and machine state 1, which is now an element in our state space approximation.

We repeat the procedure, starting from our new state; i.e., we solve the 2-step lookahead with initial $m = 1$ and inventories $\bar{s}^1 = (6.4, 8.8, 4.0)$. Again, the lookahead will suggest to setup for (but not produce) product 1, i.e. $u^1 = 0$, $q^1 = 1$, $t^1 = 0$, $\bar{s}^2 = (5.4, 7.8, 3.0)$, and then produce product 3 until one reaches a state within the approximation. So in steps (ii) and (iii), we implement this action and update the approximation to get $\tilde{S}^4(1) = \{(9.8, 1.0, 7.4), (7.4, 9.8, 5.0), (6.4, 8.8, 4.0), (5.4, 7.8, 3.0)\}$. This kind of action will also be suggested for the next two time epochs until we reach state $m = 1$, $\bar{s}^1 = (3.4, 5.8, 1.0)$, and $\tilde{S}^6(1) = \{(9.8, 1.0, 7.4), (7.4, 9.8, 5.0), (6.4, 8.8, 4.0), (5.4, 7.8, 3.0), (4.4, 6.8, 2.0), (3.4, 5.8, 1.0)\}$. In this state, the lookahead gives $u^1 = 0$, $q^1 = 3$, $t^1 = 1.4$, $\bar{s}^2 = (1.0, 5.8, 9.8)$, $q^2 = 1$, $u^2 = 0$, $t^2 = 1.4$, leading to a state in the state space approximation with $w = 1$.

Over time, the system will move to a cyclic schedule following the sequence $\vec{q} = (1, 1, 3, 1, 2, 1)$ at average total cost of 245.8 or 31% above the lower bound. This is a huge improvement over the original rotation schedule with total cost that are 314% above that lower bound.

In the postprocessing step, we take the sequence $\vec{q} = (1, 1, 3, 1, 2, 1)$ as fixed and minimize the average total cost of a cyclic schedule given $\vec{q}$ leaving inventories flexible. This is the objective stated by most of the literature on the ELSP, which does not consider initial inventories. Solving this fixed-sequence ELSP using MINOS gives (optimized) average total cost of 218.435 per time unit, only 16.7% above the lower bound.

Again, we cannot provide any guarantee that the best possible sequence is found. After further inspection of the problem, we could identify the sequence $\vec{q} = (3, 1, 2, 1)$ to yield average total cost of only 196.625 (5% above the lower bound). Both the state space approximation and the greediness (caused by the value function approximation) can be causes of this behavior. In fact, one could argue that the difference in the sequence is a result of the inability to reach the states needed for the corresponding schedule. In our case, the poor initial state space approximation clearly is the dominating factor. If we use the sequence $\vec{q} = (3, 1, 2, 1)$ as a basis for the initial state space approximation and start the heuristic in the same initial state, the heuristic will identify this sequence as desirable and follow it eventually.

The nature of the approximation could be exploited better. Figure A1 depicts the final state space approximation $\tilde{S}^{18}(1)$ in this example. At first sight, one might worry about a possible explosion of the state space approximation in terms of the values of $W_1, \dots, W_I$, especially in the early stages of the dynamic updates.

Looking at the initial approximation of $\mathcal{S}(1)$, $\tilde{\mathcal{S}}^2(1) = \{(9.8, 1.0, 7.4), (7.4, 9.8, 5.0)\}$, however, reveals a very common pattern that can be dealt with efficiently: if an inventory vector such as $\vec{s} = (6.4, 8.8, 4.0)$ dominates another inventory vector $\vec{s}'$ in the set in the sense that $s_i \leq s'_i$ for all $i \in \mathcal{I}$, one can actually dismiss $\vec{s}'$ when $\vec{s}$ is added, which increases the efficiency of the proposed heuristic. Adding $\vec{s} = (6.4, 8.8, 4.0)$ would then result in a new approximation $\tilde{\mathcal{S}}^2(1) = \{(9.8, 1.0, 7.4), (6.4, 8.8, 4.0)\}$ instead of $\tilde{\mathcal{S}}^3(1)$. Because of the structure of the state space approximation, this dismissal of $\vec{s}' = (7.4, 9.8, 5.0)$, does not change the actual approximation. It does not change Figure A1 since the dismissed inventory vector lies in the interior of $\tilde{\mathcal{S}}_{\geq}^{18}(1)$. In the example, this idea reduces the number of inventory vectors used in the final approximation of $\tilde{\mathcal{S}}(1)$ from $W_1 = 18$ to $W_1 = 13$ since 5 inventory vectors were dominated.

Appendix E: A Comparison to Dobson (1992)

If there is an initial state $(\vec{s}^1, m^1)$ and an initial state space approximation $\tilde{\mathcal{S}}_{\geq}^w(m)$ for all $m \in \mathcal{I}$, such that the price-directed heuristic suggested in Section 3 follows the optimal production schedule, we say that the price-directed heuristic *achieves optimality*. (And as a consequence, the Three-Step-heuristic will find the optimal schedule if it is initialized with a suitable initial state and state space approximation.)

As mentioned in Section 2, Adelman and Barz (2013) prove that the semi-infinite problem

$$\max_{\xi, \theta_1, V_1, \dots, \theta_I, V_I, \theta_I=0} \sum_{i=1}^I \lambda_i V_i - \xi \quad (\text{A1})$$

$$\theta_m - \theta_q + t p_q V_q - \xi \cdot (u + \tau_{mq} + t) \leq \kappa(\vec{s}, m, u, q, t) \quad \forall (\vec{s}, m) \in \mathcal{S}, (u, q, t) \in \mathcal{A}(\vec{s}, m) \quad (\text{A2})$$

$$\xi \in \mathbb{R}, V_i, \theta_i \in \mathbb{R} \quad \forall i \in \mathcal{I}, \quad (\text{A3})$$

which is obtained by plugging the approximation (7)-(8) into the optimality equations (5)-(6), gives a lower bound on the optimal average total cost and its value is equal to (9)-(10) if $\tau_{qq}^{sp} \leq \tau_{mq}(1 - \rho_q)/(1 - \sum_{i=1}^I \rho_i)$ for all $i \in \mathcal{I}$. Under a slightly stronger condition,

$$\frac{p_q V_q^* - \xi^*}{\phi_q p_q} > \tau_{qq}^{sp} \quad \text{for all } m, q \in \mathcal{I}. \quad (\text{A4})$$

they imply in the proof of Proposition 2 that the feasible sets of (9)-(10) and (A1)-(A3) are the same. They also show that there is no duality gap between the semi-infinite problem and its dual, which is given by

$$\inf_{J, (\vec{s}^j, m^j), (u^j, q^j, t^j), x^j, j=1, \dots, J} \sum_{j=1}^J \kappa(\vec{s}^j, m^j, u^j, q^j, t^j) \cdot x^j \quad (\text{A5})$$

$$\sum_{j=1}^J x^j t^j p_{qj} 1\{q_j = i\} = \lambda_i, \quad i = 1, \dots, I \quad (\text{A6})$$

$$\sum_{j=1}^J x^j \cdot (1\{m^j = i\} - 1\{q^j = i\}) = 0, \quad i = 1, \dots, I-1 \quad (\text{A6})$$

$$\sum_{j=1}^J x^j \cdot (u^j + \tau_{mq^j} + t^j) = 1 \quad (\text{A7})$$

$$J \in \mathbb{N}_0, x^j \geq 0, (\vec{s}^j, m^j) \in \mathcal{S}, (u^j, q^j, t^j) \in \mathcal{A}(\vec{s}^j, m^j) \quad j = 1, \dots, J. \quad (\text{A8})$$

Denote by $J^*, (\vec{s}^{j*}, m^{j*}), (u^{j*}, q^{j*}, t^{j*}), x^{j*}, j = 1, \dots, J^*$ an optimal solution to (A5)-(A8). Without loss of optimality, choose J^* such that all $x^{j*} > 0$ and define the frequency of state-action pair j as $\eta^j := x^{j*}/\tilde{x}$ with $\tilde{x} = \max\{x \in \mathbb{R} : x^{j*}/x \in \mathbb{N} \forall j^* = 1, \dots, J^*\}$. A solution is implementable if there exists a cyclic schedule $\{(u^{[n]}, q^{[n]}, t^{[n]})\}_{n=1, \dots, N}$ cycling through states $(\vec{s}^{[n]}, m^{[n]})$ with $m^{[n+1]} = q^{[n]}$, $s_i^{[n+1]} = s_i^{[n]} - \lambda_i \cdot (u^{[n]} + \tau_{m^{[n]}q^{[n]}} + t^{[n]}) + \delta(q^{[n]} = i)p_i t^{[n]}$ for all $n = 1, \dots, N-1$, $m^{[1]} = q^{[N]}$, and $s_i^{[1]} = s_i^{[N]} - \lambda_i \cdot (u^{[N]} + \tau_{m^{[N]}q^{[N]}} + t^{[N]}) + \delta(q^{[N]} = i)p_i t^{[N]}$ that fulfills

$$\sum_{n=1}^N \delta((s^{[n]}, m^{[n]}) = (\vec{s}^{j*}, m^{j*}), (u^{[n]}, q^{[n]}, t^{[n]}) = (u^{j*}, q^{j*}, t^{j*})) = \eta^j \quad \forall j = 1, \dots, J^*.$$

A cyclic schedule is implementable as a power-of-two schedule if there exist integers $k_q \in \mathbb{N}_0$ such that $\sum_{j=1}^{J^*} \eta^{j*} \delta(q^{j*} = q) = 2^{k_q}$ for all $q \in \mathcal{I}$.

THEOREM A1. *Assume there is a cyclic schedule with positive production times and average total cost equal to the solution of (9)-(10), i.e. the lower bound given by (9)-(10) is tight, and its solution $V_1^*, \dots, V_I^*, \theta_1^*, \dots, \theta_{I-1}^*, \xi^*$ satisfies (A4). Then, the price-directed heuristic achieves optimality.*

Proof of Theorem A1. First, note that an optimal cyclic schedule can be mapped into an optimal solution to (A5)-(A8). Without loss of generality, we assume that the optimal cyclic schedule never visits the same state $(\vec{s}, m)$ more than once. Denote this solution as $J^*, (\vec{s}^{j*}, m^{j*}), (u^{j*}, q^{j*}, t^{j*}), x^{j*}, j = 1, \dots, J^*$. Because we assumed that production times are positive, we know that $t^{j*} > 0$ for all $j = 1, \dots, J^*$.

Adelman and Barz (2013) give a dual formulation of (9)-(10), called NLP, that is based on the assumption that production of product q is started only when its inventory level is 0. This property is sometimes referred to as zero-switch. They show in their Theorem 3 that the value of NLP is equal to the value of (9)-(10), which implies that the value of NLP equals (A5)-(A8) under (A4).

Because $J^*, (\vec{s}^{j*}, m^{j*}), (u^{j*}, q^{j*}, t^{j*}), x^{j*}, j = 1, \dots, J^*$ is optimal to (A5)-(A8), the construction given in step 1 of the proof of Theorem 3 in Adelman and Barz (2013) implies $t^{j*} (s_q^{j*} (u^{j*} + \tau_{m^{j*}q^{j*}})) = 0$. As we assume $t^{j*} > 0$, we know that it satisfies $s_q^{j*} (u^{j*} + \tau_{m^{j*}q^{j*}}) = 0$, i.e. the last term in (3) must be 0.

Now, start the price-directed heuristic on a state that is visited following the optimal cyclic schedule $(\vec{s}^{[1]}, m^{[1]})$, choose a state space approximation that includes the states $(\vec{s}^{[n]}, m^{[n]})$ visited by the optimal cyclic schedule for $n = 1, \dots, N$ and maximize the net-value $\theta_{m^{[1]}}^* - \theta_q + t p_q V_q^* - \xi^* \cdot (u + \tau_{m^{[1]}q} + t) - \kappa(\vec{s}^{[1]}, m^{[1]}, u, q, t)$ subject to the constraint of not leaving the current approximation of $\mathcal{S}$.

Following the proof of Proposition 2 in Adelman and Barz (2013), one can show that if (A4) holds, an optimal solution to (9)-(10) will also be an optimal solution to (A1)-(A3). Using the Complementary Slackness Theorem for semi-infinite LPs in Glashoff and Gustafson (1983, p. 95) (and noting that the proof works for any optimal finite J^*) it follows that state-action pairs $(\vec{s}, m), (u, q, t)$ that mimic the optimal cyclic schedule will have a net-value of zero. Since the solution $V_1^*, \dots, V_I^*, \theta_1^*, \dots, \theta_{I-1}^*, \xi^*$ is feasible, all other actions have a net-value that is negative or zero.

Consider the set of optimal actions with net-values of 0. Recall that we break ties by choosing the product q with the smallest time to stock-out s_q/λ_q and that $(\vec{s}^{[1]}, m^{[1]})$ is a state on the optimal cyclic schedule, which performs action $(u^{[1]}, q^{[1]}, t^{[1]})$ next and has the zero-switch property implying that

$$s_{q^{[1]}} - \lambda_{q^{[1]}} (u^{[1]} + \tau_{m^{[1]}q^{[1]}}) = 0. \quad (\text{A9})$$

We show by contradiction that given state $(\vec{s}^{[1]}, m^{[1]})$, there cannot be a product $q \neq q^{[1]}$ with $s_q/\lambda_q \leq s_{q^{[1]}}/\lambda_{q^{[1]}}$ and therefore the price-directed heuristic will choose to next produce product $q^{[1]}$. Assume there was such a q . Then, $s_q - \lambda_q (u^{[1]} + \tau_{m^{[1]}q^{[1]}}) \leq s_{q^{[1]}} - \lambda_{q^{[1]}} (u^{[1]} + \tau_{m^{[1]}q^{[1]}}) = 0$ by (A9). This contradicts feasibility of the optimal cyclic schedule since only one product can be produced at a time.

For fixed q , the net-value problem has a unique maximum in u and t . Knowing that the net-value of $(u^{[1]}, q^{[1]}, t^{[1]})$ is 0, this must be the maximum and the next action chosen is $(u^{[1]}, q^{[1]}, t^{[1]})$ leading to $(\vec{s}^{[2]}, m^{[2]})$. Since $(\vec{s}^{[2]}, m^{[2]})$ is in the state space approximation by assumption, this action is feasible. Repeat this argument for all $n = 1, \dots, N$ to show that the optimal cyclic schedule is followed by the price-directed heuristic. $\square$

Since Dobson's heuristic only searches over the set of cyclic power-of-two schedules, we obtain the following corollary:

COROLLARY A1. *Assume (9)-(10) has a solution satisfying (A4) and there is a cyclic schedule with positive production times and average total cost equal to the solution of this lower bound problem but no power-of-two schedule obtains the same average total cost. Then, the price-directed heuristic achieves optimality, whereas Dobson's (1992) heuristic finds a suboptimal schedule.*

In the sequence-independent case, the gap between the best power-of-two schedule and the optimal schedule can be shown to be at most 6%, see Roundy (1989). In the sequence-dependent setup case, however, the gap between the performance of Dobson's heuristic and the optimal policy can be unbounded. We will illustrate this in an example, in which the price-directed heuristic will find the optimal policy outperforming Dobson's policy arbitrarily as the setup-costs c_{ii} increase. The example has the same structure as the fourth problem considered in the computational results section. Production rates are larger and setup times are lower in the following to guarantee that the lower bound is tight.

Consider a $I = 5$ product problem with identical demand rates, production rates, and sequence-independent setup times, $\lambda_i = \lambda = 1$, $p_i = p = 100$, and $\tau_{ij} = \tau = 0.1$ for all $i, j = 1, \dots, 5$. Holding costs are given by $\phi_1 = \phi_2 = \phi_3 = 0.01$, $\phi_4 = 0.04$, and $\phi_5 = 0.09$. Setups from product 5 to products 1, 2, and 3 are cheap, so are setups from products 1 and 2 to product 4, and from 3 and 4 to product 5. Setups from product i to i have intermediate costs, other setups are expensive, i.e. take

$$(c_{ij}) = \begin{pmatrix} c_2 & c_3 & c_3 & c_1 & c_3 \\ c_3 & c_2 & c_3 & c_1 & c_3 \\ c_3 & c_3 & c_2 & c_3 & c_1 \\ c_3 & c_3 & c_3 & c_2 & c_1 \\ c_1 & c_1 & c_1 & c_3 & c_2 \end{pmatrix}$$

with $0.01 \leq c_1 < c_2 \leq c_3$.

The example was constructed such that the cyclic schedule with production sequence $\vec{q} = (5, 2, 4, 5, 3, 5, 1, 4)$ is optimal. Production times of the optimal schedule are $\sqrt{c_1/(4950\phi_q)}$ every time product q is produced, idling times are chosen such that product q is setup for when its inventory level is $s_q = \tau$. The average total cost rate incurred by this optimal cyclic schedule is $8\sqrt{0.0198c_1}$.

It is easy to verify that solving (9)-(10) gives the solution of $\xi^* = 0$, $V_i^* = \sqrt{1.98\phi_i c_1}$, $\theta_i^* = 0$ for all $i = 1, \dots, 5$ and a lower bound of $8\sqrt{0.0198c_1}$. Costs of $8\sqrt{0.0198c_1}$ can be obtained by the optimal schedule outlined above. So the bound is tight and since (A4) holds the assumptions of Theorem ?? apply. Starting at inventory levels visited by the cyclic schedule, the price-directed heuristic will find the optimal solution at a cost rate of $8\sqrt{0.0198c_1}$.

In this example, Dobson's lower bound problem identifies the optimal frequencies to be $f_1 = f_2 = f_3 = 1$, $f_4 = 2$ and $f_5 = 3$ for any choice of $c_1 < c_2 \leq c_3$. Rounding to the closest power-of-two would either give $f_5 = 4$ or $f_5 = 2$ (it is not specified which closest power-of-two would be chosen). Because products 1, 2, and 3 can only be setup at c_1 when coming from $m = 5$, at least one setup with cost of c_3 is unavoidable when $f_5 = 2$. Product 5 can only be setup at c_1 when coming from products 3 or 4. Given $f_4 = 1$ and $f_4 = 2$, at least one setup will be at a cost of c_2 or c_3 if $f_5 = 4$. As a consequence, at least one setup with cost of c_2 or c_3 will be performed in any production sequence obtained by Dobson's heuristic. Because $f_1 = 1$, a weak lower bound on the average total cost produced by this heuristic is given by the one-product solution for product 1 with setup cost of c_2 plus the one-product solutions for all other products with setup cost of c_1 , $\sqrt{0.0198c_2} + 7\sqrt{0.0198c_1}$. For given $c_2 > c_1$ Dobson's heuristic will always be at least $\sqrt{(c_2/c_1 - 1)}/8 \cdot 100\%$ worse than the price directed heuristic. By changing c_2 , one can therefore construct schedules for which the price-directed heuristic arbitrarily outperforms Dobson's heuristic.

For $c_1 = 49.5$ and $c_2 = c_3 = 1000$, the optimal average cost is e.g. 7.92 and the corresponding schedule is easily identified by our heuristic. (It identifies the optimal schedule even when it is started on an inventory state that is on the schedule created by Dobson's heuristic and this schedule is used as an initial state space approximation.) From the lower bound alone, we know that Dobson's heuristic will lead to average cost of at least $\sqrt{0.0198c_2} + 7\sqrt{0.0198c_1} = 11.38$, the exact value is 14.15. Increasing c_2 and c_3 leaves the optimal sequence found by the price-directed heuristic unchanged but will increase the cost of the best schedule found by Dobson's heuristic.

Appendix F: Problem Data Used in Computational Experiments

Problem 1 is taken directly from Boctor (1982). The original data set for problem 2, stated by Sing and Foster (1987), c_{ii} and τ_{ii} are not given for $i \in I$. If one would take those values to be 0, an optimal solution would rather produce product i in tiny batches than ever produce i and then idle. Looking at Sing and Foster (1987)'s solution approach, it is clear, that this was not their intention. We therefore filled those values by choosing $\tau_{ii} = \min_{k \in I, k \neq i} \tau_{ki}$ and $c_{ii} = \min_{k \in I, k \neq i} c_{ik} + c_{ki}$. Problem 3 is given in Mallya (1992), the data for problem 4 is stated in the text. Problem 5 was originally stated in Bomberger (1966), the data presented is taken from Grznar and Riggie (1997).

#	i	λ_i	p_i	ϕ_i	$c_{ij}, j = 1, \dots, I$	$\tau_{ij}, j = 1, \dots, I$
1	1	20000	160000	.005	15 25	.0125 .0250
	2	27000	162000	.004	15 25	.0125 .0250
2	1	100000	500000	.01	80000 50000 100000	1.0 .5 2.0
	2	150000	300000	.02	30000 80000 50000	1.0 .5 2.0
	3	10000	200000	.01	10000 30000 80000	1.0 .5 2.0
3	1	431.0	1800.0	.0013265	80 140 60 100 60	.2 .35 .15 .25 .15
	2	375.0	2500.0	.000882	80 140 60 100 60	.2 .35 .15 .25 .15
	3	480.0	4000.0	.0013685	80 140 60 100 60	.2 .35 .15 .25 .15
	4	895.0	3200.0	.000987	80 140 60 100 60	.2 .35 .15 .25 .15
	5	151.0	1500.0	.000378	80 140 60 100 60	.2 .35 .15 .25 .15
4	1	1	20	1	380 570 570 190 570	.2 .2 .2 .2 2
	2	1	20	1	570 380 570 190 570	.2 .2 .2 .2 2
	3	1	20	1	570 570 380 570 190	1 1 1 1 2
	4	1	20	4	570 570 570 380 190	.2 .2 .2 .2 2
	5	1	20	9	190 190 190 570 380	.2 .2 .2 .2 2
5	1	100	30000	.0000027083	15 20 30 10 110 50 310 130 200 5	.125 .125 .250 .125 .500 .250 1.00 .500 .750 .125
	2	100	8000	.0000489583	15 20 30 10 110 50 310 130 200 5	.125 .125 .250 .125 .500 .250 1.00 .500 .750 .125
	3	200	9500	.0000531250	15 20 30 10 110 50 310 130 200 5	.125 .125 .250 .125 .500 .250 1.00 .500 .750 .125
	4	400	7500	.0000416667	15 20 30 10 110 50 310 130 200 5	.125 .125 .250 .125 .500 .250 1.00 .500 .750 .125
	5	20	2000	.0011604167	15 20 30 10 110 50 310 130 200 5	.125 .125 .250 .125 .500 .250 1.00 .500 .750 .125
	6	20	6000	.0001114583	15 20 30 10 110 50 310 130 200 5	.125 .125 .250 .125 .500 .250 1.00 .500 .750 .125
	7	6	2400	.0006250000	15 20 30 10 110 50 310 130 200 5	.125 .125 .250 .125 .500 .250 1.00 .500 .750 .125
	8	85	1300	.0024583333	15 20 30 10 110 50 310 130 200 5	.125 .125 .250 .125 .500 .250 1.00 .500 .750 .125
	9	85	2000	.0003750000	15 20 30 10 110 50 310 130 200 5	.125 .125 .250 .125 .500 .250 1.00 .500 .750 .125
	10	100	15000	.0000166667	15 20 30 10 110 50 310 130 200 5	.125 .125 .250 .125 .500 .250 1.00 .500 .750 .125

References

- Adelman, D. and Barz, C. (2013). A unifying approximate dynamic programming approach to the economic lot scheduling problem. To appear in *Mathematics of Operations Research*.
- Boctor, F.F. (1982). The two-product, single machine, static demand, infinite horizon scheduling problem. *Management Science*, Vol. 28, 787-807.
- Bomberger, E.E. (1966). A dynamic programming approach to a lot size scheduling problem. *Management Science*, Vol. 12, 778-784.
- Dobson, G. (1992). The cyclic lot scheduling problem with sequence-dependent setups. *Operations Research*, Vol. 40, 736-749.
- Glashoff, K., and Gustafson, S.-A. (1983). *Linear optimization and approximation*. Springer, New York-Heidelberg-Berlin.
- Grznar, J., and Riggie, C. (1997). An optimal algorithm for the basic period approach to the economic lot scheduling problem. *Omega*, Vol. 25, 335-364.
- Mallya, R. (1992). Multi-product scheduling on a single machine: a case study. *Omega*, Vol. 20, 529-534.
- Roundy, R. (1989). Rounding off to powers of two in continuous relaxations of capacitated lot sizing problems. *Management Science*, Vol. 35, 1433-1442.
- Sing, H., and Foster, J.B. (1987). Production Scheduling with sequence dependent setup costs. *IIE Transactions*, Vol. 1, 43-49.
- Zipkin, P.H. (2000). *Foundations of inventory management*. McGraw-Hill International Editions, Singapore.