Analytic Design of a Fuzzy Control for Mechanical Systems

D. M. PORTO, P. AMATO
Soft Computing Group
STMicroelectronics S.r.l.
Stradale Primosole 50,Catania
ITALY

Abstract: - In this paper an alternative approach for the design of a fuzzy controller is proposed. Fuzzy controllers have been successfully adopted for nonlinear plants but their design is based on human experience because it does not exists a well defined general theory to handle them. To this aim multi-values logic is introduced and, starting from a classical control law, fuzzy sets and fuzzy rules are derived in order to control a nonlinear second order damped oscillator. Since this model strictly resembles that one of a car suspension, the presented control can be very useful for automotive applications.

Key-Words: - Fuzzy Logic, Many-Valued Logic, Fuzzy Controllers, Mechanical Suspension, Nonlinear Systems.

1 Introduction

In the last decade an increasing interest has been devoted to the possibility of controlling nonlinear systems using fuzzy logic. In fact, in most cases nonlinear plants cannot be linearized in order to use linear control strategies and nonlinear control laws are not so easy to be found. Fuzzy controllers have demonstrated to be a very good solution for systems presenting strong nonlinearities. However their design is usually quite approximate because it is based on an *a priori* human knowledge and it is not rigorous from a theoretical viewpoint. To overcome the first problem, fuzzy set theory (FS) is integrated with genetic algorithms (GA) and neural network (NN). These three techniques are the main constituents of Soft Computing paradigm (SC). Nevertheless even SC lacks of solid theoretical foundations and many of the parameter choices requested in the development of an application are driven, again, by experience. In this paper an automatic method for the definition of fuzzy sets and rules is proposed in order to provide a suitable nonlinear control law. The successfulness of this method is based on theoretical results obtained in the frame of many-valued logic. This method can be seen as an alternative to SC techniques in situations where the classical model is known.

2 Classic Control for a 2nd order system

A damped oscillator is a mechanical system composed, in the simplest case, by a mass M, a coil K and a damping element C (see figure 1).

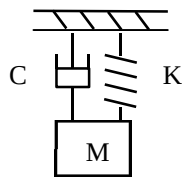

Fig.1

Considering only one-dimensional displacement x of the mass, and by applying Newton's Laws, the system in Fig. 1 can be described by the following nonlinear differential equation

$$M\ddot{x} + C(\dot{x}) + K(x) = u \qquad (1)$$

where u is a suitable control force having the same direction of x. In terms of state equation, assuming $x_1 = x,\ \ x_2 = \dot{x},$ it can be written:

$$\begin{aligned} \dot{x}_1 &= f_1(x_1, x_2) = x_2 \\ \dot{x}_2 &= f_2(x_1, x_2) = -\frac{1}{M}\left[K(x_1) + C(x_2) - u\right] \end{aligned} \qquad (2)$$

In order to avoid undesired oscillations, a fast and adequately damped response to impulsive inputs should be obtained. According to project

requirements, this fact means that the poles of the controlled system must lie in a certain region of the complex plane (see Fig. 2). In fact, in case of linearity ($K(x_1)=Kx_1$ and $C(x_2)=Cx_2$), from equation (1) or (2) it is easy to derive the following transfer function:

$$\frac{X(s)}{U(s)} = \frac{1}{Ms^2 + Cs + K} \tag{3}$$

It has usually two complex conjugate poles $s_{1,2}=s\pm jw$ whose real and imaginary part must belong to a certain range of values after applying a suitable control strategy (e.g. . $s<s_m$ $|w|<ms$)

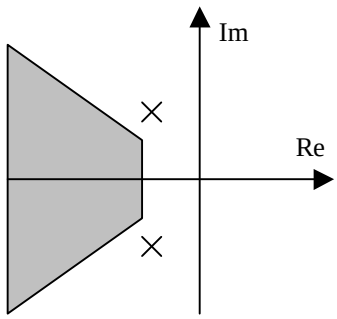

Fig.2: Pole location and (in gray) an example of region in which they should be in order to satisfy the requirements.

In general it has to be found a certain control function $u=a(x_1,x_2)$ so that, for each equilibrium point of (2), the poles of the linearized system in the neighborhood of them could satisfy the prescribed requirements. If we suppose to have only the origin as equilibrium point and assuming that

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \end{bmatrix}_{(0,0)} , \quad B = \begin{bmatrix} \frac{\partial f_1}{\partial u} \\ \frac{\partial f_1}{\partial u} \end{bmatrix}_{(0,0)}$$

and

$$k = \begin{bmatrix} \frac{\partial a}{\partial x_1} & \frac{\partial a}{\partial x_2} \end{bmatrix}_{(0,0)} \tag{4}$$

from the classical theory [1] this fact means that the eigenvalues of $A+Bk$ must lie in the gray region of figure 2. This implies that $a(x_1,x_2)$ must be chosen among those functions having certain properties in their first derivatives imposed by algebraic disequalities. Fortunately, these properties are only local and this is a great advantage when the system to be controlled has not a large number of state variables.

2.1 Example

Let us consider the system of equation (3). If we assume to have $a(x_1,x_2)=a_1(x_1)+a_2(x_2)$, a control system with regulation on state variables can be adopted (see Fig. 3).

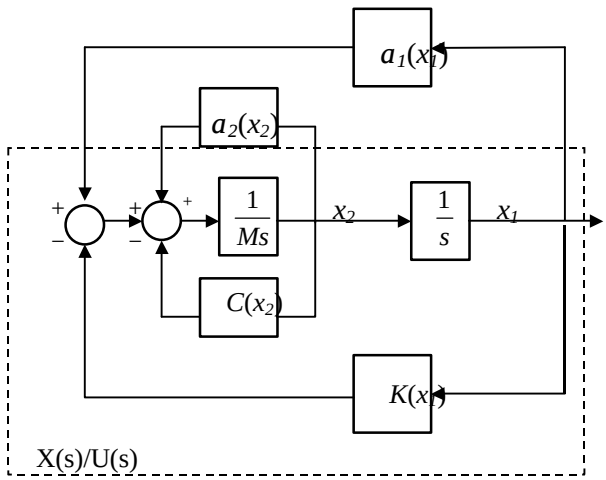

Fig.3:Block diagram of control system. There are evidenced the original system (dashed box) and the regulation actions on each state variable.

With few calculations involving (2) and (4), the characteristic polynomial of $A+Bk$ results to be

$$p(s) = s^2 + \frac{1}{M}\left(\frac{dC}{dx_2} - \frac{da_2}{dx_2}\right)_{(0,0)} s + \frac{1}{M}\left(\frac{dK}{dx_1} - \frac{da_1}{dx_1}\right)_{(0,0)} \tag{5}$$

The poles, which are the solution of $p(s)=0$, can be easily assessed by a suitable choice of $a_1(x_1)$ and $a_2(x_2)$.

3 Analytic Fuzzy Approach and Problem Solution

For clarity of discussion and figure representation, from this point we focus our attention on one dimensional control functions. However the extension of the method to multidimensional case does not present excessive difficulties. Let us assume $y=F(x)$ as the control function with $x,y\in[0,1]$. Our aim is to determine an approximation $y=F_a(x)$ of it via a fuzzy model.

3.1 Soft Computing approach

Soft computing is a new paradigm in computing based on a creative fusion of computing techniques like fuzzy computing, neural and genetic computing and similar, characterized by dealing with uncertainty, vagueness, imprecision and sub-optimality and by inspiration by human problem solving and learning. Its aim is to solve problems for which classical mathematical methods are difficult or impossible to apply due to enormous amount of data or to lack of theoretical background [2].

Let X be a set of experimental data and $f(x,y)$ a fitness function which cointains the control specifics. The SC paradigm uses Genetic Algorithms [3] to find the values y which maximizes (or minimizes) $f(x,)$, with x in X. In this way it is obtained a set of couples (x,y) which are used to train the Fuzzy Neural Network [4]. The result of this training is a fuzzy model and the searched approximation derives from defuzzifying the fuzzy model (see Figure 4).

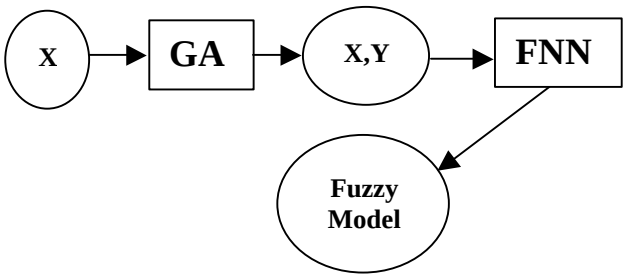

Fig.4

3.2 Fuzzy approximation of a single variable control function

SC allows to build a fuzzy model of a system without almost any knowledge about the classical formalized model. Surely this is an advantage, but there are many cases of real applications (like the one presented in this work) for which there exists a well known classical model. In these situations it is worth considering a method which, starting from the classical model, produces a better fuzzy model.

The method we propose uses only the values of the variable x and a merit function $m(x,y)$, providing a truth value whose range is between 0 and 1. This function is such that $m(x,y)=1$ if and only $y=F(x)$. In this sense, the function $m()$ give a 'vote' on how much the output value y is appropriate to the input value x. If we have the classical approximation of the control law, the merit function can be easily defined. Given the merit function, this method automatically generates the fuzzy counterpart of the classical model.

The procedure is based on several results found in the field of many-valued algebra (see [5]). It consists in partitioning in a suitable way (following the Farey's law, see [5]) the domain of x and to link each block of it to an action on y. From this correspondence derives a logical formula $y=F_a(x)$ that will approximate the control function of the system assigning to each value of x a value of the control variable y. The logical formula obtained may be seen as a 'compressed' form of the usual way of presenting fuzzy rules (*i.e.*, by a table of if-then statements).
Figures 5 and 6 shows the resulting input and output membership functions obtained with this method in approximating a quadratic function (Figure 7).

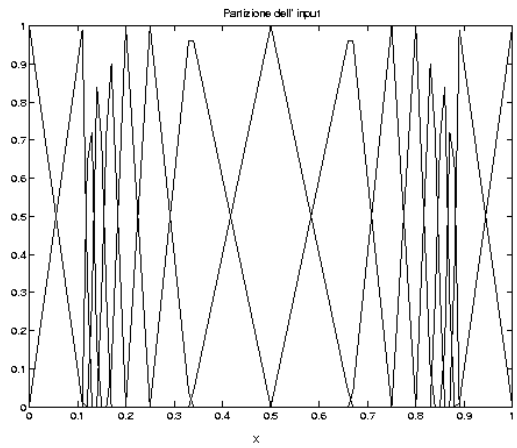

Fig.5 Input Membership functions

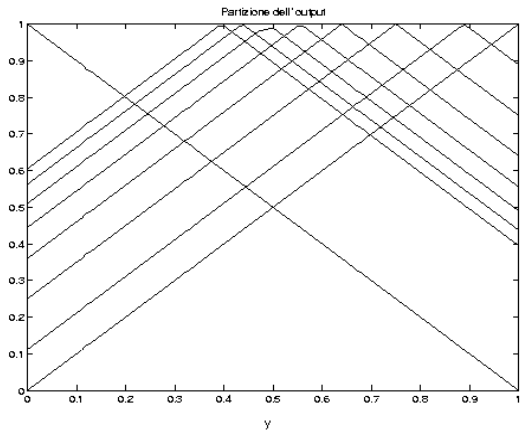

Fig.6 Output Membership functions

Looking at the control scheme depicted in Figure 3, it is possible to replace the control functions $a_1(x_1)$ and $a_2(x_2)$ with their fuzzy approximations. It is also possible, if they are not available, to use conditions derived from (5) directly in the choice of the merit function *m* which will lead to the correct approximation. A trial-and-error procedure with increasing degree polynomial approximation can be a solution in this case. Finally Figure 8 shows the impulse response of our system before and after the introduction of our fuzzy controller when $C(x)=x^3$ and $K(x)=\tanh(x)$. A quite good damping without oscillation is obtained

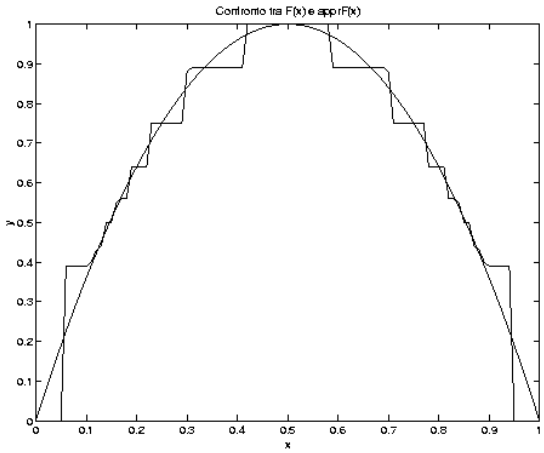

Fig.7: Approximation of a quadratic function

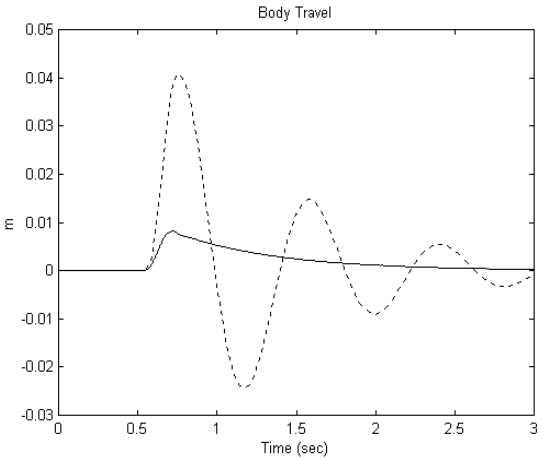

Fig.8: The mass position in case of uncontrolled (dashed) and fuzzy controlled system

4 Conclusion

In this work we used tools taken from the very abstract framework of many-valued algebra and applied it to a concrete, even if simple, control problem. Since the result obtained seems promising, in future developments the same tools could be used to invetigate relationships between classical and fuzzy models and the theoretical foundations of Soft Computing.

References:

[1] D. Cheng, A. Isidori, W. Respondek, T.J. Tarn, *Exact linearization of nonlinear systems with output*, Mathematical System Theory, 21,1988, pp. 63-83.

[2] R.R. Yager, L. Zadeh ed. *An Introduction to fuzzy Logic Applications in Intelligent System*, Kluwer Academic, 1989

[3] S.Horikawa, T. Furuhushi, Y. Uchikawa, *On Fuzzy-Modeling Using Fuzzy Neural Network with Back-Propagation Algorithm*, IEEE Transaction on Neural Networks, vol. 3, no. 5 September, 1992

[4] D.E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison Wesley, 1989

[5] R. Cignoli, I. D'Ottaviano, D. Mundici, *Algebraic Foundations of many-valued Reasoning*, Kluwer, Dordrecht, 1999