Optimal Decisions in Multi-Model Systems

M. ALEXANDRU*, K.G.ARVANITIS**, M. SERBANESCU*,

*Control System Department
Politehnica University of Bucharest,
313 Splaiul Independentei, 77206,
Bucharest 6,
ROMANIA

**Department of Electrical and Computer Engineering
National Technical University of Athens
Zographou, 15773, Athens,
GREECE

Abstract:-The paper deals with large-scale systems represented as a collection of subsystems which respect a certain order and interconnection relays. Each subsystem is described through a specific model so that the whole system can be view as a multi-model homogenous structure. The management of a system is enough complicated and in order to accomplish it we propose a decentralised decision structure having a well-defined distribution of supervisory functions. For a large scale system two strategies are outlined:; relaxation and partitioning appropriate for the weak coupling systems and decomposition technique suitable for additive separable systems. The work suggests an optimal system's functionality through a partial optimisation of each subsystem's quality criteria.

The numerical implementation of associated algorithms was improved through a parallel and hierarchical distributed computation approach. The developed software package (LSOPT) reduces the computation complexity, allowing an equilibrate calculus effort per each subsystem. Although algorithms' design enables an efficient functionality on a parallel unit net, the actual results are obtained applying a sequential procedure in a former phase. The main contribution consists in a distributed computation approach on multi-model representation.

Key-Words: large scale systems, optimisation, (non) linear problems, decomposition, relaxation, partitioning
CSCC'99 Proceedings:- Pages 2431-2435

1 Introduction

In many applications, the change of the original optimisation problem (OP) described by the linear or non linear criteria function, with or without restrictions, into an equivalent problem (or sets of equivalent subproblems) having a standard plot is necessary. The basic idea is that the solution of the latter, obtained by using the numerical methods of mathematical programming (MP), leads to the solution of the original problem. These transformations are useful in the optimisation stage of a real process also because the solution of equivalent problems needs a well-distributed computation effort. To make the computation easier, specific methods were developed to solve that problem [1], [2], [3], [4]

Several techniques may be used to simplify numerical computations used to solve large-scale optimisation problems, such as: decomposition, penalisation, relaxation, partitioning.

Decomposition results in a set of subproblems able to be solved in parallel attached to a global problem additive separable in criteria and constraints.

Penalisation changes an original problem with restrictions into a problem free of restrictions having the same solution as the original one

As usual, the relaxation and partitioning techniques are intimately correlated in the algorithms applied to obtain the appropriate solution for block diagonal problems associated to weak coupled systems.

Partitioning is applied when many variables are present. This is a procedure, which divides the variables of the problem in two subsets. At first it acts on the variables belonging to one of the subsets and afterwards on variables in the other one.

Relaxation consists in the temporary elimination of some restrictions and in solving the problem with the remaining ones. If the solution complies with relaxed restrictions, then the solution is optimal one. If not, one or several restrictions are imposed and the procedure is reiterated.

In order to solve large-scale optimisation problems a software package (LSOPT) which automatically defines and determines the optimisation problem is proposed. Options for developing the control model, which by means of the optimisation procedure leads to the solution, are available to the user.

The paper is organized as follows: First, a decomposition approach is described in Section 2 follows by the relaxation and partitioning techniques detailed in Section 3. Then, an overview presentation of a software package together with a numerical computation examples are introduced in Section 4. Finally, Section 5 traces some conclusions and future trends.

2 Decomposition approach

An overview representation of a system suitable to a decomposition approach is depicted in Fig.1, where:
u:=[u^1|u^2|...|u^N] is a set of input technological flow vectors

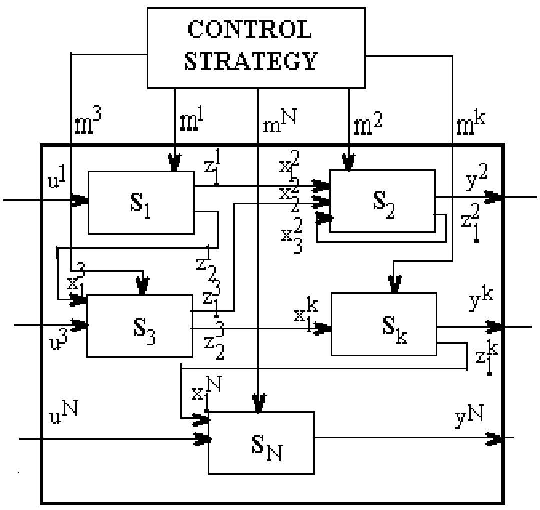

Fig.1 An overview structure of a multi-model system

m:=[m^1|m^2|...|m^N] is a set of control system vectors
x:=[x^1|x^2|...|x^N] is a set of input coupling vectors
z:=[z^1|z^2|...|z^N] is a set of output coupling vectors
y:=[y^1|y^2|...|y^N] is a set of qualitative criteria values
It is assumed the original problem is additive separable respect to criteria and constraints,

$$F(x,m) = \sum_{i=1}^{N} f^i(m^i, x^i)$$

where $f^i(m^i,x^i)$ are build up on the qualitative vectors y^i
The optimisation problem regarding the whole system *S* can be formulated as follows:

$$\max_{(x,m)} F(x,m)$$

having the input-output constraints

$$z^i = G^i(m^i,\ x^i) \qquad \mathrm{i = 1,\ 2,\ \ldots,\ N}$$

and the coupling ones,

$$x^i = H^i(z^1,\ z^2,\ \ldots,\ z^N)$$

Consequently, the problem's Lagrangean is:

$$L = \sum_{i=1}^{N} f^i(m^i, x^i) + \sum_{i=1}^{N} (m^i)^T (G^i - z^i) +$$
$$\sum_{i=1}^{N} (\rho^i)^T (H^i - x^i)$$

where μ^i, ρ^i stand for Lagrange operators
The Lagrangian is additive decomposable and the global problem is translated in N optimisation subproblems as: $\max_{(x^i,m^i)} f^i(x^i, m^i)$ keeping the same previous constraints.

Further, it was assumed that the card(μ^i) is considerable smaller then card(ρ^i) and a software module was designed to implement the algorithm based on ρ operators. The computation was distributed at two levels according to the following algorithm:
Step 1 The ρ^i vectors are estimated at higher hierarchical level through the minimisation of L(ρ^i) using a gradient method or the strategy Newton-Raphson,

$$\rho^{i(k+1)} = \rho^{i(k)} - \alpha\left(\frac{\partial L^i}{\partial \rho^i}\right)$$

respective,

$$\rho^{i(k+1)} = \rho^{i(k)}\left(\frac{\partial L^i}{\partial \rho^i}\right)^{-1(k)}\left(\frac{\partial L \rho^i}{\partial \rho^i}\right)^{(k)}$$

Step 2 Once the vectors ρ^i calculated we can minimise the optimisation problem at local (lower) level,

$$\max\left\{ f(m,x) + (\rho^{i*})^T [x - H(z^1, z^2, \ldots, z^N)] \right\},$$

$$G^i - z^i = 0, \qquad i = 1,2,\ldots,N$$

Step 3 A convergence criteria is verified and the optimisation procedure stops when the inequality becomes true.

$$\left[\sum_{i=1}^{N}\left(\frac{\partial L^i}{\partial \rho^i}\right)^2\right]^{1/2} \leq eps$$

3 Relaxation and partitioning

This section is concerned with block diagonal structure problems classified into linear and non linear sets, associated with some weak coupling systems. In the following it is presented in details the algorithms implemented in the developed software using to solve the two types of the previously mentioned problems.

3.1 Algorithm for linear problems (Ritter)

It is used for general program of the type:

- **linear problems**: $\min_{x,y}(c^T x + c_0{}^T y)$

with the coupling constraints:

Ax+D₀y=b₀

Bx+Dy=b

where:

x:=[x_1|x_2|...|x_N] is a set of n_i-dimensional vectors x_i and *y* is the coupling vector of the subsystems, $x_i \geq 0, i = 1,...,N, y \geq 0$

c:=[c_1|c_2|...|c_N] is a set of corresponding coefficients

A:=[A_1|A_2|...|A_N] is a set of $(m_0 \, x \, n_i)$ dimensional matrices A_i

D_o: (m_0 x n_0) array

$$B = \begin{bmatrix} B_1 & 0 & 0 & 0\ldots\ldots & 0 \\ 0 & B_2 & 0 & 0\ldots\ldots & 0 \\ \ldots\ldots & & & & \\ 0 & 0 & 0 & 0\ldots\ldots & B_N \end{bmatrix}$$ is a block diagonal matrice having B_i: $(m_i \, x \, n_i)$ array,

D:=[D_1|D_2|...|D_N], D_i: $(m_i \, x \, n_0)$ array,

b:=[b_1|b_2|...|b_N], b_i vectors

For the general case, the solution is given by Ritter's partitioning method and for the particular case, y = 0, by Rosen's linear method.

Let's assume rank $B_i = m_i$ and therefore the non singular array B_{i1} exists for some initially given coupling vector y_0, so that: $B_i x_i = b_i - D_i y_0 \quad x_i \geq 0 \quad (i = 1,...,N)$

Using partitioning the restrictions may be written as follows: $B_{i1}x_{i1} + B_{i2}x_{i2} + D_i y = b_i \quad (i = 1,...,N)$

By solving for x_{i1} the above equations, we get:

$$x_{i1} = B_{i1}^{-1}b_i - B_{i1}^{-1}B_{i2}x_{i2} - B_{i1}^{-1}D_i y$$

Now, both the criteria function and the coupling restrictions are properly partitioned into:

$$F(x) = \sum_{i=1}^{N}(c_{i1}^T x_{i1} + c_{i2}^T x_{i2}) + c_0^T y \, ;$$

$$\sum_{i=1}^{N}(A_{i1}x_{i1} + A_{i2}x_{i2}) + D_0 y = b_0$$

According to the classical procedure variables x_{i1} are eliminated first. In order to simplify the computation we adopt the following notations:

$$d_i^T = c_{i2}^T - c_{i1}^T B_{i1}^{-1}B_{i2}; \quad d_0^T = c_0^T - \sum_{i=1}^{N} c_{i1}^T B_{i1}^{-1} D_i$$

$$a = \sum_{i=1}^{N} c_{i1}^T B_{i1}^{-1} B_{i2} \qquad M_0 = D_0 - \sum_{i=1}^{N} A_{i1} B_{i1}^{-1} D_i$$

$$M_i = A_{i2} - \sum_{i=1}^{N} A_{i1} B_{i1}^{-1} B_{i2} \quad b = b_0 - \sum_{i=1}^{N} A_{i1} B_{i1}^{-1} b_i$$

The reduced program:

$\min(F(x) - a) = \sum_{i=1}^{N} d_i^T x_{i2} + d_o^T y$ is obtained for $\sum_{i=1}^{N} M_i x_{i2} + M_0 y = b$, $x_{i2} \geq 0 \quad y \geq 0$.

Thus we have obtained a number of equality restrictions equal to the number of coupling restrictions in the original problem. If (x_{i2}^0, y_0) is the solution of the reduced problem, the new values for x_{i1} are obtained as follows: $x_{i1}^0 = B_{i1}^{-1}b_i - B_{i1}^{-1}B_{i2}^{-1}x_{i1}^0 - B_{i1}^{-1}D_i y_0$.

Therefore, the global solution $(x_{i1}^0, x_{i2}^0, y_0)$ solves the primal problem changed by the relaxation of $x_{i1} \geq 0$ conditions. When these relationships are met, the vector $(x_{i1}^0, x_{i2}^0, y_0)$ is the appropriate solution of the problem.

3.2 Algorithm for non linear problems (Benders)

It is used for the following program type:

- **non linear problems**: $\min_{x,y}(c^T x + f(y))$

with the coupling constraints:

Ax+F(y)³ b

x³0, where:

x:=[x_1|x_2|...|x_N], is a set of n_i-dimensional vectors x_i and *y* is the coupling vector of the subsystems, $x_i \geq 0, i = 1,...,N, y \geq 0$

c:=[c_1|c_2|...|c_N] is a set of corresponding coefficients

A:=[A_1|A_2|...|A_N] is a set of $(m_0 \, x \, n_i)$ dimensional matrices A_i

f: scalar non linear function of *y;*

F Î R^m whose components are functions of *y;*

S: admissible subset in E^p selected according to the functional constraints

It may be observed that for a given value of *y* the problem is linear with respect to *x*.

Let's consider the set:

$R = \{y \mid \exists x \geq 0 \quad a.i. \quad Ax \geq b - F(y), y \in S\}$, where vectors *y* is admissible. The set *R* is determined by means of Farkas lemma, namely:

$R = \{y \mid (b - F(y))^T u_i^r \leq 0\}$ (i=1,…,n_i); y$\in S$, where u_i^r are the generators of dual elements occurring in the cone: $C = \{u \mid A^T u \leq 0, u \geq 0\}$.

The problem is rewritten as:

$$\min_{y \in R}((f(y) + \min(c^T x \mid Ax \geq b - F(y), x \geq 0)$$

For some given *y* in *R*, the minimisation between parentheses is a linear program with respect to x:

$\min(c^T x)$ for: $\begin{matrix} Ax \geq b - F(y) \\ x \geq 0 \end{matrix}$.

The dual admissibility set is:

$$\min(f(y) + \max(b - F(y))^T u \mid A^T u \le 0)$$

If the extreme points of P, noted u_i^N , i=1,…,n_N, are considered, the problem can be written:

$$\min(f(y) + \max(b - F(y))^T u_i^N) .$$

Finally, the problem is equivalent to:

$$\min z, \quad z \ge f(y) + (b - F(y))^T u_i^N ; y \in R .$$

Finding the solution of the latter problem is a very difficult task because for each extreme point a restriction arises and so a large number of constraints occurs even in middle scale problems. However, to get an optimal solution, only a few constraints are considered to be active. Therefore, the relaxation strategy is used. First a small number of restrictions is considered and the problem is partially solved. If the solution complies with the restrictions taking into account, it is an optimal one. If not, a restriction not complied with is added and the solving procedure is reiterated.

It can be shown that, if (z^0, y^0) is the solution of the problem above, and x^0 is the solution of the linear program, $\min(c^T x)$ having the restrictions $Ax \ge b - F(y^0); x \ge 0$, then the pair (x^0, y^0) solves the original problem and $z^0 = c^T x^0 + f(y^0)$.

The partitioning procedure goes on till a bellow optimality test is completed.

$$\max_u ((b - f(y^0))^T u \mid Au \le c; u \ge 0; u = z^0 - f(y^0))$$

4 Experimental results using the developed software package

The interface of the designed software package (LSOPT) is flexible and allows data to be easily taking over. The programs are developed in such way as to enable algorithms to be traced step by step. The nucleus algorithms are implemented in dynamically linked libraries to allow them to be called from various modules (e.g. SIMPLEX and BOXE algorithms) and to have more rigorous control of applications. Algorithm implementation in dynamically linked libraries has also the advantage of a better management of computer resources (e.g. stacks operation management in order to enable any kind of function to be taking over). The algorithms in the library are puts into places and for each class there are matters for algorithms control, initialization and erasing from computer storage. Since, objected oriented techniques are used new algorithms may be developed on the basis of these ones. The codes are written in C++ language.

In the following we illustrate some simple understanding numerical examples carried out by dedicated computation libraries.

•*A decomposition procedure using the Lagrange operators.*

Let's assume the following criteria functions:

$f_1(x_1,m_1)=2x_1^2+m_1^2-20x_1-40m_1+4x_1m_1+100$

$f_2(x_2,,m_2)=x_2^2+6m_2^2-30x_2-60m_2+4x_2m_2+225$

the corresponding model and coupling constraints, $x_1=z_2;\ x_2=z_1;\ z_1=x_1+2m_1;\ z_2=m_2$

associated to the system depicted in Fig.2

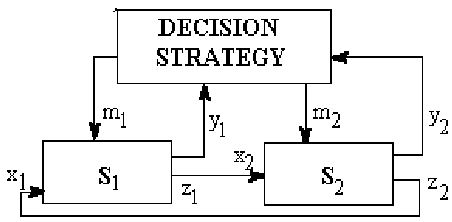

Fig.2 A simple structure of a two interconnected sub system

Consequently, the global optimisation problem is:

$$\max_{x,m} F(x) = \sum_{i=1}^{2} f^i(x^i, m^i)$$ with the same constraints mentioned above. According to Lagrange operators optimisation procedure it can be reformulate as:

$$\max_{x_i, m_i} L_i = f_i(x_i, m_i) + (r_i)^T (x_i - \sum_{j=1}^{2} c_{ij} z_j), \quad i = 1,2$$

where ρ_i it is assumed constant.

At higher hierarchical level a gradient algorithm gives us the operators' values at each step according to:

$$r^{i(k+1)} = r^{i(k)} - a \left(\frac{\frac{\partial L_i}{\partial r_i}}{\sqrt{\sum_{i=1}^{2} \frac{\partial L_i}{\partial r_i}}} \right)^{(k)}$$

where $\frac{\partial L_i}{\partial r_i} = L_r^i = x_i - \sum_{j=1}^{2} c_{ij} z_j$ and the advanced step α keeps the value of 0.05 for all iterations. The task is over when the inequality $\left[\sum_{i=1}^{2} \left(\frac{\partial L^i}{\partial r^i} \right)^2 \right]^{1/2} \le 0.001$ becomes true and the operators' values passed to the lower computation level are $\rho_1^* = -4.00$, $\rho_2^* = 6.00$. Here the BOXE method was implemented to solve the local minimisation problem. Thus we obtain $x_1^*=1.00$, $m_1^*=5.10$, $x_2^*=11.00$, $m_2^*=1.00$

· *Example of Ritter linear algorithm computation:*
Let's consider the problem:

$$\min F(x) = -x_1 - x_2 - 2x_5 - x_6$$

and the corresponding linear system:

$$\begin{bmatrix} 1\,2\,0\,0\,2\,1\,0\,0\,0\,1\,0 \\ 1\,1\,0\,0\,4\,2\,0\,0\,0\,0\,1 \\ 1\,3\,1\,0\,0\,0\,0\,0\,0\,0\,0 \\ 2\,1\,0\,1\,0\,0\,0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,1\,0\,1\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0\,1\,0\,1\,0\,0\,0 \\ 0\,0\,0\,0\,1\,1\,0\,0\,1\,0\,0 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ \vdots \\ x_{11} \end{bmatrix} = \begin{bmatrix} 40 \\ 50 \\ 30 \\ 20 \\ 10 \\ 10 \\ 15 \end{bmatrix}$$

where $x_i \ge 0$, *i=1:11*

The solutions are:

$$x_1 = x_7 = 6;\ x_2 = 8;\ x_3 = x_4 = x_8 = x_{10} = x_{11} = 0;$$
$$x_5 = 4;\ x_6 = 10;\ z = -32$$

More specific, the implementation consists in taking over input data and in defining the minimisation criteria, the number of coupling restrictions, the number of subsystems together with the estimated variables and the corresponding restrictions for each one. The output data visualised in dedicated windows consist in:

-subsystems (A_i, B_i, c_i^T, b_i, x_i, where i – subsystem's number) resulting from large scale system partitioning;

-subsystems (A_{i1}, B_{i1}, c_{i1}^T, x_{i1}, A_{i2}, B_{i2}, c_{i2}^T, x_{i2}) resulting from a new partitioning of previously partitioned subsystems;

-angular systems obtained and the pivot selected for each step;

-final solution.

· *Example of Benders algorithm computation:*
Let's consider the problem: $\min \sum_i f_i y_i + \sum_{i,j} c_{ij} x_{ij}$

for $\sum_i x_{ij} \ge 1,\ j = 1,\ldots,4$

$$0 \le x_{ij} \le y_i\ (\forall) i, j \text{ where } y_i = 0 \text{ or } 1$$

$$f_i = 7 \text{ and } c_{ij} = \begin{bmatrix} 0 & 12 & 20 & 18 \\ 12 & 0 & 8 & 6 \\ 20 & 8 & 0 & 6 \\ 18 & 6 & 6 & 0 \end{bmatrix}$$

The Table 1 shows the behaviour algorithm

Y^0	$\sum_{i,j} c_{ij} x_{ij} + \sum_i f_i y_i$	$z \ge (\ \ )$	$+(\ \ )\, y_1$	$+(\ \ )\, y_2$	$+(\ \ )\, y_3$	$+(\ \ )\, y_4$	Z^0
0 1 0 0	33	26	- 5	7	-9	-1	11
1 0 1 1	27	6	7	-11	7	7	13
0 0 1 0	41	34	-17	-9	7	-3	12
1 1 1 1	16	0	7	-5	7	7	16
1 1 1 1	16						

Table 1. Example of Bender's behaviour algorithm

The associated module receives the criteria and the restriction system selecting the start vector *y*. The dual program is solved using the SIMPLEX algorithm, which leads to the solutions. The equivalent problem built up on the previous data is solved and the solutions are tested. The algorithm is reiterated in the case of a non accomplishment of all imposed restrictions.

5 Conclusions

The management of a large scale system is still considered a complex task and an open problem for the future researches. we have presented some partial results achieved in the design of a software dedicated package. The proposed algorithms may be integrated in a decision control strategy class of large scale industrial or economical systems. Some recent attempts in applying the software tools with a satisfactory evaluation concern with a reduced manufacturing cost of cement while increasing its performance and the optimisation of the schedule of roadwork by taking into account some traffic characteristics.

References:

[1] J.Rosen and B.Primal Partition Programming for Block Diagonal Matrices, *Numerische Mathematik*,6,1964, pp. 250-260

[2] El.Ghaoui, Multi Objective Robust Measurement – Scheduling for Discrete Time Systems: an LMI Approach, Proceedings of the fourth IFAC Conference, Buc., 1997, pp.66-72

[3] D.Himelbau, *Decomposition of Large Scale Problems*, Mc.Grew-Hill Book Company,1974

[4] L.Lasdon, *Optimisation Theory for Large System*, Press House, Bucharest, 1975

[5] M.D.Grigoriadis and K.Ritter, A Decomposition Method for Structured Linear and Non-linear Programs, *Computers Science Technology Rept 10*,University of Wisconsin, 1968