

## **A model building software for optimal production and transport planning**

**Enrique Parra<sup>1</sup>**

<sup>1</sup> Dpto. de Fundamentos de Economía, Universidad de Alcalá (Madrid- Spain). C/ Ctra Canillas 103. 28043 Madrid. [Enrique.Parra@uah.es](mailto:Enrique.Parra@uah.es)

### **Abstract**

*The power calculus availability with low cost personal computers has opened wide possibilities for using mathematical programming models in industrial size production and transport planning or the global consideration of the supply, distribution and manufacturing operations. However, some challenges come up: the first is to have the disposal of the optimization algorithms that resolve problems that involve thousands or dozens thousands equations in a short period of time, in a way that make possible the practical use of the solution. The second challenge is the one referring to how the build the model. Due to the size it should be done with computer. The core of this paper is this second issue: how to construct such a large model. A software is presented to face this question. A previous version was already tested by the author in a big Spanish company with embedded optimization.*

**Keywords:** software, mathematical programming, transport optimization

### **1. Joint production and transport optimization**

#### **1.1. The problem.**

The optimum production and distribution both together are better than the optimum with both parts separately: the greater cost of distribution can make up for the greater cost of the production or vice versa. Due to that, it's necessary to create production functions and possibilities of multilevel distribution (factories, warehouses, customer).

In industrial companies the size of the problem to be solved is huge, especially if the seasonality of the production and/or demand is very noticeable. Usually consist in models with thousands of equations and dozens thousands variables.

It is common the existence of threshold quantities when we observe the real cases: either an activity it's not been made or it has to be made beneath a minimum and finally in production its common the appearance of mixtures that must fulfil some properties.

#### **1.2. Traditional solution.**

One of the most usual model types employed is mixed integer mathematical programming. To design this type of models, each company must need expert help; the expert will use any of the commercial tools available. These tools can be categorized in two groups: optimisers and modelling languages.

The first one, optimisers, solves the problem in final form, typically in matrix format. The programmer is the responsible to fill the matrix accessing to databases and using the accepted formulation. Of course, the software could allow some flexibility in the problem formulation. The modelling languages help in the job of model equations writing and, even, in the connections with databases that contain the necessary parameters for model building.

A third type of solution is a kind of systems suitable to a wide range of problems and are data driven, that is, depending on which are the data used it is possible to generate quite different models but inside the range.

It can be concluded that it is compulsory to receive the advice of an expert in the language selected, with good mathematical modelling skills capable to move into equations, the insight of the system to be modelled and, finally, translate those equations into the modelling language syntax. To finish, the data must be obtained using data files and/or databases connections. When model building ends, the system is fed by a user that can not change the model. Sometimes, the designer can allow some flexibility. In this case and depending on the options, properly speaking, the system is a model generator instead of a single model. Follows the last version features of a model generation system designed by the author.

## **2. MIP model building software**

### **2.1. A model generator.**

For the usage of optimizers, or languages that build mathematical programming models (Haverly, 2003 or Maximal, 2008), it's necessary to count with the assessment of an expert in this language who knows how to develop and test mathematical models.

Other possible answer is to think of all the different occurring possibilities in a wide type of problems and design a system that use all the data to build up the correct model in each situation and is able to generate different models without an expert help. This system then would be a model generator and not a simple model.

The system introduced in this paper (MIPG) is a new version of a previous (Parra, 99), industrially used for supply chain planning. MIPG allows an easy way to represent, for a non mathematical user, a wide range of planning models for networks in order to maximize the production and distribution variable costs margin. The system generates and solves industrial size models (aprox. 100.000 variables and 10.000 equations), and the model building process is based in attribute definition for nodes and arcs in a multiproduct and multiperiod network. A set of intuitive keys drive the model building phase, depending on the content and number of keywords a different model is generated: from didactical to industrial uses. By this way, the user is autonomous: with the same tool can afford different situations and purposes.

MIPG permit the construction of different models starting with some intuitive keys written by a non expert in mathematical programming user whose only acknowledgement is the manufacturing and distribution system.

This software has been mainly design for multilevel transport and medium complexity production systems. It allows the generation of multiproducts and multiperiods models by the usage of networks.

## 2.2. Variables and equations of the models

The network is based in **N nodes** (suppliers, factories, transshipment sites, destinations, customers) connected, by transport arcs (**A**), **P** products and **T** planning horizon periods (periods can be of variable length). They are the members, respectively, from **NODES**, **PRODUCTS** y **PERIODS** sets. Let  $i, i' \in N$  (Nodes),  $j, j' \in P$  (Products),  $k \in T$  (Periods).

The generated optimization model purpose is to find the optimal value of the following variables:

1.  $PX(i, j, k)$ . Quantity produced of product  $j$  at node  $I$  during period  $k$ .
2.  $TR(i, j, j', k)$ . Quantity transformed from product  $j$  into product  $j'$  at node  $i$  during period  $k$ .
3.  $X(i, l, j, k)$ . Quantity (if it is generated as a continuous variable) transported from origin node  $i$  to destination node  $l$  of product  $i$  during period  $k$ , or (see below), if it is generated as binary variable, indicates than node  $l$  must be provided from node  $I$  for all products.
4.  $CS(i, j, k)$ . Quantity consumed (sales = demand satisfaction) of product  $j$  at node  $i$  during period  $k$
5.  $SS(i, j, k)$ . Quantity of product  $j$  leaving node  $i$  during period  $k$ .
6.  $EF(i, j, k)$ . Quantity stocked at the end of period  $k$ , of product  $j$  at node  $i$ .
7.  $MX(i, j, j', k)$ . Quantity of ingredient  $j$  to used in the final product  $j'$  by blending in period  $k$  at node  $i$

Objective:

“Maximize the variable margin: Revenues from product sales – Variable Costs (sum of supply, manufacturing, blending, stock and transportation costs)”

MIP (mixed integer) models are generated due to two reasons: 1)  $X$  variables can be binary in order to force the model to select the same origin-destination arc for all the products required at a prefixed node type or 2) semicontinuous variables are used to force the outflows from a node to a threshold level or zero..

So, the objective function, for the continuous variables case, is (the parameter vectors are explained later):

$$\begin{aligned}
 & \sum_i \sum_j \sum_k Pr(i, j, k).CS(i, j, k) - \sum_i \sum_j \sum_k CostProd(i, j, k).PX(i, j, k) - \\
 & CostAlm. \sum_i \sum_j \sum_{k=1}^{T-1} EF(i, j, k) - \sum_i \sum_{i' \neq i} \sum_j \sum_k CosTra(i, i', j).X(i, i', j, k) - \\
 & \sum_i \sum_j \sum_{j'} \sum_k CosTR(i, j, j', k).TX(i, j, j', k) \\
 & + \sum_i \sum_j \sum_{j'} \sum_k MX(i, j, j', k) \tag{FO1}
 \end{aligned}$$

In the  $X$  binary case:

$$\begin{aligned}
& \sum_i \sum_j \sum_k \Pr(i, j, k).CS(i, j, k) - \sum_i \sum_j \sum_k CostProd(i, j, k).PX(i, j, k) - \\
& CostAlm. \sum_i \sum_j \sum_{k=1}^{T-1} EF(i, j, k) - \sum_i \sum_{i' \neq i} \sum_j \sum_k Costra(i, i', j).Dem(i, j, k).X(i, i', j, k) - \\
& \sum_i \sum_j \sum_{j'} \sum_k CosTR(i, j, j', k).TR(i, j, j', k) \\
& + \sum_i \sum_j \sum_{j'} \sum_k MX(i, j, j', k) \tag{FO2}
\end{aligned}$$

Constraints (keywords for model building are mentioned):

1. Balance for Initial stock, productions, transformations, consumption, shipments and final stock for each node (i), product (j) and period (k).

$$\begin{aligned}
EF(i, j, k) - PX(i, j, k) + CS(i, j, k) - \sum_{i'} X(i', i, j, k) + \sum_{i'} X(i, i', j, k) + \\
+ \sum_{j'} TX(i, j', j, k) - \sum_{j'} Yield(i, j, j', k). TX(i, j', j, k) \\
- \sum_{j'} MX(i, j', j, k) + \sum_{j'} MX(i, j, j', k) = ExIni(i, j) \quad \forall (i, j, k=1) \tag{1}
\end{aligned}$$

$$\begin{aligned}
EF(i, j, k) = EF(i, j, k-1) PX(i, j, k) + CS(i, j, k) - \sum_{i'} X(i', i, j, k) + \sum_{i'} X(i, i', j, k) + \\
+ \sum_{j'} TX(i, j', j, k) - \sum_{j'} Yield(i, j, j', k). TX(i, j', j, k) \\
- \sum_{j'} MX(i, j', j, k) + \sum_{j'} MX(i, j, j', k) = 0 \quad \forall (i, j, k > 1) \tag{1'}
\end{aligned}$$

2. Quantities  $PX(i, j, k)$  are produced in node  $i$ , from a lower limit,  $ProdMin(i, j, k)$ , to an upper limit,  $ProdMax(i, j, k)$ , for each product  $j$  and during each period  $k$  (keyword PROD) with an unitary cost  $CosProd(i, j, k)$ . These quantities can be seen like supplies at the node. This is the only way to generate material in the model.

$$ProdMin(i, j, k) \leq PX(i, j, k) \leq ProdMax(i, j, k) \quad \forall (i, j, k) \tag{2}$$

3. Lower limit (ProdCMin) and upper limit (ProdCMax) exists for joint production (sum for all products) for each period  $y$  node.

$$ProdCMin(i, k) \leq \sum_j PX(i, j, k) \leq ProdCMax(i, k) \quad \forall (i, k) \tag{3}$$

4. Lower limit (ProdPMin) y upper limit (ProdPMax) exists for a product production in the whole horizon and at each node.

$$ProdPMin(i, j) \leq \sum_k PX(i, j, k) \leq ProdPMax(i, j) \quad \forall (i, j) \tag{4}$$

5. Any product can be transformed into other with a yield, an unitary cost ( CosTR ) and lower and upper limits:

$$TRMin(i, j, j', k) \leq TR(i, j, j', k) \leq TRMax(i, j, j', k) \quad \forall (i, j, j', k) \tag{5}$$

$$TR1Min(i,j,k) \leq \sum_{j'} TR(i,j,j',k) \leq TR1Max(i,j,k) \quad \forall (i,j,k) \quad (6)$$

$$TR2Min(i,j) \leq \sum_{j'} \sum_k TR(i,j,j',k) \leq TR2Max(i,j) \quad \forall (i,j) \quad (7)$$

$$TR3Min(i,k) \leq \sum_j \sum_{j'} TR(i,j,j',k) \leq TR3Max(i,k) \quad \forall (i,k) \quad (8)$$

$$TR4Min(i) \leq \sum_j \sum_{j'} \sum_k TR(i,j,j',k) \leq TR4Max(i) \quad \forall (i) \quad (9)$$

6. Any product can be stocked from a period to the next one; there exists stock limits (ExMin y ExMax) for each pair product / node, and also a total for each node. At the beginning of the planning horizon, the nodes have an initial stock by product (ExIni).

$$ExMin(i,j) \leq EF(i,j,k) \leq ExMax(i,j) \quad \forall (i,j,k) \quad (10)$$

7. Sum of all product stocks at node i in each of the periods are limited (ExCMin and ExCMax)

$$ExCMin(i) \leq \sum_j EF(i,j,k) \leq ExCMax(i) \quad \forall (i,k) \quad (11)$$

8. Lower and upper stock limits for each node i, product j and period k: ExFMin and ExFMax

$$ExFMin(i,j,k) \leq EF(i,j,k) \leq ExFMax(i,j,k) \quad \forall (i,j,k) \quad (12)$$

9. Lower and upper demand limits (DemMin and DemMax), for each period, product and node; the revenues are the product of a price (Pr) and the demand. This is the way to represent the sales. They are the only usual income from the model.

$$DemMin(i,j,k) \leq CS(i,j,k) \leq DemMax(i,j,k) \quad \forall (i,j,k) \quad (13)$$

10. Lower and upper transportation limits for each arc (from node i to node i'), product j and period k (XMin and XMax). This is the way to model the transportation between nodes .

$$XMin(i,j,k) \leq X(i,j,k) \leq XMax(i,j,k) \quad \forall (i,j,k) \quad (14)$$

11. Lower and upper limits for the inflow to a node (EntMin and EntMax). To avoid very low sendings it is possible to impose a threshold level defining the outflow variable as a semicontinuous variable.

$$EntMin(i,j,k) \leq \sum_{i'} X(i',i,j,k) \leq EntMax(i,j,k) \quad \forall (i,j,k) \quad (15)$$

$$SS(i,j,k) = \sum_{i'} X(i,i',j,k) \quad \forall (i,j,k) \quad (16)$$

$$SalMin(i,j,k) \leq SS(i,j,k) \leq SalMax(i,j,k) \quad \forall (i,j,k) \quad (17)$$

If outflows are modeled as semicontinuous variable, new variables must be generated, in addition to  $SS(i,j,k)$  for the outflow from node, the model will include three auxiliary

variables,  $SS1(i,j,k)$ ,  $SS2(i,j,k)$ ,  $BS(i,j,k)$ ; the last one it is binary ( $BS(I,j,k) \in \{0,1\}$ ) and constraints (16) y (17) are reformulated

$$SS(i,j,k) = \sum_{i'} X(i,i',j,k) \quad \forall (i,j,k) \quad (16)$$

$$SS(i,j,k) - SalMin(i,j,k) \cdot SS1(i,j,k) + SalMax(i,j,k) \cdot SS1(i,j,k) = 0 \quad \forall (i,j,k) \quad (17a)$$

$$BS(i,j,k) + SS1(i,j,k) = 1 \quad \forall (i,j,k) \quad (17b)$$

12. Also it is possible to include special constraints using the variables of the model

$$R\_Specials \quad \text{Those included in the model} \quad (18)$$

In case that  $X$  must be binary all the equations must be changed to substitute  $X$  by the term  $Demanda(i,j,k)$ .  $X(i,i',j,k)$ . In addition to this, new constraints are generated:

13. It is possible, as it was mentioned above, to force that all the products will be received in a node from the same origin. This is made by adding some constraints (remember  $X(i,i',j,k)$  is binary)

$$X(i,i',1,k) - X(i,i',j,k) \leq 0 \quad \forall (i,j,k), j \neq 1 \quad [19]$$

14. Blendings. It's possible to blend several products at a node to obtain other product with certain limits in properties. It's not allowed a successive blendings (pooling problem) to avoid non linearities.

Let  $MCMax(i,j,j',k)$  and  $MCMin(i,j,j',k)$  maximum and minimum quantities admitted in product  $j'$  of the ingredient  $j$  at node  $i$  in period  $k$ .

Let  $MPMax(i,j',k)$  and  $MPMin(i,j',k)$  maximum and minimum quantities to be obtained of final product  $j'$  at node  $i$  in period  $k$ .

Let  $CalPro(i,j,k,l)$  the value of property  $l$  of product  $j$  at node  $i$  in period  $k$ .

Let  $CalMax(i,j,k,l)$  and  $CalMin(i,j,k,l)$  maximum and minimum values admitted for property  $l$  in final product  $j$  at node  $i$  in period  $k$ .

With a model includes blendings following equations must be added

$$MCMin(i,j,j',k) \leq TX(i,j,j',k) \leq MCMax(i,j,j',k) \quad \forall (i,j,j',k) \quad (19)$$

$$MPMin(i,j,k) \leq \sum_{j'} TX(i,j',j,k) \leq MPMax(i,j,k) \quad \forall (i,j,k) \quad (20)$$

$$\sum_{j'} (CalPro(i,j',k,l) - CalMax(i,j,k,l)) TX(i,j',j,k) \leq 0 \quad \forall (i,j,k,l) \quad (21)$$

$$- \sum_{j'} (CalPro(i,j',k,l) - CalMin(i,j,k,l)) TX(i,j',j,k) \leq 0 \quad \forall (i,j,k,l) \quad (22)$$

### 2.3. User approach

For the end user, the system is a black box; the user feed it with data, very intuitive for him, and gets a solution in very easy format. This is the most important MIPG feature.

Table 1 shows the type of data and the keywords used to build a model. Table 2 shows an example of a model (the keywords meaning are in table 1).

1.	<u>General</u> : "TITLE" (titles), "COSTALM" (Stock cost),
2.	<u>Structure</u> : "NODES" (multinode models), "PRODUCTS" (multiproduct), "PERIODS" (multiperiod), "NET" (transportation arcs), "JUNTOS" (all products together in the same arc)
3.	<u>Stocks</u> : "ALMC" (joint limits), "EXINI" (initial stock and limits), "EF" (stock limits).
4.	<u>Production</u> : "PROD" (limits and production cost), "PRODC", "PRODP" (limits)
5.	<u>Demand</u> : "DEM" (limits and price for the sales)
6.	<u>Transport</u> : "ENT" (inflow limits), "SAL" (outflow limits, could be semi-continuous), "X" (arc/products/periods limits)
7.	<u>Conversions / Transformations</u> : "TR" (feasibility, limits and costs fro transformation from one product to another), "TR1", "TR2", "TR3", "TR4" (other limits)
8.	<u>Blending</u> : "MX" (possibilities, limits & costs of blend some products to get another), "MC", "CA", "CP", (blending data)
9.	Special constraints : "RE", "COE" . Any other constraint over the variables.

Table 1. Keys for defining models

Title y other data	TITLE", "Test Version 3.1" "tons" "euros" "CostAlm",1
Products	"PRODUCTS",2 "Granel", "G" "Envasado", "E"
Periods	"PERIODS",2 "Ene08", "1", 31 "Feb08", "2", 28
Nodes	"NODES", 19
. Fábricas	"F.Orense", "ORENS", 0,1,0,1,1,0,2 (number 2 indicates SEMICONTINUOUS)
. Almacenes	"F.Barcelona", "BARNA", 0,1,0,1,1,0,2 "Al.Valladolid", "VALLA", 1,0,0,0,1,1,2 "Al.Teruel", "TERUE", 1,0,0,0,1,1,2
. Clientes	"Cliente 01", "CLI01", 100,0,1,0,0,1,0 ... definition of the 15 customers like nodes "Cliente 15", "CLI15", 100,0,1,0,0,1,0
Initial Stock	"EXINI", "*****", 1000,0,200004
Stocks limits	"ALMC", "*****", 0,100003 "EF", "*****" 0 6666666
Production (total)	"PRODC", "*****", 0,100001
Production (joint)	"PRODP", "*****", 0,200002
Inflows Limits	"ENT", "*****", 0,10000
Outflow Limits	"SAL", "*****", 50,100000 (will be semicontinuous)
Production limits	"SAL", "TERUEG1", 20,100000 "PROD", "*****", 0,0,0 "PROD", "ORENSG*", 0,3059,1 ...rest of limits
Transformation from G into E	"TR", "ORENSGE*", 1,11,0,3059 "TR", "BARNAGE*", 1,12,0,4210
Customer demand	"DEM", "*****", 0,0,0 "DEM", "CLI01**", 93,93,1000 .... rest of demands "DEM", "CLI15**", 30,30,1000
Network and cost	"NET", "ORENS", "VALLA", 61,31 "NET", "ORENS", "TERUE", 91,67 ... rest of tranpsortation cost NET", "TERUE", "CLI15", 94,34
Transportation limits	"X", "*****CLI15**" 1 1111 "X", "ORENSTERUEE1" 12 1000

### Table 2. A simple model

The data for any type of model can be read from databases, spreadsheets, etc. Using this data the model is generated and solved by the software (Dash Ass, 2008). Optimal solution can be analyzed using other software (OLAP type tools are recommended).

### 3. Conclusions

This paper describes a model generation software which allows the building and solving of mixed integer mathematical programming (MIP). It is possible to represent multiproduct and multiperiod networks and maximise the global economic margin of the supply chain. The system has been used for factory supply planning, for factory to customer product distribution planning, for investment analysis, etc. .. and always with the same tool. The models are built using easy keys instead of the mathematical formulation of equations.

### References

- Dash Associates (2008), "XPRESS-MP Reference Manual". Dash Associates.  
Haverly (2003) "Omni Model Management System". Annals of Operations Research.  
Maximal Software (2008). "MPL on line tutorial", <http://www.maximal-usa.com>  
Parra, E (1999). Optimización del transporte. Ed Díaz de Santos. Madrid