Learning Strategy for Optimal Fuzzy Control

Paulo Salgado
CETAV
Universidade de Trás-os-Montes e Alto Douro
Vila Real, Portugal
Email: psal@utad.pt

Getúlio Igrejas
Departamento de Electrotecnia
ESTiG - Instituto Politécnico de Bragança
Bragança, Portugal
Email: igrejas@ipb.pt

Abstract – **In this paper, a new scheme of fuzzy optimal control for discrete-time nonlinear systems based on the Pontryagin's Minimum Principle is proposed. Using back propagation from the final co-state error and gradient descent, a method which allows training an adaptive fuzzy inference system to estimate values for the co-state variables converging to the optimal ones is devised. This approach allows finding a solution to the optimal control problem on-line by training the system, rather than by pre computing it. Finally, this optimal approach is applied to nonlinear control benchmark problems. The results demonstrate the effectiveness of the approach towards achieving the optimal control objective.**

I. Introduction

In many physical and engineering systems, engineers are hindered by strong nonlinearity for successful application of optimal linear control. In many practical situations, an optimal controller that can minimize (or maximize) certain performance criterion is desired.

It is well known that exact analytical solutions exists only for specific classes of optimal control problems, such as linear systems with quadratic integral cost function. But so far few results can provide an effective way of optimal control design for general nonlinear systems.

On the other hand, a numerical solution is possible for optimal programs or open-loop controls, but it becomes very difficult for synthesis (feedback) controls if the dimension of the system is high. In this way, most of the existing results on optimal control for nonlinear systems are concerned with the optimality of a single trajectory. Given an initial condition, it is possible to use one of the many efficient available computational techniques to calculate an open-loop optimal control signal. However, these methods when applied to the mostly real control processes have a weak performance due to inaccuracies of the mathematical models and unknown disturbances affecting the systems.

In the past decade, fuzzy inference systems emerged as one of the most useful approaches to collect human knowledge and expertise on control and to transform the collected knowledge into a basis for developing controllers [1]–[3]. A fuzzy logic controller is usually a fuzzy inference system establishing a static mapping from the state variables input values to the actuators output values [3].

Considering that nowadays fuzzy logic represents one of the important techniques dealing with nonlinearities we present an alternative approach to nonlinear optimal control based on fuzzy logic. In this approach, the (adaptive) fuzzy inference system can be used to generate actuator values, but its primary function is to generate estimates of the co-state variables.

Co-state variables play a key role in finding the optimal control when using Pontryagin's Minimum Principle (PMP).

A number of stable and optimal fuzzy controllers were developed for linear systems by using the PMP with quadratic cost function. Wang [8] developed the optimal fuzzy controller for linear time-invariant systems. Based on the conventional linear quadratic optimal control theory, Wu and Lin [9] presented a design method of the optimal controllers for continuous- and discrete-time fuzzy systems. Later, Wu and Lin [10] developed a design scheme of the optimal fuzzy controller under finite- or infinite-horizon by using the calculus-of-variation method. Chen, Tseng, and Uang [11] introduced a fuzzy linear control design method for nonlinear systems with a fuzzy linear model that provides rough control to approximate the nonlinear control system, and an optimal H-infinity scheme that provides precise control to achieve the optimal robustness performance. Moreover, Wu and Lin [12][13] presented local and global approaches of optimal and stable fuzzy controller design methods for both continuous and discrete-time fuzzy systems under both finite and infinite horizons by applying traditional linear optimal control theory.

Let's begin by briefly reviewing the concepts of PMP, for the herein case of interest, i.e. for systems modelled in discrete time.

Let one consider systems described by nonlinear difference equations of the kind:

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}^k\left(\boldsymbol{x}_k, \boldsymbol{u}_k\right) \tag{1}$$

where $\boldsymbol{x}_k \in \mathbb{R}^n$ is the state and $\boldsymbol{u}_k \in \mathbb{R}^m$ the control input of the system at time kT, $\boldsymbol{f}$ is a vector valued function, possibly non-linear and time-varying, and T is the sampling period. One requires that $f\left(\boldsymbol{x},\boldsymbol{u}\right): \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$ is Lipschitz continuous and that there exists a constant $M_f > 0$ such that $\left|f\left(\boldsymbol{x},\boldsymbol{u}\right)\right| \le M_f\left(\left|\boldsymbol{x}\right| + \left|\boldsymbol{u}\right| + 1\right)$ for all $\left(\boldsymbol{x},\boldsymbol{u}\right) \in \mathbb{R}^n \times \mathbb{R}$.

The control problem is to find the control sequence $\boldsymbol{u}_k^*$ that minimizes the criterion or cost function J_i:

$$J_i = \Phi\left(N, \boldsymbol{x}_N\right) + \sum_{k=i}^{N-1} L^k\left(\boldsymbol{x}_k, \boldsymbol{u}_k\right) \tag{2}$$

In (2) [*i, N*] is the prescribed control time interval, $\Phi(N, \boldsymbol{x}_N)$ is the cost on the final state value $\boldsymbol{x}_N$, and $L^k(\boldsymbol{x}_k, \boldsymbol{u}_k)$ is the cost on both state and command at instant $k < N$.

The solution for this problem given by PMP is as follows. One defines the sequence of Hamiltonian functions H^k:

$$H^k = L^k + \boldsymbol{\lambda}_{k+1}^T \cdot \boldsymbol{f}^k \tag{3}$$

where $\boldsymbol{\lambda}_k \in \mathbb{R}^n$ is a vector of Lagrange multipliers. Accordingly to common usage one will designate $\boldsymbol{\lambda}_k$ as the co-state variables. The optimal sequence $\boldsymbol{u}_k^*$ that minimizes the criterion (2) is found by solving simultaneously the following equations:

State variable equation:

$$x_{k+1} = \frac{\partial H^k}{\partial \lambda_{k+1}} = f^k(x_k, u_k) \tag{4}$$

Co-state variable equation:

$$\lambda_k = \frac{\partial H^k}{\partial x_k} = \left(\frac{\partial f^k}{\partial x_k}\right)^T \lambda_{k+1} + \frac{\partial L^k}{\partial x_k} \tag{5}$$

Stationary conditions equation:

$$0 = \frac{\partial H^k}{\partial u_k} = \left(\frac{\partial f^k}{\partial u_k}\right)^T \lambda_{k+1} + \frac{\partial L^k}{\partial u_k} \tag{6}$$

Limit conditions equations:

$$\begin{cases} \left(\dfrac{\partial L^0}{\partial x_0} + \left(\dfrac{\partial f^0}{\partial x_0}\right)^T \lambda_1\right)^T \mathrm{d}\, x_0 = 0 \\ \left(\dfrac{\partial \Phi}{\partial x_N} - \lambda_N\right)^T \mathrm{d}\, x_N = 0 \end{cases} \tag{7}$$

In general, finding solutions for (4)-(7) is not an easy task, due to the interdependence of equations (4) and (5), which implies that forward and backward time sequences should be used.

However, looking at the problem on fuzzy logic grounds, the co-state variables appears to behave like the output of an expert system that knows which sequence of values will minimize the cost function. This means that one may devise a control strategy based on an adaptive fuzzy inference system which generates at each time *k*, in the control time interval, an estimated value for the co-state variable at time $k+1$.

Let one define a training iteration as a sequence of control actions from $k = 0$ to $k = N$. Then, along successive training iterations the fuzzy inference system rules may be changed in order to generate estimates converging to the true optimal values of the co-state variables. This will imply that the state variables values will also converge to the optimal ones.

Changing the rules of the fuzzy inference can be made by means of a learning algorithm which takes as its input the final error between state and co-state variables. In fact, it can be proved that, under the quadratic version of criterion (2), if this error goes to zero then the state and co-state trajectories will converge to the optimal ones.

This idea will be explored along the following sections. Firstly, in section 2, a brief introduction to the fuzzy inference systems used is made. The quadratic version of the optimal control problem considered and the fuzzy logic approach to its solution are described in section 3. Section 4 presents the learning algorithm. Two illustrative simulation examples are presented in section 5. Finally, the main conclusions are outlined in section 6.

II. The Fuzzy Inference System

Fuzzy systems modelling provide a framework for modelling complex nonlinear relations, using a rule-based methodology. Consider a system $y = f(\boldsymbol{x})$; *y* is the output (or consequent) variable and $\boldsymbol{x} = (x_1, \cdots, x_n)^T \in \mathbb{R}^n$ is the input vector (or antecedent) variable. Let $\boldsymbol{U} = U_1 \times \cdots \times U_n$ be the domain of the input vector $\boldsymbol{x} \in \mathbb{R}^n$ and *V* the output space.

A linguistic model relating variables *x* and *y* can be written as a collection of rules that link terms $A_{j,i} \in U_i$, $j = 1, \ldots, M$, $i = 1, \ldots, n$, and $B_j \in V$, where $A_{j,i}(x_i)$ and $B_j(y)$, respectively, represent the descriptor sets associated to variables x_i, $i = 1, \ldots, n$ and *y*. In fuzzy systems modelling, this relationship is represented by a collection $\Im$ of fuzzy IF–THEN rules

$$R_j : IF\ x_1\ is\ A_{j1}\ and \cdots x_n\ is\ A_{jn}\ THEN\ y\ is\ B_j \tag{8}$$

where *j* is a index of rule. The linguistic connective "and" of antecedent rule (8) will be defined as a t-norm operation, $\star$, where an aggregated fuzzy set A_j can be viewed as the fuzzy intersection set $\mathsf{X}_{i=1}^n A_{ji}$ with membership function $A_j(\boldsymbol{x}) = A_{j1}(x_1) \star \cdots \star A_{jn}(x_n)$. The fuzzy implication of each rule *j*, $R_j : A_j \mapsto B_j$ is a fuzzy set in product space $U \times V$ which is defined as $R_{j:A\mapsto B}(\boldsymbol{x}, y) = A_j(\boldsymbol{x}) \otimes B_j(y)$, where "$\otimes$" is an operator rule of fuzzy implication, usually min-max inference [1-2] or arithmetic inference [3].

For each rule *j*, the effective output value B_j is calculated using sup-star composition:

$$B'_j(y) = \sup_{\boldsymbol{x} \in U} \left[A'(\boldsymbol{x}) \star R_{j:A\mapsto B}(\boldsymbol{x}, y) \right].$$

The final fuzzy set *B*, which is determined by all the rules in the base, is obtained by the combination of the B_j and their associated membership functions.

In many situations, namely in series prediction and modelling applications, it is desirable to have a crisp value *y** for the output of a fuzzy system, instead of a fuzzy value

$B(y)$. The defuzzifier performs a mapping from the fuzzy sets in V to crisp points in V.

The fuzzy inference system used in this study is based on a product inference engine, singleton fuzzifier, and centre-average defuzzifier. So, the fuzzy logic inference system can be represented by:

$$f(\boldsymbol{x}) = \frac{\sum_{j=1}^{M} \bar{y}_l A_j(\boldsymbol{x})}{\sum_{j=1}^{M} A_j(\boldsymbol{x})} \quad (9)$$

where $A_j(\boldsymbol{x}) = \prod_{i=1}^{n} A_{ji}(x_i)$ is the input membership function and $\bar{y}_j$ is the centre of the output membership function.

There are three main reasons for using this fuzzy system as a basic building block for adaptive fuzzy controllers or identification systems:

–It has been showed in other works [4-7] that fuzzy logic systems given by (9) are universal function approximators;

–These fuzzy logic systems are constructed from fuzzy *IF-THEN* rules using specific fuzzy inference, fuzzification, and defuzzification strategies, which allow the incorporation of information from human experts into controllers;

–The functional form of (9) can be represented as a three-layer feedforward network. Therefore, it is possible to apply back-propagation learning methods or other neuro-fuzzy techniques, for adjusting the parameters of the membership functions of the rules [4], yielding an adaptive fuzzy system.

An adaptive fuzzy system is defined as a fuzzy logic system equipped with a learning algorithm where the fuzzy system is constructed from a set of fuzzy IF-THEN rules using fuzzy logic principles, and the learning algorithm adjusts the parameters of the fuzzy system based on the training information.

III. The Optimal Control Algorithm

In this section, a method for the optimal control of systems described by (1) under a quadratic version of criterion (2) is presented. The method is based on the ideas and framework sketched above.

Consider the rewriting of the nonlinear discrete dynamic system in (1) as:

$$x_{k+1} = g(x_k) + h(x_k)u_k \quad (10)$$

where $g:\mathbb{R}^n \to \mathbb{R}^n$ and $h:\mathbb{R}^n \to \mathbb{R}^{n\times m}$ are continuous over $\mathbb{R}^n$. Assume that $g_k + h_k u_k$ is Lipschitz continuous on a set $U \in \mathbb{R}^n$ containing the origin, and that system (10) is stabilizable in the sense that there exists a continuous control on U that asymptotically stabilizes the system. It is desired to find a sequence of u_k, which minimizes the cost function:

$$J_0 = \frac{1}{2}\sum_{k=1}^{N}(\boldsymbol{x}_k - \boldsymbol{r}_k)^T R_k(\boldsymbol{x}_k - \boldsymbol{r}_k) + \frac{1}{2}\sum_{k=0}^{N-1} u_k{}^T Q_k u_k \quad (11)$$

where r_k is the desired state at k sample, R_k and Q_k are matrices that allow to weight attainment of the desired state versus control effort.

Now, (4)-(7) can be rewritten:

$$\lambda_k = A_k\lambda_{k+1} + R_k(x_k - r_k), \quad (12)$$

$$u_k = -Q_k^{-1}B_k\lambda_{k+1}, \quad (13)$$

$$\lambda_N = R_N(x_N - r_N) \quad (14)$$

where $A_k = \left(\partial f^k / \partial x_k\right)^T$ and $B_k = \left(\partial f^k / \partial u_k\right)^T$.

Given (10), (13) allows one to write the control variable at time k as:

$$u_k = -Q_k^{-1}h(x_k)\lambda_{k+1}. \quad (15)$$

This equation may be taken as an optimal feedback control law, if the optimal value of the co-state variable, λ_{k+1}^* is known at time k.

Now the approach proposed in this paper may be made explicit. One takes λ_{k+1} as the output of a fuzzy inference system $\boldsymbol{\Lambda}$ that at instant k generates an estimate of λ_{k+1}^*, having as inputs the observed state x_k and the time to go $N-k$:

$$\lambda_{k+1} = \hat{\lambda}_{k+1}^* = \boldsymbol{\Lambda}(x_k, N-k). \quad (16)$$

This gives the feedback control law

$$u_k = -Q_k^{-1}h(x_k)\boldsymbol{\Lambda}(x_k, N-k) \quad (17)$$

which by incorporation of the h function into the fuzzy inference system can be streamlined to:

$$u_k = -Q_k^{-1}\boldsymbol{\Lambda}_h(x_k, N-k) \quad (18)$$

From equation (10) and (17) we have

$$\boldsymbol{x}_{k+1} = \boldsymbol{g}_k - H_k\boldsymbol{\lambda}_{k+1} \quad (19)$$

where $H_k = h(x_k)Q_k^{-1}h^T(x_k)$

If, by adaptation or learning of the fuzzy inference system, along successive runs or training iterations of the system from $k=0$ to N, the estimates are made to converge to the optimal ones, then any of the control laws (17) or (18) becomes optimal.

That this indeed can be done is the subject of the next section.

IV. The Learning Algorithm

To solve the equations resulting from framing a discrete optimal control problem under PMP, an off-line optimization method is usually applied. Here, one proposes a learning algorithm based on an approximate gradient descent method that, during the training iterations, progressively refines the accuracy of the co-state fuzzy estimator. This strategy reduces the necessary computing time and memory, avoiding the calculations of the exact adjoin and the directional derivatives of the cost functional. Below the first algorithm of this method is given in the theoretical form. The implementation of the algorithm will be described in future work.

From (14) it follows that for optimal trajectories one must necessarily have $\lambda_N = R_N(x_N - r_N)$ so $\lambda_N^* = R_N(x_N^* - r_N)$.

Let $E = \lambda_N - R_N(x_N - r_N)$ be the error or difference between the end value of the state and the co-state variable trajectories, as exemplified in Fig. 1 (where without loss of generality we considered that $r_N = 0$ and $R_N = 1$). As noted above, for optimal trajectories $x^*(k)$ and $\lambda^*(k)$, it is a necessary condition that $\lambda_N^* = R_N(x_N^* - r_N)$ or $E = 0$. It is also possible to prove that this is a sufficient condition. If $E \to 0$, then $x_k \to x_k^*$ and $\lambda_k \to \lambda_k^*$, i.e. the trajectories of the state and co-state variables converge to the optimal ones. Fig. 1 graphically depicts the idea for one state and co-state variable.

It follows that to attain optimal state trajectories, it is necessary that the error E converge to zero. This objective is achieved by adjusting the final λ_N co-state variables in order to minimize:

$$E^2 = (\lambda_N - R_N(x_N - r_N))^T (\lambda_N - R_N(x_N - r_N)).$$

The gradient descent algorithm was employed to determine the adjustments to the final co-state value:

$$\lambda_N^{q+1} = \lambda_N^q - 2\alpha E^q \frac{\partial E^q}{\partial \lambda_N^q} \tag{20}$$

where, $q = 0,1,2,\ldots$ is the training iteration number and α is a scalar step-size variable.

For all q one has that:

$$\frac{\partial E}{\partial \lambda_N} = I - R_N \frac{\partial x_N}{\partial \lambda_N} \tag{21}$$

where I is the identity matrix.

The summands at the right side of (21) can be solved iteratively as:

$$\frac{\partial x_{k+1}}{\partial \lambda_N} = A_k \frac{\partial x_k}{\partial \lambda_N} - H_k \frac{\partial \lambda_{k+1}}{\partial \lambda_N} \tag{22}$$

$$\frac{\partial \lambda_k}{\partial \lambda_N} = \frac{\partial \lambda_k}{\partial \lambda_{k+1}} \frac{\partial \lambda_{k+1}}{\partial \lambda_N} \tag{23}$$

with $\partial x_0 / \partial \lambda_N = 0$ and $\partial \lambda_N / \partial \lambda_N = I$.

From (12) and (19), the equation (23) can be rewrite as:

$$\frac{\partial \lambda_k}{\partial \lambda_N} = (I + R_k H_k)^{-1} A_k \frac{\partial \lambda_{k+1}}{\partial \lambda_N}$$

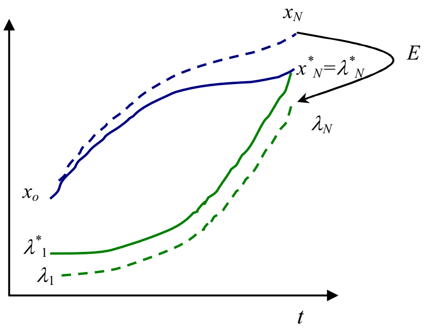

Fig. 1. Trajectories of optimal (-) and non-optimal (--) state (blue line) and co-state (green line) variables

From the new value of λ_N^{q+1} a new backward co-state trajectory is computed trough equation (12). With the new value of co-state variable λ_k^{q+1}, for k=1,..,N, a new state trajectory is also computed. As far as $E \to 0$ the trajectories of the state and co-state variables converge to the optimal ones.

V. Experimental Results

In this section, we shall design the optimal controllers for two nonlinear dynamical systems. Simulation results show that the proposed optimal fuzzy controllers can effectively drive the dynamical systems to the target trajectory in an optimal way.

Example 1: For illustrative purposes, the method described was implemented to regulate the plant:

$$x_{k+1} = x_k + T \cdot f(x_k, u_k) \tag{24}$$

with:

$$f(x_k, u_k) = 10 \frac{1 - e^{-x_k}}{1 + e^{-x_k}} + u_k \tag{25}$$

In this case, $r_N = 0$. The plant (25) is unstable if no control action exists:

$$\begin{aligned} u_k = 0 \wedge x_k > 0 \to x_{k+1} - x_k = 10 \frac{1 - e^{-x_k}}{1 + e^{-x_k}} > 0 \\ u_k = 0 \wedge x_k < 0 \to x_{k+1} - x_k = 10 \frac{1 - e^{-x_k}}{1 + e^{-x_k}} < 0 \end{aligned} \tag{26}$$

A sampling time of $T = 0.1\text{s}$ and N = 30 (i.e. $T.N$ = 1 second) was used.

For each trajectory generated the algorithm above was used to adjust the values of the co-state variable. At the end of this process, the results were stored in a fuzzy inference system.

Fig. 2 shows the values of the co-state variable outputted by the fuzzy inference system as a function of the inputs state position and time remaining to end.

This function may be linguistically interpreted as follows:

–When the initial state equals the final state, the co-state value is zero everywhere.

–The system symmetry corresponds to one co-state symmetry.

–Because the system is unstable, less energy consumption is obtained if the control system drives the state variable to the equilibrium point ($x = 0$) as soon as possible – and with more strength the shorter is the remaining time.

These conclusions could be achieved before the training process, by analysis of the system model. Incorporating them in the fuzzy inference system would result in a faster convergence process.

Fig. 3 shows test responses, comparing the described method (curve of • points) and the obtained by numerical nonlinear optimization technique (Quasi-Newton method with a mixed quadratic and cubic line search procedure) (curve of × points).

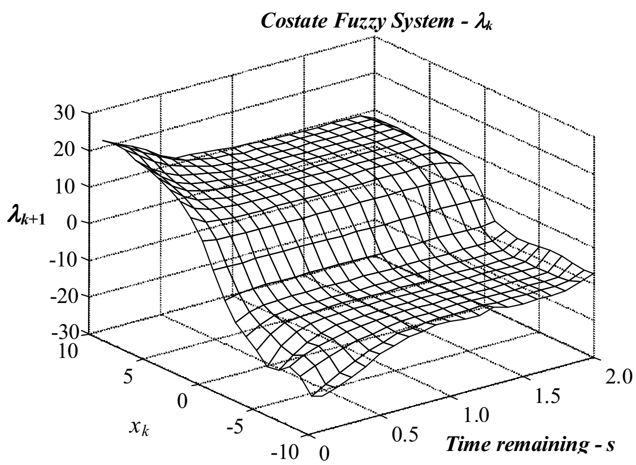

Fig. 2. Surface map of co-sate with respect to state and time position.

Example 2: Consider the following harmonic oscillator model:

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) \qquad (27)$$

Perhaps the one simple method to solve the ordinary Differential Equation $\dot{\boldsymbol{x}}(t) = f(\boldsymbol{x}(t), u(t))$ is based on Taylor series approximation to the solution of initial value $\boldsymbol{x}(t)$. If $\boldsymbol{x}^{(p+1)}(t)$ is continuous, then Taylor's formula gives

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + f(t_k, \boldsymbol{x}_k)h + \cdots + f^{(p-1)}(t_k, \boldsymbol{x}_k)\frac{h^p}{p!} + hO(h^p)$$

where $hO(h^p) = f^{(p+1)}(\xi_k, \boldsymbol{x}(\xi_k))\frac{h^{(p-1)}}{p!}$, with $t_k \le \xi_k \le t_{k+1}$.

The result of this approach is a discrete equation of the system, which can be used for optimal control strategy. In this paper we use h = 0.0345 second and p = 3, trying to minimize the linear quadratic regulator index

$$J = \sum_{k=1}^{N} \boldsymbol{x}_k^T \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \boldsymbol{x}_k + u_k^2$$

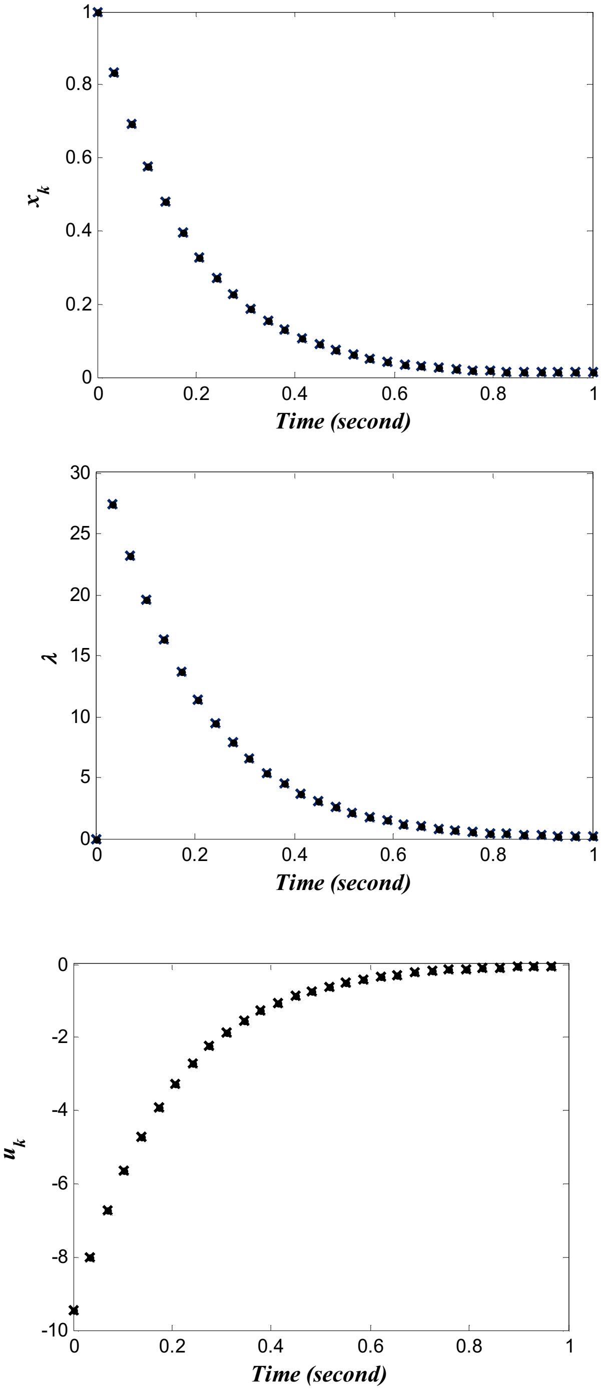

Fig. 3. Example of one trajectories of state, co-state and control variable: through numerical nonlinear optimization method (×) and co-state optimal fuzzy system (•).

Fig. 4 shows state, co-state and actuation response of the plant with the optimal controllers under initial condition $x_o(t) = [1 \quad -1]^T$ (points •). The obtained results from the nonlinear optimization method are represented by × points, which values are very near to the ones of the proposed method. Obviously, the states of both systems converge to 0 in an optimal manner.

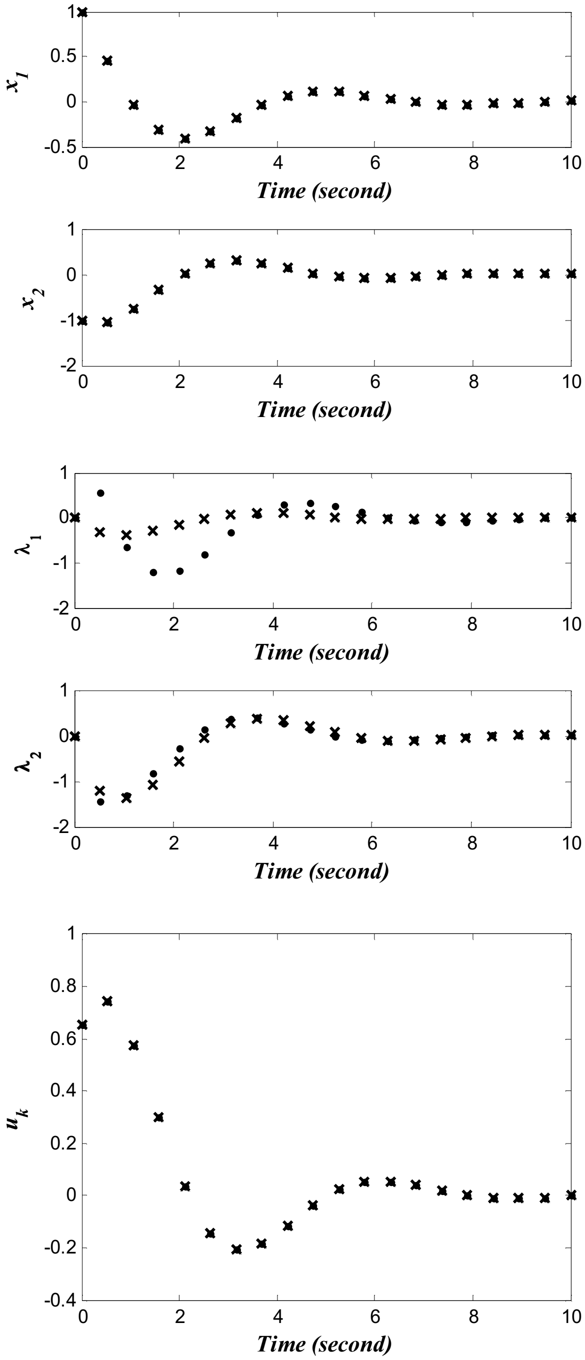

Fig. 4. Example of optimal trajectories of state (x_1 and x_2), of co-states (λ_1 and λ_2), and control variable u_k: by nonlinear optimization method (×) and by co-state optimal fuzzy system (•).

VI. Conclusion

In this paper the implementation of the non-linear quadratic optimal control is described using a fuzzy logic methodology upon Pontryagin's Minimum Principle. A learning algorithm interactively adjusts the co-state variable values in an optimal way. The found values are saved in a fuzzy inference system. The proposed methodology allows attaining on-line and in close-loop calculation of the optimal control actions. This fact should make possible to design feedback strategies more robust than standard off-line open loop optimal ones, with respect to the inaccuracies of the process model and unpredictable disturbances. Moreover, it should allow tracking process changes.

Acknowledgment

This work was supported by the Portuguese "Fundação para a Ciência e a Tecnologia (FCT)" under grant POSI/SRI/41975/2001.

References

[1] C. Chuen-Lee, "Fuzzy Logic in Control Systems: Fuzzy Logic Controller, Part I", *IEEE Trans. On Syst., Man and Cybernetics*, 20, 1990a, pp. 404-418.

[2] C. Chuen-Lee, "Fuzzy Logic in Control Systems: Fuzzy Logic Controller, Part II", *IEEE Trans. on Syst., Man, and Cybernetic*, 20, 1990b, pp. 419-435.

[3] Li-Xin Wang, *A course in fuzzy systems and control*, NJ, Prentice-Hall PTR, 1997.

[4] H. Ying, "Sufficient conditions on general fuzzy systems as function approximators," *Automatica*, vol. 30, 1994, pp. 521–525.

[5] J. A. Dickerson and B. Kosko, "Fuzzy function approximation with ellipsoidal rules," *IEEE Trans. Syst., Man, Cybern.*, vol. 26, 1996, pp. 542–560.

[6] J. L. Castro and M. Delgado, "Fuzzy systems with defuzzification are universal approximators," *IEEE Trans. Syst., Man, Cybern.*, vol. 26, 1996, pp. 149–152.

[7] L. X. Wang and J. M. Mendel, "Fuzzy basis functions, universal approximation, and orthogonal least-square learning," *IEEE Trans. Neural Networks*, vol. 3, 1992, pp. 807–814.

[8] L. X.Wang, "Stable and optimal fuzzy control of linear systems," IEEE Trans. Fuzzy Syst., vol. 6, pp. 137–143, Feb. 1998.

[9] S. J.Wu and C. T. Lin, "Optimal fuzzy controller design: Local concept approach," IEEE Trans. Fuzzy Syst., vol. 8, pp. 171–185, Apr. 2000.

[10] S. J.Wu and C. T. Lin, "Optimal fuzzy controller design: Global concept approach," IEEE Trans. Fuzzy Syst., vol. 8, pp. 713–729, Dec. 2000.

[11] Chen, B., Tseng, C., and Uang, H., "Robustness design of nonlinear systems via fuzzy linear control," IEEE Transactions on Fuzzy Systems, vol. 7, no. 5, pp. 571-585, 1999.

[12] Wu, S. J., & Lin, C. T., Optimal Fuzzy Controller Design: Local Concept Approach. IEEE Transaction on Fuzzy Systems, 8(2), 2000a,171–185.

[13] Wu, S. J., & Lin, C. T., Optimal fuzzy controller design in continuous fuzzy system: Global concept approach. IEEE Transactions on Fuzzy Systems, 8(6), 2000b, 713–729.