

# USING GENETIC ALGORITHM TO OPTIMIZE THE TUNING PARAMETERS OF DYNAMIC MATRIX CONTROL

GUSTAVO MAIA DE ALMEIDA\*, JOSÉ LEANDRO FÉLIX SALLES\*, JOSÉ DENTI FILHO\*

*\*Laboratory of the Control and Instrumentation, Department of Electric Engineering, UFES.  
Av. Fernando Ferrari C. P. 019001 Vitória, ES, Brasil*

Emails: gmaialmeida@yahoo.com.br, jleandro@ele.ufes.br, j.denti@ele.ufes.br

**Abstract**— Dynamic Matrix Control algorithms are a powerful control method widely applied to industrial process. The idea of the work is applied genetic algorithms by tuning the parameters of the Dynamic Matrix Control. The Computation method proposed here is applies in four different type of process and after that, a comparison between the method proposed and the guidelines presented by literature, where are shown advantages of the method proposed.

**Keywords**— Genetic Algorithm, Model Predictive Control, Dynamic Matrix Control, Optimization.

## 1 Introduction

Model Predictive Control (MPC) originated in the late seventies and has developed considerably since then. The term MPC does not designate a specific control strategy but a very ample range of control methods that make an explicit use of model of the process to obtain the control signal by minimizing an objective function.

The various MPC algorithms only differ amongst themselves in the model used to represent the process and the noises and the cost function to be minimized. This type of control is of an open nature within which many works have been developed, being widely received by the academic world and by industry.

Predictive control integrates optimal, stochastic, multivariable, constrained control with dead time processes to represent time-domain control problems [1], [2] and [3]. A variety of processes, ranging from those with simple dynamics to those with long delay times, non-minimum phase zeros, or unstable dynamics, can all be controlled using MPC. Extension to multivariable plants is simple, compensation for dead time is intrinsic, and feed-forward control is introduced to compensate for measured disturbances in a natural way [4]. There are many applications of predictive control successfully in use at the present time, not only in the process industry but also applications to the control of a diversity of processes ranging from robot manipulators [5], applications for developments for distillation columns, PVC plants, steam generators or servos [6]. They usually exhibit very good performance and robustness provided that the tuning parameters have been properly selected. However the adjustment of these parameters is not made of trivial form, needing a good knowledge of the process.

Shridhar & Cooper, 1997, demonstrated a tuning strategy for single-input single output (SISO), unconstrained and first order plus dead time model by dynamic matrix control (DMC).

This work considers a new technique of tuning of the parameters of the DMC, using Genetic Algorithms (GA)[7] and carries through a comparative study between the two techniques. This paper is organized as follows: section 2 describes the classical formulation of DMC; section 3 makes an overview of GA; the applications of GA for tuning the parameters is the subject in section 4; in section 5 the comparison between the two method is presented, section 6 presents the conclusions and the works future.

## 2 Dynamic Matrix Control

Dynamic matrix control is arguably the most popular MPC algorithm currently used in the chemical process industry. Quin and Badgwell (1997) reported about 600 successful applications of DMC. It is not surprising why DMC, one of the earliest formulations of MPC, represents the industry standard today. A large part of DMC appeal is drawn from a intuitive use of a finite step response (or convolution) model of the process, a quadratic performance objective over a finite prediction horizon, and optimal manipulated input moves computed as the solution to a least squares problem.

The DMC method was proposed by [9] and has been widely accepted in the industrial world, mainly by petrochemical industries [6]. This algorithm is appropriate by the control of the process with raised degree of interaction between the variable, high orders and dominant delays. The DMC presents benefits in the control of systems with the following characteristics:

- Inherently multivariable with strong interaction between the controlled variable;
- They present potential for optimization in real time;
- They possess limits in the equipment and the operational conditions.

The objective of algorithm DMC is to calculate the actions of control, represented for increments in the manipulated variable, through the minimization of cost function in the Equation 1:

$$J = \sum_{k=1}^{h_p} [\hat{y}(t+k|t) - r(t+k)]^2 + \sum_{k=1}^{h_c} \lambda [\Delta u(t+k-1)]^2 \quad (1)$$

Where:  $\hat{y}(t+k|t)$  is an optimum k-step ahead prediction of the system output at time t,  $h_p$  is prediction horizons,  $h_c$  is the control horizons associated to the control signal  $u(t)$ ,  $\Delta u(t) = u(t) - u(t-1)$ ,  $\lambda$  is weighting sequences and  $r(t+k)$  is the future evolution of reference.

So that this problem of optimization can be decided, it is necessary that if it has a model of the process to calculate the prediction of the output. The used model in algorithm DMC is known as model of response to the step.

### 2.1 Prediction

The process model employed in this formulation is the step response of the plant, while the disturbance is considered to keep constant along the horizon. The procedure to obtain the predictions is as follows:

As a step response model is given by Equation 2:

$$y(t) = \sum_{i=1}^{\infty} g_i \Delta u(t-i) + n(t) \quad (2)$$

Where:  $y(t)$  is the system output,  $u(t)$  is control signal,  $\Delta u(t) = u(t) - u(t-1)$ ,  $g_i$  is the  $i$ th unit step response coefficient of the process and  $n(t)$  is a perturbation acting in the process.

The predicted values along the horizon will be:

$$\hat{y}(t+k|t) = \sum_{i=1}^{\infty} g_i \Delta u(t+k-i) + \hat{n}(t+k|t) \quad (3)$$

The prediction equation can be rewritten, separating the referring terms to the contributions of the past, present and future [1], as described in the Equation 5.

$$\hat{y}(t+k|t) = \sum_{i=1}^k g_i \Delta u(t+k-i) + \sum_{i=k+1}^{\infty} g_i \Delta u(t+k-i) + \hat{n}(t+k|t) \quad (4)$$

The first parcel of this equation 5 represents the present and future contributions of the manipulated variable (control). The second parcel represents the contribution for the output variable,

until the instant  $t-1$ , of the increments passed in the manipulated variable. Disturbance are considered to be constant, that is,  $\hat{n}(t+k|t) = \hat{n}(t|t) = y_m(t) - \hat{y}(t|t)$ . If the process is asymptotically stable, the coefficients  $g_i$  of the step response tend to a constant value after  $N$  sampling periods, Then it can be written conformable Equation 5:

$$\hat{y}(t+k|t) = \sum_{i=1}^k g_i \Delta u(t+k-i) + \sum_{i=k+1}^N g_i \Delta u(t+k-i) + y_m(t) - \sum_{i=k+1}^N g_i \Delta u(t-i) \quad (5)$$

In this way:

$$\hat{y}(t+k|t) = \sum_{i=1}^k g_i \Delta u(t+k-i) + f(t+k) \quad (6)$$

It represents the equation of prediction of the process, where the first term represents the forced response, and  $f(t+k)$  is the free response of the system, that is, the part of the response that does not depend on the future control actions and is given by:

$$f(t+k) = y_m(t) + \sum_{i=1}^N (g_{k+1} - g_i) \Delta u(t-i)$$

Now the predictions can be computed along the prediction horizon ( $k = 1, 2, \dots, h_p$ ), considering  $h_c$  control actions.

$$\begin{aligned} \hat{y}(t+1|t) &= g_1 \Delta u(t) + f(t+1) \\ \hat{y}(t+2|t) &= g_2 \Delta u(t) + g_1 \Delta u(t+1) + f(t+2) \\ &\vdots \\ \hat{y}(t+h_p|t) &= \sum_{i=h_p-h_c+1}^{h_p} g_i \Delta u(t+h_p-i) + f(t+p) \end{aligned}$$

In the matrical form:

$$\begin{bmatrix} \hat{y}(t+1|t) \\ \hat{y}(t+2|t) \\ \vdots \\ \hat{y}(t+h_p|t) \end{bmatrix} = \begin{bmatrix} g_1 & 0 & \cdots & 0 \\ g_2 & g_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{h_p} & g_{h_p-1} & \cdots & g_{h_p-h_c+1} \end{bmatrix} * \begin{bmatrix} \Delta u(t) \\ \Delta u(t+1) \\ \vdots \\ \Delta u(t+h_c-1) \end{bmatrix} + \begin{bmatrix} f(t+1) \\ f(t+2) \\ \vdots \\ f(t+h+p) \end{bmatrix}$$

It can be written that:

$$y = \mathbf{G} \Delta \mathbf{u} + \mathbf{f} \quad (7)$$

## 2.2 Control Algorithm

The cost function can be written as Equation 8:

$$J = (\mathbf{G}\Delta\mathbf{u} + \mathbf{f} - \mathbf{r})^T (\mathbf{G}\Delta\mathbf{u} + \mathbf{f} - \mathbf{r}) + \lambda \Delta\mathbf{u}^T \Delta\mathbf{u} \quad (8)$$

The Equation 8 can be rewrite as

$$J = \frac{1}{2} \Delta\mathbf{u}^T \mathbf{H} \Delta\mathbf{u} + \mathbf{b}^T \Delta\mathbf{u} + f_0 \quad (9)$$

Where:

$$\begin{aligned} \mathbf{H} &= 2(\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I}); \\ \mathbf{b}^T &= 2(\mathbf{f} - \mathbf{r})^T \mathbf{G}; \\ f_0 &= (\mathbf{f} - \mathbf{r})^T (\mathbf{f} - \mathbf{r}). \end{aligned}$$

The minimum of  $J$ , assuming there are no constraints on the control signal, can be found by making the gradient of  $J$  equal to zero, which leads to:

$$\Delta\mathbf{u} = -\mathbf{H}^{-1} \mathbf{b} = (\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I})^{-1} \mathbf{G}^T (\mathbf{f} - \mathbf{r})$$

The control signal variation that is actually sent to the process is the first element of vector  $\Delta\mathbf{u}$ , that is given by:

$$\Delta u(t) = \mathbf{K}(\mathbf{r} - \mathbf{f})$$

Where  $\mathbf{K}$  is the first row of matrix  $(\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I})^{-1} \mathbf{G}^T$

## 3 Genetic Algorithm

Genetic algorithms are search algorithms based on the mechanics of natural selection of Darwin and natural genetics of Mendel. They combine survival of the fittest among string structures with a structures yet randomized information exchange to form a search algorithm with some of the innovative flair of human search. In every generation, a new set of artificial creatures (strings) is created using bits and pieces of the fittest of the old; an occasional new part is tried for good measure. While randomized, genetic algorithms are no simple random walk. They efficiently exploit historical information to speculate on new search points with expected improved performance. GAs were first presented by [10]. GAs have been used in many diverse areas such as function optimization [11], image processing [12] among others. A GA is a parallel global search technique that emulates natural genetic operators and works on a population representing different parameter vectors whose optimal value with respect to some (fitness) criterion is searched. This technique includes operations such as reproduction, crossover and mutation. These operators work with a number of artificial creatures called generation. By exchanging information from each individual in a population. GAs preserve a better individual and yield higher fitness generation by generation such that the performance can be improved. Next, we will briefly describe the basic operators in a GA.

## 3.1 Reproduction

Reproduction is a process in which a new generation of population is formed by selecting individuals from an existing population, according to their fitness. This process results in individuals with higher fitness values obtaining one or more copies in the next generation, while low fitness individuals may have none. Note, however, that reproduction does not generate new individuals but only favorable the percentage of fit individuals in a population of given size.

## 3.2 Crossover

This operation provides a mechanism for individual to exchange information via probabilistic process. This operation takes two parents individuals and produces two offspring who are new individual whose characteristics are a combination of those of their parents.

## 3.3 Mutation

Mutation is an operation where some characteristics of the individuals are randomly modified, yielding a new individual. Here, the operation simply consists in randomly changing the value of one bit of the string representing an individual.

## 4 Genetic Algorithm Applied to DMC

In this section, GA described above is used to optimize the parameters of a DMC controller. Firstly, GA will create the population of 40 individuals that contain the parameters necessary to minimize the objective function. The individuals of the GA are showed as:

$$[h_c \ h_p \ \lambda]$$

After that, the GA algorithm will compute the fitness of each individual in the population and will select the best individuals of this generation. The fitness functions presented in this work is:

$$Fit(h_c, h_p, \lambda) = \frac{1}{\sum abs(y - r)^2}$$

After that, the genetic operators (reproduction, crossover and mutation) will be carried through in each individual to create a new generation. GA runs iteratively during 20 generations, presenting at the end of the procedure the best individual that is considered the result of simulation.

## 5 Results

In this section, the feasibility of the proposed method is tested on four examples, where will be do the some comparison between [8] and the proposed method. For the optimal selection of the design parameters with GA, the population size,

Table 1: Summary of Simulation Results

Process		$h_p$	$h_c$	$\lambda$
$G_1(s)$	GA	1	6	26.53
	[8]	8	26	11.36
$G_2(s)$	GA	1	9	5.88
	[8]	7	19	6.91
$G_3(s)$	GA	1	9	6.22
	[8]	13	53	40.3
$G_4(s)$	GA	6	6	28.95
	[8]	6	16	4.96

the crossover rate, the mutation rate and the generation number are chosen to be 40, 0.95, 0.05 and 20, respectively. Table I shows the comparison between the method demonstrated by [8] and the proposed method, in which are shown the  $h_c$ ,  $h_p$  and  $\lambda$ .

### 5.1 Example 1

The first examples is a second order process with a relatively large dead time, where the transfer function of the process can be shown in the Equation 10.

$$G_1(s) = \frac{e^{-50s}}{(150s + 1)(25s + 1)} \quad (10)$$

The figure 1 show the response presented by two method, for the parameters shown in the table 1.

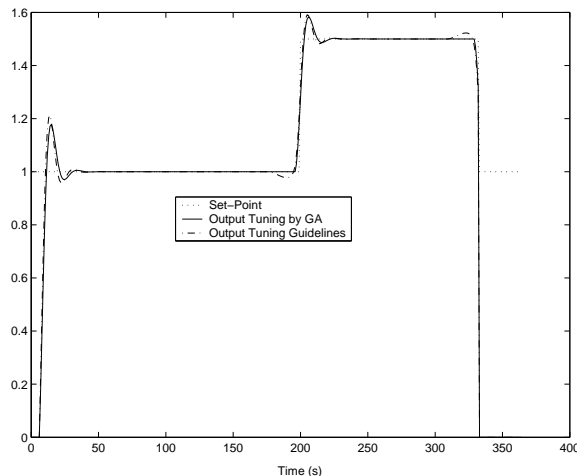


Figure 1: Closed-loop system

### 5.2 Example 2

The second example exhibits inverse response characteristics which can be shown in the Equation 11

$$G_2(s) = \frac{(1 - 50s)e^{-10s}}{(100s + 1)^2} \quad (11)$$

The figure 2 show the response presented by two method, for the parameters shown in the table 1.

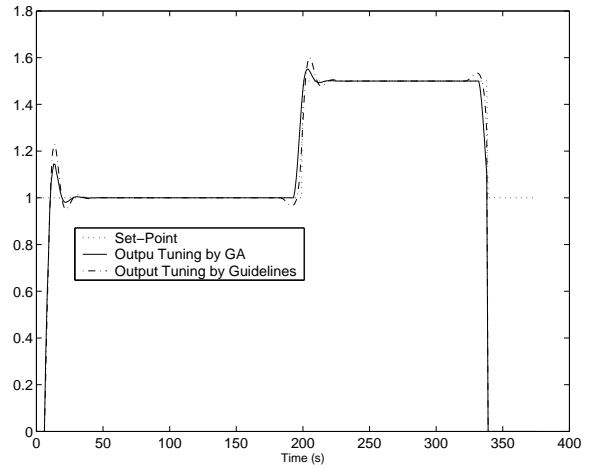


Figure 2: Closed-loop system

### 5.3 Example 3

The third example has a minimum phase behavior, which can be shown in the Equation 12.

$$G_3(s) = \frac{(50s + 1)e^{-10s}}{(100s + 1)} \quad (12)$$

The figure 3 show the response presented by two method, for the parameters shown in the table 1.

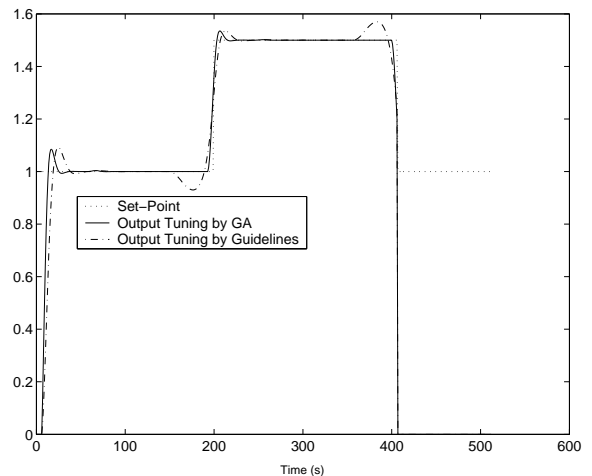


Figure 3: Closed-loop system

### 5.4 Example 4

The last example is a fourth-order process with sluggish open-loop dynamic which can be shown in the Equation 13.

$$G_4(s) = \frac{e^{-10s}}{(50s + 1)^4} \quad (13)$$

The figure 4 show the response presented by two method, for the parameters shown in the table 1

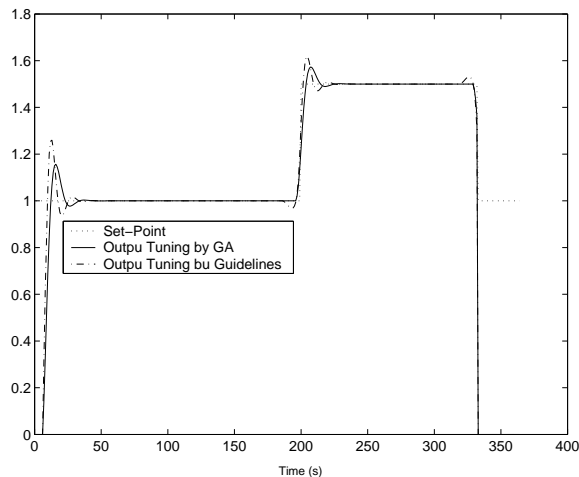


Figure 4: Closed-loop system

## 6 Conclusion

This work presented a new technique of tuning of the parameters of the DMC using Genetic Algorithms, and presented a comparison with one existing technique already in literature.

Through the presented results, one perceives that the value of the output is well next for the two techniques, however, a great difference in the values of the parameters is perceived, where the  $h_c$  and  $h_p$  shown in [8] are bigger than the  $h_c$  and  $h_p$  presented by GA.

The matrices in the predictive control depend exclusively of the size of parameters, thus, the practical implementation using GA will be simpler, a time that the predictive control also work with inversion of these matrices.

One another important fact is the case where constrained will be considered, a time that, the number of constrained depends exclusively of these parameters, being bigger how much bigger they will be these, becoming more still, when it will be to be about multivariable case.

It is important to also mention the fact of system, not being restricted the first-order plus dead time systems, as is the case of the technique shown in [8].

With this, concludes that the presented technique has a promising future in the predictive control, with perspective of use in multivariable and constrained systems.

## References

- [1] Camacho, E. F. and Bordóns, C. (2004). *Model Predictive Control*. New York: Springer.
- [2] Maciejowski, J. (2002). *Predictive Control with Constraints*. Englewood Cliffs, NJ: Prentice Hall.
- [3] Rossiter, J. A. (2003). *Model-Based Predictive Control: A Practical Approach*. Boca Raton, FL: CRC Press.
- [4] Guzmán, J. L., Berenguel, M. and Dormido, S. (2005) *Interactive Teaching of Constrained Generalized Predictive Control* IEEE Control System Magazine, pp 52–66.
- [5] Lelic, M. A. and Wellstead, P. E. (1987). *Generalized Pole Placement Self Tuning Controller. Part 1. Basic Algorithm*. Int. J. Control, 46(2):547:568.
- [6] Qin, S. J. and Badgwell, T. A. (1997) *An overview of industrial model predictive control technology* Fifth International Conference of Chemical Process Control, pp. 232–256.
- [7] Almeida, G. M., Salles, J. L. F. and Filho, J. D. (2006) *Using Genetic Algorithms for Tuning the Parameters of Generalized Predictive Control*. XVI Congresso Brasileiro de Automática.
- [8] Shridhar, R. and Cooper, D. J. (1997). *A Tuning Strategy for Unconstrained SISO Model Predictive Control* Ind. Eng. Chem. Res, 36, 729–746.
- [9] Cutler, C. R. and Ramaker, B. C. (1980). *Dynamic Matrix Control - A Computer Control Algorithm* In Automatic Control Conference, San Francisco.
- [10] Holland, J. H. (1975). *Adaptation in Neural and Artificial Systems* Ann Arbor, Michigan: University of Michigan Press.
- [11] Dejong, K. A. (1975). *An analysis of the behavior of a class of genetic adaptive systems*. PhD Dissertation, University of Michigan, Ann Arbor, Michigan.
- [12] Fitzpatrick, J. M., Grefenstette, J. J. and Van Gucht, D. (1984). *Image registration by genetic search* Proceedings of the IEEE Southeast Conference, New York: IEEE, 460–464.
- [13] Das, R. and Goldberg, D. E. (1988) *Discrete-time parameters estimation with genetic algorithm* Proceedings of the 19th Annual Pittsburg Conference on Modeling and Simulation, Pittsburg, California, USA.
- [14] Tang, K. S., Man, K. F., Kwong, S. and He, Q. (1996). *Genetic Algorithms and thier applications*. IEEE Signal Processing Magazine, 13, 22–37.