Control
and Cybernetics
Vol. 16 (1987) No. 3–4

A new algorithm for the economic lot size problem

by

JOSÉ ANTÓNIO SOEIRO FERREIRA

Faculty of Engineering
University of Oporto
Oporto, Portugal

RENÉ VICTOR VALQUI VIDAL

IMSOR
The Technical University of Denmark
DK-2800 Lyngby, Denmark

In this paper a Simplified Matrix Algorithm (SMA) is presented. It is based on some simplifications to an already known Matrix Algorithm [1].

A broad class of problems is identified, and optimality criteria are established, for which the SMA operates as an optimal algorithm. Tests performed on cases from this class indicate the efficiency of this algorithm.

1. Introduction

In this paper we present the Simplified Matrix Algorithm (SMA). SMA is based on some technical refinements of an original algorithm, the Matrix Algorithm (MA), which is directly connected with the dual of a linear formulation of the lot size problem. These refinements are not only feasible – they are a consequence of some properties of the (MA), but also desirable. The proofs of the properties are included in Appendix 2.

A broad class of problems is identified for which the SMA always gives optimal solutions. Computational results are presented and they include a comparison of results with other known lot sizing techniques.

2. Problem description and the matrix algorithm

The lot size problem (LSP), deals with the situation of selecting a production planning policy $(x_1, x_2, \ldots, x_N)$, in discrete time, to satisfy a forecasted demand $(d_1, d_2, \ldots, d_N)$ so as to minimize the total cost of production, setup and holding inventory.

We consider the following **basic model** for the LSP, which is a generalized Wagner–Whitin model [3].

$$\min\left(\sum_{k=1}^{N} c_k \cdot x_k + S_k \cdot \delta(x_k) + h_k \cdot i_k\right)$$

subject to

$$i_k = i_{k-1} + x_k - d_k, \qquad k = 1, 2, \ldots, N$$
$$i_0 = i_N = 0$$
$$i_k,\ x_k \geqslant 0, \qquad k = 1, 2, \ldots, N$$

where

$$\delta(x_k) = \begin{cases} 1, & \text{if } x_k > 0 \\ 0, & \text{otherwise} \end{cases}$$

The following parameters (all nonnegative) are defined for period k:
h_k – the inventory holding cost, in \$/unit/period,
c_k – the ordering cost, in \$/unit/period,
S_k – the setup cost, in \$,
d_k – the demand, in units.
The decision variables are:
x_k – the amount to be produced (assumed available at the beginning of period k),
i_k – the inventory level at the end of period k.
The following parameters, for

$$i = 1, 2, \ldots, N,$$
$$j = 1, 2, \ldots, N;$$
$$C_{ij} = c_{ij} \cdot d_j, \qquad (i, j), \qquad i \leqslant j$$

where

$$c_{ij} = c_{ii} + \sum_{k=i}^{j-1} h_k, \qquad (i, j), \qquad i < j$$
$$c_{ii} = c_i, \ (i)$$

and decision variables z_{ij}, (i, j), $i \leqslant j$, and $y_i = \delta(\sum_{j=i}^{N} z_{ij})$, (i), are introduced. It can be shown (see Ferreira and Vidal [1]) that the basic model is equivalent to the following linear programming model:

$$\min\left(\sum_{j=1}^{N} \sum_{i=1}^{j} C_{ij} \cdot z_{ij} + \sum_{i=1}^{N} S_i \cdot y_i\right), \qquad C_{ij},\ S_i \geqslant 0$$

subject to

$$\sum_{i=1}^{j} z_{ij} \geqslant 1, \quad (j)$$

$$y_i - z_{ij} \geqslant 0, \quad (i, j), \quad i \leqslant j$$

$$z_{ij}, y_i \geqslant 0, \quad (i, j)$$

By studying dual model (see Apendix 1) of the linear programming formulation, the MA, an algorithm always giving an optimal solution to the LSP (even in the case of time-varying cost parameters) has been developed in [1]. The SMA is a technical refinement of the MA, and is based on some insights into the cost structure of the LSP. For this reason we present here the MA but suggest the reader to get familiar with an example included in [1]. The auxiliary variables S_i^j are defined in Appendix 1.

MA

1. Set $S_i^0 = S_i, \quad i = 1, \ldots, N$
2. Set $j = 1$
3. Set $\lambda_j = \min_i \{C_{ij} + S_i^{j-1}\}, \quad (i \leqslant j)$
4. Set $S_i^j = \begin{cases} S_i^{j-1}, & \text{if } C_{ij} \geqslant \lambda_j \\ S_i^{j-1} - (\lambda_j - C_{ij}), & \text{if } C_{ij} < \lambda_j \end{cases}, \quad (i \leqslant j)$
5. If $j = N$ go to point 6, otherwise set $j = j+1$ and go to 3.
6. Find one i, at which,

$$C_{ij} \leqslant \lambda_j$$

and

$$S_i^j = 0$$

7. Set $y_i = 1$
   $z_{ik} = 1, \quad k = i, \ldots, j$
8. Set $j = i-1$ and if $j > 0$ return to point 6.
9. Stop.

3. SMA – A new algorithm

Many real problems arise in which one or more of the parameters have special characteristics such as being constant for all periods. This is one of the reasons why, in general, the methods referred to in the literature only consider or solve these special problems. That encouraged us to try some simplifications of the MA. Of course, it could also be said that the modifications to be performed were suggested by observing that some calculations made while solving different problems by the MA seemed to be redundant.

Anticipating the main conclusions, we say that for the problems having particularities such as non-increasing ordering costs or non-decreasing setup costs, the modified algorithm, the SMA, will not compromise the essential feature of exact method, while significantly reducing the computational time. On the other hand, the algorithm thus obtained proved to be an efficient heuristic for the general LSP.

Let us then start with some cost structure properties of the MA. The proofs may be found in the Appendix 2.

P.1 – In a problem with non-increasing ordering costs ($c_i \geqslant c_{i+1}$, $i = 1, \dots, N-1$), $C_{ij} \geqslant C_{lj}$, $1 \leqslant i \leqslant l \leqslant j < N$, that is, the elements of the matrix $[C_{ij}]$ do not increase in the same column from up to down.

The example given below shows that problems not satisfying the hypothesis of P.1 may not respect this property.

P.2 – In a problem with non-increasing ordering costs ($c_i \geqslant c_{i+1}$, $i = 1, \dots, N-1$), if

$$S_{i^*}^k = 0, \quad 1 < i^* \leqslant k \leqslant N$$

then

$$\lambda_j = \min_{i^* \leqslant i \leqslant j} \{C_{ij} + S_i^{j-1}\}, \quad k < j \leqslant N.$$

One could be tempted to think that in case of finding $S_i^k = 0$ for $k \geqslant i$, while following the MA, it should be optimal to start production at period i. That may not be correct as the example shows. However, if in point 6 of the MA the search for one i, such that $C_{ij} \leqslant \lambda_i$ and $S_i^j = 0$, is made from the last to the first period, it turns out that finding $S_i^k = 0$, $k \geqslant i$, is sufficient to decide that production should start at period i. The next property enters in this consideration.

P.3 – In a problem with non-increasing ordering costs ($c_i \geqslant c_{i+1}$, $i = 1, \dots, N-1$) condition $C_{ij} \leqslant \lambda_j$ can be suppressed from point 6 of the MA if for each j ($1 \leqslant j \leqslant N$) we find the greater i ($1 \leqslant i \leqslant j$) such that $S_i^j = 0$.

The example also shows that if the above mentioned assumptions are not fulfilled, then P.3 may not be satisfied.

Example

Period		1	2	3	4
Demand	(d_i)	2	2	2	2
Unit ordering cost	(c_i)	2	1	3	3
Setup cost	(S_i)	4	8	1	6
Unit inventory cost	(h_i)	1	1	1	1

We compute $[c_{ij}]$ and $[C_{ij}]$

$$[c_{ij}] = \begin{bmatrix} 2 & 3 & 4 & 5 \\ & 1 & 2 & 3 \\ & & 3 & 4 \\ & & & 3 \end{bmatrix}, \quad [C_{ij}] = \begin{bmatrix} 4 & 6 & 8 & 10 \\ & 2 & 4 & 6 \\ & & 6 & 8 \\ & & & 6 \end{bmatrix}$$

and start to find the solution of the problem:

$$\begin{bmatrix} 4 \\ 8 \\ 1 \\ 6 \end{bmatrix} \begin{bmatrix} 4 & 6 & 8 & 10 \\ & 2 & 4 & 6 \\ & & 6 & 8 \\ & & & 6 \end{bmatrix} \to \begin{bmatrix} 0 \\ 1 \\ 0 \\ 6 \end{bmatrix} \begin{bmatrix} 8 & 6 & 8 & 10 \\ & 6 & 7 & 6 \\ & & 7 & 8 \\ & & & 6 \end{bmatrix}$$

At this phase of the application of the MA, $S_3^3 = 0$ and $\lambda_3 = 7$. Neglecting the elements of the lines above line 3, for the computation of λ_4, would carry a non-optimal solution:

$$\begin{bmatrix} 0 \\ 1 \\ 0 \\ 4 \end{bmatrix} \begin{bmatrix} 8 & 6 & 8 & 10 \\ & 6 & 7 & 6 \\ & & 7 & 8 \\ & & & 8 \end{bmatrix}$$

with total cost $8+6+7+8 = 29$.
The optimal solution is

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 5 \end{bmatrix} \begin{bmatrix} 8 & 6 & 8 & 10 \\ & 6 & 7 & 7 \\ & & 7 & 8 \\ & & & 7 \end{bmatrix}$$

$$[8 \; 6 \; 7 \; 7] \; [\lambda_j]$$

with total cost $8+6+7+7 = 28$.
Also observe that if condition $C_{ij} \leqslant \lambda_j$ had been dropped from point 6 of the MA, period 3 could have been identified as a productive period, because $S_3^4 = 0$.
Although it is possible to present and prove properties analogous to P.2 and P.3 (we call them P.4 and P.5) for problems with non-decreasing setup costs (Appendix 2), the example presented above clearly illustrates the impossibility of similar statements for cases with equal holding costs or equal demand in all periods.
We are now in position of presenting the new algorithm, a simplified version of the MA, which is an optimal procedure to solve a LSP with non-increasing ordering costs or with non-decreasing setup costs.

SMA

1. Set $S_i^0 = S_i, \quad i = 1, \ldots, N$
2. Set $j = 1$ and $k = 1$
3. Set $\lambda_j = \min_i \{C_{ij} + S_i^{j-1}\}$ for $i = k, \ldots, j$
4. Set $S_i^j = \begin{cases} S_i^{j-1}, & \text{if } C_{ij} \geqslant \lambda_j \\ S_i^{j-1} - (\lambda_j - C_{ij}), & \text{if } C_{ij} < \lambda_j. \end{cases}$
5. If $j = N$ go to 8.
6. If $S_i^j = 0$ set $k = i$.
7. Set $j = j+1$ and go to 3.
8. Find the greater i such that $S_i^j = 0$.
9. Set $y_i = 1$, $z_{il} = 1$, $l = i, \ldots, j$.
10. Set $j = i-1$ and if $j > 0$ go to 8. Otherwise stop.

4. Computational experiences

The SMA inherited the characteristics of the MA. Moreover, it considerably reduced the number of comparison of costs. It is a simple technique to implement, even large-scale problems can easily be solved by hand calculations. The MA, the SMA and a well known lot-sizing heuristic, the Silver-Meal (SMH) [2], have been programmed what enabled us to make some computational experiments. In a recent paper by Wemmerlöv [4] the SMA is considered one of the best heuristics in terms of cost efficiency, consistency and model simplicity. In what concernes the Wagner–Whitin algorithm [3] it cannot be considered a serious contender in a real-life inventory system, due to the complexity and the computational requirements, [4].

It is known that the SMA may fail to give an optimal solution to problems not having any of the particularities referred to in the last section. A great number of these problems was randomly generated and the SMA always succeeded to give an optimal solution. This great efficiency was expected since we had the feeling that only the confluence of very special situations in the matrix $[C_{ij}]$ could provoke the failure of this algorithm.

Table 1 presents the average computational times in seconds, on an IBM 3033 computer, for problems with different number of periods.

Tests on 308 problems with 12 and 24 periods were performed where for each type of demand chosen different setup costs (S) and inventory holding costs (h) were fixed, with S/h varying from 25 to 400. Table 2 illustrates the main results obtained for the S.M.H.

Table 1. Computational times in sec

Periods	MA	SMA	S.M.H.
4	0.0014	0.0013	0.0007
7	0.0033	0.0028	0.0012
12	0.0085	0.0047	0.0021
24	0.0330	0.0104	0.0049
48	0.1257	0.0214	0.0079

Table 2. Test of the S.M.H.

	% of problem with optimal solution	Relative cost increase
12 periods	76	0.33
24 periods	50	0.81

The relative cost increase was obtained by

$$\text{Relative cost increase} = [C/C^* - 1] \times 100$$

where C is the total cost for the heuristic and C^* is the optimal cost obtained by the MA.

The relatively bad performance for the case of 24 periods is due to the inclusion of a few examples with frequent periods with no demand or with sharply declining demand patterns.

Although at the expense of doubling the computation time of the SMH, we should emphasize that, for the problems under consideration, a straightforward optimal solution can be obtained by the SMA.

Finally we will say that analysis of the computational results, under the relative perspective of heuristic or exact procedure, seems to confirm the SMA as a simple and very efficient method.

References

[1] FERREIRA J. A. S., R. V. V. VIDAL, Lot Sizing Algorithms with Applications to Engineering and Economics. *Int. J. Prod. Res.*, **22** (1984) 4, 575–595.

[2] SILVER E. A., H. C. MEAL, A heuristic for selecting lot size quantities for the case of a deterministic time-varying demand rate and discrete opportunities for replenishment. *Prod. Invent. Management*, **14** (1973) 2.

[3] WAGNER H. M., T. M. WHITIN, Dynamic Version of the Economic Lot Size Model. *Management Science*, **5** (1958).

[4] WEMMERLÖV U., A Comparison of Discrete, Single Stage Lot-sizing Heuristics With Special Emphasis on Rules Based on the Marginal Cost Principle. *Engineering Costs and Production Economics*, 7 (1982), 45–53.

Appendix 1

The dual of the linear programming model is

$$\max \sum_{j=1}^{N} \lambda_j$$

subject to

$$\lambda_j - \delta_{ij} \leqslant c_{ij}, \qquad (i, j), \quad i \leqslant j$$

$$\sum_{j=i}^{N} \delta_{ij} \leqslant S_i, \qquad (i)$$

$$\delta_{ij} \geqslant 0, \quad \lambda_j \geqslant 0, \quad (i, j)$$

where λ_j is the dual variable of the constraint $\sum_{i=1}^{j} z_{ij} \leqslant 1$ and δ_{ij} is the dual variable of the constraint $y_i - z_{ij} \geqslant 0$.

The MA and SMA use the following auxiliary variables S_i^j, defined by means of the dual variables δ_{ij},

$$\delta_{ij} = S_i^{j-1} - S_i^j, \quad (i, j), \quad i \leqslant j$$

with

$$S_i^0 = S_i \text{ (the setup cost at period } i).$$

S_i^j is then the part of the setup cost of production at period i that corresponds to periods after period j.

For an economical interpretation of the dual variables see [1].

Appendix 2

Properties (and their proofs) related to the MA.

P.1

Proof. From the definition of matrix $[C_{ij}]$ we have:

$$C_{ij} = (c_i + \sum_{k=i}^{j-1} h_k) \cdot d_j, \qquad i \leqslant j$$

Supposing $i < l \leqslant j$,

$$C_{ij} - C_{lj} = (c_i + \sum_{k=i}^{j-1} h_k) \cdot d_j - (c_l + \sum_{k=l}^{j-1} h_k) \cdot d_j$$

$$= (c_i - c_l) \cdot d_j + (\sum_{k=i}^{l-1} h_k) \cdot d_j \geqslant 0 \qquad (1)$$

and then,

$$C_{ij} \geqslant C_{lj}, \quad i < l \leqslant j$$

Inequality (1) is due to the assumption of non-increasing ordering costs and to the non-negativity of the variables. ■

P.2

Proof. If $S_{i^*}^k = 0$ then $S_{i^*}^j = 0$ for $j \geqslant k$ (see points 3 and 4 of the MA). P.1 implies

$$C_{ij} \geqslant C_{i^*j}, \quad i \leqslant i^*$$

so,

$$C_{ij} + S_i^{j-1} \geqslant C_{i^*j} + S_{i^*}^{j-1} = C_{i^*j}, \quad j > k, \quad i \leqslant i^*$$

and

$$\min_{i \leqslant i^*} \{C_{ij} + S_i^{j-1}\} = C_{i^*j}$$

Consequently,

$$\lambda_j = \min_{i \leqslant j} \{C_{ij} + S_i^{j-1}\} = \min_{i^* \leqslant i \leqslant j} \{C_{ij} + S_i^{j-1}\}$$

■

P.3

Proof. Point 6 in the MA starts with $j = N$. Let i^* be the greater i such that

$$S_{i^*}^N = 0 \tag{2}$$

Assume that it was not optimal to produce at period i^* to satisfy the demand till period N. Then, there should be a period i $(i < i^*)$ from which that demand could be satisfied in an optimal way. For that i obviously,

$$S_i^N = 0 \text{ and } C_{iN} \leqslant \lambda_N$$

But $C_{i^*N} \leqslant C_{iN}$ (P.1) and hence $C_{i^*N} \leqslant \lambda_N$. This last inequality together with (2) are sufficient conditions (point 6 of the MA) to start production at period i^*, what contradicts our assumption.
Following points 6, 7 and 8 of the MA we will successively have $j = i^* - 1$ till $j = 1$. A similar reasoning (case $j = N$) can still be applied for any of these j's and then complete the proof. ■

P.4

In a problem with non-decreasing setup costs $(S_i \leqslant S_{i+1},\ i = 1, \ldots, N-1)$, if

$$S_{i^*}^k = 0, \quad 1 < i^* \leqslant k \leqslant N$$

then

$$\lambda_j = \min_{i^* \leqslant i \leqslant j} \{C_{ij} + S_i^{j-1}\}, \quad k < j \leqslant N$$

Proof. We start assuming that line i^* was the first (after line 1) we found while using the MA, in which $S_{i^*}^k = 0$, $1 < i^* \leqslant k$. If $S_{i^*}^k = 0$ then $S_{i^*}^l = 0$, $k < l \leqslant N$ (after points 3 and 4 of the MA).

Let us then suppose that we have a strict minimum (the only case that interests us!) of $\{C_{il} + S_i^{l-1}\}$, for $1 \leqslant i \leqslant l$, at line $\underline{i}$ $(1 \leqslant \underline{i} < i^*)$, this is:

$$\min_{1 \leqslant i \leqslant l} \{C_{il} + S_i^{l-1}\} = C_{\underline{i}l} + S_{\underline{i}}^{l-1}, \quad 1 \leqslant \underline{i} < i^*$$

Hence,

$$C_{\underline{i}l} + S_{\underline{i}}^{l-1} < C_{i^*l} + S_{i^*}^{l-1} = C_{i^*l} \text{ (strict minimum)}$$

and

$$C_{\underline{i}l} < C_{i^*l}, \quad 1 \leqslant \underline{i} < i^* \leqslant k < l$$

This inequality is easily generalized to

$$C_{\underline{i}j} < C_{i^*j}, \quad i^* \leqslant j < l \tag{1}$$

If $\underline{i} = 1$ we could never have found $S_{i^*}^k = 0$ by points 3 and 4 of MA, by the preceding inequalities and because $S_1^m = 0$, $m \geqslant 1$.

If $\underline{i} > 1$, with $S_{\underline{i}} \leqslant S_{i^*}$ and $S_{i^*}^k = 0$ by hypothesis, then by points 3 and 4 of MA and inequalities (1), $S_{\underline{i}}^k = 0$, what contradicts the initial assumption. So, in case line i^* $(i^* > 1)$ is the first one to be found with $S_{i^*}^k = 0$, $1 < i^* \leqslant k$, we don't need to consider lines $i < i^*$ for the computation of the λ_j's, $j > k$.

To complete the proof we should analyse the cases when i^* is not the first line (after line 1) to be found with $S_{i^*}^k = 0$. Then a similar reasoning applies, with the little difference that the role of line 1 in the first part of the proof is now taken by line i $(i < i^*)$ in which $S_i^{k-1} = 0$. ∎

P.5

In a problem with non-decreasing setup costs $(S_i \leqslant S_{i+1},\ i = 1, \ldots, N-1)$ condition $C_{ij} \leqslant \lambda_j$ of point 6 of MA may be suppressed if for each j the greater i $(i \leqslant j)$ is found such that $S_i^j = 0$.

Proof. Point 6 of MA starts with $j = N$. Let i^* be the greater i such that $S_{i^*}^N = 0$.

Assume that $C_{i^*N} > \lambda_N$. Then $S_{i^*}^{N-1} = 0$, after points 3 and 4 of MA. Now using P.4 we conclude that to compute λ_N there is no need to consider the lines below line i^*. But demand of period N cannot be satisfied from any period $i > i^*$ in an optimal program because i^* is the greater i such that $S_{i^*}^N = 0$. So,

$$\lambda_N = \min_i \{C_{iN} + S_i^{N-1}\} = C_{i^*N} + S_{i^*}^{N-1}$$

and then $C_{i^*N} \leqslant \lambda_N$, what contradicts the initial assumption.
To complete the proof we just follow points 6, 7 and 8 of MA and apply a similar reasoning, first with $j = i^* - 1$, if $i^* - 1 > 0$, and so on.

Nowy algorytm rozwiązywania zadań ustalenia długości partii towaru

W pracy przedstawiono uproszczony algorytm macierzowy dla rozwiązywania zadań ustalania długości partii towaru. Jest on uproszczeniem znanego dotąd algorytmu macierzowego [1].

Określono szeroką klasę zadań i podano kryteria optymalności, przy których uproszczony algorytm macierzowy zbiega do rozwiązania optymalnego. Na przykładach numerycznych przedstawiono jego efektywność.

Новый алгоритм решения задач определения величимы партии товара

В работе представлен упрощенный матричный алгоритм для решения задач определения величимы партии товара. Он является упрощением ранее известного матричного алгоритма [1].

Определен широкий класс задач и даны критерии оптимальности, при которых упрощенный матричный алгоритм дает близкое к оптимальному решение. Представлена на численных примерах его эффективность.