A linear description of the discrete lot-sizing and scheduling problem

Citation for published version (APA):

Document status and date:
Published: 01/01/1992

Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.
Memorandum COSOR 92-47

A linear description of the discrete lot-sizing and scheduling problem

C.P.M. van Hoesel
A. Kolen

Eindhoven, November 1992
The Netherlands
Eindhoven University of Technology
Department of Mathematics and Computing Science
Probability theory, statistics, operations research and systems theory
P.O. Box 513
5600 MB Eindhoven - The Netherlands

Secretariate: Dommelbuilding 0.03
Telephone: 040-47 3130

ISSN 0926 4493
A linear description of the
discrete lot-sizing and scheduling problem

Stan van Hoesel *  Antoon Kolen †

September 18, 1992

Abstract

A new integer linear programming formulation for the discrete lot-sizing and scheduling problem is presented. This polynomial-size formulation is obtained from the model with the natural variables by splitting these variables. Its linear programming relaxation is shown to be tight, by reformulating it as a shortest path problem. The latter also provides a dynamic programming formulation for the discrete lot-sizing and scheduling problem.

1 Introduction

Production planning decisions in industry are made on two distinct levels: a strategic (long-term) level and an operational (short-term) level. On the strategic level planning systems are used to develop a rough production plan for the coming years, whereas on the operational level detailed production decisions are specified, typically for some months. On both levels a planner is faced with the same types of costs, the inventory holding costs and the production costs. The production costs usually consist of a fixed component, typical for lot-sizing, and a production-size dependent component. The most suitable class of models for strategic planning are economic lot-sizing models. These are capable of handling problems with relatively long periods, in which a large production capacity must be divided among several goods. For operational planning one usually takes a discrete lot-sizing and scheduling model, where periods are so short that only a single item can be produced in a fixed amount.

The history of the Economic Lot-Sizing Problem (ELSP) goes back to the late fifties, when the two seminal papers of Wagner and Whitin [18] and Manne [12] were published. In [18] a dynamic programming algorithm was developed to solve the single-item uncapacitated version of the economic lot-sizing problem. In [12] Manne suggests a linear programming approach for the multi-item capacitated version of the problem. Dynamic programming and linear programming are the two basic techniques that have been used for solving these

*Address: Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands. E-mail: HOESEL@BS.WIN.TUE.NL

†Address: University of Limburg, P.O. Box 616, 6200 MD Maastricht, The Netherlands. E-mail: A.KOLEN@KE.RUL.NL
and other versions of the economic lot-sizing problem. Generalizations of the problem are, among others, the case where backlogging is allowed, and the case where more machines form an assembly line. Both generalizations were introduced by Zangwill [19]. In general, the capacitated economic lot-sizing problem is \( \mathcal{NP} \)-hard. See Florian, Lenstra and Rinnooy Kan [8], and Bitran and Yanasse [4] for a detailed study on the complexity of the problem. Besides many heuristics (see Baker [1] for an overview) exact algorithms by Branch and Bound and Lagrangean relaxation have been developed by, for instance, Thizy and Van Wassenhove [17]. Recently, the polyhedral structure of the single-item economic lot-sizing problem has been investigated by Barany, Van Roy and Wolsey [2], [3]. Their valid inequalities have been implemented successfully in a cutting-plane algorithm for the multi-item problem. Krarup and Bilde [10] provide a polynomial-size complete linear description for the single-item economic lot-sizing problem by splitting the production variables. See also Eppen and Martin [5] for an evaluation of the technique of variable splitting.

Research on the discrete Lot-sizing and Scheduling Problem (DLSP) has started only recently by Schrage [16]. This late interest is due to the developments in management strategies, where short-term decisions become more and more important. Schrage [16] typified production processes with a so-called all-or-nothing policy as discrete lot-sizing and scheduling problems, and he distinguished between two different types of fixed production costs: set-up costs (typical for economic lot-sizing) and start-up or change-over costs (typical for discrete lot-sizing). Solution methods for DLSP are very similar to those for ELSP. A simple dynamic programming recursion solves the single-item version. Fleischmann [6] proposes a branch and bound algorithm by use of Lagrangean relaxation of the capacity constraints of the problem. In [7] he reformulates the problem as a travelling salesman problem with time windows. The complexity of the problem and a set of variants is discussed by Salomon [13]. The multi-item DLSP is solvable in polynomial time if the number of items is fixed, it is binary \( \mathcal{NP} \)-hard if the number of items is a problem parameter. Valid inequalities for the single-item discrete lot-sizing and scheduling problem have been developed by van Hoesel [9]. Magnanti and Vachani [11] and Sastry [15] describe facet-defining inequalities for a slightly more general problem, in which set-up costs are included. In view of the similarity of the mentioned techniques and solution methods for economic lot-sizing problems and discrete lot-sizing and scheduling problems, it is only natural to develop a formulation for the discrete lot-sizing and scheduling problem by splitting the variables. The latter formulation is the subject of this manuscript.

Section 2 contains formulations for the single-item discrete lot-sizing problem. The formulation with the natural variables (for start-ups, production, and inventory) is described and simplified by deleting the inventory variables. By splitting the variables a new extended formulation is derived. For the latter formulation a polynomial-size complete linear programming description is derived in section 3. This description is based on a shortest path model. The relation with a dynamic programming formulation will be described in section 4. Finally, in section 5 generalization of the results to the multi-item discrete lot-sizing and scheduling problem is discussed.
2 Integer programming formulations for the discrete lot-sizing and scheduling problem

Consider the single-item version of DLSP, i.e., we have one single item that must be produced. The planning horizon consists of \( T \) periods, and in each period \( t \in \{1, \ldots, T\} \) a demand of \( d_t \) units of the item occurs. This demand must be satisfied by production in one of the periods up to \( t \). Since an all-or-nothing policy is assumed in each period, the production speed can be normalized to one unit per period. Clearly, this implies that the demands can be restricted to be binary. A maximal set of consecutive periods in which production takes place is called a production batch. Such a batch must begin with a period in which a start-up takes place. The following parameters and variables are used to describe the single-item DLSP. They are defined for each period \( t \in \{1, \ldots, T\} \).

**Parameters:**
- \( d_t \): the demand of the item in period \( t \);
- \( f_t \): the start-up cost of the item in period \( t \);
- \( p_t \): the unit production cost of the item in period \( t \);
- \( h_t \): the unit inventory cost of the item in period \( t \).

**Variables:**
- \( x_t \): the production of the item in period \( t \);
- \( y_t \): \( \begin{cases} 1 & \text{if a start-up of the item is incurred in period } t; \\ 0 & \text{otherwise.} \end{cases} \)
- \( I_t \): the inventory level of the item at the end of period \( t \).

The generic formulation of the problem as suggested by Fleischmann [6] is the following.

\[
\text{(DLSP-I)} \quad \min \sum_{t=1}^{T} (f_t y_t + p_t x_t + h_t I_t) \tag{1}
\]

s.t.

\[
x_t + I_{t-1} = d_t + I_t \quad (1 \leq t \leq T) \tag{2}
\]

\[
y_t \leq x_t \leq x_{t-1} + y_t \leq 1 \quad (1 \leq t \leq T) \tag{3}
\]

\[
I_t \geq 0 \quad (1 \leq t \leq T) \tag{4}
\]

\[
x_t, y_t \in \{0, 1\} \quad (1 \leq t \leq T) \tag{5}
\]

We assume that \( x_0 \) and \( I_0 \) are equal to zero.

The constraints 2, the balance equations, ensure that the starting inventory of period \( t \) and the production at period \( t \) equal demand and ending inventory of period \( t \). The constraints 3 force a start-up when production of the item takes place in period \( t \) but not in the preceding
period \( t - 1 \). Constraints 4 ensure that the ending inventory in \( t \) is nonnegative. The problem above is, in the notation introduced by Salomon et al. [14], denoted by \( 1/1\text{SI}/G/A \), i.e., there is one machine, one item, Sequence Independent start-up costs (set-up costs in their terminology), time-dependent (General) production and inventory costs, and start-up (set-up) times are Absent.

The balance equations 2 can be used to reduce the set of variables and the number of constraints. We will use them to delete the inventory variables from the formulation.

\[
(DLSP) \quad \min_{t=1}^{T} \sum_{t} (f_t y_t + c_t x_t) \quad (6)
\]

\[
s.t. \quad \sum_{t=1}^{t} x_t \geq d_{1,t} \quad (1 \leq t \leq T) \quad (7)
\]

\[
y_t \leq x_t \leq x_{t-1} + y_t \leq 1 \quad (1 \leq t \leq T) \quad (8)
\]

\[
x_t, y_t \in \{0, 1\} \quad (1 \leq t \leq T) \quad (9)
\]

Here, \( c_t \) (\( 1 \leq t \leq T \)) denote the inventory incremented costs. Their relation to the cost coefficients in the original formulation is \( c_t = p_t + h_t + \ldots + h_T \). By \( d_{1,t} \) (\( 1 \leq t \leq T \)) we denote the cumulative demand of the first \( t \) periods, i.e., \( d_{1,t} = \sum_{\tau=1}^{t} d_{\tau} \). Note that \( d_{1,T} = D \).

By splitting the variables one can usually create a tighter integer linear programming reformulation with less constraints but more variables. Splitting the variables is done by specifying the demand period. Therefore, the demand periods are numbered, namely in increasing order of appearance on the planning horizon as follows: \( t_1, t_2, \ldots, t_D \), where \( D \) is the cumulative demand of the periods \( \{1, \ldots, T\} \). The variables, defined for \( i : 1 \leq i \leq D \), and \( t; i \leq t \leq t_i \) are as follows.

\[
x_{t,i} \begin{cases} 
1 & \text{if there is production in period } t \text{ for demand in period } t_i; \\
0 & \text{otherwise.}
\end{cases}
\]

\[
y_{t,i} \begin{cases} 
1 & \text{if there is a start-up in period } t \text{ for production of demand from } t_i; \\
0 & \text{otherwise.}
\end{cases}
\]

Clearly, \( x_t = \sum_{i:t_i \geq t} x_{t,i} \) and \( y_t = \sum_{i:t_i \geq 1} y_{t,i} \). In contrast with the economic lot-sizing problem, where splitting the production variables led to a tight linear reformulation (Krarup and Bilde [10]), for DLSP we need in addition the splitting of the start-up variables. The integer linear programming formulation with the split variables is the following.

\[
(DLSP-S) \quad \min \sum_{i=1}^{D} \sum_{t=i}^{t_i} (f_t y_{t,i} + c_t x_{t,i}) \quad (10)
\]

\[
s.t. \quad \sum_{t=i}^{t_i} x_{t,i} = 1 \quad (1 \leq i \leq D) \quad (11)
\]
The constraints 11 ensure that each demand is satisfied. The constraints 12 model the startup structure. The constraints 13 are added to the formulation to restrict the production in each period. The constraints 14 are part of the formulation to avoid certain unnecessary startups. In fact, they ensure that the demand periods increase with the production periods in a feasible solution.

The main advantage of formulation DLSP-S is that production can be related to demand much more effectively. It can be shown that constraints 7 are implied by the constraints 11. Thus, DLSP-S is a stronger formulation than DLSP, in the sense that its linear programming relaxation has a value that is not larger than the value of the linear programming relaxation of DLSP. The linear programming relaxation of DLSP-S still allows for fractional solutions. However, DLSP-S can be viewed as a shortest-path problem on an acyclic network. This viewpoint will enable us to find the set of constraints which, when added to DLSP-S will result in a formulation with integral extreme points.

3 A shortest-path formulation of DLSP

Consider an instance of DLSP. Any feasible solution of formulation DLSP-S specifies production-demand pairs \((t, i)\). The production periods \(t\) in these pairs can be reordered to an increasing sequence for increasing \(i\). We will only consider solutions with this property in the following shortest-path model for DLSP. An instance of the shortest-path problem is defined as follows.

The graph is a 2-dimensional structure that consists of \(D\) vertical layers and \(T\) horizontal layers. Moreover, there is a source node \(S_a\) and a target node \(T_a\). The intersection of vertical layer \(i\) and horizontal layer \(t\) contains at most four vertices \(P_{t,i}, Q_{t,i}, R_{t,i},\) and \(S_{t,i}\). The arcs between these vertices as well as the connection with other vertices is given in figure 1, together with the name of the corresponding variables.

Vertex \(P_{t,i}\) exists for \(i = 2, \ldots, D; t = i + 1, \ldots, t_i\).
Vertex \(Q_{t,i}\) exists for \(i = 2, \ldots, D; t = i, \ldots, t_{i-1} + 1\).
Vertex \(R_{t,i}\) exists for \(i = 2, \ldots, D; t = i, \ldots, t_i - 1\).
Vertex \(S_{t,i}\) exists for \(i = 1, \ldots, D; t = i, \ldots, t_i\).

The arcs with corresponding variables and costs are:
Figure 1: Part of the network.

\[(S_a, s_{t,1})\] with flow \(y_{t,1}\) and cost \(c_t\) \((1 \leq t \leq t_1)\);  
\[(p_{t,i}, r_{t,i})\] with flow \(\delta_{t,i}\) and cost 0 \((2 \leq i \leq D; i < t < t_i)\);  
\[(p_{t,i}, s_{t,i})\] with flow \(y_{t,i}\) and cost \(f_t\) \((2 \leq i \leq D; i < t \leq t_i)\);  
\[(q_{t,i}, r_{t,i})\] with flow \(\alpha_{t,i}\) and cost 0 \((2 \leq i \leq D; i \leq t \leq \min\{t_{i-1} + 1, t_i - 1\})\);  
\[(q_{t,i}, s_{t,i})\] with flow \(\beta_{t,i}\) and cost 0 \((2 \leq i \leq D; i \leq t \leq t_{i-1} + 1)\);  
\[(s_{t,i}, q_{i+1,i+1})\] with flow \(x_{t,i}\) and cost \(c_i\) \((1 \leq i \leq D - 1; i \leq t \leq t_i)\);  
\[(r_{t-1,i}, p_{t,i})\] with flow \(\gamma_{t,i}\) and cost 0 \((2 \leq i \leq D; i < t \leq t_i)\);  
\[(s_{t,D}, T_a)\] with flow \(x_{t,D}\) and cost \(c_l\) \((D \leq t \leq t_D)\).

The complete graph related to the following instance of DLSP is given in figure 2.

EXAMPLE: \(T = 6\)

\[
\begin{array}{c|ccccccc}
 t & 1 & 2 & 3 & 4 & 5 & 6 \\
 d_t & 0 & 0 & 1 & 0 & 1 & 1 \\
\end{array}
\]

We now describe the one-to-one correspondence between feasible solutions to DLSP and paths from \(S_a\) to \(T_a\) in the graph. This will also clarify why some of the \(p, q, r,\) and \(s\) vertices do not exist.

Let us first concentrate on vertical layer one. We will leave this layer through vertex \(s_{t,1}\) if and only if the first start-up period is \(t\). In this case the flow variables \(y_{t,1}\) and \(x_{t,1}\) are both equal to one. In general, we enter vertical layer \(i\) at vertex \(q_{t,i}\), if \(t - 1\) is the period in which the demand of period \(t_{i-1}\) is produced. We leave layer \(i\) at some vertex \(s_{t',i}\) with \(t' \geq t\). There is a unique path from \(q_{t,i}\) to \(s_{t',i}\).

\(t' = t\) if and only if we produce in period \(t\) for the \(i\)-th demand period;
Figure 2: Shortest path network of example.
\( t' > t \) if and only if we do not produce in periods \( t, \ldots, t' - 1 \) and start up production in period \( t' \) for production of the \( i \)-th demand period.

The shortest-path linear programming formulation is constrained by the following arguments. The flow leaving \( S_a \) is one unit. The flow through each arc must be nonnegative. Finally, in each node we have flow conservation. The flow conservation constraints are used to eliminate the \( \alpha, \beta, \gamma, \) and \( \delta \) variables. The nonnegativity constraints are then used to create new linear inequalities. Note that since there are no cycles in the graph the flow through each arc is automatically bounded by one.

Flow = 1: \[ \sum_{t=1}^{t_1} y_{t,1} = 1 \] (16)

Flow conservation constraints:

At \( s_{t,1} \):
\[ y_{t,1} = x_{t,1} \] (17)

At \( p_{t,i} \):
\[ \gamma_{t,i} = \delta_{t,i} + y_{t,i} \] (18)
\[ \gamma_{t,i} = y_{t,i} \] (19)

At \( q_{t,i} \):
\[ \alpha_{t,i} + \beta_{t,i} = x_{t-1,i-1} \] (20)
\[ \beta_{t,i} = x_{t-1,i-1} \] (21)

At \( r_{t,i} \):
\[ \alpha_{t,i} = \gamma_{t+1,i} \] (22)

At \( r_{t,i} \):
\[ \alpha_{t,i} + \delta_{t,i} = \gamma_{t+1,i} \] (23)
\[ \delta_{t,i} = \gamma_{t+1,i} \] (24)

At \( s_{t,i} \):
\[ \beta_{t,i} = x_{t,i} \] (25)

At \( s_{t,i} \):
\[ \beta_{t,i} + y_{t,i} = x_{t,i} \] (26)
\[ y_{t,i} = x_{t,i} \] (27)

From the flow conservation constraints we get explicit expressions for the \( \alpha_{t,i}, \beta_{t,i}, \gamma_{t,i} \) and \( \delta_{t,i} \) as follows.

Flow conservation at \( s_{t,i} \):

From 25: \[ \beta_{t,i} = x_{t,i} \] (28)

8
From 26:  \[ \beta_{t,i} = x_{t,i} - y_{t,i} \quad (2 \leq i \leq D; i < t \leq t_{i-1} + 1) \]  

(29)

Flow conservation at \( q_{t,i} \) and \( s_{t,i} \):

From 20 and 25:

\[ \alpha_{i,i} = x_{i-1,i-1} - x_{i,i} \quad (2 \leq i \leq D) \]  

(30)

From 20 and 26:

\[ \alpha_{t,i} = x_{t-1,i-1} + y_{t,i} - x_{t,i} \quad (2 \leq i \leq D; i < t \leq \min\{t_{i-1} + 1, t_i - 1\}) \]  

(31)

We now consider \( \gamma_{t+1,i} \) (\( i \leq t < t_i \)). Consider the vertices \( p_{r,i}, q_{r,i}, r_{r,i} \) and \( s_{r,i} \) (\( r > t \)). If \( t \leq t_{i-1} \), then the total flow entering these vertices is equal to \( \gamma_{t+1,i} \) (at vertex \( p_{t+1,i} \)) and \( x_{r-1,i-1} \) (at vertex \( q_{r,i} \)). The total flow leaving is equal to \( x_{r,i} \) (at vertex \( s_{r,i} \)). By flow conservation we obtain

\[ \gamma_{t+1,i} = \sum_{r=t+1}^{t_i} x_{r,i} - \sum_{r=t}^{t_{i-1}} x_{r,i-1} \quad (2 \leq i \leq D; i < t \leq t_{i-1}) \]  

(32)

If \( t > t_{i-1} \), then the total flow entering is equal to \( \gamma_{t+1,i} \) (at vertex \( p_{t+1,i} \)). The total flow leaving is equal to \( x_{r,i} \) (at vertex \( s_{r,i} \)). By flow conservation we obtain

\[ \gamma_{t+1,i} = \sum_{r=t+1}^{t_i} x_{r,i} \quad (2 \leq i \leq D; t_{i-1} < t < t_i) \]  

(33)

With these inequalities and flow conservation at \( r_{t,i} \) we get:

\[ \delta_{t,i} = \gamma_{t+1,i} - \alpha_{t,i} \]

\[ = \sum_{r=t}^{t_i} x_{r,i} - \sum_{r=t-1}^{t_{i-1}} x_{r,i-1} - y_{t,i} \quad (2 \leq i \leq D; i < t \leq t_{i-1} + 1) \]  

(34)

\[ \delta_{t,i} = \gamma_{t+1,i} = \sum_{r=t+1}^{t_i} x_{r,i} \quad (2 \leq i \leq D; t_{i-1} < t < t_i) \]  

(35)

The following constraints are implied by the nonnegativity of the variables \( \alpha_{t,i}, \beta_{t,i} \) and \( \delta_{t,i} \). Note that 22 to 24 imply that \( \gamma_{t,i} \geq 0 \). We get the following complete linear description of DLSP-S.

\[ \sum_{t=1}^{t_i} y_{t,i} = 1 \]  

(36)
\[ x_{t,1} = y_{t,1} \geq 0 \quad (1 \leq t \leq t_1) \]  
(37)

\[ x_{i,i} \geq 0 \quad (2 \leq i \leq D) \]  
(38)

\[ x_{t,i} \geq y_{t,i} \geq 0 \quad (2 \leq i \leq D; i < t \leq t_{i-1} + 1) \]  
(39)

\[ x_{t,i} = y_{t,i} \geq 0 \quad (2 \leq i \leq D; t_{i-1} + 1 < t \leq t_i) \]  
(40)

\[ x_{i-1,i-1} \geq x_{i,i} \geq 0 \quad (2 \leq i \leq D) \]  
(41)

\[ x_{t-1,i-1} + y_{t,i} \geq x_{t,i} \geq 0 \quad (2 \leq i \leq D; i < t \leq \min\{t_{i-1} + 1, t_i - 1\}) \]  
(42)

\[ x_{t-1,i-1} + y_{t,i} = x_{t,i} \geq 0 \quad (2 \leq i \leq D; t = t_{i-1} + 1 = t_i) \]  
(43)

\[ \sum_{\tau=t}^{t_i} x_{\tau,i} \geq y_{t,i} + \sum_{\tau=t-i-1}^{t_i-1} x_{\tau,i-1} \quad (2 \leq i \leq D; i < t \leq t_{i-1} + 1) \]  
(44)

Except for the last inequality 44 most constraints can easily be verified. The latter inequality is a strengthening of the inequalities 14 with \( y_{t,i} \) added to the right hand-side. Concluding, the model defined by 36 - 44 gives a complete linear description of DLSP-S.

4 Dynamic programming formulation of DLSP

In the shortest path formulation of DLSP the length of an arc corresponding to flow variable \( x_{t,i} \) is \( c_i \); the length of an arc corresponding to flow variables \( y_{t,i} \) is \( f_t \); all other arcs have length zero. We define

\[ S(t, i) = \text{length of the shortest path from } S_a \text{ to } s_{t,i} \text{ for } i = 1, \ldots, D; t = i, \ldots, t_i; \]

\[ R(t, i) = \text{length of the shortest path from } S_a \text{ to } r_{t,i} \text{ for } i = 2, \ldots, D; t = i, \ldots, t_i - 1. \]

Let us consider vertices \( s_{t,i} \) (\( t = 1, \ldots, t_1 \)). Clearly,

\[ S(t, 1) = f_t \quad (1 \leq t \leq t_1) \]  
(45)

Each path through \( s_{i,i} \) (\( 2 \leq i \leq D \)) passes through \( s_{i-1,i-1} \) and therefore

\[ S(i, i) = S(i - 1, i - 1) \quad (2 \leq i \leq D) \]  
(46)

The paths to \( s_{t,i} \) \( i = 2, \ldots, D; t = i + 1, \ldots, t_i - 1 + 1 \) pass either through \( s_{t-1,i-1} \) or \( r_{t-1,i} \). The length of the path from \( s_{t-1,i-1} \) to \( s_{t,i} \) is \( c_{t-1} \); the length of the path from \( r_{t-1,i} \) to \( s_{t,i} \) is \( f_t \). Therefore
\[ S(t, i) = \min\{S(t - 1, i - 1) + c_{t-1}, R(t - 1, i) + f_t\} \]
\[ (2 \leq i \leq D; i < t \leq t_{i-1} + 1) \] (47)

The paths to \( s_{t,i} \) \( i = 2, \ldots, D; t = t_{i-1} + 2, \ldots, t_i \) pass only through \( r_{t-1,i} \). Thus
\[ S(t, i) = R(t - 1, i) + f_t \]
\[ (2 \leq i \leq D; t_{i-1} + 1 < t \leq t_i) \] (48)

Each path through \( r_{t,i} \) \( (2 \leq i \leq D) \) passes through \( s_{i-1,i-1} \) and therefore
\[ R(i, i) = S(i - 1, i - 1) + c_{t-1} \]
\[ (2 \leq i \leq D) \] (49)

The paths to \( r_{t,i} \) \( i = 2, \ldots, D; t = i + 1, \ldots, t_{i-1} + 1 \) pass either through \( s_{t-1,i-1} \) or \( r_{t-1,i} \). The length of the path from \( s_{t-1,i-1} \) to \( r_{t,i} \) is \( c_{t-1} \); the length of the path from \( r_{t-1,i} \) to \( r_{t,i} \) is zero. Therefore
\[ R(t, i) = \min\{S(t - 1, i - 1) + c_{t-1}, R(t - 1, i)\} \]
\[ (2 \leq i \leq D; i < t \leq t_{i-1} + 1) \] (50)

The paths to \( r_{t,i} \) \( i = 2, \ldots, D; t = t_{i-1} + 2, \ldots, t_i - 1 \) pass only through \( r_{t-1,i} \). Thus, since the path from \( r_{t-1,i} \) to \( r_{t,i} \) has zero length
\[ R(t, i) = R(t - 1, i) \]
\[ (2 \leq i \leq D; t_{i-1} + 1 < t < t_i) \] (51)

Finally, the sought value is
\[ \min\{S(t, D) + c_t | t = D, \ldots, T\} \] (52)

Clearly, we can interpret \( R(t, i) \) as the minimum cost of producing the first \( i - 1 \) demands in the periods 1, \ldots, \( t - 1 \) and not producing in period \( t \), and \( S(t, i) \) as the minimum cost of producing the first \( i - 1 \) demands in the periods 1, \ldots, \( t - 1 \) and producing the \( i \)-th demand in period \( t \). The running time for the algorithm based on the recursion 45 to 52 is \( O(DT) \), since this is an upperbound on the total number of variables in the dynamic programming recursion.

5 Concluding remarks

A complete linear description for the model with disaggregated variables of the single-item DLSP has been derived and its relation to a dynamic programming formulation has been shown. Both the linear description and the dynamic programming recursion can be generalized to the multi-item case. The shortest path model then extends to an \((N + 1)\)-dimensional
network, \( N \) being the number of items. Since the number of variables and the number of constraints increases to \( O(D^N T) \) (\( D \) is the cumulative demand over all items) this model grows quickly with the number of items. Therefore the most practical way to use this formulation is to split a multi-item DLSP in \( N \) single-item problems and solve each with the linear program as defined in section 3. Note that the separation problem for the single-item DLSP can also be modelled as a shortest-path problem. By then, if any non-integral variables are still left specific multi-item constraints may be added.

Acknowledgment.
The authors would like to thank Y. Pochet and M. Salomon for their remarks on section 3.
References


<table>
<thead>
<tr>
<th>Number</th>
<th>Month</th>
<th>Author</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>92-01</td>
<td>January</td>
<td>F.W. Steutel</td>
<td>On the addition of log-convex functions and sequences</td>
</tr>
<tr>
<td>92-02</td>
<td>January</td>
<td>P. v.d. Laan</td>
<td>Selection constants for Uniform populations</td>
</tr>
<tr>
<td>92-04</td>
<td>February</td>
<td>H.J.C. Huijberts H. Nijmeijer</td>
<td>Strong dynamic input-output decoupling: from linearity to nonlinearity</td>
</tr>
<tr>
<td>92-05</td>
<td>March</td>
<td>S.J.L. v. Eijndhoven J.M. Soethoudt</td>
<td>Introduction to a behavioral approach of continuous-time systems</td>
</tr>
<tr>
<td>92-06</td>
<td>April</td>
<td>P.J. Zwietering E.H.L. Aarts J. Wessels</td>
<td>The minimal number of layers of a perceptron that sorts</td>
</tr>
<tr>
<td>92-07</td>
<td>April</td>
<td>F.P.A. Coolen</td>
<td>Maximum Imprecision Related to Intervals of Measures and Bayesian Inference with Conjugate Imprecise Prior Densities</td>
</tr>
<tr>
<td>92-08</td>
<td>May</td>
<td>I.J.B.F. Adan J. Wessels W.H.M. Zijm</td>
<td>A Note on “The effect of varying routing probability in two parallel queues with dynamic routing under a threshold-type scheduling”</td>
</tr>
<tr>
<td>92-10</td>
<td>May</td>
<td>P. v.d. Laan</td>
<td>Subset Selection: Robustness and Imprecise Selection</td>
</tr>
<tr>
<td>92-11</td>
<td>May</td>
<td>R.J.M. Vlaessens E.H.L. Aarts J.K. Lenstra</td>
<td>A Local Search Template (Extended Abstract)</td>
</tr>
<tr>
<td>92-12</td>
<td>May</td>
<td>F.P.A. Coolen</td>
<td>Elicitation of Expert Knowledge and Assessment of Imprecise Prior Densities for Lifetime Distributions</td>
</tr>
<tr>
<td>92-13</td>
<td>May</td>
<td>M.A. Peters A.A. Stoorvogel</td>
<td>Mixed $H_2/H_\infty$ Control in a Stochastic Framework</td>
</tr>
<tr>
<td>Number</td>
<td>Month</td>
<td>Author</td>
<td>Title</td>
</tr>
<tr>
<td>--------</td>
<td>-------</td>
<td>--------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>92-14</td>
<td>June</td>
<td>P.J. Zwietering E.H.L. Aarts J. Wessels</td>
<td>The construction of minimal multi-layered perceptrons: a case study for sorting</td>
</tr>
<tr>
<td>92-15</td>
<td>June</td>
<td>P. van der Laan</td>
<td>Experiments: Design, Parametric and Nonparametric Analysis, and Selection</td>
</tr>
<tr>
<td>92-16</td>
<td>June</td>
<td>J.J.A.M. Brands F.W. Steutel R.J.G. Wilms</td>
<td>On the number of maxima in a discrete sample</td>
</tr>
<tr>
<td>92-17</td>
<td>June</td>
<td>S.J.L. v. Eijndhoven J.M. Soethoudt</td>
<td>Introduction to a behavioral approach of continuous-time systems part II</td>
</tr>
<tr>
<td>92-18</td>
<td>June</td>
<td>J.A. Hoogeveen H. Oosterhout S.L. van der Velde</td>
<td>New lower and upper bounds for scheduling around a small common due date</td>
</tr>
<tr>
<td>92-19</td>
<td>June</td>
<td>F.P.A. Coolen</td>
<td>On Bernoulli Experiments with Imprecise Prior Probabilities</td>
</tr>
<tr>
<td>92-20</td>
<td>June</td>
<td>J.A. Hoogeveen S.L. van de Velde</td>
<td>Minimizing Total Inventory Cost on a Single Machine in Just-in-Time Manufacturing</td>
</tr>
<tr>
<td>92-21</td>
<td>June</td>
<td>J.A. Hoogeveen S.L. van de Velde</td>
<td>Polynomial-time algorithms for single-machine bicriteria scheduling</td>
</tr>
<tr>
<td>92-22</td>
<td>June</td>
<td>P. van der Laan</td>
<td>The best variety or an almost best one? A comparison of subset selection procedures</td>
</tr>
<tr>
<td>92-23</td>
<td>June</td>
<td>T.J.A. Storcken P.H.M. Ruys</td>
<td>Extensions of choice behaviour</td>
</tr>
<tr>
<td>92-25</td>
<td>July</td>
<td>P.J. Zwietering E.H.L. Aarts J. Wessels</td>
<td>Exact Classification With Two-Layered Perceptrons</td>
</tr>
<tr>
<td>92-26</td>
<td>July</td>
<td>M.W.P. Savelsbergh</td>
<td>Preprocessing and Probing Techniques for Mixed Integer Programming Problems</td>
</tr>
<tr>
<td>Number</td>
<td>Month</td>
<td>Author</td>
<td>Title</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>-------------------------------</td>
<td>--------------------------------------------------------------</td>
</tr>
<tr>
<td>92-27</td>
<td>July</td>
<td>I.J.B.F. Adan</td>
<td>Analysing $E_k</td>
</tr>
<tr>
<td></td>
<td></td>
<td>W.A. van de Waarsenburg</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>J. Wessels</td>
<td></td>
</tr>
<tr>
<td>92-28</td>
<td>July</td>
<td>O.J. Boxma</td>
<td>The compensation approach applied to a $2 \times 2$ switch</td>
</tr>
<tr>
<td></td>
<td></td>
<td>G.J. van Houtum</td>
<td></td>
</tr>
<tr>
<td>92-29</td>
<td>July</td>
<td>E.H.L. Aarts</td>
<td>Job Shop Scheduling by Local Search</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P.J.M. van Laarhoven</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>J.K. Lenstra</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>N.L.J. Ulder</td>
<td></td>
</tr>
<tr>
<td>92-30</td>
<td>August</td>
<td>G.A.P. Kindervater</td>
<td>Local Search in Physical Distribution Management</td>
</tr>
<tr>
<td></td>
<td></td>
<td>M.W.P. Savelsbergh</td>
<td></td>
</tr>
<tr>
<td>92-31</td>
<td>August</td>
<td>M. Makowski</td>
<td>MP-DIT Mathematical Program data Interchange Tool</td>
</tr>
<tr>
<td></td>
<td></td>
<td>M.W.P. Savelsbergh</td>
<td></td>
</tr>
<tr>
<td>92-32</td>
<td>August</td>
<td>J.A. Hoogeveen</td>
<td>Complexity of scheduling multiprocessor tasks with</td>
</tr>
<tr>
<td></td>
<td></td>
<td>S.L. van de Velde</td>
<td>prespecified processor allocations</td>
</tr>
<tr>
<td></td>
<td></td>
<td>B. Veltman</td>
<td></td>
</tr>
<tr>
<td>92-33</td>
<td>August</td>
<td>O.J. Boxma</td>
<td>Tandem queues with deterministic service times</td>
</tr>
<tr>
<td></td>
<td></td>
<td>J.A.C. Resing</td>
<td></td>
</tr>
<tr>
<td>92-34</td>
<td>September</td>
<td>J.H.J. Einmahl</td>
<td>A Bahadur-Kiefer theorem beyond the largest observation</td>
</tr>
<tr>
<td>92-35</td>
<td>September</td>
<td>F.P.A. Coolen</td>
<td>On non-informativeness in a classical Bayesian inference</td>
</tr>
<tr>
<td>92-36</td>
<td>September</td>
<td>M.A. Peters</td>
<td>A Mixed $H_2/H_\infty$ Function for a Discrete Time System</td>
</tr>
<tr>
<td>92-37</td>
<td>September</td>
<td>I.J.B.F. Adan</td>
<td>Product forms as a solution base for queueing systems</td>
</tr>
<tr>
<td></td>
<td></td>
<td>J. Wessels</td>
<td></td>
</tr>
<tr>
<td>92-38</td>
<td>September</td>
<td>L.C.G.J.M. Habets</td>
<td>A Reachability Test for Systems over Polynomial Rings using</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Gröbner Bases</td>
</tr>
<tr>
<td>92-39</td>
<td>September</td>
<td>G.J. van Houtum</td>
<td>The compensation approach for three or more dimensional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>I.J.B.F. Adan</td>
<td>random walks</td>
</tr>
<tr>
<td></td>
<td></td>
<td>J. Wessels</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>W.H.M. Zijm</td>
<td></td>
</tr>
<tr>
<td>Number</td>
<td>Month</td>
<td>Author</td>
<td>Title</td>
</tr>
<tr>
<td>--------</td>
<td>-----------</td>
<td>-------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>92-40</td>
<td>September</td>
<td>F.P.A. Coolen</td>
<td>Bounds for expected loss in Bayesian decision theory with imprecise prior probabilities</td>
</tr>
<tr>
<td>92-41</td>
<td>October</td>
<td>H.J.C. Huijbers, H. Nijmeijer, A.C. Ruiz</td>
<td>Nonlinear disturbance decoupling and linearization: a partial interpretation of integral feedback</td>
</tr>
<tr>
<td>92-42</td>
<td>October</td>
<td>A.A. Stoovogel, A. Saberi, B.M. Chen</td>
<td>The discrete-time $H_{\infty}$ control problem with measurement feedback</td>
</tr>
<tr>
<td>92-43</td>
<td>October</td>
<td>P. van der Laan</td>
<td>Statistical Quality Management</td>
</tr>
<tr>
<td>92-44</td>
<td>November</td>
<td>M. Sol, M.W.P. Savelsbergh</td>
<td>The General Pickup and Delivery Problem</td>
</tr>
<tr>
<td>92-45</td>
<td>November</td>
<td>C.P.M. van Hoesel, A.P.M. Wagelmans, B. Moerman</td>
<td>Using geometric techniques to improve dynamic programming algorithms for the economic lot-sizing problems and extensions</td>
</tr>
<tr>
<td>92-46</td>
<td>November</td>
<td>C.P.M. van Hoesel, A.P.M. Wagelmans, L.A. Wolsey</td>
<td>Polyhedral characterization of the Economic Lot-sizing problem with Start-up costs</td>
</tr>
<tr>
<td>92-47</td>
<td>November</td>
<td>C.P.M. van Hoesel, A. Kolen</td>
<td>A linear description of the discrete lot-sizing and scheduling problem</td>
</tr>
</tbody>
</table>