Nonlinear Model Based Predictive Control using Fuzzy Models and Genetic Algorithms

M. Boumehraz[1], K. Benmahammed[2], D. Saigaa[1]

(1) Laboratoire MSE, Département d'Electronique, Université de Biskra (medboumehraz@netcourrier.com)

(2) Département d'Electronique, Université de Sétif.

Abstract : Nonlinear model base predictive control (MBPC) is one of the most powerful techniques in process control, however, two main problems need to be considered : obtaining a suitable nonlinear model and an efficient optimization procedure. In this paper, fuzzy Takagi-Sugeno (TS) models are used for nonlinear systems modeling and the optimization routine is based on genetic algorithms(GAs). First a fuzzy TS model of the non-linear system is derived from input-output data by means of fuzzy clustering and least squares parameter estimation. Next, the fuzzy model is used in an MBPC structure where the critical element is the optimisation routine which is nonconvex and thus difficult to solve. A genetic algorithm based approach is proposed to deal with this problem. The efficiency of this approach had been demonstrated with a simulation example.

1. Introduction

Model based predictive control MBPC was developed in the process industries in the 1960's and 70's, based primarily on heuristic ideas and input-output step and impulse response models[1,2]. The basic principle is to solve an open-loop optimal control problem at each time step. The decision variables are a set of future manipulated variable and the objective function is to minimize deviations from a desired trajectory; constraints on manipulated, state and output variables are naturally handled in this formulation. Feedback is handled by providing a model update at each time, and performing the optimization again[3,4].

The classical MBPC algorithm use linear models of the process to predict the output of the process over the prediction horizon. When no model of the system is available, the classical system identification theory provides possible solutions to the problem, but when the process is non-linear and it is driven over a wide dynamic operating range, the use of linear models becomes impractical, and the use of non-linear models for control becomes a necessity[5].

In recent years, the use of fuzzy systems for nonlinear system modelling has proved to be extremely successful[6,7]. In this paper we propose to use fuzzy systems to model non-linear systems in an MBPC structure.

An additional difficulty is that the optimization problems to be solved on line are generally nonlinear programms without any redeeming features, which implies that converegence to global optimum cannot be assured[2]. Often the nonlinear optimization problem is solved by iterative methods such as sequential quadratic rogramming (SQP), which is computationally very expensive with no guarantee of convergence to a global optimum. Genetic Algorithms (GAs) [5] are potential methods as optimisation techniques for complex problems. The aim of this paper is to use fuzzy systems as models for the plant in an MBPC strategy and to solve the non-linear constrained optimisation problem by genetic Algorithms. The paper is organized as follows. Section 2 provide elementary ground on MBPC. Section 3 describes the concept of non-linear system modelling by fuzzy systems. Section 4 deals with the use of genetic algorithms to solve constrained optimisation problems in MBPC. Section 5, presents a simulation example to demonstrate the effectiveness of the proposed approach. Section 6 draws some conclusions from the presented work.

2. Basic elements of model based predictive control

MBPC also known as receding horizon control (RHC) is a general methodology for solving control problems in the time

domain. It is based on three main concepts [3,9]:
1. Explicit use of a model to predict the process output.
2. Computation of a sequence of future control actions by minimizing a given objective function.
3. The use of the receding horizon strategy: only the first control action in the sequence is applied, the horizons are moved one sample period towards the future, and optimization is repeated.

Because of the optimization approach and the explicit use of the process model, MBPC can realize multivariable optimal control, deal with nonlinear processes and handle constraints efficiently. The three basic elements of MBPC: (i) a model which describes the process, (ii) a goal, defined by an objective function and constraints (optional), and (iii) an optimization procedure.

The future process outputs are predicted over the prediction horizon H_p using the model of the process : $\hat{\mathbf{y}}(k+i)$ for $i=1,\dots,H_p$. Theses values depend on the current process state, and the future control signal $\mathbf{u}(k+i)$ for $i=0,\dots,H_c-1$, where $H_c \le H_p$ is the control horizon. The control variable is manipulated only within the control horizon and remains constant afterwards.

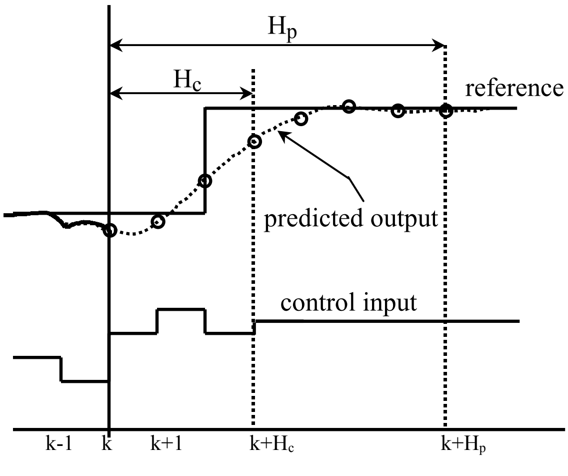

Figure 1. The basic principle of model based predictive control

2.1. Process model

The model must describe the system well and it does not matter what type of model is used to this end: a black-box, a gray-box, or a white-box one[9,10]. The future process outputs $\hat{\mathbf{y}}(k+i)$ for $i=1,\dots,H_p$, are predicted over the prediction horizon H_p using a model of the process.

2.2. Objective function

The objective function mathematically describes the control goal. In general, good tracking of the reference trajectory is required, with low control energy consumption. These requirements can be expressed by the general form [ ]:

$$J=\sum_{i=0}^{H_p}[r(k+i)-y(k+i)]Q[r(k+i)-y(k+i)]^T+\sum_{i=1}^{H_c}u(k+i)Pu^T(k+i)+$$
$$\sum_{i=0}^{H_p}[\Delta r(k+i)-\Delta y(k+i)]\Delta Q[\Delta r(k+i)-\Delta y(k+i)]^T+$$
$$\sum_{i=1}^{H_c}\Delta u(k+i)\Delta P\Delta u^T(k+i) \qquad (1)$$

Where r(k) is the reference, P, ΔP, Q and ΔQ are positive definite weight matrices. Level and rate constraints of the control input and/or other process variables can be specified as a part of the optimization problem.

In MBPC equation (1) is usually used in combination with input and output constraints:

$$u_{\min}\le u\le u_{\max}$$
$$\Delta u_{\min}\le \Delta u\le \Delta u_{\max}$$
$$y_{\min}\le y\le y_{\max} \qquad (2)$$
$$\Delta y_{\min}\le \Delta y\le \Delta y_{\max}$$

Other constraints can be implemented in a straightforward way, e.g. state constraints for state space models [2].

2.3 Optimisation

Model predictive control requires an optimization procedure by which a sequence of optimal control signals can be found at each step.

Linear MBPC problem with constraints form a convex optimization problem, that can be efficiently solved by numerical methods[2].

In the presence of nonlinearities and constraints, a non-convex optimization problem must be solved at each sampling period. This hampers the application of nonlinear MBPC to fast systems where iterative optimization techniques cannot be properly used, due to short sampling periods and extensive computation times[9].

Moreover, iterative optimization algorithms, such as the Nelder-Mead method, the multi-step Newton-type algorithm[11], or sequential quadratic programming(SQP)[12], usually converge to local minima, which results in poor solutions of the optimization problem. For efficiency many vendors use heuristic methods, for example, by using dynamic matrices[2].

In this paper, a genetic algorithm based approach is used to solve the MBPC constrained optimisation problem.

3. Fuzzy modeling and identification

3.1 Fuzzy modeling

Takagi-Sugeno fuzzy models are universal approximators[7], so they are suitable to model a large class of non linear systems.

A general TS fuzzy model can be represented by a set of fuzzy rules having the following form:

R_i : if x_1 is A_{i1} and ….. x_n is A_{in} then

$$\hat{y}_i = f(x_1,x_2,\ldots,x_n) \quad i=1,\ldots,R \quad (3)$$

Here $\mathbf{x} = [x_1,x_2,\ldots,x_n]^T$ is the input vector, $\hat{y}_i$ is the output of the i^{th} rule, and $A_{i1},\ldots,A_{in}$ are fuzzy sets defined in the antecedent space by membership functions $\mu_{Aij}(x_j) : \mathbb{R} \rightarrow [0,1]$, f is the rule consequent and R is the number of rules. The rule consequents are often taken to be linear functions of inputs :

$$f(x_1,x_2,\ldots,x_n) = C_{i0} + C_{i1}x_1 + \ldots + C_{in}x_n.$$

Where the parameters C_{ij} i=1,…,R and j=0,..,n are the consequent parameters.

The total output of the model is computed by aggregating the individual contributions of the rules :

$$y^m = \sum_{i=1}^{N} \alpha_i(x) y_i^m \quad (4)$$

where $\alpha_i(x)$ is the normalized firing strength of the i^{th} rule:

$$\alpha_i = \frac{\prod_{j=1}^{n_i} \mu_{Aij}(x_j)}{\sum_{i=1}^{N}\prod_{j=1}^{n_i} \mu_{Aij}(x_j} \quad (5)$$

3.2 Identification

Given N input-output data pairs $\{\mathbf{x}_k,\mathbf{u}_k\}$, the typical identification of the TS model is done in two steps : first the fuzzy rule antecedents are determined and then least squares parameter estimation is applied to determine the consequents.

In this paper, the antecedents of the initial fuzzy rule bases are obtained by fuzzy clustering in the product space of the sampled input-output data. Each cluster represent a certain region in the systems input-output state space , and correspond to a rule in the rule base. The fuzzy sets in the rule antecedents are obtained by projecting the cluster into the domain of the various inputs.

4. Optimisation

Genetic Algorithms (GAs) as an optimization method have been lately applied as an alternative to classical optimization methods. Their ability to find the optimum of functions where classical methods have difficulties (e.g. non derivative functions), is one of the most properties of this technique. In this paper, a genetic algorithm is used to solve the MBPC optimization problem. The algorithm is derived from the steady-state GA and utilizes floating point encoding. A fitness function of the optimizer is defined by the objective function of the model predictive control formulation.

4.1. Encoding

Every individual $\{p_i ; i=1,\ldots,N_{pop}\}$ in the population of a genetic algorithm determines a control sequence:

$$p_i = \{u_i(k), u_i(k+1), \ldots, u_i(k+H_c-1)\} \quad (6)$$

the elements of which are represented as floating point numbers. An individual p_i is described by a set of H_c numbers which are selected within the admissible interval [u_{min} , u_{max}] with absolute differences {Δu_i (k+j); j =1,…,H_c-1 } not exceeding the prescribed value $\Delta u_{i\,max}$.

4.2 Initialization

In order to provide for faster convergence of the genetic algorithm, suitable initialization procedure should be specified. In this paper we combine random initialisation with the interevolution steady-state principle :

Randomly Initialization : Random control trajectories are generated in accordance with the constraints presented in Eq.(2).

Inter-evolution exchange : The best solutions of the last optimization cycle are used in the next period.

4.2 Termination conditions

The termination function is used to determine when the optimization loop should be finished. Selection of a fixed number of generations is not very suitable because evolution may converge earlier. Therefore we introduce a new convergence measure to determine the termination condition. Deviations of all signals of the best individual in the population are scanned for the last N_{conv} generations. The termination condition is fulfilled when either the relative maximum deviation becomes smaller than a prescribed value p_{conv} or the maximum number of generations N_{gen} is exceeded.

4.3 Constraints handling

Manipulated variables (MVs) Constraints are directly handled in the AG reproduction procedure. Each individual p_i is described by a set of H_c numbers which are selected within the admissible interval [$\mathbf{u}_{min}$, $\mathbf{u}_{max}$] with absolute differences {$\Delta \mathbf{u}_i$ (k+j); j =1,…,H_c-1 } not exceeding the prescribed value $\Delta \mathbf{u}_{i\,min}$ and $\Delta u_{i\,max}$.

Controlled variables (CVs) constraints are handled by penalizing infeasible individuals[13]. The fitness function is modified and the violation of constraints is specified by penalties. The modified fitness function for an individual p is evaluated by :

$$eval(p) = f(p) + Q(p) \qquad (7)$$

where f(p) is the fitness function without constraints and Q(p) is a penalty function corresponding to constraints violation. The value of Q(p) is proportional to the amplitude and the time of the constraint violation.

5. Simulation

Consider the non-linear discrete system described by the equation :

$$y(k+1) = \frac{y(k)}{1+y^2(k)} + u(k) \qquad (8)$$

A fuzzy model is obtained using input/output data sets generated by random values of $u(k) \in [-1.0, 1.0]$. Fuzzy model membership functions are obtained by the Gustafson-Kessel[14] clustering algorithm and the consequent parameters are derived with a least squares algorithm.

Figure 2 represent the obtained membership functions for u(k) and y(k).

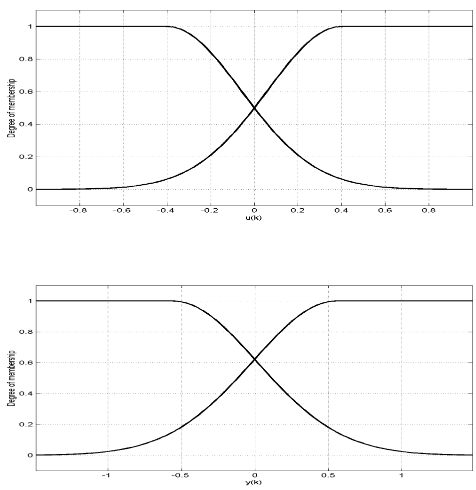

Figure 2. Membership functions for u(k) and y(k)

The fuzzy model is described by four the following rules:

1. If y(k) is Ω_{11} and u(k) is Ω_{21} then
y(k+1) = -0.1537 + 1.00y(k) - 0.6896u(k)
2. If y(k) is Ω_{11} and u(k) is Ω_{22} then
y(k+1) = -0.1539 + 1.00y(k) - 0.6902u(k)
3. If y(k) is Ω_{12} and u(k) is Ω_{21} then
y(k+1) = -0.1501 + 1.00y(k) 0.6857u(k)
2. If y(k) is Ω_{12} and u(k) is Ω_{22} then
y(k+1) = -0.1501 + 1.001y(k) 0.6857u(k)

(9)

The goal of the predictive control is to generate suitable sequence of actions $u(k)\in[-1.0,1.0]$ so to minimize the objective function given by equation (1) where the reference signal is : r(k)=0.5 for k=1,…,50; r(k)=-0.5 for k=51,…,100 and r(k) = 0.2 for k=101,…,200.
The constraints are :

$$-1.0\leq u(k)\leq 1.0$$
$$-1.0\leq y(k)\leq 1.0 \qquad (10)$$

The prediction horizon $H_p=5$ and the control horizon is $H_c = 3$. The weight matrices in equation (1) are P = 1.0 , Q =1.0 ΔP=0 and ΔQ =0.
Figure 3 represents the system output and the reference, the corresponding control input is represented in figure 4.

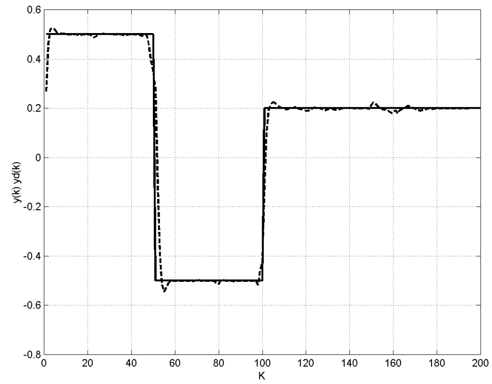

Figure 3. System output (dashed line) and the desired response (solid line)

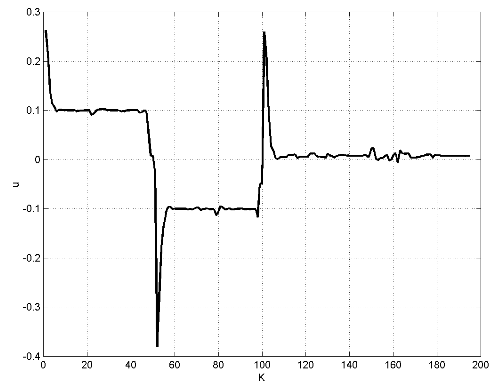

Figure 4. Control input

6. Conclusions

A non-linear model based predictive control strategy based on fuzzy models and genetic algorithms had been presented. This strategy is a very efficient non-linear model based predictive control approach.

Future work should be done to improve the computation time of the optimiser by choosing special operators to enhance the convergence of the genetic algorithm. A combination with iterative methods may decrease the computational time and avoid the convergence to local minima.

References :

[1]- E. S. Camacho, C. Bordons, “Model predictive control in process industry”, Springer, London, U.K. 1995.

[2]-M. Morari, J. H. Lee, “Model predictive control: past, present and future”, Computers and chemical engineering, 23-(1999), 667-682.

[3]- J. Richalet, “Industrial applications of model based predictive control”, Automatica 29(1993) pp 1251-1274.

[4]-C. E. Garcia, D. M. Prett, M. Morari, “Model predictive control theory and practice : a survey”, Automatica 25 (1989), 335-348.

[5]-D. E. Goldberg, “Genetic algorithms in search, optimisation and machine learning”, Addison Wesley, Reading, MA, 1989.

[6]- L. Magni, “Non-linear receding horizon control: theory and applications”, PhD thesi, University degli studi di Paria, Italy, 1998.

[7]- T. Takagi, M. Sugeno, “Fuzzy identification of systems and its applications to modeling and control”, IEEE transactions on Systems Men and Cybernetics, Vol. 15,(1995), pp 116-132.

[8]- R. Babuska, J. A. Roubos, H. B. Verbruggen, “Identification of MIMO systems by inputoutput TS fuzzy models”, in FUZZ-IEEE, vol. 1 Anchorage, Alaska, 1998, pp 657-662.

[9]- J. A. Roubos, S. Moullov, R. Babuska, H. B. Verbruggen, “Fuzzy model predictive control using Takagi-Sugeno models”, International Journal of Approximate reasoning, 22 (1999) pp 3-30.

[10]- J. Sjoberg, Q. Zhang, L. Ljung, A. Benveniste, “Non-linear black-box modeling in system identification : A unified approach”, Automatica, 31(1995) pp 1691-1724.

[11]- M. M. C. Oleviera, L. T. Biegler, " An extension of Newton-type algorithms for non-linear process control", Automatica, 31(1995), pp 281-285.
[12]- L. T. Biegler, "Advances in non-linear programming, concepts for process control", IFAC Adchem Conference, Banff, Canada, 1997, pp 857-598.
[13]- M. Schoenauer, S. Xanthakis, " Constrained GA optimisation", proceedings of the 5th ICGA, Morgan Kaufman, 1993, pp 573-580.
[14] –A. Fiordaliso, "Systèmes flous et prévision de séries temporelles", Hermes, Paris, 1999.