

Technical note

Optimal and heuristic policies for lot sizing with learning in setups

Ram Rachamadugu^{a,*}, Thomas J. Schriber^b

^a *ISOM Department, College of Business Administration, The University of Toledo, Toledo, OH 43606, USA*

^b *CIS Department, Graduate School of Business, The University of Michigan, Ann Arbor, MI 48109, USA*

Abstract

We develop heuristic and optimal methods for determining lot sizes when setup cost reductions occur over time due to emphasis on continuous improvement, learning effects and incremental process changes. Our heuristic methods are intuitively appealing, easy to implement, require little information about setup costs, and have low computational burden. Computational studies show that choosing the appropriate heuristic yields nearly optimal solutions. Recommendations for choosing the appropriate heuristic are also provided. The optimal method developed in this paper is also useful to managers for evaluating investments in hardware and/or worker training for setup reduction. Concepts and methods are illustrated with numerical examples. Managerial implications of using our policies are discussed.

1. Introduction

Over the last several years, focus on the high productivity of the Japanese manufacturing sector has resulted in efforts to identify and adapt some of these methods to other environments. One of the factors cited for the responsiveness and efficiency of Japanese manufacturing methods is the small lot sizes resulting from a constant emphasis on reducing setup costs. The adoption of Japanese manufacturing techniques and emphasis on continuous improvement initiatives has resulted in large reduction in setup times and costs being achieved in many industries. For example, consider Table 1 which shows the trends in die setup times for the same stamping

presses over years in two automotive plants (McElroy, 1986, 1987, 1988, 1989, 1990, 1991).

Suzaki (1987) mentions how setup time reductions at a household goods manufacturer were achieved over time through engineering efforts such as externalizing setup times. Plossl (1985) suggests that the benefits of reforms in JIT related manufacturing activities accrue over several years with annual improvements in the range 2–40%.

Many initiatives and factors such as emphasis on continuous improvement (Schonberger, 1982), incremental process improvements (Fine and Porteus, 1989) and learning effects (Chand, 1989; Mekler, 1993) contribute to decreasing setup costs. Citing empirical studies by Rosenberg (1982) and Hollander (1965) from industry, Fine and Porteus (1989) suggest that the cumulative benefits of incremental process changes occurring over time can be a major source of improvement in manufacturing systems.

* Corresponding author.

Table 1
Trends in die setup times in automotive stamping plants (in minutes)

Plant (location)	Equipment (type)	1986	1987	1988	1989	1990	1991
Honda II (Marysville, OH)	Semi-automatic	5:14	4:38	3:57	2:49	2:52	2:36
GM-BOC (Buick City, MI)	Semi-automatic	N/A	N/A	18:16	15:33	10:32	10:47

Learning effects also play an important role in setup reductions. Chand (1989) refers to the comments by practitioners at an APICS conference: “We in the USA do not setup as frequently as the Japanese do because our setup costs are high; and, our setup costs are high because we do not learn to setup by setting up as frequently as the Japanese do”.

The above empirical and anecdotal evidence suggests that practitioners need appropriate decision tools to aid in determining lot sizes when setup costs decrease over time due to factors such as emphasis on continuous improvement, learning effects and incremental process improvements. Further, such tools are also useful in evaluating investments in such activities. In this paper, we develop both heuristic and optimal methods to determine lot sizes for these environments.

First we develop a procedure for determining optimal lot sizes using net present value methods. This procedure requires the use of a dynamic programming based procedure, an approach more difficult to comprehend than simple lot sizing procedures. Like any other optimal procedure, it also requires full information regarding all future setup costs, which may be difficult to estimate. Under such circumstances, practitioners prefer to use effective heuristic policies which are easy to comprehend, and are modest in their information and computational requirements. Hence we also investigated two heuristic policies which are intuitively appealing, easy to understand, require little future information, and also are of low computational burden. One of them is based on the minimum achievable setup cost. The other is the continuous time version of the well known Part Period Balancing method (Zoller and Robrade, 1988). Computational results show that our heuristic policies are very effective. We also present insights into their worst case behavior which help in

identifying situations where the use of a heuristic may not be appropriate. Further, our two heuristics supplement each other in the sense that while one of them performs well in machine intensive (automated) situations, the other is suitable for labor intensive situations.

Our paper is organized as follows. The next section provides a review of the relevant literature. In Section 3, we provide a mathematical formulation of the problem, and develop a method for determining optimal lot sizes. It is illustrated with numerical examples. We also demonstrate how our method can be used to evaluate investments in improving learning and/or lowering the limit on the minimum achievable setup cost. In Section 4, we describe two heuristic policies and provide insights into their worst case behavior. Section 5 provides details of our computational study which shows how the heuristics perform across a broad range of possible scenarios, and also suggests when one policy is better than the other. Finally, in Section 6, we provide conclusions. Secondary aspects such as optimal procedures, analytical derivations and detailed computational results are provided in the appendices.

2. Literature review

There are many early studies on the effects of learning on lot sizes. Keachie and Fontana (1966), Adler and Nanda (1974), Fisk and Ballou (1982), Smunt and Morton (1985), Klastorin and Moinzadeh (1989), Karwan et al. (1988), among others, addressed the problem of determining optimal lot sizes when the runtime costs (process costs such as material cost) and/or holding costs decrease due to learning effects. Some of these studies used dynamic programming procedures to determine lot sizes. Be-

cause they used either the cumulative number of units produced and/or the cumulative number of setups to define the state space, large state space requirements limited applicability of their analysis to finite horizon situations.

Chand (1989) addressed a similar problem in a pure setup learning environment in which only setup costs are susceptible to improvements. Karwan et al. (1988) also studied this situation for a special case of learning in setups, in which setup costs decrease according to the well known power function (Argote and Epple, 1990), and for finite horizons. Chand's model (Chand and Sethi, 1990) is useful in environments where runtime operations are machine paced (hence little scope exists for learning) and setup operations are labor intensive. Chand denotes these situations as "semi-automatic" environments. His study is also limited to finite horizons, but his analysis permits any decreasing setup cost structure. He suggests a binary search procedure for determining the optimal number of setups. Extending his analysis to long planning periods (or infinite horizon situations) leads to the conclusion that learning effects should be disregarded, and only the minimum achievable setup cost should be the basis for lot sizing. Chand also suggested a variable lot sizing policy (similar to (Replogle, 1988)) as a heuristic, but did not explore its effectiveness. Based on average cost analysis, Cheng (1991) suggested an optimal lot sizing policy for finite horizons using numerical approximation methods.

Chand and Sethi (1990) recently extended Chand's study (Chand, 1989) to a discrete time, dynamic demand version of the problem. This formulation permits modelling varying demand. Though the basic Wagner–Whitin model (Wagner and Whitin, 1958) can be used in concept to analyze setup learning with varying demand situations, Chand and Sethi (1990) developed an efficient forward dynamic programming procedure which can be used effectively on a rolling horizon basis for infinite horizon problems. Also, they present decision/forecast horizon results which can further reduce both computational effort and the amount of demand forecast information needed.

There is also a parallel stream of research for determining optimal investments in setup cost reduction. Porteus (1985a, 1986) studied the effects on lot

sizes of investing in setup cost reduction. However, the framework used in his studies makes them applicable to situations in which the entire reduction in setup costs occurs immediately (or at some point in time). Porteus' results help to identify optimal investment strategies when setup cost can be characterized as a logarithmic or power function of investment in setup reduction. Billington (1987) extended the analysis to situations in which the setup cost is a linear or exponential (continuous) function of the investment in setup reduction for infinite horizon, constant demand problems. However, Mekler (1993) suggested that the investment opportunities to reduce setup costs are discrete, and shows how the Wagner–Whitin procedure (Wagner and Whitin, 1958) can be used to evaluate investments for discrete time, varying demand situations. Her approach is useful for finite horizon, varying demand situations.

Our study extends earlier research in two ways. First, we develop and then characterize optimal policies for lot sizing in infinite horizon situations (such as the manufacture of commodity products which have stable demand over long periods of time) when additional reductions in setup cost persist to occur due to factors such as emphasis on continuous improvement, incremental process changes and learning. Unlike earlier studies, we use net present value methods to capture the effects of variations in setup costs over long horizons. Second, we investigate and evaluate two heuristic lot sizing policies which are intuitively appealing and easy to understand. Our methods are also useful for evaluating investments in initiatives such as worker learning and training where payoffs are not necessarily instantaneous.

Our research also provides an opportunity to verify a suggestion made by Chand (1989) and Karwan et al. (1988) about the use of lot size corresponding to the minimum possible setup cost as a heuristic. Also, Hadley (1964), based on his computational experiments, suggested that the average cost analysis is an excellent surrogate for the conceptually rigorous net present value analysis for lot sizing decisions. Because we use the net present value analysis in our study, our paper provides an opportunity to validate his conjecture, albeit in the context of non-stationary parameter situations.

The notation used in the paper is shown in Table 2. Whenever a parameter is invariant, we suppress

Table 2
Notation

Acronyms	
CURS	current setup cost lot sizing policy
MINS	minimum setup cost lot sizing policy
npv	net present value
PPB	part period balancing heuristic
Parameters and variables	
P	price per unit
D	demand rate
N	the setup index (number) at which minimum setup cost is first realized
S_{\min}	asymptotic limit on the setup costs due to technological constraints
S_i	setup cost for the i th setup, $S_1 \geq S_2 \geq \dots \geq S_i \geq S_{i+1} \dots \geq S_{\min} = S_N = S_{N+1} \dots$
r	discount rate
h	noncapital holding charges per unit per period
ϕ	learning rate. This indicates how fast setup costs decrease. For example, a 90% learning rate implies that when the cumulative number of setups doubles, the cost for the last setup equals 90% of the median setup cost.
b	learning index, $\phi = 2^{-b}$
R	setup cost reduction factor, S_{\min}/S_1
T_i	replenishment interval corresponding to the i th setup, i.e., units ordered equals DT_i
$F(i)$	net present value of all future costs in the optimal policy, discounted to the time at which the i th setup occurs (the value of the the optimal policy is given by $F(1)$)
$V(T_i)$	npv of costs incurred in the i th setup cycle, discounted to the time at which the i th setup occurs
T_i^*	optimal replenishment interval for the i th setup
T_{\min}^*	optimal replenishment interval if all setup costs are equal to S_{\min}
$J(i)$	net present value of the optimal policy for the problem in which each setup cost equals S_i
\tilde{T}_i	optimal replenishment interval for the problem in which all setup costs equal S_i
$G_{\text{CURS}}(i)$	net present value of all costs incurred in CURS policy beginning from the i th setup, discounted to the time at which the i th setup occurs
$G_{\text{MINS}}(i)$	net present value of all costs incurred in MINS policy beginning from the i th setup, discounted to the times at which the i th setup occurs

the subscript. For example, the setup cost for the i th setup is denoted by S_i . However, if all setup costs are equal, then the setup cost is denoted by S .

3. Problem statement and optimal lot sizing

Consider a manufacturing environment in which setup costs decrease because of investments in worker learning and/or emphasis on continuous improvement. Since the process is assumed to be machine intensive, very little or no improvement in runtime costs is expected to occur (Chand, 1989). S_i denotes the setup cost for the i th setup. Because of continuous improvement and learning effects,

$$S_1 \geq S_2 \geq S_3 \geq \dots \geq S_i \geq S_{i+1} \geq \dots \geq S_N (= S_{\min}) = S_{N+1} = S_{N+2} \dots, \tag{1}$$

where S_{\min} is the minimum achievable setup cost. This is realized at the N th setup. Let D , P and r denote the demand rate, price per unit and the discount (interest) rate, respectively. h represents the noncapital holding charges per unit per period. Let T_i represent the replenishment interval for the i th replenishment cycle, i.e., the amount ordered or produced in that cycle will be DT_i units. $V(T_i)$, the npv of all costs associated with the i th cycle discounted to the time at which the i th setup takes place, is given as follows:

$$V(T_i) = S_i + DPT_i + \int_0^{T_i} Dh(T_i - t)e^{-rt} dt. \tag{2}$$

The first term in (2) represents the setup cost for the i th cycle. The second term represents the runtime costs (such as materials) incurred for the cycle. The

third term represents the npv of inventory carrying costs (noncapital charges such as material handling and warehousing) discounted to the time at which the i th setup takes place (note that the interest charges for the capital tied up in inventory is implicitly recognized through discounting). We need to determine optimal replenishment intervals. The problem can be mathematically stated as follows:

$$\text{minimize } \sum_{i=1}^{\infty} \exp\left\{-r \sum_{j=1}^{i-1} T_j\right\} V(T_i), \quad (3)$$

$$\text{subject to } T_i \geq 0 \quad \forall i = 1, 2, \dots \quad (4)$$

Eq. (3) states that the sum of net present values of costs related to each replenishment cycle ($V(T_i)$) discounted to time zero should be minimized. Eq. (4) states that the replenishment intervals are nonnegative. We solve this optimization problem using a dynamic programming procedure. Details of the dynamic programming procedure are provided in Appendix A. We next illustrate the method with numerical examples.

3.1. Numerical examples

Suppose the demand (D) equals 2000 units per year, the initial setup cost (S_1) is \$310, $S_{\min} = \$81.26$, $P = \$10$ per unit, $r = 20\%$ per year and $h = \$1.95$ per unit per year. Setup costs are expected to decrease due to emphasis on worker training and continuous improvement. For convenience, assume that the setup costs can be computed using the well known power function shown below (Argote and Epple, 1990; Yelle, 1979):

$$S_i = S_1 i^{-b} \quad \forall \{i | S_i \geq S_{\min}\}, \quad (5)$$

where S_1 is the initial setup cost and b is an index characterizing the learning rate (ϕ). Prior experience indicates that the setup costs decline at an 80% learning rate. An 80% learning rate implies that when the cumulative number of setups is doubled, the marginal setup cost reduces to 80% of the cost for the median setup. We need to determine optimal lot sizes, and the npv of the optimal policy. From (5),

Learning rate (ϕ)

$$= \frac{S_i}{S_{2i}} = 0.8 = 2^{-b} \quad \forall \{i | S_{2i} \geq S_{\min}\}. \quad (6)$$

From (6), b equals 0.321928. Empirical studies (Argote and Epple, 1990) suggest that in most cases $0 \leq b \leq 1$. Further, it can be verified that the minimum setup cost is achieved at the 64th setup. Because no further reductions in setup costs are anticipated, optimal lot sizes for the 64th and subsequent setups can be found by solving the discounted cash flow version of the classical economic order quantity problem. Eq. (7) is used to find T_{\min}^* ($= \check{T}_{64}$, the optimum replenishment interval if all setup costs are equal to S_{64}):

$$\frac{S_{\min} r^2}{D(h + Pr)} = e^{r\check{T}_{64}} - 1 - r\check{T}_{64}. \quad (7)$$

Eq. (7) is the same as (A.7) in Appendix A. Eq. (7) can be solved efficiently for \check{T}_{64} using the iterative procedure suggested by Porteus (1985b), starting with the classical square root formula (also known as the economic order quantity) as the initial solution. Hence $\check{T}_{64} = T_{64}^* = 0.142754$. Next we find $F(64)$, the npv of all costs incurred in all future replenishment cycles starting at the 64th setup, discounted to the time at which the 64th setup occurs. This is found using Eq. (A.6) in Appendix A.

$$F(64) = \left(S_{\min} + DP\check{T}_{64} + \frac{Dh}{r^2} (r\check{T}_{64} - 1 + e^{-r\check{T}_{64}}) \right) \times (1 - e^{-r\check{T}_{64}})^{-1}. \quad (8)$$

$F(64)$ equals \$105720. Using \check{T}_{64} , and $F(64)$, and applying (A.5) and (A.4) iteratively backwards we find optimal lot sizes for setups 63 to 1. We terminate with $F(1)$, npv of the optimal policy. Computational results are shown in Table 3. Since Eqs. (A.5) and (A.4) are applied backwards, results in the table are shown in decreasing order of the setup index. The value of the optimal policy ($F(1)$) equals \$107299. It is clear from Table 3 that the lot sizes decrease rapidly. We elaborate on this characteristic of the optimal policy in Section 3.2.

Note that $F(1)$ includes the npv of inventory related costs, and also variable raw material costs. Clearly, irrespective of the type of lot sizing decisions made, we need to spend \$100000 on raw

Table 3
Results for the numerical example

Setup no. (<i>i</i>)	Setup cost (S_i)	Optimal lot size DT_i^*	$F(i)$
≥ 64	81.26	285.5	105720
63	81.67	285.5	105720
62	82.09	285.5	105721
61	82.53	285.5	105722
60	82.97	285.6	105724
59	83.42	285.7	105726
58	83.88	285.8	105729
57	84.35	285.9	105731
56	84.83	286.0	105735
55	85.32	286.2	105738
54	85.83	286.4	105742
53	86.35	286.6	105747
52	86.88	286.8	105752
51	87.42	287.0	105757
50	87.98	287.3	105763
49	88.56	287.6	105769
48	89.15	287.9	105775
47	89.75	288.2	105782
46	90.38	288.5	105789
45	91.02	288.9	105797
44	91.68	289.3	105805
43	92.36	289.7	105814
42	93.06	290.1	105823
41	93.79	290.5	105833
40	94.53	291.0	105843
39	95.31	291.5	105853
38	96.11	292.0	105865
37	96.94	292.6	105876
36	97.80	293.1	105888
35	98.69	293.7	105901
34	99.61	294.3	105914
33	100.57	295.0	105928
32	101.58	295.7	105942
31	102.62	296.4	105957
30	103.71	297.1	105973
29	104.85	297.9	105989
28	106.04	298.7	106006
27	107.29	299.5	106024
26	108.60	300.4	106042
25	109.98	301.3	106062
24	111.43	302.3	106082
23	112.97	303.3	106103
22	114.60	304.3	106126
21	116.33	305.4	106149
20	118.17	306.5	106173
19	120.14	307.7	106199
18	122.25	309.0	106226
17	124.52	310.3	106254
16	126.97	311.7	106284
15	129.64	313.2	106316
14	132.55	314.7	106350

Table 3 (continued)

Setup no. (<i>i</i>)	Setup cost (S_i)	Optimal lot size DT_i^*	$F(i)$
13	135.75	316.4	106385
12	139.29	318.1	106424
11	143.25	320.0	106465
10	147.71	322.0	106509
9	152.81	324.2	106557
8	158.72	326.5	106609
7	165.69	329.1	106666
6	174.12	331.9	106730
5	184.64	335.1	106803
4	198.40	338.6	106887
3	217.65	342.7	106987
2	248.00	347.6	107114
1	310.00	353.8	107299

^a Setup costs and optimal lot sizes do not change after the 64th setup.

^b Value of the optimal policy is given by $F(1)$.

materials ($DP/r = \text{npv of material costs}$). Hence the npv of the costs related to lot sizing in the optimal policy is $\$7299(F(1) - (DP/r))$. Netting out variable material costs in our analysis is exactly analogous to the classical economic order quantity analysis in which material costs are netted out, since these costs are incurred irrespective of the type of lot sizing decisions. This is also true of studies in heuristic lot sizing in MRP systems where direct material costs are excluded from consideration.

Now reconsider the above problem data, except that now the minimum achievable setup cost (S_{\min}) is \$31 – a reduction to 10% of the initial setup cost. It can be verified that 1279 setups take place before the setup cost levels off at this value. Using the optimal procedure described in Appendix A, the cost of the optimal policy ($F(1)$) is \$107244. Hence the inventory related costs in the optimal policy is \$7244. Suppose that the scheduler ignored the learning effects, and simply set all lot sizes equal to the optimum lot size corresponding to the minimum setup cost (DT_{\min}^*). It can be verified that the inventory related costs for this policy is \$7949 or 9.7% above the optimum. This example illustrates the benefits of incorporating learning effects in lot sizing decisions.

Preceding numerical illustrations used the well known power function from learning curves literature to characterize setup cost reductions. However,

because the optimal procedure does not make any such restrictive assumption about setup cost reduction patterns, the methodology developed in this paper can be used for any arbitrary setup cost trends.

3.2. Characterization of optimal lot sizes

In our numerical example (see Table 2), as setup costs decrease, so do optimal lot sizes. Detailed proof is provided in Appendix B. Though this result is intuitively obvious, earlier researchers concluded that the effect of investment in setup cost reduction (such as worker training and or process enhancements) is to reduce the lot sizes, but all lots should be kept equal to each other (Porteus, 1985a, 1986; Billington, 1987; Karwan et al., 1988; Replogle, 1988; Chand, 1989; Cheng, 1991). The divergence in conclusions is due to either (i) implicit assumption about immediate and all at-one-time reduction in setup costs (Porteus, 1985a, 1986; Billington, 1987), and/or (ii) using average cost analysis instead of the conceptually rigorous npv method (Karwan et al., 1988; Replogle, 1988; Chand, 1989; Cheng, 1991). *Our methodology for determining optimal lot sizes eliminates approximations due to both these factors.*

3.3. Evaluation of investments in learning and / or minimum setup cost reduction

Porteus (1985a, 1986) and Billington (1987) developed methods to determine optimal investment when the setup cost can be expressed as a special function (continuous) of the investment in setup cost reduction, and implicitly assumed that the benefits of the investment are realized by an all-at-one-time reduction (permanent) in setup costs. Our methodology is more general in the sense that it can be used to evaluate the investments when additional reductions in setups continue to occur at later times, as illustrated below.

Consider a product with $D = 8074$ units per period, $P = \$10$ per unit, $r = 5\%$ per period, $h = 0.4875$ per unit per period, and initial setup cost $(S_1) = \$400$. Assume that the current learning rate is 95% – i.e., the cost for the last setup reduces to 95% of the median setup cost, until technological limits preclude further cost reductions. Also, the current equipment configuration results in a minimum setup

cost (S_{\min}) of \$294.50. However, studies indicate that an additional investment of \$20000 in worker training and equipment reconfiguration will immediately reduce the setup cost to \$310. Further, these efforts will also improve the worker learning rate to 55%, and reduce the minimum achievable setup cost (S_{\min}) to \$124. We want to determine if the investment is economically justified.

Using the optimal procedure described in Appendix A, npv of the optimal policy for the current system configuration (exclusive of material costs, $F(1) - DP/r$) is \$44691. Similarly, with the new parameters based on the proposed investment, npv of the optimal policy (exclusive of material costs) is \$28438. Hence the resulting cost saving ($44691 - 28438 = \$16253$) is less than the required marginal investment. Unless there are additional nonquantifiable benefits, the investment in training and equipment reconfiguration is not economically justified.

4. Heuristic policies

Computation of optimal lot sizes requires cost information on each future setup cost. These costs may be difficult to forecast. Also, the optimal policy is based on dynamic programming, a procedure difficult to comprehend when compared to most heuristic lot sizing procedures. Hence we now propose and evaluate two heuristic policies which are intuitively appealing, easy to understand, and require little information regarding future setup costs. These are (i) the current setup cost lot sizing policy, and (ii) the minimum setup cost lot sizing policy.

4.1. Current setup cost lot sizing policy (curs)

Under this policy, the lot size at each setup is computed assuming that all future setup costs will be equal to the current setup cost. This policy results in decreasing lot sizes until no further reduction in setup cost occurs. It disregards any beneficial effects of future setup cost reductions on the current lot size. Since this policy only uses information about the current setup cost, we call it “current setup cost lot sizing policy” (CURS). The lot size for the i th setup will be equal to $D\tilde{T}_i$, where \tilde{T}_i equals the optimum replenishment interval if all future setup costs are

Table 4
Description of CURS policy

Step 1:	Using Eq. (9), determine the lot size for each setup (DT_i), $i = 1, 2, \dots, N$.
Step 2:	Determine $G_{\text{CURS}}(N)$, the npv of all future costs discounted to the time at which the N th setup occurs: $G_{\text{CURS}}(N) = \left(S_{\min} + DP\check{T}_N + \frac{Dh}{r^2} (r\check{T}_N - 1 + e^{-r\check{T}_N}) \right) (1 - e^{-r\check{T}_N})^{-1}.$
Step 3:	Do $i = N - 1, 1$, step -1 $G_{\text{CURS}}(i) = S_i + DP\check{T}_i + \frac{Dh}{r^2} (r\check{T}_i - 1 + e^{-r\check{T}_i}) + e^{-r\check{T}_i} G_{\text{CURS}}(i + 1).$

equal to S_i . The exact value of \check{T}_i can be calculated using the following formula (Trippi and Levin, 1974; Porteus, 1985b; Rachamadugu, 1988):

$$\frac{S_i r^2}{D(h + Pr)} = e^{r\check{T}_i} - 1 - r\check{T}_i. \quad (9)$$

Porteus (1985b) suggested an effective iterative procedure for finding the value of \check{T}_i , using the lot size provided by the classical square root formula as the initial upper bound. The procedure for determining lot sizes and the npv of the CURS policy is summarized in Table 4.

Step 1 in Table 4 determines the optimal lot size for each setup. The net present value of the CURS policy is found in Steps 2 and 3. In Step 2, we determine $G_{\text{CURS}}(N)$, the npv of all costs discounted to the time at which the minimum setup cost is realized for the first time. Step 3 discounts recursively costs of all earlier setups to the start of the first setup, and gives the npv of the CURS policy.

It is interesting to note that the CURS policy has the same intuitive appeal as the Part Period Balancing (PPB) heuristic. PPB and CURS set the current lot size disregarding all future setup cost information. PPB is well known in the Materials Requirements Planning literature (Zoller and Robrade, 1988). Both these procedures set the current lot size so that the setup cost for any lot is approximately equal to the inventory carrying costs incurred during its depletion. When setup costs are decreasing, lot sizes under CURS will also be decreasing. However, CURS overestimates lot sizes. Details are given in Appendix C.

Computational results reported later in the paper show that CURS performs very well over a broad range of scenarios. However, we are also able to provide some insights into its worst case behavior.

Though worst case analysis is generally based on pathological cases, it provides bounds on the performance of a heuristic. Also, it can provide insights into when it is inappropriate to use a heuristic policy, and it can be a guide in developing improved heuristics. Earlier studies by Axsater (1982) and Rachamadugu (1994) showed that policies very similar to CURS can result in twice the optimal cost in the worst case. Though their results were based on average cost analysis for finite horizon problems, we conjecture that the same result is also valid for our case. Appendix C provides an instance of the problem in which CURS results in twice the optimal cost. It also provides insights into when CURS may not perform well; in particular, CURS performs poorly in rapid learning environments (low values of ϕ). Our conclusion is also corroborated by the computational studies reported in Section 5.

4.2. Minimum setup cost lot sizing policy (MINS)

Under this policy, we disregard learning effects and use the economic lot size corresponding to the estimated minimum setup cost (S_{\min}). This lot size is used for all setups from the beginning. Hence all lot sizes are set equal to DT_{\min}^* , where T_{\min}^* is given by

$$\frac{S_{\min} r^2}{D(h + Pr)} = e^{rT_{\min}^*} - 1 - rT_{\min}^*. \quad (10)$$

As discussed in Section 4.1, an excellent approximation to DT_{\min}^* is provided by the classical square root formula. Its exact value can be determined by using the procedure suggested by Porteus (1985b).

Table 5 provides a summary of MINS policy. In Step 1, we determine the lot size corresponding to the minimum setup cost. This lot size is used at all setups from the beginning. Steps 2 and 3 are used to

Table 5
Description of MINS policy

Step 1:	Using (10), determine DT_{min}^* , the lot size for all setups: $\frac{S_{min}r^2}{D(h + Pr)} = e^{rT_{min}^*} - 1 - rT_{min}^*$
Step 2:	Determine $G_{MINS}(N)$, the npv of all future costs discounted to the time at which the N th setup occurs: $G_{MINS}(N) = \left(S_{min} + DPT_{min}^* + \frac{Dh}{r^2} (rT_{min}^* - 1 + e^{-rT_{min}^*}) \right) (1 - e^{-rT_{min}^*}) - 1.$
Step 3:	Do $i = N - 1, 1$, step - 1 $G_{MINS}(i) = S_i + DPT_{min}^* + \frac{Dh}{r^2} (rT_{min}^* - 1 + e^{-rT_{min}^*}) + e^{-rT_{min}^*} G_{CURS}(i + 1).$

evaluate the npv of the MINS policy. Their description is similar to their counterparts in Table 4 for the CURS policy.

All lot sizes in the MINS policy are equal. This feature is useful in developing cyclic schedules. Also, practitioners may favor the stability of constant lot sizes. When the learning is rapid (i.e., ϕ is low) and/or technological constraints preclude significant reductions in setup costs, optimal lot sizes quickly decrease to DT_{min}^* . Hence MINS may perform well in those scenarios. Further, it consistently underestimates the optimal lot sizes until setup costs stabilize. Though this makes the policy suboptimal, small lots provide quick feedback on quality. This feature can be beneficial, particularly in the initial stages of new product introduction.

MINS policy has a few disadvantages. Though it does not require forecasts of future setup costs, it does require an estimate of S_{min} , which may be difficult to forecast. While our earlier arguments point out that the policy may perform well under some circumstances (such as rapid learning), we show in Appendix D that it has unbounded worst case performance – i.e., under some circumstances, the cost of using the policy can be arbitrarily large.

5. Computational experiments

We conducted computational experiments to gain a better understanding of how well our heuristics perform across a broad spectrum of scenarios, and also to determine when one is better than the other. For our computational experiments, we chose the demand, price, setup cost and inventory holding cost

parameters so that the initial replenishment interval (using the classical square root formula) would be 2, 5.4 or 14.5 weeks. These parameter settings are consistent with earlier research in lot sizing (Wemerlov and Whybark, 1984; Blackburn and Millen, 1980; Baker, 1977). The initial setup cost was set at \$310. The price of the product was set at \$10 per unit. The discount rate was set at two levels, 5% and 20% per year, to represent low and high capital costs. Since we are using discounted cashflow methods for lot sizing (Hadley, 1964; Rachamadugu, 1988), we chose the parameters so as to control for the relative ratio of nonfinancial holding charges to the total inventory charges ($h/(h + Pr)$). h represents the noncapital related holding charges such as material handling and warehousing costs. P times r represents the holding charges attributable to the capital costs associated with carrying the product in inventory. The ratio ($h/(h + Pr)$) represents the extent to which cash flows arising from items such as material handling, warehousing, etc., influence the total inventory carrying costs. Controlling this ratio helps to see if that component of noncapital holding charges has any significant effect on the performance of the procedures. Since capital related costs tend to dominate inventory holding costs, we chose the parameters so that the ratio ($h/(h + Pr)$) was set at 0.25. Preliminary investigations with this ratio as large as 0.5 showed that it did not have significant influence on the performance of the CURS and MINS heuristics. Hence we did not vary the ($h/(h + Pr)$) ratio in our computational study.

The benefits of continuous improvement efforts, worker learning and incremental process improvements on setup costs are characterized by two pa-

parameters in our study – the learning rate (ϕ), and the setup cost reduction factor (R). Eq. (5) is a typical representation of the learning phenomenon (Argote and Epple, 1990). Earlier research by Baloff (1966, 1971) and Conway and Schultz (1959) suggests that levelling in learning takes place in machine intensive situations. We model this effect with the parameter R , which is defined as follows:

$$R = \frac{S_{\min}}{S_1} \quad (11)$$

$R = 1$ implies that the setup costs are not susceptible to any reduction at all. When $R = 0$, there is scope for entirely eliminating setup costs. However, how fast these reductions occur depends on the learning rate. The setup costs in our experiment are as follows:

$$S_i = \max\{S_1 i^{-b}, S_{\min}\} = S_1 \max\{i^{-b}, R\}, \quad (12)$$

where b is an index representing the learning rate. Eq. (12) implies that the setup costs cannot be less than S_{\min} . Learning effects reduce the setup cost until it reaches S_{\min} . Technological constraints limit any further reductions. Eq. (12) is a generalization of the classical learning model. Setting $S_{\min} = 0$ yields the classical learning model.

Given a learning rate (ϕ), b can be determined using (6), $\phi = 2^{-b}$. Note that *high values of learning rate (ϕ) denote slow reductions in setup costs (slower learning or none at all)*. For example, $\phi = 1$ (highest possible value) implies that there is no learning at all, and S_{\min} will never be achieved. $\phi = 0$ (lowest possible value) implies that the minimum setup cost is achieved in the second setup! This anomalous and counterintuitive notation for the learning rate (ϕ) has been used extensively in learning theory and its applications, and we use the same notation to be consistent with earlier research.

In our computational experiments, R was varied from 0.1 to 0.9 in steps of 0.1 so that we could analyze the performance of the CURS and MINS heuristics for a broad range of setup cost reductions. We also varied the learning rate (ϕ) from 0.5 to 0.95 in our experiments. This range is representative of the field studies cited in Argote and Epple (1990).

We tested a total of 480 scenarios ($\{8 \text{ settings for } R\} \times \{10 \text{ settings for } \phi\} \times \{2 \text{ settings for } r\} \times \{3 \text{ settings for the initial reorder interval}\}$). As discussed earlier, h and D were varied so as to hold the ratio of noncapital related holding charges to total holding charges ($h/(h + Pr)$) constant at 0.25, while achieving necessary initial reorder interval. Procedures were coded in Fortran77, and run on an IBM 9021-720 mainframe computer.

5.1. Analysis

Performance of the heuristic policies relative to the optimum (in percent) was evaluated using the following criterion:

Excess cost(%)

$$\begin{aligned} & \text{NPV of the heuristic policy} - \frac{DP}{r} \\ &= \frac{\text{NPV of the heuristic policy} - \frac{DP}{r}}{\text{NPV of the optimal policy} - \frac{DP}{r}} \times 100. \end{aligned} \quad (13)$$

In the above expression, DP/r represents the npv of material costs (i.e., npv obtained by discounting the infinite stream of annual material cost at the discount rate). Note that the material price (P) is constant in our analysis. Hence the effect of material costs should be excluded from the analysis, since those costs are invariant with respect to lot sizing decisions. For example, it is common practice to exclude material costs from consideration in evaluating the heuristics in Material Requirements Planning. Similarly, in performing sensitivity analysis of lot sizing decisions, and also when evaluating the effect of nonoptimal order quantities in the classical lot sizing analysis, material costs are excluded (see (Fogarty et al., 1991)). Hence DP/r is netted out from both the optimal and heuristic policies in (13) to evaluate the effectiveness of the heuristics.

Table 6 shows the relative performance (excess cost in % over optimum) of the MINS and CURS heuristics. Each cell represents the average value for six problems for a specific combination of R and ϕ , and different values of r and initial reorder interval. A bold figure in a cell shows the average performance of CURS policy. Bold figures in parentheses show the performance of the MINS poli . Entries

R	0.1	0.2	0.3	0.4	0.5	0.6
ϕ						
0.5	1.7 (< 0.1)	0.7 (< 0.1)	0.3 (< 0.1)	0.2 (< 0.1)	0.1 (< 0.1)	< 0.1 (< 0.1)
0.55	1.9 (< 0.1)	0.7 (< 0.1)	0.3 (< 0.1)	0.2 (< 0.1)	0.1 (< 0.1)	< 0.1 (< 0.1)
0.6	2.1 (< 0.1)	0.8 (< 0.1)	0.4 (< 0.1)	0.2 (< 0.1)	0.1 (< 0.1)	< 0.1 (< 0.1)
0.65	2.5 (0.1)	0.9 (< 0.1)	0.4 (< 0.1)	0.2 (< 0.1)	0.1 (< 0.1)	< 0.1 (< 0.1)
0.7	2.8 (0.2)	1.0 (< 0.1)	0.5 (< 0.1)	0.2 (< 0.1)	0.1 (< 0.1)	< 0.1 (< 0.1)
0.75	2.8 (0.8)	1.2 (0.1)	0.6 (< 0.1)	0.3 (< 0.1)	0.1 (< 0.1)	< 0.1 (< 0.1)
0.8	2.2 (3.2)	1.3 (0.3)	0.7 (0.1)	0.3 (< 0.1)	0.2 (< 0.1)	0.1 (< 0.1)
0.85	1.1 (10.6)	1.0 (1.7)	0.7 (0.1)	0.4 (< 0.1)	0.2 (< 0.1)	0.1 (< 0.1)
0.9	*	0.4 (6.9)	0.4 (1.9)	0.4 (0.4)	0.2 (0.1)	0.1 (< 0.1)
0.95	*	*	*	*	< 0.1 (1.1)	0.1 (0.2)

* indicates that no computational tests were performed due to excessive computational requirements. Shaded cells indicate situations in which CURS performs better than the average cost policy.

marked with * indicate the instances when optimal values could not be found due to excessive computational time requirements (i.e., number of setups required to reach S_{min} exceeded 160,000). This occurred in instances for which the learning rate was high (slow learning), and also R was low. This is not surprising, because these parameter combinations imply that too many setups are required to reach S_{min} . Results for $R = 0.7$ and 0.8 were deleted since both heuristics performed equally well ($\leq 0.1\%$ above optimum) in all cases.

The results in Table 6 indicate that the CURS policy performs well. It provides nearly optimal solutions in an average sense, with its cost within 2.8% above the optimum for any combination of learning rate (ϕ) and setup cost reduction factor (R) considered in our experiments. Among 450 problems with known optima, its worst performance was 5.7% above the optimum. In other cases (30 problems – cells with * in Table 6), trends in the table indicate that its performance could only be better. It is interesting to note that the CURS policy performed so well without the benefit of any information about future setup costs.

Table 6 also shows that MINS also performs well for most parameter combinations, except when the learning rate is high and R is low. For example, when the learning rate is 0.85 and $R = 0.1$, the average excess cost resulting from the MINS policy was 10.6% above the optimum. Further, detailed analysis of all 450 optimally solved cases showed that the MINS policy cost could be as high as 19.5% above the optimum. Trends in Table 6 indicate that the performance of MINS policy deteriorates as the learning rate increases (slow learning). This is also evident from the worst case analyzed in Appendix D. The MINS policy sets lot sizes too small under these circumstances, resulting in excessive setups in early stages (which leads to high npv). These computational results suggest that MINS is better suited for fast learning environments (such as manual work environments) which are characterized by low learning rate (ϕ).

In our computational experiments, choosing the best heuristic for any problem resulted in excess costs being at worst 3.4% above the optimum. Choosing the best heuristic is not difficult either. In Table 6, the shaded cells indicate combinations of R

and ϕ for which CURS is a better choice than MINS. Note that they form an unbroken block. Hence at high values of ϕ and low values of R , CURS is a better choice. For all other situations MINS is a better choice. Note that if S_{\min} cannot be forecast, or if one is in doubt about which heuristic to choose, we recommend using CURS policy. The computational results in Table 6 show that it is quite robust. Further, worst case analyses of the heuristics also support our conclusion.

6. Conclusions

We addressed the problem of determining optimal lot sizes when reductions in setup costs persist due to emphasis on continuous improvement, worker learning and incremental process improvements. We developed a procedure for determining optimal lot sizes. This methodology is both general and practical in the sense that it assumes no special functional form for successive setup costs. In addition to its primary use as a tool for determining optimal lot sizes, it is also useful in evaluating investments in hardware that lower the minimum achievable setup costs and/or investments which enhance the pace of worker learning.

Determination of optimal lot sizes requires information on how fast setup costs decrease and also on the minimum achievable setup cost. We suggested two heuristic procedures (CURS and MINS policies) which can be used when information about setup cost reduction trends is not available. The current setup cost lot sizing policy (CURS) is better suited for situations in which improvements in setup costs occur at a slow pace (high learning rate – such as highly automated environments). For other situations, the MINS policy is appropriate. In the absence of any information about future setup costs, we recommend the use of CURS policy due to its robustness and better worst case performance.

Also, our analysis has implications for the use of average cost analysis. Hadley (1964) showed in his computational experiments that average cost analysis is an adequate substitute for the conceptually rigorous net present value approach for lot sizing. This was further substantiated by Porteus (1985b) in his

analytical studies. Since the results of long run average cost analysis are not influenced by transient effects, using the average cost analysis in our situation would result in the conclusion that all lot sizes should be equal to the optimal lot size corresponding to the minimum setup cost (which is the MINS policy). However, our computational results have shown that such a policy can result in lot sizing costs that exceed the optimum by as much as 19.5% or more. This implies that the average cost analysis could be inadequate for nonstationary cost parameter situations, such as when a production process is used for long periods of time while the reductions in setup costs continue to occur over time due to emphasis on *kaizen* (continuous improvement – (Stephanou and Spiegl, 1992)) and worker learning. Such situations call for more accurate decision making tools based on npv analysis instead of approaches based on average costs.

Acknowledgements

We thank the anonymous referees, the associate editor and the editor for their constructive comments on earlier versions of the paper. We are also grateful to Dr. Sita Bhaskaran, Operating Sciences Department, General Motors Research Laboratories, Warren, Michigan for her comments and help with empirical data reported in the paper.

Appendix A. Dynamic programming procedure for optimal lot sizing

In this appendix we first characterize the optimal policy, and then describe how the characterization can be used to find the optimal lot sizes.

Optimal policy

Reconsider the costs incurred in the i th replenishment interval. These costs, discounted to the start of the i th replenishment cycle, are given by Eq. (2) which is stated once again below for the ease of exposition:

$$V(T_i) = S_i + DPT_i + \int_0^{T_i} Dh(T_i - t)e^{-rt} dt. \quad (\text{A.1})$$

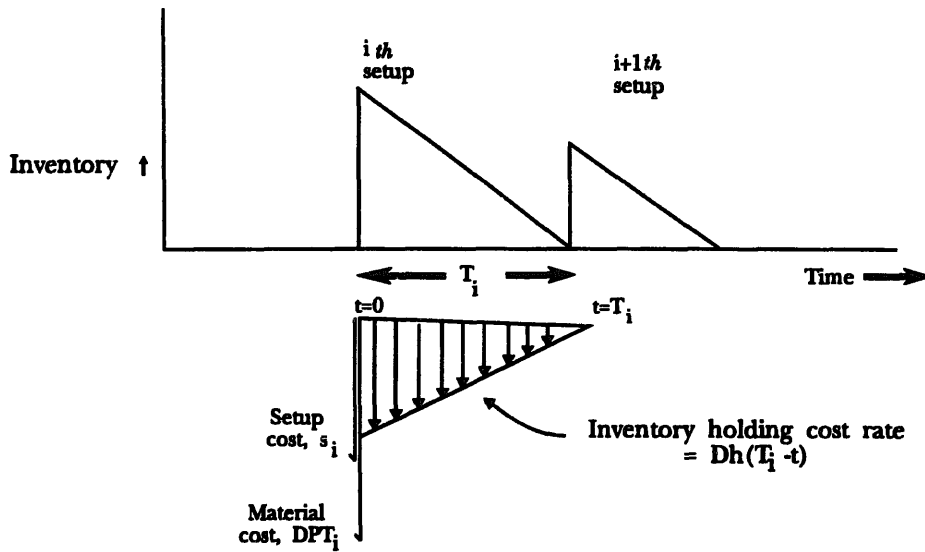


Fig. 1. Costs incurred in the *i*th setup.

The first term in (A.1) represents the setup cost for the *i*th cycle. The second term represents the variable costs such as materials incurred in the cycle. The third term represents the npv of inventory carrying costs (noncapital charges such as material handling and warehousing) discounted to the time at which the *i*th setup takes place (note that the interest expense for the capital tied up in inventory is implicitly recognized through discounting). We need to determine optimal replenishment intervals. The tem-

poral occurrence of these costs is shown in Fig. 1. Integrating (A.1), we have

$$V(T_i) = S_i + DPT_i + \frac{Dh}{r^2} (rT_i - 1 + e^{-rT_i}). \quad (A.2)$$

Let $F(i)$ represent the npv of costs incurred in all replenishment cycles $i, i + 1, i + 2, \dots$ discounted to the start of the *i*th cycle. Note that *i* represents the setup index, and not time. Hence, by definition, the

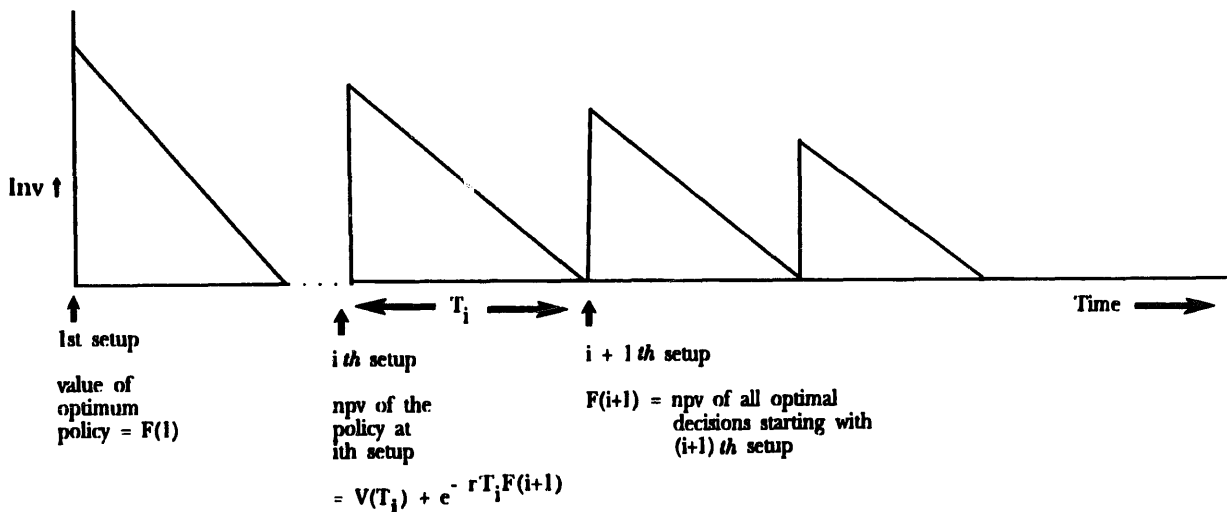


Fig. 2. Structure of the optimal policy.

value of the optimal policy is given by $F(1)$. By the principle of optimality (Bellman and Dreyfus, 1962),

$$F(i) = \min(V(T_i) + e^{-rT_i}F(i + 1)), \tag{A.3}$$

where $0 \leq T_i$. Fig. 2 explains the recursion graphically. The first term in (A.3) is the npv of all costs incurred in the i th replenishment cycle. The second term is the npv of all optimal future lot sizing decisions discounted to the start of the i th cycle. Substituting (A.2) in (A.3),

$$F(i) = \min\left(S_i + DPT_i + \frac{Dh}{r^2}(rT_i - 1 + e^{-rT_i}) + e^{-rT_i}F(i + 1)\right), \tag{A.4}$$

where $0 \leq T_i$. It can easily be verified that (A.4) is convex in T_i . Therefore, the optimum replenishment interval (T_i^*) is obtained by setting the derivative of (A.4) equal to zero:

$$T_i^* = \frac{1}{r} \ln\left(\frac{h}{h + Pr} + \frac{r^2}{D(h + Pr)}F(i + 1)\right). \tag{A.5}$$

It is also interesting to note that the optimal lot sizes do not depend on the initial setup cost. This is evident from (A.5) by setting $i = 1$. A similar result was derived by Mekler (1993) for dynamic lot sizing with setup reduction. Optimal replenishment intervals (or lot sizes) depend on all parameters, except the initial setup cost. This is not surprising because the initial setup cost has to be incurred at time zero, and hence constitutes a sunk cost from the decision making point of view. Therefore, the optimal lot sizes depend only on how fast setup costs decrease, and also the minimum setup cost (S_{\min}) that can be realized. Next, we describe how the above optimal policy characterization can be used to determine both the optimal lot sizes and the associated optimal cost.

Procedure for determining optimal lot sizes and cost

Note that the setup costs for the N th and subsequent setups equal S_{\min} . Hence each of these lot sizes is equal to $D\check{T}_N$. Therefore, $F(N)$ is given by

$$F(N) = \left(S_{\min} + DPT_N + \frac{Dh}{r^2}(r\check{T}_N - 1 + e^{-r\check{T}_N})\right) \times (1 - e^{-r\check{T}_N})^{-1}. \tag{A.6}$$

The terms in the first set of large brackets in (A.6) represent the npv of costs for each of the replenishment cycles in all setups indexed N and beyond, discounted to the start time of each respective cycle. The terms in the second set of large brackets discount this infinite stream of costs to the time at which the N th setup occurs. \check{T}_N is found by the following:

$$\frac{S_{\min}r^2}{D(h + Pr)} = e^{r\check{T}_N} - 1 - r\check{T}_N. \tag{A.7}$$

Eq. (A.7) is the discounted cash flow formulation of the classical economic order quantity model. Details of deriving (A.6) and (A.7) are given in (Trippi and Levin, 1974; Rachamadugu, 1988). Eq. (A.7) can be solved very efficiently for \check{T}_N using the iterative procedure suggested by Porteus (1985a,b), starting with the classical square root formula (also known as the economic order quantity) as the initial solution. We find all optimal replenishment intervals by working backwards with Eqs. (A.5) and (A.4) iteratively for each setup. We terminate with finding $F(1)$, the npv of the optimal policy.

Appendix B. Characterization of optimal policies

Proposition B.1. *If the setup costs are nonincreasing, then the optimal lot sizes are also nonincreasing.*

Proof. It is intuitively clear that

$$F(1) \geq F(2) \geq F(3) \geq \dots F(i) \geq F(i + 1) \dots, \tag{B.1}$$

However, we provide a rigorous proof below. Using (A.5) and rewriting (A.4),

$$F(i) = \sum_{j=i}^{\infty} e^{-rZ_j} \left(S_j + DPT_j^* + \frac{Dh}{r^2}(rT_j^* - 1 + e^{-rT_j^*}) \right), \tag{B.2}$$

where

$$Z_j = \begin{cases} 0 & \text{if } j = i \\ \sum_{l=i}^j T_l^* & \text{if } j > i \end{cases} \tag{B.3}$$

and $S_j = S_{j+1} + \delta_j, \delta_j \geq 0$.

Using (B.3) in (B.2),

$$F(i) = \sum_{j=i}^{\infty} e^{-rZ_j} (S_{j+1} + DPT_j^* + \frac{Dh}{r^2} (rT_j^* - 1 + e^{-rT_j^*})) + \sum_{j=i}^{\infty} e^{-rZ_j} \delta_j. \tag{B.4}$$

Note that the first summation on the right-hand side of the expression in (B.4) represents the net present value of a feasible policy for all future setups starting with the (i + 1)th set up. Hence it is $\geq F(i + 1)$. Clearly, the second summation on the right-hand side of Eq. (B.4) is non-negative. Therefore, (B.4) can be rewritten as

$$F(i) \geq F(i + 1) + \sum_{j=i}^{\infty} e^{-rZ_j} \delta_j \geq F(i + 1). \tag{B.5}$$

From (A.5), it is clear that T_i^* is monotonic in $F(i)$. Using (B.5) in (A.5),

$$S_i \geq S_{i+1} \Rightarrow F(i) \geq F(i + 1) \Rightarrow T_{i+1}^* \geq T_{i+2}^*. \tag{B.6}$$

Recursive application of the above completes the proof. \square

Appendix C. Characteristics of curs policy

In this appendix, we discuss the characteristics of CURS policy. First, we compare the lot sizes resulting from CURS policy with the optimal lot sizes. Next, we comment on its worst case performance.

Lot sizes

Proposition C.1. *In the CURS policy, the lot size for any particular setup is not less than the optimal lot size corresponding to that setup.*

Proof. By definition, the lot size for the i th setup in the CURS policy equals $D\check{T}_i$. Let $J(i)$ represent the net present value of the optimal policy for the problem in which all future setup costs are equal to S_i . Using (6),

$$\check{T}_i = \frac{1}{r} \ln \left(\frac{h}{h + Pr} + \frac{r^2}{D(h + Pr)} J(i + 1) \right). \tag{C.1}$$

But $J(i + 1) = J(i) \geq F(i + 1)$. Hence $\check{T}_i \geq T_i^*$. This completes the proof. \square

Worst case performance

In this appendix, we show that the current setup cost lot sizing policy (CURS) can result in costs twice as high as the optimal policy. The performance measure (see Eq. (13) in Section 5.1) used to evaluate the heuristic policies is repeated below:

$$\frac{\text{NPV of the heuristic policy} - \frac{DP}{r}}{\text{NPV of the optimal policy} - \frac{DP}{r}} \times 100.$$

The term DP/r represents the minimum net present value of the material costs. It is netted out from both the optimal and heuristic policies to make the performance measure comparable to the classical studies on lot sizing policies.

Now we present a scenario in which the worst case performance of the current setup cost lot sizing policy (CURS) is twice the cost of the optimal policy. Consider a situation in which the initial setup cost is S_1 and all subsequent setup costs are zero. Note that contrived as it may seem, we identify a situation here in which the performance of CURS can be at its worst. The cost of the CURS policy in this case is given by

$$\left\{ S_1 + DPT_1 + \frac{Dh}{r^2} (r\check{T}_1 - 1 + e^{-r\check{T}_1}) \right\} + e^{-r\check{T}_1} \frac{DP}{r}. \tag{C.2}$$

The first term in braces in (B.1) is the setup cost for the first setup. The second term accounts for the variable material costs incurred in the first setup. The third term shows the net present value of the non-financial holding costs during the first setup, discounted to time zero. The term outside the braces accounts for the present value of all future costs. Clearly, the optimal policy is to set up the machine once, and then treat the process as a continuous process. The net present value of the optimal policy is given by $S_1 + (DP/r)$. Hence the performance of

the CURS policy is given by (using (C.2) in Eq. (13))

$$S_1 + DP\check{T}_1 + \frac{Dh\check{T}_1}{r} - (1 - e^{-r\check{T}_1}) \frac{D(h + Pr)}{r^2} \quad (C.3)$$

Earlier, Rachamadugu (1988) showed that

$$\frac{e^{r\check{T}_1} - 1 - r\check{T}_1}{S_1} = \frac{r^2}{D(h + Pr)} \quad (C.4)$$

Using (C.4) in (C.3), and with a little algebraic manipulation, the performance measure for CURS reduces to the following expression:

$$\frac{e^{r\check{T}_1} + e^{-r\check{T}_1} - 2}{e^{r\check{T}_1} - 1 - r\check{T}_1} \quad (C.5)$$

The maximum of the above expression occurs as $r\check{T}_1 \rightarrow 0$, and equals 2. Hence, CURS can be twice as costly as the optimal policy. Computational results in this paper, as well as earlier worst case results for variants of our problem (Axsater, 1982; Rachamadugu, 1988) lend strong support to the conjecture that at worst CURS is likely to cost twice the optimal policy. However, as our computational experiments show, its performance in a typical situation is far superior to its worst case performance.

Appendix D. Worst case analysis of MINS policy

We present an instance of the minimum setup cost lot sizing policy (MINS) which has unbounded worst case performance. Let S_i equal S for the first n setups, and then the process becomes a continuous one, i.e., $S_i = 0 \forall i > n$. In this case the MINS policy suggests that the lot sizes should be infinitesimal – i.e., repeatedly perform all the setups at the start of the process. An upper bound on the optimal policy is given by assuming that setup costs do not decrease at all. Clearly, the worst case performance of the MINS policy can be no better than its performance relative to the upper bound suggested here. Using (13), the

worst case performance of the MINS policy cannot be less than the following expression:

$$\frac{\left\{ nS + \frac{DP}{r} \right\} - \frac{DP}{r}}{\left\{ \frac{S + DP\check{T} + \frac{Dh}{r^2} (r\check{T} - 1 + e^{-r\check{T}})}{1 - e^{-r\check{T}}} \right\} - \frac{DP}{r}} \quad (D.1)$$

where \check{T} is the optimal replenishment interval for the infinite horizon problem in which the setup costs remain invariant at S . With algebraic manipulation, (D.1) can be rewritten as

$$S + \frac{d(h + Pr)}{r^2} (r\check{T} - 1 + e^{-r\check{T}}) \quad (D.2)$$

Rachamadugu (1988) has shown that

$$e^{r\check{T}} - 1 - r\check{T} = \frac{Sr^2}{D(h + Pr)} \quad (D.3)$$

Using (D.3), (D.2) can be rewritten as

$$n \left\{ \frac{1 - e^{-r\check{T}}}{r\check{T} - 1 + e^{-r\check{T}}} \right\} \quad (D.4)$$

As $r\check{T} \rightarrow \infty$, (D.4) $\rightarrow n$. As $n \rightarrow \infty$, performance of the MINS policy becomes arbitrarily bad.

References

Adler, G.I. and R. Nanda (1974). "The effects of learning on optimal lot size determination", *AIIE Transactions*, vol. 6, no. 1, pp. 14–27.
 Argote, L. and D. Epple (1990). "Learning curves in manufacturing", *Science*, vol. 247, pp. 920–924.
 Axsater, S. (1982). "Worst case performance for lot sizing heuristics", *European Journal of Operational Research*, vol. 9, no. 4, pp. 339–343.
 Baker, K.R. (1977). "An experimental study of the effectiveness of rolling schedules in production planning", *Decision Sciences*, vol. 8, pp. 20–27.
 Baloff, N. (1966). "Startups in machine-intensive production

- systems", *Journal of Industrial Engineering*, vol. 17, no. 1, pp. 25–32.
- Baloff, N. (1971). "Extension of the learning curve – Some empirical results", *Operations Research Quarterly*, vol. 22, no. 4, pp. 329–340.
- Bellman, R.E. and S.E. Dreyfus (1962). *Applied Dynamic Programming*, 1st Edition, Princeton University Press, Princeton.
- Billington, P.J. (1987). "The classic economic production quantity model with setup cost as a function of capital expenditure", *Decision Sciences*, vol. 18, no. 1, pp. 25–42.
- Blackburn, J.D. and R.A. Millen (1980). "Heuristic lot size performance in a rolling schedule environment", *Decision Sciences*, vol. 11, no. 4, pp. 691–701.
- Chand, S. (1989). "Lot sizes and setup frequency with learning in setups and process quality", *European Journal of Operational Research*, vol. 42, pp. 190–202.
- Chand, S. and S. Sethi (1990). "A dynamic lot sizing model with learning in setups", *Operations Research*, vol. 38, no. 4, pp. 644–655.
- Cheng, T.C.E. (1991). "An EOQ model with learning effect on setups", *Production and Inventory Management Journal*, vol. 32, no. 1, pp. 83–84.
- Conway, R. and A. Schultz (1959). "The manufacturing progress function", *Journal of Industrial Engineering*, vol. 10, no. 1, pp. 39–53.
- Fine, C.H. and E.L. Porteus (1989). "Dynamic process improvement", *Operations Research*, vol. 37, no. 4, pp. 580–591.
- Fisk, J.C. and D.P. Ballou (1982). "Production lot sizing under a learning effect", *IIE Transactions*, vol. 14, no. 4, pp. 257–264.
- Fogarty, D.W., J.H. Blackstone and T.R. Hoffmann (1991). *Production and Inventory Management*, 2nd Edition, South-Western Publishing Company, Cincinnati.
- Hadley, G. (1964). "A comparison of order quantities computed using the average annual cost and the discounted cost", *Management Science*, vol. 10, pp. 472–476.
- Hollander, S. (1965). *The Source of Increased Efficiency: The Study of Dupont Rayon Plants*, MIT Press, Cambridge, MA.
- Karwan, K.R., J.B. Mazzola and R.C. Morey (1988). "Product lot sizing under setup and worker learning", *Naval Research Logistics*, vol. 35, pp. 159–175.
- Keachie, E. and R. Fontana (1966). "Effects of learning on optimal lot size", *Management Science*, vol. 13, no. 2, pp. 102–108.
- Klasterin, T.D. and K. Moinzadeh (1989). "Production lot-sizing under learning effects: An efficient solution technique", *IIE Transactions*, vol. 21, no. 1, pp. 1–10.
- McElroy, J. (1986). "The best damn die change team in America", *Automotive Industries*, November, pp. 46–51.
- McElroy, J. (1987). "Die change challenge! And you thought they were good last year", *Automotive Industries*, November, pp. 32–38.
- McElroy, J. (1988). "Die change challenge '88", *Automotive Industries*, November, pp. 52–63.
- McElroy, J. (1989). "Die change 1989", *Automotive Industries*, November, pp. 25–35.
- McElroy, J. (1990). "1990 die change challenge", *Automotive Industries*, November, pp. 43–52.
- McElroy, J. (1991). "Sixth annual die change challenge international", *Automotive Industries*, November, pp. 26–39.
- Mekler, V.A. (1993). "Setup cost reduction in the dynamic lot-size model", *Journal of Operations Management*, vol. 11, no. 1, pp. 35–43.
- Plossl, G.W. (1985) *Just-in-Time: A Special Roundtable*, George Plossl Educational Services Inc., Atlanta.
- Porteus, E.L. (1985a). "Investing in reduced setups in the EOQ model", *Management Science*, vol. 31, no. 8, pp. 988–1010.
- Porteus, E.L. (1985b). "Undiscounted approximations of discounted regenerative models", *Operations Research Letters*, vol. 3, no. 8, pp. 293–300.
- Porteus, E.L. (1986). "Investing in new parameter values in the discounted EOQ model", *Naval Research Logistics Quarterly*, vol. 33, no. 1, pp. 39–48.
- Rachamadugu, R. (1988). "Error bounds for EOQ", *Naval Research Logistics*, vol. 35, no. 5, pp. 419–425.
- Rachamadugu, R. (1994). "Performance analysis of a myopic lot size policy", *IIE Transactions*, vol. 26, no. 5, pp. 85–91.
- Replogle, S.H. (1988). "The strategic use of smaller lot sizes through a new EOQ model", *Production and Inventory Management Journal*, vol. 29, no. 2, pp. 41–44.
- Rosenberg, N. (1982). *Inside the Black Box: Technology and Economics*, Cambridge University Press, New York.
- Schonberger, R.J. (1982). *Japanese Manufacturing Techniques: Nine Hidden Lessons in Simplicity*, The Free Press, New York.
- Smunt, T.L. and T.E. Morton (1985). "The effect of learning on optimal lot sizes: Further developments on the single product case", *IIE Transactions*, vol. 17, no. 1, pp. 33–37.
- Stephanou, S.E. and F. Spiegl (1992). *The Manufacturing Challenge – From Concept to Production*, Van Nostrand Reinhold, New York, pp. 300–301.
- Suzaki, K. (1987). *The New Manufacturing Challenge – Techniques for Continuous Improvement*, Free Press, New York.
- Trippi, R.R. and D.E. Levin (1974). "A present value formulation of the classical EOQ problem", *Decision Sciences*, vol. 5, pp. 30–35.
- Wagner, H.M. and T.M. Whitin (1958). "Dynamic version of the economic lot size model", *Management Science*, vol. 5, pp. 89–96.
- Wemmerlov, U. and D.C. Whybark (1984). "Lot sizing under uncertainty in a rolling schedule environment", *International Journal of Production Research*, vol. 22, no. 3, pp. 467–484.
- Yelle, L.E. (1979). "The learning curve: Historical review and comprehensive survey", *Decision Sciences*, vol. 10, pp. 302–32.
- Zoller, K. and A. Robrade (1988). "Dynamic lot sizing techniques: Survey and comparison", *Journal of Operations Management*, vol. 7, no. 4, pp. 125–148.