On The Inverse Design Problem of GMBPC

S. Tzafestas and G. Vagelatos
National Technical University of Athens
Dept. of Electrical & Computer Engineering
Intelligent Robotics & Control Unit

ABSTRACT

In this paper the relation of generalized model based predictive control (GMBPC) and pole-placement control is investigated through the solution of the inverse GMBPC design problem, i.e. through the establishment of the conditions under which a design exists that results in a given pole-placement state feedback gain matrix **K**. The result is that the set of achievable **K** is determined solely by the inherent structure of the system. In particular, if the system is observable with no eigenvalue at zero any K is achievable.

INTRODUCTION

The problem posed by GMBPC can be stated as follows: Given a linear dynamic discrete time system in state space form

$$\boldsymbol{x}(t+1) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \quad , \quad \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) \tag{1}$$

and a reference trajectory $\boldsymbol{r}(t)$, determine the input-vector $\boldsymbol{u}(t)$ so as to minimize a performance criterion of the form

$$J = \sum_{j=1}^{T} \{ [\boldsymbol{y}(t+j) - \boldsymbol{r}(t+j)]^{\mathrm{T}} \boldsymbol{\lambda}_c(j) [\boldsymbol{y}(t+j) - \boldsymbol{r}(t+j)] + \boldsymbol{u}^{\mathrm{T}}(t+j-1) \boldsymbol{\lambda}_u(j) \boldsymbol{u}(t+j-1) + \Delta\boldsymbol{u}^{\mathrm{T}}(t+j-1) \boldsymbol{\lambda}_d(j) \Delta\boldsymbol{u}(t+j-1) \} \tag{2}$$

The length of the prediction-horizon T and the weight-matrices $\boldsymbol{\lambda}_c(j)$, $\boldsymbol{\lambda}_u(j)$, $\boldsymbol{\lambda}_d(j)$ are assumed to be design parameters. One can verify that almost all of the established Predictive Control algorithms, such as GPC, DMC, etc. (ref. [6],[7]) can be reduced to the form of GMBPC, and so the results presented here are valid for all of these algorithms.

Since our attention is focused to the case of unconstrained regulator (i.e. $\boldsymbol{r}(t) = \mathbf{0}$), it is sufficient to set $\lambda_d(j) = \mathbf{0}$. Under these assumptions the solution is given by the constant state feedback law:

$$\boldsymbol{u} = -\boldsymbol{K}\boldsymbol{x}(t) \tag{3}$$

where the feedback gain matrix $\boldsymbol{K}$ is given by:

$$\boldsymbol{K} = \boldsymbol{E}_1^{\mathrm{T}} (\boldsymbol{G}^{\mathrm{T}} \boldsymbol{\Lambda}_c \boldsymbol{G} + \boldsymbol{\Lambda}_u)^{-1} \boldsymbol{G}^{\mathrm{T}} \boldsymbol{\Lambda}_c \boldsymbol{F} \tag{4}$$

with

$$\boldsymbol{\Lambda}_c = \mathrm{diag}(\boldsymbol{\lambda}_c(1), \dots, \boldsymbol{\lambda}_c(T)) \quad , \quad \boldsymbol{\Lambda}_u = \mathrm{diag}(\boldsymbol{\lambda}_u(1), \dots, \boldsymbol{\lambda}_u(T)) \tag{5}$$

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{F}_1 \\ \vdots \\ \boldsymbol{F}_T \end{bmatrix} \quad , \quad \boldsymbol{G} = \begin{bmatrix} \boldsymbol{G}_0 & & & \\ \boldsymbol{G}_1 & \boldsymbol{G}_0 & & \\ \vdots & & \ddots & \\ \boldsymbol{G}_{T-1} & \boldsymbol{G}_{T-2} & \cdots & \boldsymbol{G}_0 \end{bmatrix} \tag{6}$$

$$\boldsymbol{F}_i = \boldsymbol{C}\boldsymbol{A}^i \quad , \quad \boldsymbol{G}_i = \boldsymbol{C}\boldsymbol{A}^i\boldsymbol{B} \tag{7}$$

$$\boldsymbol{E}_1^{\mathrm{T}} = [\ \mathbf{I}_{n_u}\ \mathbf{0}_{n_u}\ \cdots\ \mathbf{0}_{n_u}\ ] \tag{8}$$

More details on the GMBPC algorithm can be found in [1].

The fact that in the unconstrained case an MBPC algorithm results in a constant state feedback law indicates that it cannot permorm better than a direct pole-placement technique. Then the question "*whether MBPC is inferior to pole-placement control, in the sense that it cannot produce* ***any*** *feedback-gain matrix* K *, that direct pole-placement can do, and hence under certain circumstances a direct pole-placement technique leads to control of better quality*" arises. This question can obviously be answered if the inverse design problem is solved: "*Under which conditions does a design exist, that results in a given feedback-gain matrix* K *, and which are the corresponding design parameters* T, Λ_c , Λ_u ". A byproduct of the solution of this problem is that a pole-placement design can be interpreted in terms of a finite horizon optimal control.

The relation of GPC and pole-placement control is also treated in other works such as [2-3], but their results concern only certain limiting cases through the achievement of dead-beat control of a reference system. In this paper we develop conditions and an algorithm that solves the problem of selecting the horizon T and the weights in a straitforward manner.

2. SOLUTION FOR A SISO SYSTEM AND DESIGN WITH WEIGHTS $\lambda_c(j)$, $\lambda_u(j)$ ONLY

In the present work we treat only the useful and revealing special case of SISO systems. Let us write the control law in transpose form:

$$K^{T} = F^{T}\Lambda_c G(G^{T}\Lambda_c G + \Lambda_u)^{-1}E_1^{T} \tag{9}$$

As long as the matrices E_1 and $(G^{T}\Lambda_c G+\Lambda_u)$ are certainly of full rank, sufficient conditions for (9) to be valid are

$$\operatorname{rank}(\Lambda_c) = T \;\leftrightarrow\; \lambda_c(j) \neq 0 \quad (j=1,\dots,T) \tag{10}$$

$$\operatorname{rank}G = T \;\leftrightarrow\; G_0 = CB \neq 0 \tag{11}$$

$$\operatorname{rank}[F^{T}|K^{T}] = \operatorname{rank}F^{T} \tag{12}$$

The occurence of delay is a common reason for condition (11) not to be fulfilled. If this is the case, i.e. a delay $\tau>1$ exists, then the minimization horizon can be structured as $\{t+\tau,\dots,t+\tau+T\}$, and the condition (11) falls into the assumption $G_{\tau-1}\neq 0$, and all the indices are shifted forward by $\tau-1$. More generaly, the same trick can be applied to any system that really couples the input to the output. From now on, we shall assume that the minimization horizon is structured such that condition (11) holds.

It must be mentioned that in general some special condition, that preserves the symmetric structure of $G^{T}\Lambda_c G+\Lambda_u$, must be added. But in the single input case, this condition is covered by the above conditions (10)-(12). This fact can be validated if one observe that the multiplication by E_1^{T} of (9) just keeps the leftmost column of $G^{T}\Lambda_c G+\Lambda_u$.

If the previous conditions are fulfilled, then matrix equation (9) can be inverted according to the following scheme:

$$\Phi^{T}K^{T} + V_{F0}L_F = \Lambda_c G(G^{T}\Lambda_c G + \Lambda_u)^{-1}E_1 \tag{13}$$

$$G^{-1}\Lambda_c^{-1}(\Phi^{T}K^{T} + V_{F0}L_F) = (G^{T}\Lambda_c G + \Lambda_u)^{-1}E_1 \tag{14}$$

$$(K\Phi + L_F^{T}V_{F0}^{T})\Lambda_c^{-1}\Gamma^{T}(G^{T}\Lambda_c G + \Lambda_u) = E_1^{T} \tag{15}$$

The columns of V_{F0} constitute a basis for the null-space of F^{T} . The vector L_F consists of free parameters that produce the space of multiple solutions of (9). If rank$F= T$, then the term

$V_{F0}L_F$ does not exist at all, i.e we have a unique solution. That is, if both F and G are of full rank then the solution is unique.

These matrices can be calculated easily with the use of singular value decomposition (in partitioned form) of F^T (ref. [4]):

$$F^{T} = U_F \Sigma_F V_F^T = \begin{bmatrix} U_{FR} & U_{F0} \end{bmatrix} \begin{bmatrix} \Sigma_{FR} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_{FR}^T \\ V_{F0}^T \end{bmatrix} \tag{16}$$

where $U_F^T U_F = I$, $V_F V_F^T = I$, $U_{FR} \in \mathbb{R}^{n_x \times r_F}$, $U_{F0} \in \mathbb{R}^{n_x \times (n_x - r_F)}$, $V_{FR} \in \mathbb{R}^{T \times r_F}$, $V_{F0} \in \mathbb{R}^{T \times (T - r_F)}$ and

$\Sigma_{FR} = \mathrm{diag}(\sigma_{F1}, \dots, \sigma_{Fr_F})$ with $\sigma_{F1} \geq \dots \geq \sigma_{Fr_F} > 0$ and $r_F = \mathrm{rank}(F) \leq T$

Hence

$$\Phi^{T} = V_{FR} \Sigma_{FR}^{-1} U_{FR}^T \tag{17}$$

With the definitions

$$M = \begin{bmatrix} K & L_F^T \end{bmatrix} \begin{bmatrix} \Phi \\ V_{F0}^T \end{bmatrix} = \begin{bmatrix} M_1 & \cdots & M_T \end{bmatrix} \tag{18}$$

equation (15) is equivalent to

$$M \Lambda_c^{-1} = (E_1^T - MG) \Lambda_u^{-1} G^T \tag{19}$$

which after some algebraic manipulations can be written in partitioned form

$$\mathrm{diag}(M_1, \dots, M_T) \begin{bmatrix} \lambda_c^{-1}(1) \\ \vdots \\ \lambda_c^{-1}(T) \end{bmatrix} = G\, \mathrm{diag}\Big((1 - \sum_{l=1}^{T} M_l G_{l-1}), \dots, \sum_{l=j}^{T} M_l G_{l-j}, \dots, M_T G_0 \Big) \begin{bmatrix} \lambda_u^{-1}(1) \\ \vdots \\ \lambda_u^{-1}(T) \end{bmatrix} \tag{20}$$

Equation (20) can be solved for either $\lambda_c^{-1}(j)$ or $\lambda_u^{-1}(j)$. We choose the first way, which with the definition

$$p_{ji} = \begin{cases} (1 - \sum_{l=1}^{T} M_l G_{\%w1}) G_0 & , \quad \text{if } i = 1 \\ -(\sum_{l=j}^{T} M_l G_{l-j}) G_{l-j} & , \quad \text{if } i = 2, \dots, j \end{cases} \qquad (j = 1, \dots, T) \tag{21}$$

gives the solution

$$\lambda_c^{-1}(1) = M^{-1} p_{11} \lambda_u^{-1}(1)$$

$$\lambda_c^{-1}(j) = \begin{cases} M_j^{-1} \sum_{i=1}^{j} p_{ji} \lambda_u^{-1}(i) & , \quad \text{if } M_j \neq 0 \\ \text{any positive value} & , \quad \text{if } M_j = 0 \end{cases} \tag{22}$$

In order to be valid such a solution, the following constraints must be satisfied:

$$M_1 p_{11} \geq 0 \quad , \quad M_1 \neq 0 \tag{23}$$

$$\left.\begin{array}{ll} M_j \sum_{i=1}^{j} p_{ji} \lambda_u^{-1}(i) \geq 0 & \text{, if } M_j \neq 0 \\ \sum_{i=1}^{j} p_{ji} \lambda_u^{-1}(i) = 0 & \text{, if } M_j = 0 \end{array}\right\} \quad j=2,\dots,T \tag{24}$$

$$\lambda_u^{-1}(i) \geq 0 \quad , \quad j=1,\dots,T$$

Introducing a criterion, such as

$$\max \sum_{i=1}^{T} \lambda_u^{-1}(i) \tag{25}$$

in order to force $\lambda_u(i)$ to the minimum possible values, and the extra constraints

$$\lambda_u^{-1}(j) \leq \lambda_{MIN}^{-1} \tag{26}$$

in order to prevent the criterion from taking an infinite value, an **always well posed** and **solvable** Linear Programming problem is formulated, from which one can determine $\lambda_u(i)$. (For details on the solution of LP solving refer to standard textbooks such as [5]).

The constraints (23) seem to constrain the feedback-gain vector $\boldsymbol{K}$. But one can observe that these constraints are fulfiled if the following sufficient conditions hold

$$\begin{bmatrix} K & L_F^T \end{bmatrix} \begin{bmatrix} \Phi_1 \\ V_{F0,1}^T \end{bmatrix} G_0 > 0 \quad , \quad \begin{bmatrix} K & L_F^T \end{bmatrix} \begin{bmatrix} \Phi \\ V_{F0}^T \end{bmatrix} \begin{bmatrix} G_0 \\ \vdots \\ G_T \end{bmatrix} \leq 1 \tag{27}$$

These two relations can be fulfiled simultaneously for any K, if L_F consists of two or more free parameters. That is, if the length of the minimization horizon is

$$T \geq r_F + 2 \tag{28}$$

Hence, the set of achievable K is determined only by the inherent structure of the system through condition (12). It is worthwile of mention that for an observable system with no eigenvalue at zero, the matrix F is of full-rank, and hence any K is achievable. It is also clear that there is no way to remove any zero eigenvalue by means of MBPC.

References

[1] S.Tzafestas, G.Vagelatos, G.Kapsiotis: *A New Generalized Model-Based Predictive Control Algorithm*, Proc. 30th IEEE Conf. on Decision & Control - Brighton U.K. 1991

[2] D.W.Clarke, C.Mohtadi: *Properties of Generalized Predictive Control*,Automatica v.25, n.6, pp.859-875, 1989

[3] C.Mohtadi-Haghighi: *Studies in Advanced Self-Tuning Algorithms*, Report No.1689/87 Univ. of Oxford

[4] G.H. Golub, C.F.Van Loan: *Matrix Computations 2nd ed.*, The John Hopkins University Press, 1989

[5] R.Fletcher: *Practical Methods of Optimization, Vol.2 Constrained Optimization*, John Wiley & Sons, 1981

[6] R.M.C. De Keyser: *Model Based Predictive Control Toolbox*, Computer Integrated Design of Controlled Idustrial Systems (J.Richalet, S.Tzafestas eds.)

[7] R.M.C.De Keyser, P.G.A.Van De Velde, F.A.G.Dumortier: *A Comparative Study of Self-adaptive Long-range Predictive Control Methods*, Automatica, v.24, n.2.

The effect of interconnections on the BAS regulator of large scale systems and the BAS regulator of N identical subsystems.

Andreas V. Pagalos and Peter P. Groumpos.
Laboratory of Automation & Robotics.
Department of Electrical Engineering.
University of Patras.
Rion 26500 GREECE.
Phone : +30-61-997295.
Fax : +30-61-997309.
E-mail: lar@grpatvx1.bitnet

Keywords: Large Scale Systems, Decentralized control, Structural control, Overlapping subsystems.

Summary.

Decomposition methodologies play the central role in solving problems in large scale systems. This role has been considerably strengthened by the recent trend of control and estimation techniques toward decentralized computation and control technology. Local estimators are built to provide information about the subsystems, which is used by the decentralized controllers to drive the subsystems to achieve the objective of the overall system.
In this paper we consider the regulation of a decentralized large scale system which consists of N linear subsystems S_1, S_2, S_N interconnected with a common linear subsystem S_o as shown in figure 1.

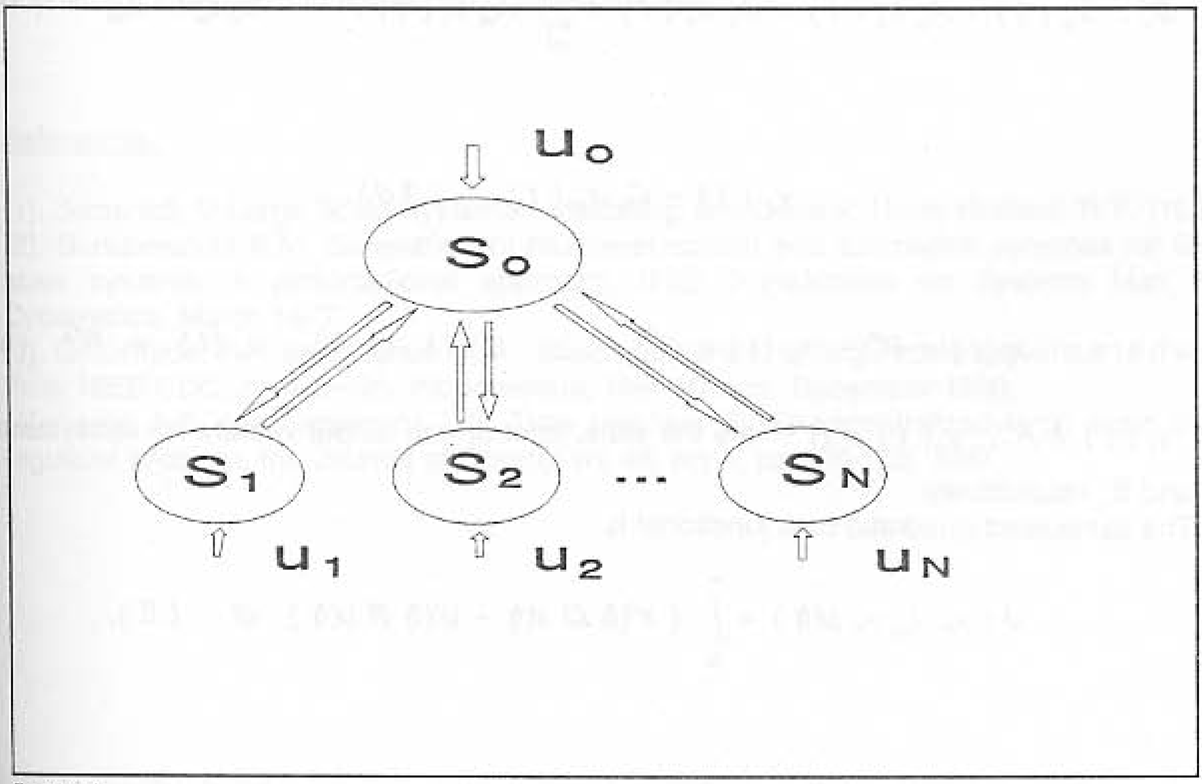

figure 1.

Since in the state space description, as will be shown, the matrix describing the dynamics of this system consists of block elements depicting an arrow structure, this system is called a Block Arrow Structure (BAS) decentralized large scale system. This class of large scale systems may appear in many cases of multilevel hierarchical systems (MHS). The regulation of the above system can be obtained by using large scale techniques proposed by many authors (Singh, Siljak, Sundareshan, Ozguner, Jamshidi). An approach that uses the structure of the system and has the advantages of parallel implementation and structural flexibility has been proposed by Groumpos and Loparo [3], Leros and Groumpos [4].

In this paper we study the effect of interconnections between the N subsystems S_i and the coordinator S_o.

The objective is to find the controllers u_i in a time interval $[t_0, \infty]$ in order to minimize an associated quadratic cost functional. Following the formulation of reference [4], which extensively deals with this problem, we have the following mathematical model:

$$S_i: \ \dot{x}_i(t) = A_i x_i(t) + B_i u_i(t) + A_{io} x_o(t), \quad x_i(t_0) = x_{i0} \qquad (1a).$$

$$y_i(t) = C_i x_i(t) \qquad (1b).$$

$$S_o: \ \dot{x}_o(t) = A_o x_o(t) + B_o u_o(t) + \sum_{i=1}^{N} A_{oi} x_i(t), \qquad x_o(t_0) = x_{o0} \qquad (1c)$$

$$y_o(t) = C_o x_o(t) \qquad (1d).$$

where $x_i(t) \in R^{n_i}$, $x_o(t) \in R^{n_o}$, $u_i(t) \in R^{r_i}$, $u_o(t) \in R^{r_o}$ and $y_i(t) \in R^{k_i}$, $y_o(t) \in R^{k_o}$ are the state, control and output vectors for subsystems S_i and S_o respectively.

The assosiated quadratic cost functional is

$$J(x_0, t_0, \infty, u(t)) = \int_{t_0}^{\infty} (x'(t)\, Q\, x(t) + u'(t)\, R\, u(t))\, dt \qquad (2).$$

where $Q \in R^{nxn}$ and $R \in R^{rxr}$ are constant, symmetric, block diagonal positive semidefinite and positive definite weighting matrices respectively.

Then the matrices $A_i \in R^{n_i x n_i}$, $A_o \in R^{n_o x n_o}$, , $B_i \in R^{n_i x r_i}$, $B_o \in R^{n_o x r_o}$ and

$C_i \in R^{k_i x n_i}$, $C_o \in R^{k_o x n_o}$ describe the dynamics, control and output distribution for S_i and S_o respectively. The interconnections (or information transfer) from S_o to S_i and S_i to S_o are represented by the matrices $A_{io} \in R^{n_i x n_o}$ and $A_{oi} \in R^{n_o x n_i}$ respectively.

It can be shown that there are cases where the interconnections play no role in system's performance (so they can be omitted), while in other cases they play significant role not only in system's performance but also in system's stability.

Another special case of the BAS regulator worth investigating is the case where all the subsystems S_1, S_2, S_N are identical. In that case, using the BAS algorithm we have a very fast solution of the problem, compared with other methods which use the overall system. Additionally we have the ability to say a priori how many subsystems S_i can be connected to the coordinator S_o in order for the overall system to be stable. In other words we can find a critical integer N_c, so the N_c identical subsystems S_i interconnected with the coordinator S_o constitute a stable system. By adding another subsystem S_i to the previous system, the overall system becomes unstable.

In the conference, the BAS algorithm and the above theoretical developments will be presented, together with some numerical examples.

References.

[1]. Jamshidi, M.Large Scale Systems - Modelling and Control. North Holland, N.Y. (1983)

[2]. Sundareshan K.M, Generation of multilevel control and estimation schemes for large scale systems. A perturbational approach, IEEE Transactions on Systems Man and Cybernetics, March 1977.

[3]. Groumpos, P.P. and Loparo, K.A., Structural control of large scale systems, Proc. IEEE-CDC, pp.422-426, Albuquerque, New Mexico, December 1980.

[4]. Leros A.P. and Groumpos P.P. Time Invariant (BAS) decentralized large scale linear regulator problem, Int. Journal of Control vol 46, No 1, pp 129-152, 1987.