KM 10 (1983)
pag 159 - 176

Design and Optimization of Multi-Echelon Assembly Networks: Savings and Potentialities

by

Paul van Beek*
Arie Bremer**
Cornelis van Putten***

Abstract

In the last decade considerable attention has been paid to the analysis and optimization of multi-echelon assembly networks. Research contributions in this field are mainly focussed on the determination of optimal batch quantities. In this paper two other issues concerning multi-echelon assembly networks are also discussed. The first point concerns the notion of *flexibility* in assembly operations. Flexibility can be achieved by spending on assembly engineering to cut set-ups costs. The second point has to do with *design* of multi-level assembly systems. Furthermore the notion of *minimum network leadtime* is introduced. This turns out to be (besides minimal average cost per period) a performance indicator of considerable importance. To conclude an existing Philips product (called MEDSYS) is presented in order to illustrate the ideas discussed in this paper.

Key-words: dynamic programming, leadtime, lot-sizing, manufacturing resource planning, multi-echelon networks.

1. Introduction

In the last few years considerable attention has been paid to the optimization of multi-echelon assembly/production networks under *infinite* loading conditions. These conditions turn out to be quite unrealistic in practice because capacity at the various assembly/production stages will invariably be *finite*.
Recently some attention has been focussed on models for simultaneous lot-sizing and capacity planning in multi-stage assembly networks (Blackburn and Millen). This area of research is both for practice and theory of crucial importance.

* Agricultural University, Wageningen, The Netherlands
** Philips'Industries, MSD, Eindhoven, The Netherlands
*** Philips'Industries, CQM, Eindhoven, The Netherlands.

In the frame-work of Manufacturing Resource Planning (MRP) the need for "closing the loop" in the assembly/production process is evident. This means that in the process of exploding the requirements defined through the Master Production Schedule (MPS) to all subassemblies and parts involved, constraints on the capacity of all assembly operations should be taken into account. In this way the rescheduling of the entire assembly network with a modified MPS until a feasible schedule has been found becomes superfluous.

The majority of the contributions in the field of simultaneous lot-sizing and capacity planning in these networks, is concentrated on the determination of batch quantities. Although this is considered to be of utmost importance, there are many more relevant questions to be posed. Two other very important issues in this field are discussed.

The first point concerns *flexibility* in these assembly operations.

To achieve flexibility in the assembly operations the assembly lead times should be *short*. This is the only way to tune final assembly with the ultimate sales as intimate as possible, resulting in a drastic reduction of stocks.

By driving down set-up costs it becomes economical to run *smaller* batches, ideally a batch of one. This can be achieved by spending on assembly engineering to cut these set-up costs (Schonberger, (1981)). A batch size of one corresponds to the notion of *flow assembly*, a common mode of operation,for instance,in aircraft industry.

In an assembly environment set-up costs are not only caused by preparing a machine; they are for the major part due to the supply of materials at each assembly station.

The second point is concerned with the *design* of multi-level assembly systems. Typical questions to be answered are:

- What is the effect on performance and cost of *merging* two or more assembly stages into one new stage?
- Should the materials needed for a certain assembly stage be supplied with the same frequency as the set-ups occur in that stage? Or will it be favourable to change the structure of the assembly network by disconnecting the material flow towards the assembly stage and the flow out of the materials store by means of a buffer?

The above aspects of assembly systems are discussed in this paper on the basis of a real highly complex product (called MEDSYS) assembled at Philips'Industries in Eindhoven.

The environment in which this product is assembled will be discussed thoroughly in section 2.
The model SIMLOT was developed especially for this environment. It enables management to get insight in

i) lot-size determination for an assembly network with tree structure;
ii) assembly lead time vs. costs;
iii) design of multi-level, multi-stage assembly networks with tree structure.

This model will be the subject of section 3.
In section 4 the optimization aspects are discussed.
Results and conclusions are presented in section 5 en 6, respectively.
Furthermore, all monetary quantities are expressed in dutch guilders (Dfl.).

2. The Problem Environment

Philips is a multinational company (with turnover of approximately Dfl. 40.10^9 producing and selling a wide variety of electronic products in a large number of countries.
The industry at which SIMLOT has been developed is the Medical System Division (MSD) of Philips'Industries at Eindhoven (The Netherlands).
The program of the Medical Systems Division covers a large range of products like X-ray diagnostic imaging, Thermal and Ultrasonic diagnostic systems, Radiation Therapy and Nuclear Medicine systems, computerized X-ray Scanners etc.. Each (system) product is available in a great variety to meet as close as possible the application-requirements of the different customers. Obviously, the MSD products are *highly specialized* and rely heavily upon advanced mechanics, electronics and software. The quantities produced per option are relatively low.
The customers of MSD are, for the major part, hospitals which are in general confronted with serious budget restrictions. Due to the world-wide recession the market is somewhat stabilizing.
To maintain the market position the price/performance ratios of the products as well as the flexibility of the supply organization are getting great emphasis. These circumstances caused organizational changes from a *functional* to a *product-oriented* structure. In view of these changes the model SIMLOT is very important: it has to support management with evaluations of different proposals concerning the way of assembling.

The production consists mainly of the assembly of components and subassemblies into *standard system modules* *). These standard system modules can be small (a printed circuit board) or very large (a mechanical patient handling device); they are the building blocks for the system to be installed at customer-site, in meeting all agreed specifications.
A typical standard system module possesses a product structure of 5 levels, consists of about 100 subassemblies and has a yearly turnover of about 100. Each assembly is manufactured and tested.
Until now the assembly operations are mainly *batch* oriented. The standard system modules are stored in the warehouse after assembly.
Assembly batches cover the requirement of a month up to a year, dependent on the type of the system module. A well-proved calculation method for determining lot-sizes of subassemblies is not available.
Some attempts were made to use the Wilson lot-size formula, not because it is correct in a lumpy-demand environment but because of its convenience to work with.
Usually the lot-size determination (on all levels) was based on experience: price, technology and other aspects thereby played a role. Because of the absence of an formal calculation tool to determine lot-sizes, an objective discussion of this subject was virtually impossible and the suggestion to shorten assembly leadtime by introducing the concept of *flow assembly* could not be evaluated financially.
In this situation MSD has decided to start with the development of the model SIMLOT to get more insight in cost and leadtime of the assembly operations. One function of SIMLOT consists of the determination, in a multi-level, multi-stage environment, of optimal assembly lots.
To cope with uncertainty in the demand for the final product, the concept of *quadratic obsolescence cost* has been introduced in SIMLOT. This means that, besides costs in connection with assembly set-up and stock-holding of a certain subassembly, a cost item that is proportional to the *square* of the sojourn time in the assembly system is added to the total cost related to that subassembly.
This quadratic obsolescence cost has the effect of a *downward* pressure on assembly leadtimes with emphasis on the *long* ones.

---

*) In the subsequent sections we will use the more common name *finished product* or *final product* instead of *standard system module*.

In SIMLOT several minor simplifications were made, the most important being:

- Learning curve effects are neglected. This means that the learning curve for short, often repeated workcycles is assumed to be the same as for long workcycles that are repeated less frequently.
- Flow assembly is equivalent to batch assembly with lot-size one.

3. The Model

In this section the model on which SIMLOT is based is discussed thoroughly.

3.1 Preliminary Remarks

Recently, Blackburn and Millen (1981) reported on an exploratory study of an alternative approach to the problem of how to determine optimal lot-sizes in capacitated assembly networks having a directed tree structure*). Figure 1 shows a typical example of such a network with 4 levels and 9 assembly stages, a so-called 4-level, 9-stage assembly network.

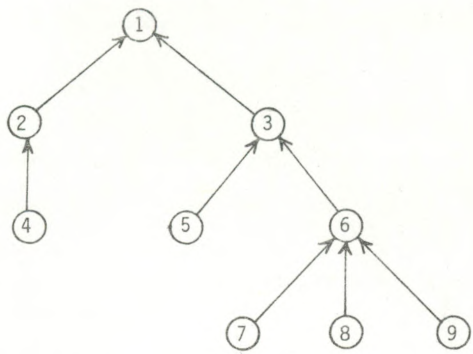

Figure 1 : A 4-level, 9-stage assembly network

The *nodes* of the network represent the various assembly stages whilst the *arcs* indicate the flow of goods. Node 1 always corresponds to the *finished product* stage. The other nodes correspond to the other stages where subassemblies are produced.

Batches (or *lots*) are generated according to a *pull system* initiated by the demand for the finished product. By this we mean that the assembly at each stage is scheduled on the basis of the requirements of its successor stage.

---

*) The multiple finished product version (arborescence) is not treated here. We restrict ourselves to directed networks with *tree* structure.

The assembly operation in stage i consist of increasing the value of the preceding subassemblies with the value of *materials* and *labour* added in stage i. An example (figure 1): the value of the subassembly 3 consists of the value of the two subassemblies 5 and 6 augmented with the value of materials and labour added in stage 3. The latter value is the very function of the assembly operation at stage 3. Starting point for our contribution is the *constant* demand, *infinite* horizon, *capacitated* network model of Blackburn and Millen (1981).
In section 3.3 we shall illustrate our approach by means of a very simple 2-level, 2-stage assembly network (see figure 2).

Figure 2 : A 2-level, 2-stage assembly network

In the sequel we make use of the following definitions:

D : = demand per period for the finished product (number of items)

J : = total number of assembly stages

n_j : = number of periods between two set-ups of batches in stage j

P_j : = the set of immediate predecessor stages of stage j (in figure 1 : $P_6 = \{7,8,9\}$)

α_j : = costcoefficient of stockholding per period due to the *material* added in stage j to *one* item of subassembly j

β_j : = costcoefficient of stockholding per period due to the *labour* added in stage j to *one* item of subassembly j

γ_j : = proportionality factor related to the obsolescence cost

F_j : = fixed set-up cost incurred each time the assembly of a new batch is started in stage j

M_j : = value of the *material* added in stage j to one item of subassembly j

L_j : = value of the *labour* added in stage j to one item of subassembly j

$\overline{B}_j$: = upper bound for the lot-size n_jD at stage j

$\underline{B}_j$: = lower bound for the lot-size n_jD at stage j

v_j : = the maximum number of items that can be assembled in stage j during one period

$p(j)$: = the total number of nodes on the path from node j to node 1 (including node j and node 1. In figure 1: $p(8) = 4$, $p(3) = 2$)

$s(j,i)$: = the $(i-1)^{th}$ *successor stage* of stage j, $i = 1,2,\dots,p(j)$ (in figure 1: $s(7,1) = 7$, $s(7,2) = 6$, $s(7,3) = 3$ and $s(7,4) = 1$).

3.2 Cost relations

In this section we discuss the following cost relations: cost of stockholding, cost of set-up and cost related to the obsolescence risk.

A. *Cost of stockholding*

Consider figure 3, derived for the 2-level, 2-stage network of figure 2 (we assume $v_j = D$, $j = 1,2$). At time $t = 0$ the assembly of a lot of subassembly 2 starts. Material (with value M_2n_2D) for assembling this lot (with lot-size n_2D) enters the system at $t = 0$.

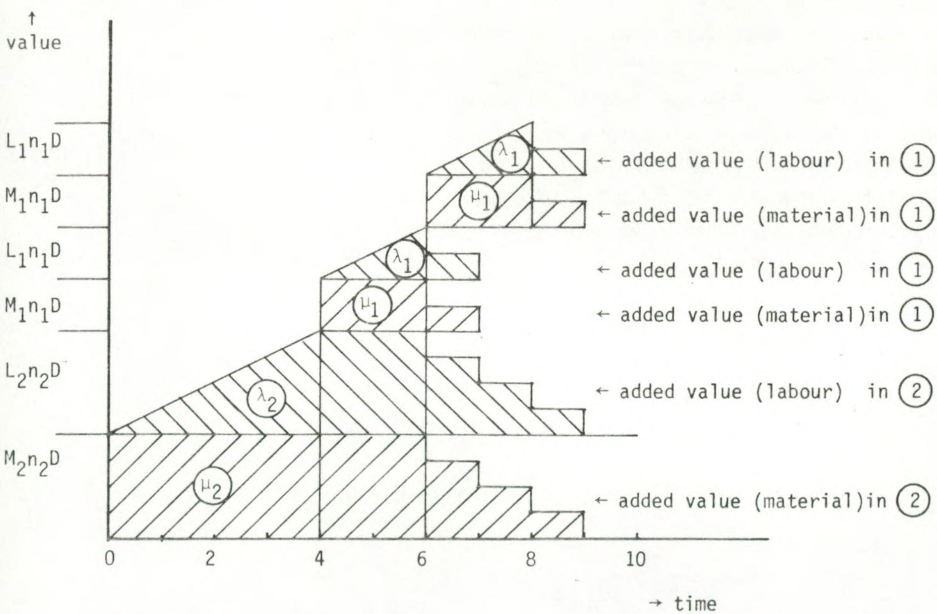

Figure 3: A 2-level, 2-stage assembly network ($n_1 = 2$, $n_2 = 4$)
(The costs are proportional to the shaded area).

At time $t = n_2 = 4$ this lot is completed and forwarded (with no delay) to the next stage 1.
At time $t = n_2 = 4$ the assembly of the *first* lot of the finished product 1 can start. Material (with value M_1n_1D) for assembling this lot (with lot-size n_1D) enters the system at $t = n_2$.
From $t = n_1 + n_2 = 6$ on the finished product can be delivered to the customers in the desired portions of D items per period.
At time $t = n_2 + n_1 = 6$ the assembly of the *second* lot of the finished product 1 can start. Material (with value M_1n_1D) for assembling this second lot (with lot-size n_1D) enters the system at $t = n_1 + n_2 = 6$.
In contrast to the fact that *material* necessary for making a lot n_2D of sub-assembly 2 should be available at time $t = 0$, the *labour* (with value L_2n_2D) necessary for making that lot is brought into the system *gradually* during the periods 1, 2, ..., n_2 (see figure 3).
The value M_2n_2D remains in the assembly system during the periods 1,2,..., $n_1 + n_2$. During the periods $n_1 + n_2 + 1, \ldots, n_1 + 2n_2$ this value leaves the system in portions of M_2D per period (as part of the value of a finished product item). An analogous reasoning can be held for the value L_2n_2D, M_1n_1D and L_1n_1D (see figure 3).
Total costs of stockholding (related to the value M_2n_2D of the material added in stage 2 at $t = 0$) are proportional to the shaded area (indicated in figure 3 as: μ_2) with proportionality factor α_2.
As the value M_2n_2D is supplied to the system once per n_2 periods, the average cost of stockholding (concerning the material added in stage 2) per period can be obtained by dividing this total cost by n_2.
Now let us consider the situation of a general network.
The shaded area μ_j, corresponding to the *material* supplied to stage j amounts to (compare figure 3):

$$(1) \quad \mu_j = M_jn_jD\,(t_j + \tfrac{1}{2}n_j - \tfrac{1}{2})$$

with

$$(2) \quad t_j := \sum_{i=1}^{p(j)} \frac{n_{s(j,i)}D}{v_{s(j,i)}}$$

(t_j is called the *minimum assembly leadtime for stage* j).

The shaded area λ_j, corresponding to the *labour* supplied to stage j gradually, amounts to:

$$(3) \quad \lambda_j = L_j n_j D \left(t_j - \frac{n_j D}{2v_j} + \tfrac{1}{2}n_j - \tfrac{1}{2}\right).$$

Note that the factor 2 in $\frac{n_j D}{2v_j}$ stems from the fact that labour is supplied *gradually* in stage j. Now it is possible to draw up the average cost per period related to stockholding ($C_j^{(1)}$):

$$(4) \quad C_j^{(1)} = \alpha_j M_j D(t_j + \tfrac{1}{2}n_j - \tfrac{1}{2}) + \beta_j L_j D\left(t_j - \frac{n_j D}{2v_j} + \tfrac{1}{2}n_j - \tfrac{1}{2}\right).$$

Remark: The quantity $D(t_j + \frac{1}{2}n_j - \frac{1}{2})$ corresponds to the notion of *average echelon stocklevel of stage j* (see Clark and Scarf (1960)).

B. *Cost of set-ups*

This cost component is very easy to obtain. As only *one* set-up occurs during n_j periods the average cost per period related to set-up ($C_j^{(2)}$) is as follows:

$$(5) \quad C_j^{(2)} = \frac{F_j}{n_j}.$$

C. *Cost related to the obsolescence risk*

It is felt that the obsolescence riks of subassembly j increases *more* than proportional to the sojourn time of that subassembly in the system.
The *sojourn time* of an item of subassembly j is defined as the time spent in the system between the instant of a set-up of subassembly j and the instant the finished product item in which an item of subassembly j is incorporated, leaves the system by delivery to the warehouse. To quantify this risk a cost component $C_j^{(3)}$ is introduced that is proportional to the product of the value $L_j + M_j$ added to one item in stage j and the average *squared* sojourn time of that item in the system.

Suppose a lot of $n_j D$ items of subassembly j is produced during the time interval $[0, n_j D/v_j]$. The finished product item, in which the *first* item of the lot of $n_j D$ was built, leaves the system at time t_j. The *last* item leaves the system at time $t_j + n_j - 1$ (compare figure 3).

Now the *obsolescence cost* per period of subassembly j, $C_j^{(3)}$, can be expressed as follows:

$$(6) \quad C_j^{(3)} = \frac{\gamma_j(L_j + M_j)}{n_j} \sum_{k=0}^{n_j-1} (t_j + k)^2$$

(For the sake of clarity we do not substitute (in (6)) the relation
$\frac{1}{n_j} \sum_{k=0}^{n_j-1} (t_j + k)^2 = n_j \{t_j^2 + (n_j - 1)t_j + \frac{1}{6}(n_j - 1)(2n_j - 1)\}$).

The average total cost per period (C_{tot}) then becomes:

$$(7) \quad C_{tot} = \sum_{j=1}^{J} \{C_j^{(1)} + C_j^{(2)} + C_j^{(3)}\}.$$

4. Optimization

In this section we make use of the *Integrality Property* of an assembly system with directed tree structure.

Integrality Property

In minimizing expression (7) with respect to $n_1, n_2, \ldots, n_J$ and subject to capacity constraints we can restrict ourselves to solutions satisfying

$$(8) \quad n_j = k_j n_i, \; j \in P_i, \; i = 0, 1, \ldots, J.$$

The variable k_j is a *positive* integer, $n_0 := 1$ and $P_0 := \{1\}$ (see Crowston et al. (1973) and Schwarz and Schrage (1975)).

Remark: The Integrality Property also finds its 'justification' in practice.

Our first aim is optimization of the average total cost C_{tot} subject to the restriction that the minimum leadtimes t_j, $j = 1,2,\ldots,J$, should not exceed a prescribed value T:

$$(9) \quad \text{minimize } \{C_{tot}\}$$

subject to the following restrictions

$$\begin{cases} n_j = k_j n_i, \; n_i \text{ and } k_j \in \mathbb{N}, \; j \in P_i, \; i = 0,1,\ldots,J \\ \underline{B}_j \leq n_j D \leq \overline{B}_j, \; j \in P_i, \; i = 0,1,\ldots,J \\ t_j \leq T, \; j = 1,2,\ldots,J . \end{cases}$$

One way of solving this problem is a recursive enumeration procedure in which for every subtree with stage i as top-level the same kind of problems are solved for subtrees with top-level $j \in P_i$, thus reducing the number of levels by at least one for every subproblem (in figure 1: the *subtree* with *top-level* 3 consists of the nodes 3, 5, 6, 7, 8 and 9).

The numerical results presented in section 5 are obtained by using the program SIMLOT which is based on this enumeration procedure and solves (9).

By closer inspection of (9) one can see that the optimal solution of the problem restricted to a subtree structure with top-level i only depends on n_i and t_i. Therefore a Dynamic Programming approach might be favourable (compare also Crowston et al. (1973) and Schwarz and Schrage (1975)).

Problem (9) can be converted directly to the following DP recursion:

$$(10)\quad V_i(n_i,t_i) = \sum_{j\in P_i} \min_{\substack{\underline{B}_j \leq n_jD \leq \overline{B}_j \\ t_j \leq T}} \Big\{\frac{F_j}{n_j} + \alpha_j M_j D(t_j + \tfrac{1}{2}n_j - \tfrac{1}{2}) + \beta_j L_j D(t_j - \frac{n_jD}{2v_j} + \tfrac{1}{2}n_j - \tfrac{1}{2}) + \frac{\gamma_j(L_j+M_j)}{n_j}\sum_{k=0}^{n_j-1}(t_j+k)^2 + V_j(n_j,t_j)\Big\}\ ,\quad i = J,J-1,\ldots,0$$

with

(i) $V_j(n_j,t_j) = 0$ for all n_j and $t_j \leq T$ if $P_j = \emptyset$

(ii) $n_j = k_j n_i$ and $t_j = t_i + \frac{n_jD}{v_j}$, $i = 0,1,\ldots,J,\ j \in P_i$.

V_i is as usual in Dynamic Programming the *value function* and (10) the *Bellman recursion formula*.

We have assumed tacitly that the numbering of the stages is such that $j > i$ as soon as $j \in P_i$. This is always possible in directed tree networks.

In evaluating (10) we need only carry out $\#P_i$ minimizations of functions of *one* discrete variable k_j, $j \in P_i$, for a number of relevant combinations of n_i and t_i ($i = J,J-1,\ldots,0$).

The Dynamic Programming version (10) of problem (9) has not yet been implemented in SIMLOT.

Nevertheless, the authors believe that the DP version (10) of (9) can handle very efficiently the *large* assembly networks that arise in the environment of MSD.

In the sequel we shall make use of the following definition .

We define the *minimum network leadtime* t as follows:

$$t := \max_{j \in R} \{t_j\} \quad \text{with} \quad R := \{j | P_j = \emptyset\}.$$

5. Results

Four SIMLOT calculations have been made for MEDSYS, a really existing finished product. The result will be discussed in this section.

MEDSYS can be described as a 4-level, 26-stage assembly network (see figure 4). In contrast with figures 1 and 2 we use only *odd* numbers to indicate the different assembly stages. The reason for this will be made clear shortly.

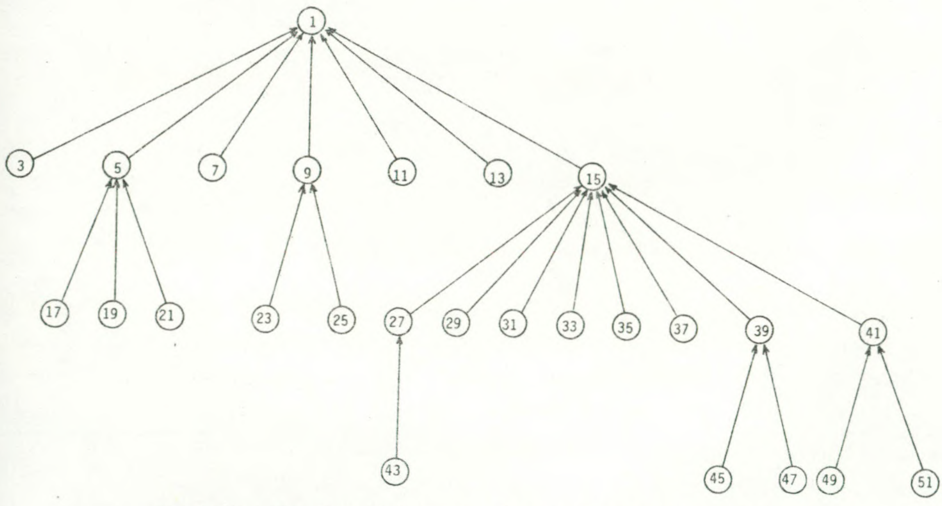

Figure 4: The MEDSYS 4-level, 26-stage assembly network

As already has been discussed earlier, in every assembly stage i labour and material are added to the operation in that stage.

The material needed for the operation in stage i is supplied by the materials store*).

---

*) The material consists of a number of socalled *codenumbers* which correspond to the different types of material needed in stage i.

This store has to deliver the different material codenumbers only in quantities necessary for assembling $n_i D$ items in stage i (i.e. with the same frequency as the set-ups in stage i).
An alternative way to organize the delivery of material to stage i consists of *disconnecting* materials supply and assembly operation, so that the material can be supplied to stage i in portions whose magnitude are an arbitrary *integer* multiple of $n_i D$.
This can be achieved by introducing a stage i + 1 where the materials needed for the operation in i can be stored temporarily (i + 1 is an *even* integer).
This stage i + 1 is connected only with stage i (see figure 5).

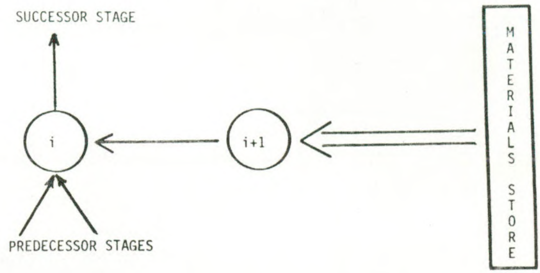

Figure 5: Material supply and assembly operation in stage i are disconnected by introducing stage i + 1 (i *odd* integer).

In such a way another J stages are introduced (J = 26).
Formally, the "even" nodes are treated in the same way as the "odd" nodes.
The difference is that only in the *odd* nodes *labour* is added whilst in the *even* nodes only *material* is supplied.
The set-up cost in the even stages can be calculated from the following relation suggested by practice:

(11) F_j = 10 * {number of *different* material codenumbers needed to carry out the operation in stage j - 1}
(j = 2,4,...,2J).

Before we discuss the four SIMLOT runs we give in Table 1 the parameters which determine the problem defined in (9) for MEDSYS (it is assumed that $T = \infty$).

The finished product MEDSYS possesses a yearly turnover of 100 items.
For the sake of simplicity we choose the length of the planning period as $\frac{250}{100} = 2{,}5$ days (we assume 250 working days per year).
As a consequence we have $D = 1$.
Furthermore we assume:

$$\alpha_i = 0{,}0018$$
$$\beta_i = 0{,}0018$$
$$\gamma_i = 0{,}0014 \qquad i = 1,2,\ldots,2J.$$

ASSEMBLY OPERATIONS					SUPPLY OF MATERIALS				
i	v_i	F_i	L_i	M_i	i	v_i	F_i	L_i	M_i
1	2.60	100.	452.	1401.	2	1000.	800.	0.	5991.
3	40.00	100.	29.	0.	4	1000.	400.	0.	161.
5	6.40	100.	185.	361.	6	1000.	610.	0.	258.
7	14.60	100.	81.	186.	8	1000.	350.	0.	224.
9	16.00	100.	74.	0.	10	1000.	540.	0.	161.
11	4.70	100.	255.	40.	12	1000.	700.	0.	1108.
13	12.30	100.	99.	0.	14	1000.	500.	0.	1001.
15	1.25	100.	1891.	0.	16	1000.	1750.	0.	3771.
17	533.00	100.	2.	0.	18	1000.	10.	0.	61.
19	18.00	100.	66.	0.	20	1000.	490.	0.	251.
21	160.00	100.	7.	0.	22	1000.	140.	0.	62.
23	26.70	100.	44.	0.	24	1000.	460.	0.	211.
25	160.00	100.	7.	0.	26	1000.	120.	0.	43.
27	6.20	100.	384.	0.	28	1000.	1180.	0.	5343.
29	2.00	100.	593.	0.	30	1000.	890.	0.	3172.
31	2.30	100.	519.	0.	32	1000.	1110.	0.	2668.
33	2.00	100.	593.	0.	34	1000.	1810.	0.	4773.
35	10.70	100.	111.	0.	36	1000.	160.	0.	960.
37	14.80	100.	81.	0.	38	1000.	40.	0.	1010.
39	5.70	100.	207.	0.	40	1000.	600.	0.	1395.
41	13.30	100.	177.	0.	42	1000.	450.	0.	360.
43	21.00	100.	56.	0.	44	1000.	100.	0.	1217.
45	6.70	100.	170.	0.	46	1000.	690.	0.	858.
47	2.60	100.	452.	0.	48	1000.	1220.	0.	2311.
49	6.50	100.	363.	0.	50	1000.	1110.	0.	988.
51	4.10	100.	586.	0.	52	1000.	1190.	0.	1230.

Table 1: The parameters of MEDSYS

Remark: The column corresponding to M_i in Table 1 (Assembly Operations) contains some values *different* from zero. The reason for this is that MEDSYS encompasses some less important assembly stages which have been neglected in figure 4 and are taken into account as *material* added in stage i but *not* from the materials store.

The following computerruns concerning MEDSYS have been carried out by solving problem (9) (with $T = \infty$) using SIMLOT:

A. The assembly batches n_iD are calculated by factory planners using their own experience ($i = 1,3,\ldots,2J-1$).
The disconnection of material supply and assembly operation has *not* been carried out, so that:

$$n_i = n_{i+1},\ i = 1,3,\ldots,2J-1.$$

In this case the minimal average total cost C_{tot} can be calculated by putting (in (9)):

$$\underline{B}_i = \overline{B}_i = n_iD,\ i = 1,2,\ldots,2J.$$

B. The assembly batches n_iD are calculated using SIMLOT ($i = 1,3,\ldots,2J-1$).
The disconnection of material supply and assembly operation has *not* been carried out or:

$$n_i = n_{i+1},\ i = 1,3,\ldots,2J-1.$$

In this case we have:

$$\underline{B}_i = 1 \text{ and } \overline{B}_i = 100,\ i = 1,2,\ldots,2J.$$

C. The assembly batches n_iD are calculated using SIMLOT ($i = 1,3,\ldots,2J-1$).
The disconnection of material supply and assembly operation has been carried out. In this case we have:

$$n_{i+1} = k_{i+1}n_i\ \text{, where } k_{i+1} \text{ is a positive integer } (i = 1,3,\ldots,2J-1).$$

Also in this case we have:

$$\underline{B}_i = 1 \text{ and } \overline{B}_i = 100,\ i = 1,2,\ldots,2J.$$

D. The only difference with alternative C concerns the set-up costs F_i, $i = 1,2,\ldots,2J$.
These costs have been *halved* in this alternative.
From the total cost related to this alternative one can obtain insight in how much can be invested in flexibility (for instance by spending on manufacturing engineering to reduce set-up costs).

Now we give, in Table 2, the results obtained by applying SIMLOT on the alternatives A, B, C and D (recall that for the alternative A the application of SIMLOT means: $\underline{B}_i = \overline{B}_i = n_iD$ where n_i has been determined by the factory planner).

Computing time for each of the alternatives A, B, C and D was moderate: alternative A took 0.2 seconds whilst alternatives B, C and D each took about 10 seconds on the AMDAHL AV8 system (VSPC-FORTRAN).

ASSEMBLY OPERATIONS				
i	A	B	C	D
1	5	5	2	1
3	100	45	20	10
5	25	20	10	7
7	50	25	18	10
9	50	40	18	10
11	10	20	12	8
13	10	20	20	9
15	5	5	2	1
17	100	40	40	28
19	100	40	40	28
21	100	40	50	35
23	100	40	36	30
25	100	40	54	40
27	5	10	6	5
29	10	10	4	4
31	10	15	6	4
33	10	15	4	3
35	10	15	14	10
37	20	10	10	8
39	10	15	6	5
41	25	20	10	7
43	10	10	12	10
45	10	30	12	10
47	10	15	6	5
49	25	20	10	7
51	25	20	10	7

SUPPLY OF MATERIALS				
i	A	B	C	D
2	5	5	12	8
4	100	45	40	30
6	25	20	40	28
8	50	25	36	30
10	50	40	54	40
12	10	20	24	16
14	10	20	20	18
16	5	5	20	10
18	100	40	40	28
20	100	40	40	28
22	100	40	50	35
24	100	40	36	30
26	100	40	54	40
28	5	10	12	10
30	10	10	16	12
32	10	15	18	16
34	10	15	20	12
36	10	15	14	10
38	20	10	10	8
40	10	15	18	15
42	25	20	30	21
44	10	10	12	10
46	10	30	24	20
48	10	15	24	15
50	25	20	30	21
52	25	20	30	21

Table 2: The quantities n_i , $i = 1, 2, 3, \ldots, 2J$ for each of the alternatives A, B, C and D.

Now the minimal average total cost per period (or, alternatively, per product as $D = 1$) can be obtained by minimizing C_{tot} in (9) with $T = \infty$. Besides this cost we give, in Table 3, the minimum network leadtime t.
These figures can be calculated for each of the alternatives A, B, C and D (see Table 3).

	A	B	C	D
t	14	14	5	4
C_{tot}	3200	2940	2214	1530

Table 3: The quantities t and C_{tot}.

6. Conclusions

Analyzing the figures of MEDSYS (Tables 1, 2 and 3) we can draw the following conclusions:

(i) The total cost of one item of MEDSYS can be reduced with 3200 - 2940 = 260, without affecting the minimum network lead time t. In this case the *actual* production organization remains unaltered.

(ii) If material supply from the store is disconnected from the assembly process a further reduction of cost can be achieved with magnitude 2940 - 2214 = 726. Furthermore, t is reduced substantially.

(iii) Reduction of all set-up costs ($F_i \to 0.5F_i$, $i = 1,2,\ldots,2J$) reduces total cost with an amount of 684 from 2214 to 1530. Again a shortening of the minimum network lead time t can be achieved. To say it in another way: for investment in reduction of set-up costs to 50% of these values an amount of 68400 per year is available.

(iv) The costprice of MEDSYS ($\sum_{i=1}^{2J} (L_i + M_i)$) amounts to 49060. This means that (comparing alternative A and C) in relation to the cost price savings of 2% can be achieved.

Concluding we can assert that:

- SIMLOT is capable to compare (with performance indicators C_{tot} and t) different organization structures for assembling a complex product.
- SIMLOT can calculate optimal batches taking into account restrictions on batches and minimum network lead time t (see (9)).
- SIMLOT can calculate the financial consequences of investments in reduction of set-up costs.

SIMLOT also offers the opportunity to obtain insight into the relation of C_{tot} and T.
Choosing the "best" combination of C_{tot} and T is in fact a *bicriterion decision problem* with which the logistics manager is faced.
All combinations of the minimal cost C_{tot} and T constitute the *efficient frontier* or *Pareto optimal set* for this decision problem. We shall not discuss this issue further but content ourselves with giving Table 4 where, for alternative C, some figures are given.

T	∞	5	4	3	2
t	5.13	4.57	3.83	2.99	1.99
C_{tot}	2214	2215	2223	2262	2562

Table 4: Some combinations of C_{tot} and T (alternative C).

Acknowledgement: The authors wish to express their gratitude to Dr. L. Fortuin for his useful remarks and J. Agterberg for programming SIMLOT.

7. References

1. Blackburn, J.D. ; Millen, R.A.:
Simultaneous lot-sizing and Capacity Planning in Multi-Stage Assembly Processes, Working Paper No. 81-130 Owen Graduate School of Management, VanderBilt University.

2. Clark, A.J. ; Scarf, H.:
Optimal policies for a multi-echelon inventory problem,
Management Science, Vol. 6, No. 4, July 1960, 475 - 490.

3. Crowston, W.B. ; Wagner, M. ; Williams, J.F.:
Economic lot-size determination in multi-stage assembly systems,
Management Science, Vol. 19, No. 5, January 1973, 517 - 527.

4. Schonberger, R.J.:
Kanban applications at Kawasaki, USA,
American Production & Inventory Control Society 1981 Conference Proceedings, pp. 188 - 191.

5. Schwarz, L.B. ; Schrage, L.:
Optimal and System myopic policies for multi-echelon production/inventory assembly systems,
Management Science, Vol. 21, No. 11, July 1975, 1285 - 1294.