# OPTIMAL DESIGN
# OF
# MERGE-IN-TRANSIT DISTRIBUTION NETWORKS

**Michael H. Cole, Ph.D.**
**Assistant Professor**

**Mukundh Parthasarathy**
**M.S.I.E. Graduate Student**

**Department of Industrial Engineering**
**University of Arkansas**
**Fayetteville, Arkansas**

**July 1998**

# TABLE OF CONTENTS

## TABLE OF FIGURES

## TABLE OF TABLES

# CHAPTER 1 – INTRODUCTION

Merge-in-transit operations work as follows: carriers (e.g., trucks) pick up separate loads from two or more sources (e.g., ports, factories, or distribution centers) and transport the loads to a location near their final destination where a "merge" operation is performed. At the simplest, the merge operation comprises consolidating the loads in a cross-docking operation. In more complex systems, the merge operation includes a value-added step such as assembly. A key advantage of merge-in-transit networks over traditional distribution networks is that merge points can often be opened or closed quickly and cheaply, depending on the type of merge operation being performed. In general, customer service will be improved by having merge points that are closer than existing warehouses. Merge-in-transit is especially useful to companies expanding into new geographical regions that are not well served by their existing distribution network

This research has two objectives. One, to extend existing production models to account for the merge-in-transit environment. Two, to develop a user-friendly GIS (geographic information system) based decision support system for designing merge-in-transit networks.

The modeling portion of this research involves developing a simple mathematical model for designing distribution networks that enables consideration of merge-in-transit costs simultaneously with production, warehousing, and

**inventory costs. Outputs of the model include number, location and type of merge points, selection of transportation channels, allocation of customers/retailers to merge points,**

**and disposition of inventories. This model is a straightforward extension of existing mathematical models**

**The user interface portion of this research involves prototyping the model in a user-friendly Microsoft Windows ™ -based PC environment so that the model will be accessible to industry practitioners. The prototype decision support system enables rapid "what-if" analyses.**

**The research is useful to both carriers and shippers. Carriers (and third party logistics providers) can use the model to rapidly design their own merge-in-transit networks and to show potential customers the benefits of partnering. Shippers can use the model to consider alternatives to their traditional distribution networks.**

# CHAPTER 2 – LITERATURE REVIEW

This chapter reviews selected literature related to merge-in-transit distribution. The objective is to detail selected recent and significant material, not to provide an exhaustive review. Section 2.1 details relevant logistics network models. Section 2.2 covers decision support systems (DSS) for logistics network design.

## *2.1. Logistics Network Models*

Williams (1990) and Fourer, et al (1993) provide good overviews of basic production and distribution models. Vidal and Goetschalckx (1997) describe more recent logistics network models. Cole (1995) provides extensive coverage on an integrated logistics model that simultaneously considers inventory, transportation, and facility costs.

Glover and Klingman (1977) and Glover et al (1978) are seminal papers on the practical application of generalized networks. Generalized networks differ from pure networks in that they permit commodity transformations on arcs and at nodes. For instance, a group of commodities might be transformed into a different commodity at a node representing an assembly point. A merge-in-transit network is an example of a generalized network.

Garg and Tang (1997) and Lee and Tang (1997) discuss the use of delayed product differentiation to provide relatively inexpensive service to markets that demand high levels of product variety. Their analysis concentrates on inventory

**levels in a multi-level production process. One of their key findings is that delayed product differentiation can lead to reduced buffer inventories. For further research they suggest developing models for studying the relationship between the benefits of delayed product differentiation and the costs of distribution.**

**Van Hoek and Commandeur (1998) describe case study research on the use of postponement strategies that enable "mass customization" of products. They study a number of European and American companies. They conclude that postponement strategies are an important means of improving customer service, but that the anticipated benefits can be hard to obtain. Achieving the benefits requires that management have a firm vision of the future and be committed to process reengineering. As a supplement to their case study research, they recommend development of "calculation models" for cost benefit analysis.**

**Slats, et al, (1995) cautions that logistics network analysis that starts with the mathematical model can be inadequate. Instead, they suggest starting the analysis with a view to data availability. They also stress the need to be able to rapidly develop new models and extend old ones when working with real-world logistics systems.**

### *2.2. Decision Support Systems for Logistics Network Design*

Goeschalckx, et al, (1994) discuss the development of a graphical-based computer system for designing logistics networks. They developed their own simple GIS (geographic information system) and used a flat file database. ESRI provides MapObjects™, a simple geographic information system that can be easily integrated in a user-defined Microsoft Visual Basic™ program. GIS programs such as this are significant in that they enable researchers to develop quality prototypes rapidly and inexpensively.

Fourer, et al (1993) describe AMPL, an algebraic mathematical modeling language. AMPL enables the rapid development and debugging of models that can be easily hooked to commercial database programs. Such capabilities are very important since the design of decision support system is an ongoing process subject to revisions dictated by users.

## CHAPTER 3 – MATHEMATICAL MODEL

This chapter details a mathematical model for designing optimal merge-in-transit distribution networks. Section 3.1 lists some basic assumptions. Section 3.2 covers the mathematical notation. Section 3.3 presents the mathematical formulation in traditional form. Section 3.4 discusses computation issues.

### *3.1. Assumptions*

The model is an uncapacitated multicommodity generalized network. There are two classes of products: inputs and outputs. The nodes represent three types of nodes: sources, customers, and merge points. The set of potential sources, potential merge points, and customers is known. Sources supply input products. Merge points assemble input products into output products. Customers consume output products.

### *3.2. Notation*

Sets and Indexes

C – set of customers

I, J, K – sets of nodes (e.g., ports, factories, warehouses, customers)

M – set of potential merge points

S – set of sources

IP – set of input products

**OP – set of output products**

**Parameters**

$r_{p\hat{p}}$ **- units of input product p required to produce one unit of output product** $\hat{p}$

$A_{jp}$ **- cost to produce one unit of output product p at node j**

$C_{ijp}$ **- cost to transport one unit of product p from node i to node j (**$p \in IP \cup OP$**)**

$D_{cp}$ **– demand for output product p at customer c**

$S_{jp}$ **– supply of input product p at node j (**$j \in S$**)**

$OpenCost_j$ **– cost to open facility j (**$j \in S \cup M$**)**

$CloseCost_j$ **– cost to close facility j (**$j \in S \cup M$**)**

**Independent (Decision) Variables**

$x_{ijp}$ **- units of product p shipped from i to j (**$p \in IP \cup OP$**)**

$z_j$ **- 1 if facility j is open, 0 if closed (**$j \in S \cup M$**)**

**Dependent Variables**

$w_{jp}$ **- units of input product p consumed at node j (**$j \in M$**)**

$y_{jp}$ **- units of output product p assembled at node j (**$j \in M$**)**

### *3.3. Mathematical Formulation*

**Minimize:**

$$\begin{aligned}&\sum_i \sum_j \sum_p C_{ijp} x_{ijp} + \\ &\sum_j \sum_{p \in OP} A_{jp} y_{jp} + \\ &\sum_j OpenCost_j z_j + \sum_{\mathrm{j}} CloseCost_j (1 - z_j)\end{aligned} \qquad (1)$$

**Subject to:**

$$y_{jp} = \sum_k x_{jkp} - \sum_i x_{ijp} \qquad \forall \quad j \in M, p \in OP \qquad (2)$$

$$w_{jp} = \sum_i x_{ijp} - \sum_k x_{jkp} \qquad \forall \quad j \in M, p \in IP \qquad (3)$$

$$w_{jp} \geq \sum_{\hat{p} \in OP} r_{p\hat{p}} \, y_{j\hat{p}} \qquad \forall \quad j \in M, p \in IP \qquad (4)$$

$$\sum_j x_{ijp} \leq S_{ip} z_i \qquad \forall \quad i \in S, p \in IP \qquad (5)$$

$$\sum_i x_{ijp} = D_{jp} \qquad \forall \quad j \in C, p \in OP \qquad (6)$$

$$x_{ijp} \leq D_{jp} z_i \qquad \forall \quad i \in M, j \in C, p \in OP \qquad (7)$$

$$x_{ijp} \geq 0 \qquad \forall \quad i, j, p \qquad (8)$$

$$y_{jp} \geq 0 \qquad \forall \quad j \in M, p \in OP \qquad (9)$$

$$w_{jp} \geq 0 \qquad \forall \quad j \in M, p \in IP \qquad (10)$$

**Equation 1 is the objective function which comprises transportation costs, assembly costs, and facility (opening and closing) costs. The transportation cost on a channel is determined by multiplying the total weight shipped by a cost factor. There are no quantity discounts. The assembly cost at a merge point is determined**

**by multiplying the unit assembly cost by the amount of output product assembled. The fixed facility cost comprises two elements: cost to close an existing open facility and cost to open a new (closed) facility. Note that the previous facility state (opened or closed) is not embedded in the model; it must be accounted for in the particular data on a case by case basis. For instance, if a particular merge point is currently closed, its closing cost would probably be set to zero.**

**Equation 2 determines the amount of each output product assembled at a merge point by subtracting the amount that enters a node from the amount that leaves the node. If merge points were allowed to consume or supply (rather than assemble) output products then this constraint would need to be modified. If the assembly process generates substantial scrap, then the related assembly cost in the objective function would need to be modified.**

**Equation 3 determines the amount of each input product consumed at a merge point by subtracting the amount leaving the node from the amount entering the node.**

**Equation 4 regulates assembly flow conservation. According to this constraint, the amount of input products consumed at a merge point limits the amount of output products assembled.**

**Equation 5 ensures supply capacity is not exceeded and that supplies come only from open sources. Equation 6 ensures that customer demand is satisfied.**

**Equation 7 states that only open merge points can ship to customers.**

**Equation 8 prevents negative flows. Equation 9 prevents disassembly. Equation 10 prevents reverse consumption.**

### *3.4. Computation*

**The time required to solve an uncapacitated problem is often directly related to the problem size. If we let each set name stand for the cardinality of that set, then the problem size is approximately:**

**Number of variables: number of channel-product combinations**

**Number of constraints: OP * (M + C) + IP * (2M + S).**

**Several test cases were run in which the model was programmed using the AMPL modeling language and solved via XLSOL. Appendix B gives a representation of the model in the AMPL modeling language. Appendix C shows a sample AMPL data file. In general, solution times were less than a few minutes. For capacitated problems, solution times can be expected to be longer and much more variable. This is an area for future researchers to examine.**

# CHAPTER 4 – DECISION SUPPORT SYSTEM

**Figure 1 shows the conceptual framework for the DSS (decision support system). The user interacts with the GIS (geographic information system) which functions as a graphical user interface. In the prototype software the GIS is ESRI MapObjects™, an add-on to Microsoft Visual Basic™. The GIS provides an interface to the database, in this case Microsoft Access™. When the data are organized, the GIS interfaces to the math model, in the form of AMPL™. AMPL™ in turn calls the solver, XLSOL™ in the prototype.**

**Figure 1. Conceptual Framework for Decision Support System**

## *4.1. Geographic Information System*

**The GIS program is based on moView, a sample application program that is included free with ESRI MapObjects. Figure 2 shows the main screen. The user is able to click on a city (or pair of cities for channels) to input or edit data. Communication with AMPL is effected via the menu.**

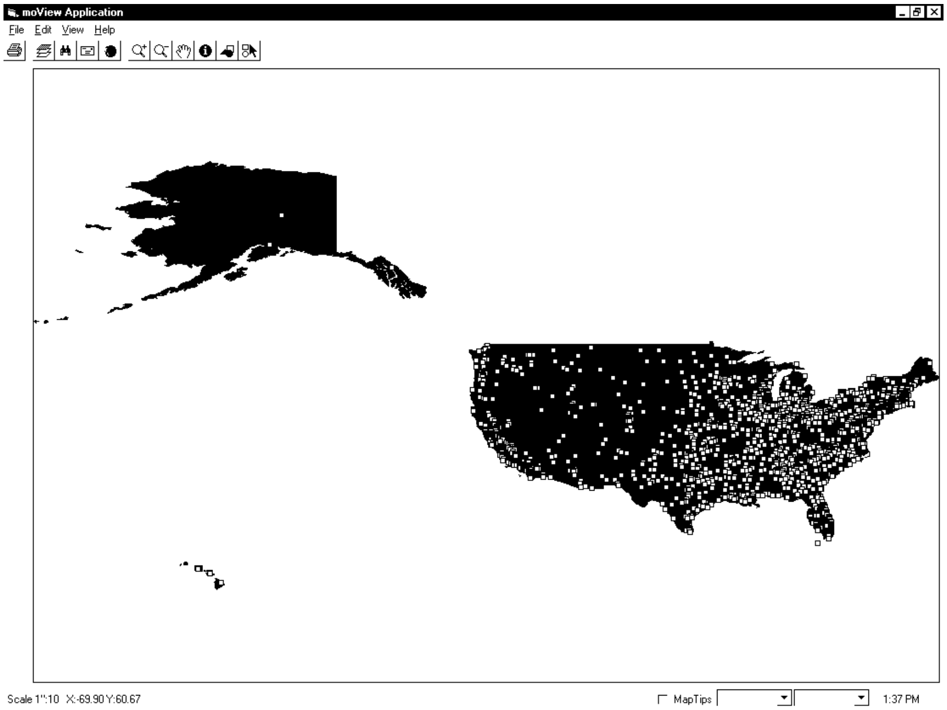
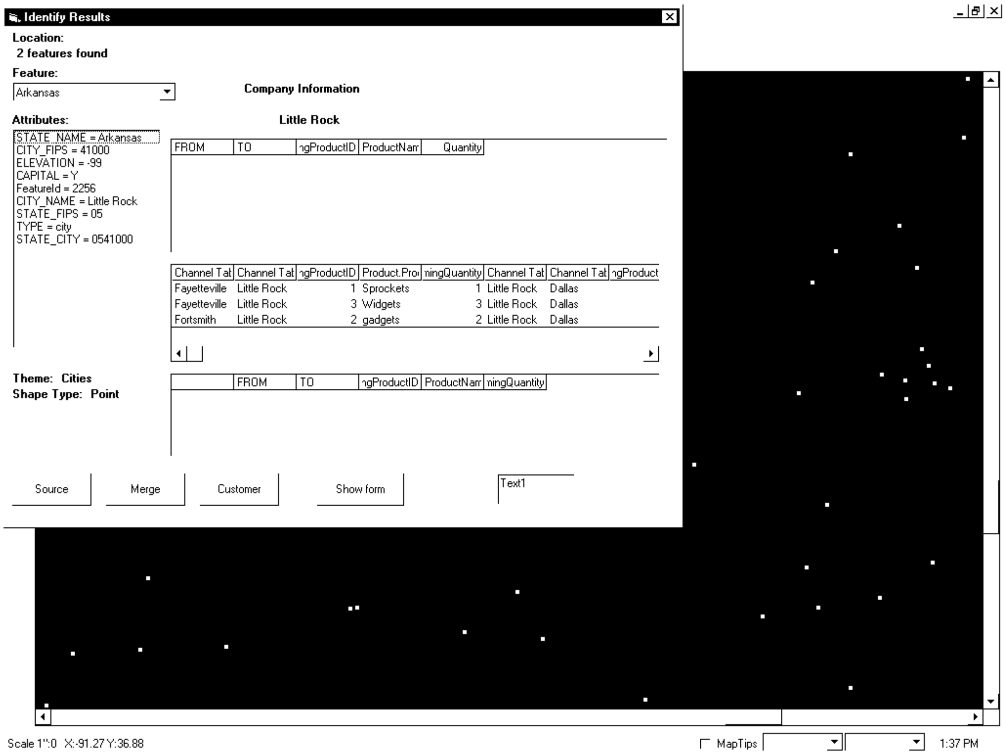


**Figure 2. GIS Main Screen**

**Figure 3 shows the dialog box that pops up when the user clicks on a city. The dialog box enables the user to query the database and edit the database with respect to the node-related data tables. Similar dialog boxes are used to edit the channel and product related data tables.**



Location:
2 features found

Feature: Arkansas

Company Information

Attributes:

STATE_NAME = Arkansas
CITY_FIPS = 41000
ELEVATION = -99
CAPITAL = Y
FeatureId = 2256
CITY_NAME = Little Rock
STATE_FIPS = 05
TYPE = city
STATE_CITY = 0541000

Little Rock

| FROM | TO | ngProductID | ProductNam | Quantity |
|---|---|---|---|---|

| Channel Tat | Channel Tat | ngProductID | Product.Pro | ningQuantity | Channel Tat | Channel Tat | ngProduct |
|---|---|---|---|---|---|---|---|
| Fayetteville | Little Rock | 1 | Sprockets | 1 | Little Rock | Dallas | |
| Fayetteville | Little Rock | 3 | Widgets | 3 | Little Rock | Dallas | |
| Fortsmith | Little Rock | 2 | gadgets | 2 | Little Rock | Dallas | |

Theme: Cities
Shape Type: Point

| | FROM | TO | ngProductID | ProductNam | ningQuantity |
|---|---|---|---|---|---|

Scale 1":0 X:-91.27 Y:36.88

1:37 PM

Figure 3. Node Dialog Box

## *4.2. Database*

**The database should be structured in the following tables: Node, Product, Merge Formula, Channel, Node-Product, and Channel-Product. In general, each table is keyed to one or more names. For instance, the Node table is keyed to unique node names and the Channel-Product table is keyed to unique channel names and unique product names.**

**Table 1 lists the fields of the Node data table. The Status is the only output field; the other fields are inputs. The fields for Fixed Cost have no meaning for customer**

nodes.

Table 1. Node Data Table

Name
Type (source, merge point, customer)
Location (zip code)
Fixed Cost if Open
Fixed Cost if Closed
Status (open or closed)

Table 2 lists the fields of the Product data table. The part weight is used to calculate transportation costs. All of the fields are inputs.

Table 2. Product Data Table

Name
Type (input or output)
Weight

Table 3 lists the fields of the Merge Formula data table. For each output product, there is a record for each associated input product. The field Number of Inputs tells how many units of the input product are needed to assemble one unit of the output product. In general, each output product requires several different input products. All of the fields are inputs.

Table 3. Merge Formula Data Table

Output Product Name
Input Product Name
Number of Inputs per Output

Table 4 lists the fields of the Channel data table. All of the fields are inputs.

Table 4. Channel Data Table

Name
Origin node
Destination node
Cost (per weight)

Table 5 lists the fields in the Node-Product data table. All of the elements in this table are inputs. The Cost and Amount fields require some explanation. For source nodes, Cost is simply the cost and Amount is the supply capacity. For customer nodes, Cost could represent the revenue and Amount is the demand. For merge nodes, the Cost and Amount fields have no meaning for merge nodes.

Table 5. Node-Product Data Table

Node Name
Product Name
Cost ($ per product)
Amount (supply or demand)

**Table 6 lists the fields in the Channel-Product data table. The Amount is the amount of product that flows on the channel; it is an output of the model.**

**Table 6. Channel-Product Data Table**

**Channel Name**
**Product Name**
**Amount**

**An earlier version of the database structure is described an appendix. The prototype software currently still uses that obsolete version of the database.**

### *4.3. Solution Package*

**The solution package used in the prototype comprised AMPL+, a mathematical modeling language, and XLSOL, a solver.**

# CHAPTER 5 – CONCLUSIONS

This research developed a simple mathematical model and decision support system for designing merge-in-transit distribution networks. Such networks are increasingly important in today's competitive business environment as companies invest in efforts to provide good customer service rather than invest in conventional warehouse-dominated distribution networks.

Future research should include a full parametric analysis of the model to determine how sensitive solutions are to the various parameters. Future research should concentrate on extending the model to account for capacities, additional costs, and different cost structures (e.g., economies-of-scale). Future research should also consider modeling the system dynamically using simulation or multi-period network models. Researchers are advised to keep their models parsimonious since data collection can be a real challenge.

# REFERENCES

**Bramel, J. and Simchi-Levi, D. (1997) *The Logic of Logistics*, Springer-Verlag, New York, New York.**

**Cole, M.H. (1995) *Service Considerations and the Design of Strategic Distribution Systems*, Ph.D. Dissertation (Technical Report MHRC-TD-96-02), Georgia Institute of Technology, Atlanta, Georgia.**

**Daganzo, C.F. (1991) *Logistics Systems Analysis*, Springer-Verlag, Berlin, Germany.**

**ESRI (Environmental Systems Research Systems Institute). (1996) *Building Applications with MapObjects*™.**

**Fourer, D., Gay, D.M., and Kernighan, B.W. (1993) *AMPL: A Modeling Language for Mathematical Programming*, Boyd & Fraser, Danvers, Massachusetts.**

**Garg, A., and Tang, C.S. (1997) On Postponement Strategies for Product Families with Multiple Points of Differentiation, *IIE Transactions*, 29, 641-650.**

**Geoffrion, A.M., Morris, J.G., and Webster, S.T. (1995) Distribution System Design, in *Facility Location* (Z. Drezner, ed.), Springer-Verlag, New York, New York.**

**Glover, F., Hultz, J., Klingman, D., and Stutz, J. (1978) Generalized Networks: A Fundamental Computer-Based Tool, *Management Science*, 24 (12), 1209-1220.**

**Glover, F., and Klingman, D. (1977) Network Application in Industry and Government, *AIIE Transactions*, 9 (4), 363-376.**

**Goetschalckx, M., Nemhauser, G., Cole, M.H., Wei, R., Dogan, K., and Zhang, X., (1994) "Computer Aided Design of Industrial Logistics Systems," *Proceeding of the Third Triennial Symposium on Transportation Analysis* (TRISTAN III), Capri, Italy, 151-178.**

**Lee, H.L., and Tang, C.S. (1997) Modelling the Costs and Benefits of Delayed Product Differentiation, *Management Science*, 43 (1), 40-53.**

**Popken, D.A., (1994) An Algorithm for the Multiattribute, Multicommodity Flow**

**Problem with Freight Consolidation and Inventory Costs, *Operations Research*, 42 (2), 274-286.**

**Van Hoek, R.I., and Commandeur, H.R., "Reconfiguring Logistics Systems through Postponement Strategies," *Journal of Business Logistics*, 19 (1), 33-54.**

**Verter, V., and Dincer, M.C., (1995) Global Manufacturing Strategy, in *Facility Location* (Z. Drezner, ed.), Springer-Verlag, New York, New York.**

**Slats, P.A., Bhola, B., Evers, J.J.M., and Dijkhuzien, G., (1995) "Logistics chain modelling," *European Journal of Operational Research*, 87, 1-20.**

**Vidal, C.J., and Goetschalckx, M. (1997) "Strategic production-distribution models: a critical review with emphasis on global supply chain models," *European Journal of Operational Research*, 98, 1-18.**

**Williams, H.P. (1990) *Model Building in Mathematical Programming*, John-Wiley & Sons, New York, New York.**

# APPENDIX A – AMPL MODEL FILE

```
#############
# merge-in-transit network design
#
# Mike Cole and Mukundh Parthasarathy
# June 3, 1998
#

set I_PROD;     # input product
set O_PROD;     # output product
set PROD := I_PROD union O_PROD;

set S_CITY;     # supply points
set M_CITY;     # merge points
set C_CITY;     # customer points
set CITY := S_CITY union M_CITY union C_CITY;

set LINKS within {CITY cross CITY};

param opening_cost {S_CITY union M_CITY} ;
param closing_cost {S_CITY union M_CITY} ;

param prod_weight {PROD} >=0;       # product weight

param link_cost {LINKS} >=0;        # transportation cost / weight
param link_capy {LINKS} >=0;        # transport capacity (weight)

param assy_rate {p1 in I_PROD, p2 in O_PROD} >= 0; # units p1 per p2
param assy_cost {M_CITY, O_PROD} >=0;           # unit assembly cost
param assy_capy {M_CITY, O_PROD} >=0;           # assembly capacity

param supply {S_CITY,PROD} >= 0;    # amt available at supplier

param demand {C_CITY,O_PROD} >= 0;  # amt demanded at customer

var Open {i in S_CITY union M_CITY} binary >= 0;

var Ship {(i,j) in LINKS, p in PROD} >= 0;

var Consume {j in M_CITY, p in I_PROD} >= 0;
var Assemble {j in M_CITY, p in O_PROD} >= 0;


minimize Total_cost:
      sum {(i,j) in LINKS, p in PROD}
         link_cost[i,j] * prod_weight[p] * Ship[i,j,p] +
      sum {j in M_CITY, p in O_PROD} assy_cost[j,p] * Assemble[j,p] +
      sum {j in S_CITY union M_CITY} opening_cost[j] * Open[j] +
```

```
      sum {j in S_CITY union M_CITY} closing_cost[j] * (1-Open[j])
      ;


subject to Supply_capacity {i in S_CITY, p in PROD}:
      sum {(i,j) in LINKS} Ship[i,j,p] <= supply[i,p] * Open[i];

subject to Amount_assembled {j in M_CITY, p in O_PROD}:
      Assemble[j,p] = (sum {(j,k) in LINKS}
         Ship[j,k,p]) - (sum {(i,j) in LINKS} Ship[i,j,p]);

subject to Amount_consumed {j in M_CITY, p in I_PROD}:
      Consume[j,p] = (sum {(i,j) in LINKS} Ship[i,j,p]) -
                     (sum {(j,k) in LINKS} Ship[j,k,p]);

subject to Assembly_flow_conservation {j in M_CITY, p1 in I_PROD}:
      Consume[j,p1] >=
          sum {p2 in O_PROD} (assy_rate[p1,p2] * Assemble[j,p2]);

subject to Demand_satisfaction {k in C_CITY, p in O_PROD}:
      sum {(j,k) in LINKS} Ship[j,k,p] = demand[k,p];

subject to Ship_from_open_merge_points
   {j in M_CITY, k in C_CITY, p in O_PROD: (j,k) in LINKS}:
      Ship[j,k,p] <= demand[k,p] * Open[j];
```

# APPENDIX B – SAMPLE AMPL DATA FILE

```
#################################
# merge in transit data file
## Mike Cole and Mukundh Parthasarathy
# June 3, 1998

set I_PROD := i_1, i_2;
set O_PROD := o_1;

set S_CITY := PITT ;
set M_CITY := NE SE ;
set C_CITY := BOS EWR BWI ATL MCO ;

set LINKS :=
      (PITT,NE) (PITT,SE)
      (NE,BOS) (NE,EWR) (NE,BWI)
      (SE,EWR) (SE,BWI) (SE,ATL) (SE,MCO);

param:      opening_cost      closing_cost      :=

PITT        1000                    200
SE          300                     300
NE          1000                    100         ;

param:      prod_weight :=
      i_1         1
      i_2         1
      o_1         1 ;

param:      link_cost   link_capy  :=
  PITT NE         1.5       250
  PITT SE         3.5       250
  NE BOS          1.7       100
  NE EWR          0.7       100
  NE BWI          1.3       100
  SE EWR          1.3       100
  SE BWI          0.8       100
  SE ATL          0.2       100
  SE MCO          2.1       100 ;

param assy_rate:        o_1 :=
            i_1         2
            i_2         1;

param:            assy_cost   assy_capy :=
      NE o_1            2     1000
      SE o_1            3     1000 ;
```

```
param supply:      i_1         i_2         o_1   :=
            PITT  12000       10000       0     ;


param demand:                  o_1 :=
                        BOS      90
                        EWR     120
                        BWI     120
                        ATL      70
                        MCO      50 ;
```

## APPENDIX C – PROTOYPE DATABASE TABLES

**The recommended (revised) database structure was detailed in Chapter 3. The actual prototype database consists of (the now obsolete) tables described in Tables 7 through 12.**

**Table 7. Product Data Table (Prototype)**

**ID Number**
**Name**
**Weight**

**Table 8. Source Data Table (Prototype)**

**ID Number**
**Name**
**Outgoing Product ID**
**Outgoing Product Quantity (Supply)**
**Outgoing Channel ID**

**Table 9. Customer Data Table (Prototype)**

**ID Number**
**Name**
**Incoming Product ID**
**Incoming Product Quantity (Demand)**
**Incoming Channel ID**

**Table 10. Channel Data Table (Prototype)**

**ID Number**
**Origin**

Destination
Cost

**Table 11. Merge Point Data Table (Prototype)**

ID Number
Name
Status (Open/Closed)
Incoming Product ID
Incoming Product Quantity
Incoming Channel ID
Outgoing Product ID
Outgoing Product Quantity
Outgoing Channel ID

**Table 12. Merge Formula Data Table (Prototype)**

Finished Product ID
Raw Material Product ID
Units of Raw Material Product Required