LOGS: AN OPTIMIZATION SYSTEM
FOR LOGISTICS PLANNING*

by

William D. Northup
and
Jeremy F. Shapiro

OR 089-79 October 1979

*This is a revision of an earlier version dated June 1979.
ABSTRACT

This paper reports on the design, implementation and use in a large process manufacturing firm of a general purpose logistics planning system, called LOGS. A novel and essential component of LOGS is a set of basic decision elements for describing logistics models; they provide an easy to use interface between the data in the firm's information systems and mathematical programming modeling and optimization capabilities. Included in the paper are discussions of experience with the system applied to production allocation, supply contracting, analysis of an acquisition, and other logistics planning problems.
LOGS: AN OPTIMIZATION SYSTEM FOR LOGISTICS PLANNING

Introduction

Computer hardware and software have developed to the point where all large manufacturing companies have extensive information systems containing data about present and projected operations at all levels from purchasing to sales. In spite of the availability of data, logistics planning decisions remain difficult to evaluate because of the complex interactions between manufacturing levels and the geographical regions in which the company operates. Typical decisions are those about the size and location of new facilities, product mix, terms for contracts with suppliers and customers, as well as many operational decisions of production and distribution.

In recent years, a variety of mathematical programming models have been proposed to aid in logistics planning. The models are attractive because they describe a manufacturing system in its entirety, thereby facilitating a global optimization that attempts to reconcile important trade-offs between the profit contributions of different products, production and distribution costs at different manufacturing facilities, contractual arrangements with different suppliers, and so on. However, the models have not yet been widely integrated into logistics planning at the corporate level, in part because of the technical difficulties in constructing them from the relevant data and in using them in a flexible and expeditious manner.

This paper gives details about a logistics planning system, called LOGS, that has been successfully implemented in a large process manufacturing firm. LOGS analyzes a given logistics planning problem by automatically generating an appropriate mixed integer programming (MIP) model from the relevant data, and then optimizing it. This model is generated by the Logistics Modeling
Program that provides an easy to use interface between the data in the company's information systems and mathematical programming modeling and optimization capabilities. An analyst using LOGS is not required to know anything about mathematical programming modeling techniques or algorithms in order to specify and analyze a logistics planning problem. LOGS uses cost, activity and resource data about the company's distribution network, purchasing, manufacturing and sales activities plus additional management policy constraints that cut across individual facilities.

LOGS also writes reports based on the MIP solutions. Many of these reports are straight translations of the solutions to other formats although some further accounting calculations are also performed. An example of the latter are fair transfer prices reflecting accrued costs on products as they flow through the stages of the manufacturing process.

Moreover, the existence of LOGS permits and encourages the establishment, maintenance and use of common data bases in analyzing related logistics planning problems. Models for strategic, tactical and operational planning may share a description of system-wide interactions. For example, the production and transportation data used to determine monthly production plans can also be used to determine and revise quarterly budgets and production standards. The form of the models used in these two cases is identical. The differences are in the lengths of the planning horizons and the degree of certainty in sales projections.

LOGS was developed for a specific company but its applicability is much more general. It is capable of generating and optimizing mixed integer programming models for a wide range of decision problems faced by many manufacturing firms. LOGS was implemented in PL/I for interactive or batch use on IBM 370 computers. Figure 1 gives a schematic representation of the system; the ellipses correspond to data and the rectangles to programs.
The plan of this paper is the following. The next section contains a detailed discussion of the Logistics Modeling Program. A sample problem illustrating many of the modeling constructs is given in the section after that. We then discuss features of the MIP optimization performed by LOGS. The following section is devoted to our experience with LOGS applied to actual logistics planning problems. The final section contains a few concluding remarks directed mainly at future extensions. An important aspect of the use of LOGS that we will not discuss in this paper are the front end programs that feed data to LOGS in the formats accepted by the Logistics Modeling Program. The management of these information systems, as well as the implications and consequences to the company of centralized logistics planning, will be discussed in a separate paper.

Automated Logistics Modeling

The Logistics Modeling Program accepts basic decision elements describing a logistics planning problem and creates an appropriate MIP model for optimizing it. These basic decision elements are a novel and essential feature of LOGS that permits a problem to be described in natural terms by an analyst. The exact specification of the basic decision elements resulted from extensive discussions with analysts experienced in the use of logistics planning models in the company and elsewhere. As a result, LOGS has the capability to create MIP models describing a wide variety of logistics planning problems.

The generic logistics planning problem that can be analyzed by LOGS is comprised of three components. First, there is a distribution network that describes product flows at all levels from the suppliers of raw materials through intermediate stages of manufacturing and warehousing to the
Figure 1

PURCHASING, MANUFACTURING, SALES ACTIVITIES
AND
FAIR TRANSFER PRICES ON PRODUCTS
markets. Superimposed on the distribution network are production activities that describe how the products are bought, transformed, stored and sold. Finally, there are logical constraints cutting across the network that link and restrict production activities. An example is a constraint stating that a particular product can be produced in at most three plants.

Network terminology is used extensively by the Logistics Modeling Program in the description of logistics planning problems. However, the general class of models that can be created by the program includes many that are not pure network optimization models because of the presence of non-convex cost functions and constraints that are not easily represented as flow balance constraints. The generic model is defined over a distribution network consisting of nodes and arcs \((i,j;p)\) describing directed connections between nodes \(i\) and \(j\) by product type \(p\). The nodes in the distribution network represent facilities considered in a broad sense to include plants, machines, suppliers, warehouses and markets. Products include raw materials, work in process and finished goods to be sold on the market. Products are converted at nodes by processes. Let \(\mathcal{A}\) denote the set of arcs and let \(x_{ij}^p\) denote the variable flow of product \(p\) in the arc \((i,j;p)\). Let \(y_q^i\) denote the variable production activity for process \(q\) at node \(i\).
As shown in Figure 2, each node $i$ in the distribution network can be expanded to a set of nodes $Q_i$ corresponding to the processes at $i$. Arcs for flows within a node and their associated flow conservation constraints are generated automatically as needed. For simplicity, these arcs will be considered elements of $Q$ in the following discussion.

Processes include conversion of input product to output products at intermediate manufacturing nodes, conversion of finishing product to income at the market nodes, and conversion of costs to raw materials at the supply nodes. Five basic decision elements can be used to define and constrain these process activities and to describe their costs:
For expositional reasons, we will discuss these relationships using standard mathematical programming terminology but we wish to emphasize again that the analyst does not describe a logistics planning problem in this way. The following section gives a sample logistics planning problem and its description in terms of its basic decision elements.

The node activity is the product flow or combination of product flows and process activities that are used to compute costs at the node. If there is no explicit specification of node activity, material balance, recipe or product conservation, then by default the node activity is the variable $y_i^0$ given by

$$y_i^0 = \sum_{(j,i;p) \in Q} x_{ji}^p = \sum_{(i,k;p) \in Q} x_{ik}^p$$

Process 0 (and other processes specified for a node) can be used to specify a general cost function of the type shown in Figure 3. The equality between total flow into node $i$ and out of $i$ indicates that the default option also includes an implicit material balance equation for the node.
The explicit (non-default) specification of node activity (NA) can be used to define a new activity for node i

\[ y_i^r = \sum_{(j,i;p) \in Q} w^p x_{ji} + \sum_{q \in Q_1} x_{qi}^q + \sum_{(i,k;p) \in Q} v^p x_{ik} \]

where the weights \( w^p, z^q, v^p \) are specified by the analyst. The variable \( y_i^r \) can be used to compute costs as in Figure 3 and upper and lower bounds on it.
can be specified. Note that this broad definition of node activities permits the accumulation of costs incurred at several levels of a physical facility; e.g., indirect plant costs, indirect machine costs and direct costs for specific processes.

Alternatively, the Logistics Modeling Program accepts the specification of a material balance (MB) equation

$$\sum (i,k;\epsilon)_{\epsilon \in A} v_{ik}^p = \sum (j,i;\epsilon)_{\epsilon \in A} w_{ji}^p$$

When the material balance option is used for process \( q \), the weighted sum of inputs (equal to the weighted sum of outputs) is automatically used as the node activity \( y^q_i \).

The recipe option (RP) is used to describe process \( q \) whereby input products are converted to output products at node \( i \). Let \( s_1, \ldots, s_\alpha \) denote a set of input products and \( t_1, \ldots, t_\beta \) denote a set of output products. One product is a reference product; we assume this to be \( s_1 \). A recipe is given by specifying the quantities \( a_{1l}, l = 2, \ldots, \alpha \) and \( b_{im}, m = 1, \ldots, \beta \), thereby implying the equation

$$a_{1l} \sum (j,i;\epsilon)_{\epsilon \in A} x_{ji}^1 = \sum (j,i;\epsilon)_{\epsilon \in A} x_{ji}^l \text{ for } l = 2, \ldots, \alpha$$

$$b_{im} \sum (i,k;\epsilon)_{\epsilon \in A} x_{ik}^1 = \sum (i,k;\epsilon)_{\epsilon \in A} x_{ik}^m \text{ for } m = 1, \ldots, \beta.$$

Moreover, we define the variable \( y^q_i = \sum (j,i;\epsilon)_{\epsilon \in A} x_{ji}^1 \).
The fixed ratio (FR) option constrains flows of different products into or out of a node, or processes at a node. This option is needed to model, for instance, constraints imposed by outside suppliers. The constraints have the form

\[ \sum_{(i,k,s) \in A_{ik}} x_{ik}^s = r_{st}^i (\sum_{(i,k,t) \in A_{ik}} x_{ik}^t) \]

The final option, product conservation, permits the specification of individual constraints by product of the form

\[ \sum_{(i,s) \in A_j} x_{ji}^s = \sum_{(i,k) \in A_{ik}} x_{ik}^t \]

Upper and lower bounds can be imposed on these sums. Product conservation is used to model, for example, the flows of individual products through a warehouse.

Additional information required to create a logistics planning problem is a specification of unit costs, upper and lower bounds for each arc \((i,j;p) \in A\). The current implementation of LOGS does not permit nonlinear costs on arc flows. These can be modeled by the creation of dummy nodes, one for each arc for which the costs are nonlinear.

The final type of input is a specification of side constraints that cut across nodes such as multiple choice, logical implication and logical
equivalence constraints. These constraints are optional and need not be
given for a particular logistics planning problem.

The Logistics Modeling Program reads the nodes, arcs and logical
constraints files discussed above and creates an MIP model file to repre-
sent it in standard MPS format. The translation is done one node at a
time for which the required variables and constraints are defined as indi-
cated by the basic decision elements. Then variables corresponding to
specified arcs are created. The rules used to generate the MIP models
are straightforward although sometimes complicated. A significant effort
was made to find effective and compact model representations; for example,
a minimal number of zero-one variables are used to represent a non-convex
cost curve such as the one shown in Figure 3. More importantly, if the
MIP problem implied by the input files is too large for efficient optimiza-
tion, then a second, smaller MIP problem is created as a surrogate to be
used in optimizing the true one. More details on how this is done are
given in the section on MIP optimization.

Sample Problem

Figure 4 shows a logistics planning problem that we use for illus-
trative purposes. The SP nodes are the suppliers of the input products AA,
BB, CC. These are converted at the mill node MIL to the finished products
DD and EE which are transshipped through the warehouse WAR to be sold at
the markets MR1 and MR2.
SAMPLE LOGISTICS PLANNING PROBLEM

Figure 4
Table 1 is a listing of the input files (nodes and arcs) for a logistics planning problem defined over the network of Figure 4. The first line of the nodes file gives the costs of supply of AA and BB from SP1. In addition, the FR specification states that AA and BB must be bought in equal quantities. Figure 5 depicts the implied cost function for SP1.

![Fixed Product Ratio](image)

Fixed Product Ratio
AA = BB

Cost Function - Supplier 1

Figure 5

The cost function for SP2 uses flow of BB as the node activity. It consists of a fixed cost of 3.75 with a unit cost of 1.625 up to 10 units followed by a unit cost of 1.875 up to the upper limit of 20 units.

There are two processes specified at MIL denoted by $P1 and $P2. These
Listing of Input Files - Nodes and Arcs - for Logistics Planning Problem

Table I

<table>
<thead>
<tr>
<th>FILE: DUBOIS NODES</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPL 1  1.278</td>
</tr>
<tr>
<td>FR 1  &lt;AA</td>
</tr>
<tr>
<td>SP2  1.875</td>
</tr>
<tr>
<td>SP3  2.35</td>
</tr>
<tr>
<td>SP4  2.48</td>
</tr>
<tr>
<td>MIL  0.33</td>
</tr>
<tr>
<td>NA  &lt;SP1  1</td>
</tr>
<tr>
<td>RP  AAA</td>
</tr>
<tr>
<td>RP  &gt;BB</td>
</tr>
<tr>
<td>WAR  0.05</td>
</tr>
<tr>
<td>NA  &lt;DD</td>
</tr>
<tr>
<td>PC  &lt;DD</td>
</tr>
<tr>
<td>EE</td>
</tr>
</tbody>
</table>

| MRL  > |
| SDD  >DD | -193.8 | 50 |
| = 17 | -175.7 | 28 |
| SEE  >EE | -223.8 | 29 |
| = 20 | -190.0 |

| MRL  > |
| SDD  >DD | -192.2 | 40 |
| SEE  >EE | -235.4 | 27 |

<table>
<thead>
<tr>
<th>FILE: DUBOIS ARCS</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPL AA  MIL  1.1</td>
</tr>
<tr>
<td>SPL BB  MIL  1.23</td>
</tr>
<tr>
<td>SP2 BB  MIL  2.03</td>
</tr>
<tr>
<td>SP2 CC  MIL  2.11</td>
</tr>
<tr>
<td>SP3 AA  MIL  1.79</td>
</tr>
<tr>
<td>SP4 BB  MIL  2.67</td>
</tr>
<tr>
<td>MIL DD  WAR  0.0</td>
</tr>
<tr>
<td>MIL EE  WAR  0.0</td>
</tr>
<tr>
<td>WAR DD  MRL  3.42</td>
</tr>
<tr>
<td>WAR EE  MRL  3.57</td>
</tr>
<tr>
<td>WAR EE  MR2  2.86</td>
</tr>
<tr>
<td>WAR DD  MR2  2.80</td>
</tr>
</tbody>
</table>
are depicted in Figures 6 and 7. The node activity at the MIL level is equal to \( 1 \times \text{node activity of } P1 + 1.2 \times \text{node activity of } P2 = 1AA + 1.2BB \). Thus, indirect costs at the MIL level are given by 0.33 (1AA + 1.2BB).

The income (negative cost) functions at the market nodes are given in the final lines of the nodes file. The functions consist of a number of pieces and for MRL, there are shut-down costs of 38.7 and 47.3 for not supplying DD and EE, respectively, at levels equal to the minimal specifications of 4 and 2 units. The arcs file gives the unit transportation costs for flows in the indicated arcs; e.g., 1.1 for each unit of AA sent from SPL to MIL.

The Logistics Modeling Program created a mixed integer programming problem from the input files that consisted of 32 rows, 9 zero-one variables and 35 continuous variables. MPSX/370 was used to optimize the problem and also to locate alternative optima within 50% of the maximal profit attainable.
PROCESS 1:

RecipE: Inputs 1.0 AA, 0.8 BB, 1.1 C
Outputs 0.4 DD, 0.33 EE

Figure 6

PROCESS 2:

RecipE: Inputs 1.05 AA, 1.0 BB, 1.17 CC
Outputs 0.37 DD, 0.41 EE

Figure 7
Mixed Integer Programming Optimization

LOGS uses the IBM system MPSX/370 to perform the mixed integer programming (MIP) optimization on the problems created by the Logistics Modeling Program. Alternatively, any mathematical programming system for 370 computers that accepts problems in MPS format could be used. It is important to emphasize that our experience thus far indicates that many MIP problems can be predictably and dependably solved if the number of integer variables is not too large (less than 150) and if the number of constraints is not too large (less than 2000). Detailed results are discussed in the following section. This is somewhat contrary to the widely held feeling that MIP computation is almost always extremely difficult.

Nevertheless, MIP problems with many zero-one variables, or those with several thousand rows, can be implicitly proposed by users of LOGS with large logistics planning problems to be analyzed. For this reason, LOGS will automatically aggregate an MIP problem if it is judged to be too large. The analyst can also tell LOGS to aggregate the problem if desired. The criterion for automatic aggregation is the anticipated size of the linear programming subproblems to be analyzed during the branch and bound stage of MPSX/370. Aggregation is desirable only if the resulting linear programming subproblems are sufficiently reduced in size to more than compensate for the additional overhead. The aggregated MIP problem is used as a surrogate to optimize the original MIP problem.

LOGS aggregates large MIP models by reducing the number of distinct market-product locations. A typical warehouse distribution problem could have 150 markets to which 40 products are distributed. This would contribute 6000 rows to the model, and it is usually possible to reduce this
number to 1000 by aggregation without introducing great approximation errors. Specifically, nodes are distributed into aggregate nodes on the basis of similar transportation costs from plants and warehouses to the markets, and products are clustered into aggregate products on the basis of similar market sales functions. Unit costs for arcs in the aggregate network are always selected to understate the true cost relative to the original logistics planning problem. Bounds on aggregate arc flows reflect bounds on market demand. Thus, the aggregation is always constructed so that it provides lower bounds on the minimal cost in the original MIP problem. The aggregate MIP problem is optimized using MPSX/370 and each feasible solution is disaggregated to generate a solution to the original MIP problem. The reader is referred to Geoffrion [1] and Zipkin [4] for similar approaches to aggregating logistics planning problems.

Experience with the System

LOGS is fully implemented and being used in a batch mode by a large process manufacturing firm to study a variety of logistics planning problems. Table 2 gives a few representative results. The purchasing problems 1 and 2 address company strategy in making contracts with suppliers of primary inputs to production. A minimal cost strategy is difficult to find because of complicated supply cost functions involving fixed costs and price breaks for high volume purchases. In addition, some suppliers impose fixed ratio (FR) restrictions on supplies due to the nature of their own processes for making these supplies. It is well worth noting that the optimal solution to problem 1 was $4 million lower than the cost of the
<table>
<thead>
<tr>
<th>Problem Number</th>
<th>Description</th>
<th>Rows</th>
<th>Zero-One Variables</th>
<th>Continuous Variables</th>
<th>Seconds to Generate</th>
<th>Seconds to Optimize (MPSX/370)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Purchasing - Item one</td>
<td>290</td>
<td>40</td>
<td>340</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>Purchasing - Item two</td>
<td>330</td>
<td>38</td>
<td>600</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>Monthly Production Planning</td>
<td>2000</td>
<td>0</td>
<td>8000</td>
<td>53</td>
<td>288</td>
</tr>
<tr>
<td>4a</td>
<td>Monthly Production Planning</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>With Fixed Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Unaggregated</td>
<td>2150</td>
<td>70</td>
<td>10400</td>
<td>56</td>
<td>540</td>
</tr>
<tr>
<td>4b</td>
<td>Aggregated</td>
<td>1410</td>
<td>70</td>
<td>7000</td>
<td>110</td>
<td>925</td>
</tr>
</tbody>
</table>

Table 2

Times are on an IBM 370/168
proposed strategy obtained prior to the use of LOGS. This large savings
represented real dollars to be saved by the company in its purchases for
the coming year. Moreover, the Purchasing Department was able to use the
results of the model to negotiate better terms with some of the suppliers.

The monthly production planning models are aggregate models used to
assign forecasted demand for all products to specific plants and machines
across the U.S. Problem 3 is a linear programming (LP) version of the
problem that had previously been generated and solved by a commercially
available matrix generation language and LP optimizer with a generalized
upper bound (GUB) option. This run was made to validate LOGS and to com-
pare its performance with the other system. We found that MPSX/370 without
the GUB option was able to find an optimal solution to the model in approxi-
mately the same length of time as the LP optimizer with GUB despite the
fact that 80% of the rows were GUB rows. This relative efficiency of
MPSX/370 without GUB is not sustained, however, for larger GUB problems.
We were able to overcome this deficiency of MPSX/370 by procedures for choos-
ing better starting bases and also by putting into the objective function
those constraints that are merely accounting balances. In general, LOGS
generates these LP and MIP models at least 10 times faster than the generator
written in the matrix generation language.

Problem 4 addresses the same company-wide production planning problem
as problem 3, but with fixed costs and minimal activity levels on certain
production activities. These fixed costs contribute significantly to total
production costs. More generally, it is hoped that the richer MIP models
generated and optimized by LOGS will result in a better reconciliation of
the company's medium term (monthly) production planning and short term (weekly)
production planning at the plant- and machine levels. MIP modeling is also needed for certain logical constraints such as one permitting production of product B at a certain plant only if at least a minimal amount of product A is made.

The aggregated version of problem 4 was run for experimental purposes. The added time to generate the aggregated MIP model is a certainty because LOGS generates it after generating the unaggregated version. The longer time to optimize indicates either that aggregation is not attractive for MIP models of the stated size or that more work is required to efficiently use the aggregation to solve the unaggregated model. However, an optimal solution to the aggregated MIP model easily provided an optimal solution to the unaggregated model.

At the time of this writing, LOGS has been in operational use in the firm for almost a year. During that time, it has been used to generate and optimize hundreds of LP and MIP models in the above applications areas and several others. One application establishes budgeting standards and variances by use of quarterly versions of the monthly production planning models. Another application was an analysis of the impact on the company of a proposed acquisition. Other applications include process design, cost/service tradeoff analysis and plant layout. There has naturally been some evolution of LOGS as the result of this experience with it. The evolution is exclusively in the optimization program and in the corresponding model representations. There have been no changes needed thus far in LOGS' modeling capability.

Conclusions

We have discussed in some detail the design, implementation and use of
LOGS within a single company. We feel its success thus far in over a half-dozen applications indicates its potential value for other logistics planning problems. Some new applications currently being contemplated would require extensions of the descriptive capability of the basic decision elements. For example, the use of LOGS to analyze multi-period distribution models would require the addition of a time dimension and implied inventory balance equations.

The efficiency and ease of use of LOGS is derived in part from the fact that it addresses a specific class of MIP models, rather than the general class of models addressed by languages for matrix generators. There are classes arising in other applications that would appear to be good candidates for the same approach. One such class includes models of U. S. coal supply and demand markets. In their full form, these are multi-period, multi-regional LP and MIP models that describe the logistics of coal extraction, transportation and use in generating electricity (see Shapiro and White [3]).

Although the general purpose code MPSX/370 has generally performed well, extensions to the optimization programs of LOGS may be worthwhile. Included are algorithms and methods for exploiting GUB and pure network structures (see Glover et al [2]). Special structures such as these are particularly easy for LOGS to identify from the basic decision elements describing a logistics planning model.
References


