New methods for the estimation of Takagi–Sugeno model based extended Kalman filter and its applications to optimal control for nonlinear systems

Basil M. Al-Hadithi^{*,†}, Agustín Jiménez and Fernando Matía

Intelligent Control Group, Universidad Politécnica de Madrid, Madrid, Spain

SUMMARY

This paper describes new approaches to improve the local and global approximation (matching) and modeling capability of Takagi–Sugeno (T-S) fuzzy model. The main aim is obtaining high function approximation accuracy and fast convergence. The main problem encountered is that T-S identification method cannot be applied when the membership functions are overlapped by pairs. This restricts the application of the T-S method because this type of membership function has been widely used during the last 2 decades in the stability, controller design of fuzzy systems and is popular in industrial control applications. The approach developed here can be considered as a generalized version of T-S identification method with optimized performance in approximating nonlinear functions. We propose a noniterative method through weighting of parameters approach and an iterative algorithm by applying the extended Kalman filter, based on the same idea of parameters' weighting. We show that the Kalman filter is an effective tool in the identification of T-S fuzzy model. A fuzzy controller based linear quadratic regulator is proposed in order to show the effectiveness of the estimation method developed here in control applications. An illustrative example of an inverted pendulum is chosen to evaluate the robustness and remarkable performance of the proposed method locally and globally in comparison with the original T-S model. Simulation results indicate the potential, simplicity, and generality of the algorithm. An illustrative example is chosen to evaluate the robustness. In this paper, we prove that these algorithms converge very fast, thereby making them very practical to use. Copyright © 2011 John Wiley & Sons, Ltd.

KEY WORDS: nonlinear systems; fuzzy systems; Takagi–Sugeno fuzzy model; extended Kalman filter; linear quadratic regulator

1. INTRODUCTION

Nonlinear control systems based on the Takagi–Sugeno (T-S) fuzzy model [1] have attracted lots of attention during the last 20 years (e.g., see [2–7]). It provides a powerful solution for the development of function approximation, systematic techniques to stability analysis and controller design of fuzzy control systems in view of a fruitful conventional control theory and techniques. They also allow relatively easy application of powerful learning techniques for their identification from data. This model is formed by using a set of fuzzy rules to represent a nonlinear system as a set of local affine models which are connected by fuzzy membership functions. This fuzzy modeling method presents an alternative technique to represent complex nonlinear systems [8] and reduces the number of rules in modeling higher order nonlinear systems [1] and [2].

Some papers focus on the proof of the universal approximation property of the T-S fuzzy models as they are able to approximate any smooth nonlinear functions to any degree of accuracy in any

*Correspondence to: Basil M. Al-Hadithi, Intelligent Control Group, Universidad Politécnica de Madrid, José Gutiérrez Abascal 2, E-28006 Madrid, Spain.

†E-mail: basil@etsii.upm.es

convex compact region [9], [4], [10], and [8]. But it was clearly shown by several researchers that the number of fuzzy rules increases as the approximation error tends to zero [11]. This exponential growing cannot be eliminated, so it becomes difficult to make use of the universal approximation property of the T-S fuzzy modeling for practical purposes. In addition to that, it was shown in [12] that the set of functions, consisting of polytopic or the T-S models constructed from finite number of rules, is nowhere dense in the approximation model space, if that is defined as a subset of continuous functions. As a consequence, by means of the mentioned models, it is difficult to approximate in general, continuous functions arbitrarily well if the number of rules are restricted. Thus, only the functions satisfying certain conditions can be approximated by such models, or alternatively, an unbounded number of rules are required. This result is considered as a limitation for such models. The solution suggested by the authors for this discrepancy can be the trade-off between the accuracy and the number of rules.

These conflicting objectives have motivated the researchers to find a balance between the specified accuracy and the computational complexity of the resulting fuzzy model.

Recently, a great interest has been shown to Tensor Product Distributed Compensation (TPDC) as an established controller design framework that links the TP model transformation and parallel distributed compensation framework. The TP model transformation converts different models to a common representational form [13]. In [14], the approximation capabilities of the TP model forms are analyzed, because the universal applicability of the TPDC framework strongly relies on it. It has been shown that the practically important subset of the TP model forms that uses bounded number of rules has limited approximation capability: the set of such TP model forms lie nowhere dense in the space of continuous functions. Such cases require the use of trade-off techniques between accuracy and the TP model forms complexity. It has been shown that the uniform applicability of the TPDC framework is guaranteed, because the TPDC enables the use of trade-off methods. This is a great advantage of over analytic transformation, where such considerations are hardly possible to apply.

The definition of higher order singular-value-decomposition (HOSVD) canonical form was firstly proposed in [15]. The authors introduced how the concept of tensor HOSVD can be carried over to the TP dynamic models, namely, how the ‘HOSVD like’ decomposition of linear parameter varying dynamic models can be defined and generated. The authors name this decomposition as HOSVD based canonical form of the TP model or polytopic model form. The key idea and the basic concept of this decomposition was proposed recently with the TP model transformation based control design methodology. The novelty of this work is to present the mathematical background of this concept. The paper shows convergency theorems how the TP model transformation is capable of reconstructing this HOSVD based canonical form numerically. The proofs of the theorems are lengthy, therefore they are omitted. The paper also presents numerical examples to show the applicability, efficiency, and uniformity of the numerical reconstruction.

In [16], the HOSVD based canonical form of linear parameter varying models [15] is analyzed and a models’s most important invariant characteristics is shown. The numerical reconstructibility of the HOSVD canonical form using the TP model transformation is studied. Convergency theorems for given numerical reconstruction constraint are presented.

In [17], a singular value-based method for reducing a given fuzzy rule set was discussed. The method conducts singular value decomposition of the rule consequents and generates certain linear combinations of the original membership functions to form new ones for the reduced set. The work characterizes membership functions by the conditions of sum normalization, non-negativeness, and normality. The proposed method is applicable regardless of the adopted inference paradigms. The present work discusses three specific applications of fuzzy reduction: fuzzy rule base with singleton support, fuzzy rule base with nonsingleton support, and the T-S model.

In [18], the authors proposed to obtain the best features of the Mamdani and T-S models by using an affine global model with function approximation capabilities which maintains local interpretation. The suggested model is composed of variant coefficients which are independently governed by a zeroth order fuzzy inference system. This model may be interpreted as a generalization of the T-S model in which dynamics coefficients have been decoupled. They have shown that the Mamdani and T-S models can be combined so that local and global interpretations are preserved.

Great attention has been paid to the identification of the T-S fuzzy models and several results have been obtained [19], [9], and [20]. They are based upon two kinds of approaches, one is to linearize the original nonlinear system in various operating points when the model of the system is known, and the other is based on the input-output data collected from the original nonlinear system when its model is unknown. The authors in [19] use a fuzzy clustering method to identify the T-S fuzzy models, including identification of the number of fuzzy rules and parameters of fuzzy membership functions and identification of parameters of local linear models by using a least squares method [21]. The goal is to minimize the error between the T-S fuzzy models and the corresponding original nonlinear systems.

In [22], Klawonn *et al.* explained how fuzzy clustering techniques could be applied to design a fuzzy controller (FC) from the training data. In [23], Hong and Lee have analyzed that the disadvantages of most fuzzy systems are that the membership functions and fuzzy rules should be predefined to map numerical data into linguistic terms and to make fuzzy reasoning work. They suggested a method based on the fuzzy clustering technique and the decision tables to derive membership functions and fuzzy rules from numerical data. However, Hong and Lee's algorithm presented in [23] needs to predefine the membership functions of the input linguistic variables and it simplifies fuzzy rules by a series of merge operations. As the number of variables becomes larger, the decision table will grow tremendously and the process of the rule simplification based on the decision tables becomes more complicated. The authors in [9] suggest a method to identify the T-S fuzzy models. Their method aims at improving the local and global approximation of the T-S model. However, this complicates the approximation in order to obtain both targets. It has been shown that constrained and regularized identification methods may improve interpretability of constituent local models as local linearizations, and locally weighted least squares method may explicitly address the trade-off between the local and global accuracy of the T-S fuzzy models.

In [21], a new method of interval fuzzy model identification was developed. The method combines a fuzzy identification methodology with some ideas from linear programming theory. The idea is then extended to modeling the optimal lower and upper bound functions that define the band which contains all the measurement values. The method can be efficiently used in the case of the approximation of the nonlinear functions family.

In [24], a new approach to fuzzy modeling using the relevance vector learning mechanism (RVM) based on a kernel-based Bayesian estimation is introduced. The main concern is to find the best structure of the T-S fuzzy model for modeling nonlinear dynamic systems with measurement error. The number of rules and the parameter values of membership functions can be found as optimizing the marginal likelihood of the RVM in the proposed Fuzzy Inference System (FIS). Because the RVM is not necessary to satisfy Mercer's condition, selection of kernel function is beyond the limit of the positive definite continuous symmetric function of Support Vector Machine (SVM). The relaxed condition of kernel function can satisfy various types of membership functions in fuzzy model.

In [25], the authors suggest a transformation method that offers a trade-off between the modeling complexity and accuracy of the T-S fuzzy model. The suggested transformation method is aimed at finding the minimal number of fuzzy rules for a given accuracy of a given T-S fuzzy model. Optimization of the approximation trade-off is conceptually obtained by the discarding of those rules, which have weak or no contribution at all to the output.

Interpolation is an important subject for reducing dense rule bases as well for deriving useful information from sparse ones. In [11], the authors show that rule interpolation helps in reducing the identification complexity as it allows rule bases with gaps. It is suggested that only the minimal necessary number of rules remain which contain the essential information, and all other rules are replaced by the interpolation algorithm.

The approach proposed in [11] has some drawbacks such as subnormal conclusion for certain configuration of the involved fuzzy sets and does not always lead to interpretable fuzzy membership functions. In [26], a new technique for fuzzy rule interpolation was investigated. The aim is to combine the advantageous computational behavior of [11] and at the same time reduces the disadvantage of subnormality.

In spite of its advantageous properties in various approximation aspects and in complexity reduction, it was shown that α -cut distance based fuzzy rule base interpolation has some essential

deficiencies, for instance, it does not always result in immediately interpretable fuzzy membership functions. In [27], an approach is presented to get rid of these disadvantages. It is based on the interpolation of relations instead of interpolating α -cut distances and which offers a way to derive a family of interpolation methods capable of eliminating some typical deficiencies of fuzzy rule interpolation techniques.

In [28], a generalized version of the previous Cartesian approach [29] for interpolating fuzzy rules of membership functions with finite number of characteristic points is presented. Instead of representing membership functions as points in Cartesian spaces, they now become elements in the space of square, integrable function. Interpolation is thus made between the antecedent and consequent function spaces. The generalized representation allows an extended class of membership functions satisfying two monotonicity conditions to be accommodated in the interpolation process.

In [1], the authors develop an interesting method to identify nonlinear systems using input-output data. They divide the identification process in three steps; premise variables, membership functions, and consequent parameters. With respect to the membership functions, they apply nonlinear programming technique using the complex method for the minimization of the performance index.

In [30], a study has outlined a new min-max approach to the fuzzy clustering, estimation, and identification with uncertain data. The proposed approach minimizes the worst-case effect of data uncertainties and modeling errors on estimation performance without making any statistical assumption and requiring a priori knowledge of uncertainties.

Several methods are used to deal with the problem of optimizing membership functions, which are either derivative-based or derivative-free methods. The derivative free approaches are desirable because they are more robust than derivative-based methods with respect to finding global minimum and with respect to a wide range of objective function and Membership Functions (MFs) types. The main drawback is that they converge more slowly than derivative-based techniques [31].

On the other hand, derivative-based methods tend to converge to local minimums. In addition, they are limited to specific objective functions and types of inference and MFs. The most common approaches are: gradient descent [32], least squares [21], back propagation [33], and Kalman filtering [34], [35].

Kalman filter, is a powerful mathematical tool for stochastic estimation from noisy environment. It has various applications in optimizing fuzzy systems. For example, it has been used to extract fuzzy rules from a given rule base [36] and to optimize the output function parameters of the T-S fuzzy systems. Fuzzy logic has been used to compute the gains of a bank of parallel Kalman filters in order to combine their outputs [37]. The fuzzy logic has also been used to tune the parameters of the Kalman filter [38].

The use of the Kalman filter training to optimize the MFs of a fuzzy system was introduced by Simon [34] for motor winding current estimation. The used MFs were assumed as symmetric triangular forms. The Kalman filter training was extended to asymmetric triangles in [35], and a matrix was defined relating the parameters of the MFs together based on the sum-normality conditions, then projecting this matrix in each iteration of optimization to constrain the MFs to sum normal types.

Because the derivatives of the functions are used in Kalman filtering, it is limited to special type of MFs because of complicated and time consuming calculations. So far only triangular types are optimized for both inputs and outputs of an Fuzzy Logic Controller (FLC) [34, 35].

As we will demonstrate in this article, the T-S model cannot be applied when the membership functions are overlapped by pairs. The methods presented here are characterized by the high accuracy obtained in approximating nonlinear systems locally and globally in comparison with the original T-S model.

The rest of the paper is organized as follows. Section 2 recalls the estimation of the T-S fuzzy model. Section 3 introduces restrictions of the T-S identification method. In Section 4, the proposed noniterative approach is explained. The proposed iterative approach by applying extended Kalman filter is explained in Section 5. In Section 6, the design of an optimal controller based on the proposed estimation method is detailed. Section 7 entails an example of an inverted pendulum to demonstrate the validity of the proposed approach. The results show that the proposed approach is less conservative than those based on (standard) the T-S model and illustrate the utility of the proposed approach in comparison with the T-S model. The conclusions of the effectiveness and validity of the proposed approach are explained in Section 8.

2. ESTIMATION OF THE FUZZY T-S MODEL'S PARAMETERS

An interesting method of identification is presented in [1]. The idea is based on estimating the non-linear system parameters minimizing a quadratic performance index. The method is based on the identification of functions of the following form:

$$f : \Re^n \longrightarrow \Re$$

$$y = f(x_1, x_2, \dots, x_n)$$

Each IF-THEN rule $R^{i_1 \dots i_n}$, for an n th order system can be written as follows:

$$\begin{aligned} S^{(i_1 \dots i_n)} : & \text{If } x_1 \text{ is } M_1^{i_1} \text{ and } \dots x_n \text{ is } M_n^{i_n} \text{ then} \\ \hat{y} = & p_0^{(i_1 \dots i_n)} + p_1^{(i_1 \dots i_n)} x_1 + p_2^{(i_1 \dots i_n)} x_2 + \dots + p_n^{(i_1 \dots i_n)} x_n \end{aligned} \quad (1)$$

where the fuzzy estimation of the output is:

$$\hat{y} = \frac{\sum_{i_1=1}^{r_1} \dots \sum_{i_n=1}^{r_n} w^{(i_1 \dots i_n)}(\mathbf{x}) \left[p_0^{(i_1 \dots i_n)} + p_1^{(i_1 \dots i_n)} x_1 + p_2^{(i_1 \dots i_n)} x_2 + \dots + p_n^{(i_1 \dots i_n)} x_n \right]}{\sum_{i_1=1}^{r_1} \dots \sum_{i_n=1}^{r_n} w^{(i_1 \dots i_n)}(\mathbf{x})} \quad (2)$$

where,

$$w^{(i_1 \dots i_n)}(\mathbf{x}) = \mu_{1i_1}(x_1) \mu_{2i_2}(x_2) \dots \mu_{ni_n}(x_n)$$

being $\mu_{ji_j}(x_j)$, the membership function that corresponds to the fuzzy set $M_j^{i_j}$.

Let M be a set of input/output system samples $\{x_{1k}, x_{2k}, \dots, x_{nk}, y_k\}$. The parameters of the fuzzy system can be calculated as a result of minimizing a quadratic performance index:

$$J = \sum_{k=1}^m (y_k - \hat{y}_k)^2 = \|Y - XP\|^2 \quad (3)$$

where

$$\begin{aligned} Y &= [y_1 \ y_2 \ \dots \ y_m]^T \\ P &= \begin{bmatrix} p_0^{(1 \dots 1)} & p_1^{(1 \dots 1)} & p_2^{(1 \dots 1)} & \dots & p_n^{(1 \dots 1)} & \dots & p_0^{(r_1 \dots r_n)} & \dots & p_n^{(r_1 \dots r_n)} \end{bmatrix}^T \\ X &= \begin{bmatrix} \beta_1^{(1 \dots 1)} & \beta_1^{(1 \dots 1)} x_{11} & \dots & \beta_1^{(1 \dots 1)} x_{n1} & \dots & \beta_1^{(r_1 \dots r_n)} & \dots & \beta_1^{(r_1 \dots r_n)} x_{n1} \\ \beta_m^{(1 \dots 1)} & \beta_m^{(1 \dots 1)} x_{1m} & \dots & \beta_m^{(1 \dots 1)} x_{nm} & \dots & \beta_m^{(r_1 \dots r_n)} & \dots & \beta_m^{(r_1 \dots r_n)} x_{nm} \end{bmatrix} \end{aligned} \quad (4)$$

and

$$\beta_k^{(i_1 \dots i_n)} = \frac{w^{(i_1 \dots i_n)}(\mathbf{x}_k)}{\sum_{i_1=1}^{r_1} \dots \sum_{i_n=1}^{r_n} w^{(i_1 \dots i_n)}(\mathbf{x}_k)} \quad (5)$$

If X is a matrix of full rank, the solution is obtained as follows:

$$\begin{aligned} J &= \|Y - XP\|^2 = (Y - XP)^T (Y - XP) \\ \nabla J &= X^T (Y - XP) = X^T Y - X^T X P = 0 \\ P &= (X^T X)^{-1} X^T Y \end{aligned} \quad (6)$$

3. RESTRICTIONS OF THE T-S IDENTIFICATION METHOD

The method proposed in [1] arises serious problems as it cannot be applied in the most common case where the membership functions are those shown in Figure 1.

The membership functions $\mu_{i1}(x_i) = \frac{b_i - x_i}{b_i - a_i}$ and $\mu_{i2}(x_i) = \frac{x_i - a_i}{b_i - a_i}$ are defined in an interval $[a_i, b_i]$ which should verify:

$$\begin{aligned}\mu_{i1}(a_i) &= 1 & \mu_{i1}(b_i) &= 0 \\ \mu_{i2}(a_i) &= 0 & \mu_{i2}(b_i) &= 1 \\ \mu_{i1}(x_i) + \mu_{i2}(x_i) &= 1\end{aligned}$$

For this case which is widely used, it can be easily demonstrated [39] that the matrix X is not of full rank and therefore $X^T X$ is not invertible, which makes the mentioned method of T-S invalid. This result can be easily proven as follows:

Supposing that it exists:

$$f : \mathfrak{R}^n \longrightarrow \mathfrak{R}$$

$$y = f(x)$$

in which each row of the matrix X is of the form:

$$X_k = \begin{bmatrix} \mu_1(x_k) & \mu_1(x_k)x_k & \mu_2(x_k) & \mu_2(x_k)x_k \end{bmatrix} = \begin{bmatrix} \frac{b-x_k}{b-a} & \frac{b-x_k}{b-a}x_k & \frac{x_k-a}{b-a} & \frac{x_k-a}{b-a}x_k \end{bmatrix} \quad (7)$$

verifying that:

$$\begin{bmatrix} \frac{b-x_k}{b-a} & \frac{b-x_k}{b-a}x_k & \frac{x_k-a}{b-a} & \frac{x_k-a}{b-a}x_k \end{bmatrix} \begin{bmatrix} -a \\ 1 \\ -b \\ 1 \end{bmatrix} = 0$$

The rank of X in this case is 3. In other words, the columns of X are linearly dependent which in turn makes impossible the use of the aforementioned identification method proposed in [1]. Analyzing another example of two variables:

$$f : \mathfrak{R}^2 \longrightarrow \mathfrak{R}$$

$$y = f(x_1, x_2)$$

Each row of the matrix X is of the form:

$$\begin{aligned}X_k = & [\mu_{11}\mu_{21} \quad \mu_{11}\mu_{21}x_{1k} \quad \mu_{11}\mu_{21}x_{2k} \quad \mu_{11}\mu_{22} \quad \mu_{11}\mu_{22}x_{1k} \quad \mu_{11}\mu_{22}x_{2k} \\ & \mu_{12}\mu_{21} \quad \mu_{12}\mu_{21}x_{1k} \quad \mu_{12}\mu_{21}x_{2k} \quad \mu_{12}\mu_{22} \quad \mu_{12}\mu_{22}x_{1k} \quad \mu_{12}\mu_{22}x_{2k}] \end{aligned} \quad (8)$$

It can be noticed that the columns 1, 3, 4, and 6 have the same form as in the previous example multiplied by a constant μ_{11} and therefore, they are linearly dependent as well. The same thing happens with the columns 6, 9, 10, 12, and so on. In fact, the rank of the matrix in this case is 8.

Figure 1. Membership functions of the fuzzy system.

The solution proposed in [1] avoids the occurrence of this situation. In order to identify a function in the interval $[a_i, b_i]$ using the T-S method, certain intermediate points are chosen of the form:

$$a_i^* \in [a_i, b_i] \text{ and } b_i^* \in [a_i, b_i]$$

and they use membership functions which verify:

$$\mu_{i1}(x) = \begin{cases} \frac{x_i - b_i^*}{a_i - b_i^*} & a_i \leq x \leq b_i^* \\ 0 & b_i^* \leq x \leq b_i \end{cases} \quad (9)$$

$$\mu_{i2}(x) = \begin{cases} 0 & a_i \leq x \leq a_i^* \\ \frac{x_i - a_i^*}{b_i - a_i^*} & a_i^* \leq x \leq b_i \end{cases} \quad (10)$$

and thus:

$$\begin{aligned} \mu_{i1}(a_i) &= 1 & \mu_{i1}(b_i^*) &= 0 \\ \mu_{i2}(a_i^*) &= 0 & \mu_{i2}(b_i) &= 1 \end{aligned}$$

which impedes that the domains of these functions being overlapped and therefore it can be observed that, except for some isolated points,

$$\mu_{i1}(x_i) + \mu_{i2}(x_i) \neq 1$$

and thus, in general, the matrix X will be of full rank and the method is applicable. This solution can be clearly seen in [1] where the authors find the optimum membership functions minimizing the performance index and reducing the problem to a nonlinear programming one. For this reason, they use the well-known complex method for the minimization. This can obviously be observed in the illustrative examples selected by the authors in [1] where all the identified memberships are nonoverlapping ones.

4. NONITERATIVE APPROACH

The restriction of the T-S identification method for the case presented in the previous section does not mean the nonexistence of solutions rather than an incentive for their search. As it has been seen, the problem comes from the fact that the solution should fulfil:

$$\nabla J = X^t Y - X^t X P = 0 \quad (11)$$

But as it was shown earlier, the columns of the matrix X are linearly dependent and consequently $X^t X$ is not an invertible matrix, therefore it is impossible to calculate P through:

$$P = (X^t X)^{-1} X^t Y \quad (12)$$

Nevertheless, as the rows of X^t are linearly dependent, the independent term in (11) $X^t Y$ will have the same dependence among its rows and thereupon, the rank of the system matrix will be the same as the rank of the extended matrix with the independent term.

$$\text{rank}(X^t X) = \text{rank}(X^t X | X^t Y)$$

And so, the system has a solution. In other words, the system is a compatible indeterminate one, that is, if P is a solution of (11) and K belongs to the Kernel of $X^t X$,

$$K \in \text{Ker}(X^t X)$$

then $P^* = P + K$ will also be a solution. Therefore, the problem is not the lack of a solution rather, the existence of infinite solutions and the key idea is the ability to find one of them. Several proposals can be made to select a solution. In our case, the aim is to find solutions with lower norm.

4.1. Parameters' weighting method

An effective approach with few computational effort, based on the well known parameters' weighting method, is proposed. The main target is to improve the choice of the performance index and minimize it. It is characterized by extending the objective function by including a weighting γ of the norm of P vector.

$$J = \sum_{k=1}^m (y_k - \hat{y}_k)^2 + \gamma^2 \sum_j p_j^2 = \|Y - XP\|^2 + \gamma^2 \|P\|^2 \quad (13)$$

This can be rewritten as follows:

$$J = \|Y - XP\|^2 + \gamma^2 \|P\|^2 = \left\| \begin{bmatrix} Y \\ 0 \end{bmatrix} - \begin{bmatrix} X \\ \gamma I \end{bmatrix} P \right\|^2 = \|Y_a - X_a P\|^2 \quad (14)$$

Now, the extended matrix X_a is of full rank, and the vector P can be computed as:

$$P = (X_a^t X_a)^{-1} X_a^t Y_a \quad (15)$$

Obviously, the solution that minimizes this index is not optimum. However, for a small value of γ , it will be close to the optimum one and it will be unique as well.

The weighting of γ does not need to have a unique value, rather than some values can be weighted more than others to choose the most suitable one.

$$J = \sum_{k=1}^m (y_k - \hat{y}_k)^2 + \sum_j \gamma_i^2 p_j^2 = \|Y - XP\|^2 + \|\Gamma P\|^2 = \left\| \begin{bmatrix} Y \\ 0 \end{bmatrix} - \begin{bmatrix} X \\ \Gamma \end{bmatrix} P \right\|^2 \quad (16)$$

where Γ is a diagonal matrix formed by the values γ^2 which should necessarily have a value other than zero to guarantee that the extended matrix is invertible.

4.2. Parameters tuning using the parameters' weighting method

The parameters' weighting method can also be used for parameters tuning of the T-S model from local parameters obtained through the identification of a system in an operating region or from any physical input/output data.

We suppose that in this case, we have a first estimation

$$P_0 = [p_0^0 \ p_1^0 \ p_2^0 \ \dots \ p_n^0]^T$$

of the T-S model parameters. In order to obtain such an estimation, the classical least square method can be used around the equilibrium point. The objective is to obtain a global approximation of the system.

$$\hat{y} = p_0^0 + p_1^0 x_1 + p_2^0 x_2 + \dots + p_n^0 x_n$$

Let us analyze a set of input/output system samples $\{x_{1k}, x_{2k}, \dots, x_{nk}, y_k\}$. The parameters of the global approximation can be calculated by minimizing the following quadratic performance index:

$$J = \sum_{k=1}^m (y_k - \hat{y}_k)^2 = \|Y - X_g P_0\|^2 \quad (17)$$

where

$$\begin{aligned} Y &= [y_1 \ y_2 \ y_3 \ \dots \ y_m]^T \\ P_0 &= [p_0^0 \ p_1^0 \ p_2^0 \ \dots \ p_n^0]^T \\ X_g &= \begin{bmatrix} 1 & x_{11} & x_{21} & \dots & x_{n1} \\ 1 & x_{12} & x_{22} & \dots & x_{n2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{1m} & x_{2m} & \dots & x_{nm} \end{bmatrix} \end{aligned} \quad (18)$$

In this case, if we select a sufficient number of points distributed in the region where it is required to obtain the approximation, then the matrix X_g will be of a full rank and therefore, the solution becomes unique, which can be calculated as follows:

$$P_0 = (X_g^T X_g)^{-1} X_g^T Y \quad (19)$$

This first approximation can be utilized as reference parameters for all the subsystems. Then, the parameters' vector of the fuzzy model can be obtained minimizing:

$$\begin{aligned} J &= \sum_{k=1}^m (y_k - \hat{y}_k)^2 + \gamma^2 \sum_{i_1=1}^{r_1} \dots \sum_{i_n=1}^{r_n} \sum_{j=0}^n (p_j^0 - p_j^{(i_1 \dots i_n)})^2 = \|Y - Xp\|^2 + \gamma^2 \|p_0 - p\|^2 \\ &= \left\| \begin{bmatrix} Y \\ \gamma p_0 \end{bmatrix} - \begin{bmatrix} X \\ \gamma I \end{bmatrix} p \right\|^2 = \|Y_a - X_a p\|^2 \end{aligned} \quad (20)$$

where

$$p_0 = \underbrace{[P_0 \ P_0 \ \dots \ P_0]^T}_{r_1 \cdot r_2 \dots r_n}$$

In this case, the factor γ represents the degree of confidence of the parameters initially estimated. In a similar way to the previous case, different weight factors of $\gamma_j^{(i_1 \dots i_n)}$ can be used to each one of the parameters $p_j^{(i_1 \dots i_n)}$ depending on the reliability of the initial parameter p_j^0 in the specific rule.

$$\begin{aligned} J &= \sum_{k=1}^m (y_k - \hat{y}_k)^2 + \sum_{i_1=1}^{r_1} \dots \sum_{i_n=1}^{r_n} \sum_{j=0}^n \gamma_j^{(i_1 \dots i_n)2} (p_j^0 - p_j^{(i_1 \dots i_n)})^2 = \|Y - Xp\|^2 + \|\Gamma(p_0 - p)\|^2 \\ &= \left\| \begin{bmatrix} Y \\ \Gamma p_0 \end{bmatrix} - \begin{bmatrix} X \\ \Gamma \end{bmatrix} p \right\|^2 = \|Y_a - X_a p\|^2 \end{aligned} \quad (21)$$

where Γ is a diagonal matrix with the weight factor $\gamma_j^{(i_1 \dots i_n)}$. It is not necessary to apply this process for all the parameters. If the values of some of them are known, they can be fixed beforehand or we can assign them a weighting factor $\gamma_j^{(i_1 \dots i_n)}$ comparatively high.

5. ITERATIVE PARAMETERS' IDENTIFICATION

The inconvenient feature of the aforementioned method is the amplification of the matrix X throughout the time, so that it becomes inappropriate to be used in real time applications as adaptive control for example. The solution is find an iterative method so that the dimension of the calculation will not be augmented for each sample. As explained in detail in Section 3 that the solution developed in [1] to find the optimum membership functions is invalid when memberships are overlapping ones. In this section, we use an iterative method based on the extended Kalman filter to solve with this problem.

5.1. The Kalman filter

The Kalman filter is widely used for stochastic estimation. It is developed by Rudolph E. Kalman, through a recursive method for the discrete data linear filtering [40]. The Kalman filter is known to be optimum for linear systems with white process and measurement noises. It is assumed that the system is described by the following sampled model:

$$\begin{aligned} x(k+1) &= \Phi x(k) + \Gamma u(k) + v(k) \\ y(k) &= Cx(k) + e(k) \end{aligned} \quad (22)$$

$$x(k), x(k+1), v(k) \in \mathbb{R}^n \quad u(k) \in \mathbb{R}^m \quad y(k), e(k) \in \mathbb{R}^p$$

where $x(k)$ represents the state of the dynamic system, $u(k)$ is the input vector and $y(k)$ is the output vector. The vector $v(k)$ represents the Gaussian-white noise of the system and $e(k)$ is the measured Gaussian-white noise. Both of them are independent from each other with zero mean. The objective of the Kalman filter is to obtain an optimum estimation $\hat{x}(k)$ of the state $x(k)$ from measurements of the input/output vectors. The covariance matrices are supposed to be known and are given as:

$$\begin{aligned} R_1 &= E(v(k), v^t(k)) \\ R_{12} &= E(v(k), e^t(k)) \\ R_2 &= E(e(k), e^t(k)) \end{aligned} \quad (23)$$

It is also assumed that the initial condition $x(0)$ is a Gaussian distributed with

$$\begin{aligned} m_0 &= E(x(0)) \\ R(0) &= E((x(0) - m_0)(x(0) - m_0)^t) \end{aligned} \quad (24)$$

where $E(\cdot)$ is the expectation operator. It is supposed that $\hat{x}(k/k-1)$, $u(k)$ and $y(k)$ are known, and the objective is to estimate $\hat{x}(k+1/k)$. The prediction problem can be improved by introducing the difference between the measured and estimated outputs, $(y(k) - C\hat{x}(k/k-1))$ as a feedback gain:

$$\hat{x}(k+1/k) = \Phi\hat{x}(k/k-1) + \Gamma u(k) + K(k)(y(k) - C\hat{x}(k/k-1))$$

The resultant prediction error is the difference between the state of the real system and the estimated one which can be stated as the following:

$$\epsilon(k+1) = x(k+1) - \hat{x}(k+1/k) \quad (25)$$

It should be observed that as the aforementioned Gaussian errors $v(k)$ and $e(k)$ are with zero mean, it can be verified that:

$$\hat{\epsilon}(k+1) = (\Phi - K(k)C)\hat{\epsilon}(k) \quad (26)$$

Thus, if $\bar{\epsilon}(0) (\Rightarrow \hat{x}(0) = m_0) \Rightarrow \forall k > 0 \quad \epsilon(k) = 0 \quad (\hat{x}(k) = m_k)$

And if the dynamics of (26) is stable, then:

$$\forall x(0) \quad \lim_{k \rightarrow \infty} \bar{\epsilon}(k) = 0 \Rightarrow \lim_{k \rightarrow \infty} \hat{x}(k) = m_k$$

The secondary objective is to minimize the covariance matrix which is denoted as $P(k)$,

$$P(k) = E((\epsilon - \bar{\epsilon})(\epsilon - \bar{\epsilon})^t)$$

in the sense that it approaches its minimum for:

$$\min(\alpha^t P(k) \alpha) \quad \forall \alpha \in \mathbb{R}^n$$

The algorithm of the Kalman filter can be summarized by the following iterative process:

$$\begin{aligned} K(k) &= (\Phi P(k/k-1)C^t + R_{12}) (CP(k/k-1)C^t + R_2)^{-1} \\ \hat{x}(k+1/k) &= \Phi\hat{x}(k/k-1) + \Gamma u(k) + K(k)(y(k) - C\hat{x}(k/k-1)) \\ P(k+1/k) &= (\Phi P(k/k-1)\Phi^t + R_1 - K(k)) (CP(k/k-1)\Phi^t + R_{12}^t) \end{aligned} \quad (27)$$

This process is initialized with $\hat{x}(0) = m_0$ and $P(0) = R_0$ which have been initially estimated. The classic formulation of the Kalman filter can be complemented with an additional useful filtering process for certain applications.

5.2. Extended Kalman filter

The Kalman filter can also be used for state estimation of nonlinear systems. It is based on the same idea of the parameters' weighting method. For nonlinear systems, for example fuzzy systems, the Kalman filter cannot be applied directly; but if the nonlinearity of the system be sufficiently smooth, then we can linearize it about the current mean and covariance of the state estimation. This is called extended Kalman filter with white process and measurement noises. Derivations of the extended Kalman filter are widely available in the literature [41]. The extended Kalman filter can be considered as a predictor-corrector type linear estimator obtained by the linearization of a nonlinear model at each time step. It is used to estimate the states and parameters of a nonlinear system through the measurements using a function of the linearized model with additive Gaussian white noise. The extended Kalman filter procedure consists of two steps: time update step and measurement update step. The time update step projects forward the current state and error covariance estimates to obtain the a priori estimates for the next time step. The measurement update step incorporates a new measurement into the a priori estimate to get an improved a posteriori estimate. In other words, the time update step is a model prediction and the measurement update step is a measurement correction. In this section, we briefly outline the algorithm and show how it can be applied to fuzzy system optimization. Consider a nonlinear discrete time system of the form:

$$\begin{aligned} x(k+1) &= f(x(k), u(k)) + v(k) \\ y(k) &= g(x(k)) + e(k) \end{aligned} \quad (28)$$

In this case, the Jacobian matrices are those which represent the nonlinear systems:

$$\begin{aligned} \Phi(x(k), u(k)) &= \frac{\partial f}{\partial x} \Big|_{x=x(k), u=u(k)} \\ \Gamma(x(k), u(k)) &= \frac{\partial f}{\partial u} \Big|_{x=x(k), u=u(k)} \\ C(x(k)) &= \frac{\partial g}{\partial x} \Big|_{x=x(k)} \end{aligned} \quad (29)$$

Moreover, the prediction formula for the nonlinear case is the following:

$$\hat{x}(k+1) = \Phi \hat{x}(k/k-1) + \Gamma(k) + K(k)(y(k) - g(\hat{x}(k/k-1), u(k))) \quad (30)$$

It must be noted that in this case, the system matrices in this depend on both the state and input of the system in each instant. Thus, it becomes necessary the calculation of these matrices in each iteration of the algorithm.

5.3. The Kalman filter for parameters' estimation

One of the applications of the Kalman filter is the identification of parameters. Let us suppose that a function depends on q parameters $p_1, p_2 \dots p_q$

$$\begin{aligned} f : \mathbb{R}^n &\rightarrow \mathbb{R}^m \\ y &= f(x_1, x_2, \dots, x_n, p_1, p_2 \dots p_q) = f(x, p) \end{aligned} \quad (31)$$

The problem of the identification of parameters can be explained as a problem of estimation of the systems' states.

$$\begin{aligned} p(k+1) &= p(k) \\ y(k+1) &= f(p(k)) + e(k) \end{aligned} \quad (32)$$

Then, if we have a set of m samples $\{x_{1k}, x_{2k}, \dots, x_{nk}, y_k\}$ of the function to be identified, the Kalman filter can be used with the following particularities. The matrix Φ will be an identity matrix in this case. It is assumed a free system without an external input so the matrix Γ is null and the matrix C can be calculated as follows:

$$C(p(k)) = \frac{\partial f}{\partial p} \Big|_{p=\hat{p}(k)}$$

The matrices R_1 y R_{12} become null, whereas R_2 is selected based on trial and error. If $y \in \Re$ and we suppose that $R_2 = I$, which would correspond to Gaussians error functions $N(0,1)$, and the function is a linear one, the algorithm becomes equivalent to the recursive minimum square one. The initial covariance state matrix is supposed to be $P(0) = cI$, where C is a number relatively large with respect to the data of the problem. The formulation of the algorithm becomes:

$$\begin{aligned} K(k) &= (P(k/k-1)C^t) (CP(k/k-1)C^t + R_2)^{-1} \\ \hat{p}(k+1/k) &= \hat{p}(k/k-1) + K(k)(y_k - f(x_{1k}, x_{2k}, \dots, x_{nk}, \hat{p}(k/k-1))) \\ P(k+1/k) &= P(k/k-1) - K(k)CP(k/k-1) \end{aligned} \quad (33)$$

5.4. Application of the Kalman filter for the T-S fuzzy model

To overcome the problems caused by the inaccuracy of the model or changes of operating conditions, the process model has to be updated with appropriate parameter identification techniques [42]. In the light of control view points, the dynamic models for various model-based controllers involve state variables. However, in most cases, the full states are not available online for the state feedback control. If the states for the state feedback control are not available, then they should resort to state estimators. Motivated by the successful use of the Kalman filter for training neural networks [43] and for defuzzification strategies [44], we can apply a similar method to the training of fuzzy systems. In general, we can view the identification of fuzzy systems as a weighted least-squares minimization problem, where the error vector is the difference between the fuzzy model outputs and the target values for those outputs. As mentioned previously, this algorithm cannot be applied directly for the identification of the fuzzy T-S models when the membership functions are overlapped by pairs. The proposed solution for its application is to combine it with a minimization of a weighting of the norm of the vector of parameters p .

Let a function be represented as:

$$\begin{aligned} f : \Re^n &\rightarrow \Re \\ y &= f(x_1, x_2, \dots, x_n) \end{aligned} \quad (34)$$

The optimum approximation of the function is searched by describing the function as a fuzzy system represented in the following form:

$$\begin{aligned} S^{(i_1 \dots i_n)} : \text{If } x_1 \text{ is } M_1^{i_1} \text{ and } \dots x_n \text{ is } M_n^{i_n} \text{ then} \\ \hat{y} &= p_0^{(i_1 \dots i_n)} + p_1^{(i_1 \dots i_n)} x_1 + p_2^{(i_1 \dots i_n)} x_2 + \dots + p_n^{(i_1 \dots i_n)} x_n \end{aligned} \quad (35)$$

In order to cast the fuzzy system identification problem in a form suitable for the Kalman filtering, we let the parameters of the rules constitute the state of a nonlinear system, and we consider the output of the fuzzy system as the output of the nonlinear system to which the Kalman filter is applied.

$$\begin{aligned} p(k+1) &= p(k) \\ y_k &= f(x_{1k}, x_{2k}, \dots, x_{nk}, p(k)) + e(k) \end{aligned} \quad (36)$$

Nevertheless, if the membership are overlapped by pairs, it becomes impossible to apply this method directly to the system, because as mentioned in Section (3), the problem does not have a unique solution which leads to convergence problem of the algorithm. The solution presented here is equivalent to weighting of parameters approach explained in Section (4), which is characterized by extending the objective function by including a weighting of the norm of P vector. We propose to describe the problem as an optimum estimation of the state of the system.

$$\begin{aligned} p(k+1) &= p(k) \\ \begin{bmatrix} y_k \\ \delta p_0 \end{bmatrix} &= \begin{bmatrix} f(x_{1k}, x_{2k}, \dots, x_{nk}, p(k)) \\ \delta p(k) \end{bmatrix} + e(k) \end{aligned} \quad (37)$$

where δ is a relatively small positive number in order not to disturb the principal objective of the problem. It should be noted that the matrix C of the algorithm vary as follows:

$$C(\hat{p}(k)) = \begin{bmatrix} \frac{\partial f}{\partial p} |_{p=\hat{p}(k)} \\ \delta I \end{bmatrix}$$

It should be noticed that the function f is a linear one with respect to the parameters and therefore it can be calculated in a direct form as follows:

$$\frac{\partial f}{\partial p_0^{(i_1 \dots i_n)}} |_{p=\hat{p}(k)} = \frac{w^{(i_1 \dots i_n)}(x_k)}{\sum_{i_1=1}^{r_1} \dots \sum_{i_n=1}^{r_n} w^{(i_1 \dots i_n)}(x_k)}$$

and

$$\frac{\partial f}{\partial p_j^{(i_1 \dots i_n)}} |_{p=\hat{p}(k)} = \frac{w^{(i_1 \dots i_n)}(x_k)}{\sum_{i_1=1}^{r_1} \dots \sum_{i_n=1}^{r_n} w^{(i_1 \dots i_n)}(x_k)} x_{jk} \text{ for } j = 1 \dots n \quad (38)$$

Therefore, the Jacobian coincides with the row k of the matrix defined in Section (2).

$$\frac{\partial}{\partial p} |_{p=\hat{p}(k)} = X_k = \left[\beta_k^{(1 \dots 1)} \beta_k^{(1 \dots 1)} x_{1k} \dots \beta_k^{(1 \dots 1)} x_{nk} \dots \beta_k^{(r_1 \dots r_n)} \dots \beta_k^{(r_1 \dots r_n)} x_{nk} \right] \quad (39)$$

And thus, the problem can be formulated as an estimation of the state of the linear system

$$\begin{aligned} p(k+1) &= p(k) \\ \begin{bmatrix} y_k \\ \delta p_0 \end{bmatrix} &= C(\hat{p}(k)) \cdot p(k) + e(k) \end{aligned} \quad (40)$$

where

$$C(\hat{p}(k)) = \begin{bmatrix} X_k \\ \delta I \end{bmatrix}$$

The prediction formula in this case becomes:

$$\hat{p}(k+1/k) = \hat{p}(k/k-1) + K(k) \left(\begin{bmatrix} y_k \\ \delta p_0 \end{bmatrix} - C(p(k)) \cdot p(k) \right)$$

In order that this iterative method becomes equivalent to the noniterative one based on weighting of parameters mentioned in Section (4), we should taken into consideration that the factor γ of the non-iterative method weights the norm of the parameters in front of the quadratic sum of errors resulted for the modeling of samples. On the other hand, the factor weights the norm of parameters in front of each one of the errors of samples. If m is the number of used samples, then the equivalence is produced supposing that $\delta = \gamma/m$ and in general $\delta \ll \gamma$ should be fulfilled.

6. DESIGN OF AN OPTIMAL CONTROLLER

In order to show the effectiveness of the proposed estimation methods, a design of an optimal controller is carried out for a dynamic system whose model is of the following form:

$$x^{(n)} = f(x, x', \dots, x^{(n-1)}, u)$$

Applying the proposed estimation method mentioned, the T-S model can be adjusted as follows: The IF-THEN rules are as follows:

$$\begin{aligned} &S^{(i_1 \dots i_n)} : \text{If } x \text{ is } M_1^{i_1} \text{ and } x' \text{ is } M_2^{i_2} \text{ and } \dots \text{ and } x^{(n-1)} \text{ is } M_n^{i_n}, \\ \text{then } x^{(n)} &= a_0^{(i_1 \dots i_n)} + a_1^{(i_1 \dots i_n)} x + a_2^{(i_1 \dots i_n)} x' + \dots + a_n^{(i_1 \dots i_n)} x^{(n-1)} + b^{(i_1 \dots i_n)} u \end{aligned} \quad (41)$$

where $M_1^{i_1}$ ($i_1 = 1, 2, \dots, r_1$) are fuzzy sets for x , $M_2^{i_2}$ ($i_2 = 1, 2, \dots, r_2$) are fuzzy sets for x' and $M_n^{i_n}$ ($i_n = 1, 2, \dots, r_n$) are fuzzy sets for $x^{(n-1)}$.

The fuzzy system is described as:

$$x^{(n)} = \frac{\sum_{i_1=1}^{r_1} \dots \sum_{i_n=1}^{r_n} w^{(i_1 \dots i_n)}(\mathbf{x}) \left[a_0^{(i_1 \dots i_n)} + a_1^{(i_1 \dots i_n)} x \right]}{\sum_{i_1=1}^{r_1} \dots \sum_{i_n=1}^{r_n} w^{(i_1 \dots i_n)}(\mathbf{x})} + \frac{\sum_{i_1=1}^{r_1} \dots \sum_{i_n=1}^{r_n} w^{(i_1 \dots i_n)}(\mathbf{x}) \left[a_2^{(i_1 \dots i_n)} x' + \dots + a_n^{(i_1 \dots i_n)} x^{(n-1)} + b^{(i_1 \dots i_n)} u \right]}{\sum_{i_1=1}^{r_1} \dots \sum_{i_n=1}^{r_n} w^{(i_1 \dots i_n)}(\mathbf{x})} \quad (42)$$

The controller fuzzy rule is represented in a similar form:

$$C^{(i_1 \dots i_n)} : \text{If } x \text{ is } M_1^{i_1} \text{ and } x' \text{ is } M_2^{i_2} \text{ and } \dots \text{ and } x^{(n-1)} \text{ is } M_n^{i_n}, \\ \text{then } u = r - (k_0^{(i_1 \dots i_n)} + k_1^{(i_1 \dots i_n)} x + k_2^{(i_1 \dots i_n)} x' + \dots + k_n^{(i_1 \dots i_n)} x^{(n-1)}) \quad (43)$$

In order to design the control system, it will be supposed that the system can be modelled in closed-loop by substituting (43) in (41), assuming that the resultant error is negligible:

$$SC^{(i_1 \dots i_n)} : \text{If } x \text{ is } M_1^{i_1} \text{ and } x' \text{ is } M_2^{i_2} \text{ and } \dots \text{ and } x^{(n-1)} \text{ is } M_n^{i_n}, \\ \text{then } x^{(n)} = a_0^{(i_1 \dots i_n)} + a_1^{(i_1 \dots i_n)} x + a_2^{(i_1 \dots i_n)} x' + \dots + a_n^{(i_1 \dots i_n)} x^{(n-1)} \\ + b^{(i_1 \dots i_n)} [r - (k_0^{(i_1 \dots i_n)} + k_1^{(i_1 \dots i_n)} x + k_2^{(i_1 \dots i_n)} x' + \dots + k_n^{(i_1 \dots i_n)} x^{(n-1)})] \quad (44)$$

6.1. Modeling error of the feedback system

In this section, we will analyze the errors produced in the previous formulation. Consider the following first order monovariate system:

$$x' = f(x, u)$$

By modeling the system as a fuzzy one as we have explained in Section (2), we obtain:

$$S^1 : \text{If } x \text{ is } M_1^1, \text{ then } x' = a_{10} + a_{11}x + b_1u \\ S^2 : \text{If } x \text{ is } M_1^2, \text{ then } x' = a_{20} + a_{21}x + b_2u \quad (45)$$

and so the resultant fuzzy system is,

$$x' = \mu_1(x)(a_{10} + a_{11}x + b_1u) + \mu_2(x)(a_{20} + a_{21}x + b_2u) \quad (46)$$

Supposing that the controller to be designed has the following form:

$$R^1 : \text{If } x \text{ is } M_1^1, \text{ then } u = k_{10} + k_{11}x \\ R^2 : \text{If } x \text{ is } M_1^2, \text{ then } u = k_{20} + k_{21}x \quad (47)$$

Therefore, the FC is:

$$u = \mu_1(x)(k_{10} + k_{11}x) + \mu_2(x)(k_{20} + k_{21}x) \quad (48)$$

Substituting (48) in (46), we get the feedback fuzzy system:

$$x' = \mu_1(x)[a_{10} + a_{11}x + b_1(\mu_1(x)(k_{10} + k_{11}x) + \mu_2(x)(k_{20} + k_{21}x))] \\ + \mu_2(x)[a_{20} + a_{21}x + b_2(\mu_1(x)(k_{10} + k_{11}x) + \mu_2(x)(k_{20} + k_{21}x))] \\ = \mu_1(x)(a_{10} + a_{11}x) + \mu_2(x)(a_{20} + a_{21}x) \\ + [\mu_1(x)]^2 b_1(k_{10} + k_{11}x) + \mu_1(x)\mu_2(x)b_1(k_{20} + k_{21}x) \\ + [\mu_2(x)]^2 b_2(k_{20} + k_{21}x) + \mu_1(x)\mu_2(x)b_2(k_{10} + k_{11}x) \quad (49)$$

On the other hand, if the feedback system is modelled as explained in the previous section, we obtain:

$$\begin{aligned} RS^1 : \text{If } x \text{ is } M_1^1, \text{ then } x' &= a_{10} + a_{11}x + b_1(k_{10} + k_{11}x) \\ RS^2 : \text{If } x \text{ is } M_1^2, \text{ then } x' &= a_{20} + a_{21}x + b_2(k_{20} + k_{21}x) \end{aligned} \quad (50)$$

The fuzzy feedback controlled system becomes:

$$\begin{aligned} x' &= \mu_1(x)(a_{10} + a_{11}x + b_1(k_{10} + k_{11}x)) + \mu_2(x)(a_{20} + a_{21}x + b_2(k_{20} + k_{21}x)) \\ &= \mu_1(x)(a_{10} + a_{11}x) + \mu_2(x)(a_{20} + a_{21}x) + \mu_1(x)b_1(k_{10} + k_{11}x) + \mu_2(x)b_2(k_{20} + k_{21}x) \end{aligned} \quad (51)$$

The modeling error of this formulation in comparison with the first one which was more accurate can be obtained by subtracting (51) from (49):

$$\begin{aligned} e &= [\mu_1(x)]^2 b_1(k_{10} + k_{11}x) + \mu_1(x)\mu_2(x)b_1(k_{20} + k_{21}x) \\ &\quad + [\mu_2(x)]^2 b_2(k_{20} + k_{21}x) + \mu_1(x)\mu_2(x)b_2(k_{10} + k_{11}x) \\ &\quad - \mu_1(x)b_1(k_{10} + k_{11}x) - \mu_2(x)b_2(k_{20} + k_{21}x) \\ &= \mu_1(x)(\mu_1(x) - 1)b_1(k_{10} + k_{11}x) + \mu_2(x)(\mu_2(x) - 1)b_2(k_{20} + k_{21}x) \\ &\quad + \mu_1(x)\mu_2(x)b_1(k_{20} + k_{21}x) + \mu_1(x)\mu_2(x)b_2(k_{10} + k_{11}x) \end{aligned} \quad (52)$$

Because triangular overlapping membership functions are used, it can be confirmed that $\mu_1(x) + \mu_2(x) = 1$. By substituting this equation in (52), we get:

$$\begin{aligned} e &= -\mu_1(x)\mu_2(x)b_1(k_{10} + k_{11}x) - \mu_1(x)\mu_2(x)b_2(k_{20} + k_{21}x) \\ &\quad + \mu_1(x)\mu_2(x)b_1(k_{20} + k_{21}x) + \mu_1(x)\mu_2(x)b_2(k_{10} + k_{11}x) \\ &= \mu_1(x)\mu_2(x)[b_1(k_{20} + k_{21}x) + b_2(k_{10} + k_{11}x) - b_1(k_{10} + k_{11}x) - b_2(k_{20} + k_{21}x)] \\ &= \mu_1(x)\mu_2(x)(b_1 - b_2)[(k_{20} - k_{10} + (k_{21} - k_{11})x)] \end{aligned} \quad (53)$$

In order to analyze this error, we will compare the terms that correspond to the control part in (51)

$$c = \mu_1(x)b_1(k_{10} + k_{11}x) + \mu_2(x)b_2(k_{20} + k_{21}x) \quad (54)$$

and the resultant error obtained in (53) is:

$$e = \mu_1(x)\mu_2(x)(b_1 - b_2)[k_{20} - k_{10} + (k_{21} - k_{11})x] \quad (55)$$

we should observe that:

$$\left. \begin{aligned} 0 \leq \mu_1(x) \leq 1 \\ 0 \leq \mu_2(x) \leq 1 \end{aligned} \right\} \implies \left\{ \begin{aligned} 0 \leq \mu_1(x)\mu_2(x) \leq \mu_1(x) \\ 0 \leq \mu_1(x)\mu_2(x) \leq \mu_2(x) \end{aligned} \right. \quad (56)$$

Thus, the weighting of the error term is always less than any other term of control. Moreover, as the membership functions are triangulares which verifies that $\mu_1(x)\mu_2(x) \leq 0.25$, whereas in the control action, there is always a term whose weighting value is greater than 0.5.

On the other hand, if the distance between two defined models in two adjacent rules is very small, we can verify that,

$$b_1 \approx b_2$$

It should be noted that the systems modelled by applying the T-S model are smooth without large variations between adjacent subsystems. Moreover, the identification with the methodology of parameters tuning penalizes the distance to the initial parameters of the system. Therefore this criterion is always fulfilled.

Furthermore, if equivalent criteria are used for the control design, the distance between the two controllers becomes small and it will verify:

$$k_{10} \approx k_{20}$$

$$k_{11} \approx k_{21}$$

we can then conclude that the modeling error resulted by applying (51) is very small with respect to the control effect; therefore, it can be considered negligible and thus we can use (51) for the design of the controller. In other words, each control rule can be designed in an individual way associated to the corresponding system rule. This result can be generalized in a similar manner to any system of any dimension.

It is interesting to note that if this formulation is correct, then the stability of the system will be ensured, because in general, it can set the values of the matrix K so that for any pair of desired values a_0, a_1

$$\begin{aligned} a_0 &= a_{10} + b_1 k_{10} = a_{20} + b_2 k_{20} \\ a_1 &= a_{11} + b_1 k_{11} = a_{21} + b_2 k_{21} \end{aligned} \quad (57)$$

and therefore

$$x' = \mu_1(x)(a_0 + a_1 x) + \mu_2(x)(a_0 + a_1 x) = (\mu_1(x) + \mu_2(x))(a_0 + a_1 x). \quad (58)$$

Taking into account that overlapping triangular membership functions are used, it can be verified that:

$$\mu_1(x) + \mu_2(x) = 1 \quad (59)$$

then (58) becomes

$$x' = (\mu_1(x) + \mu_2(x))(a_0 + a_1 x) = a_0 + a_1 x \quad (60)$$

And because a_0, a_1 can be arbitrarily set, then the stability of the feedback system can be guaranteed.

6.2. Calculation of the affine term

The proposed methodology of design is based on the possibility to formulate the feedback system as shown previously in (44),

$$\begin{aligned} SC^{(i_1 \dots i_n)} : \text{ If } x \text{ is } M_l^{i_l} \text{ and } x' \text{ is } M_2^{i_2} \text{ and } \dots \text{ and } x^{(n-1)} \text{ is } M_n^{i_n}, \\ \text{then } x^{(n)} &= a_o^{(i_1 \dots i_n)} + a_1^{(i_1 \dots i_n)} x + a_2^{(i_1 \dots i_n)} x' + \dots + a_n^{(i_1 \dots i_n)} x^{(n-1)} \\ &+ b^{(i_1 \dots i_n)} [r - (k_0^{(i_1 \dots i_n)} + k_1^{(i_1 \dots i_n)} x + k_2^{(i_1 \dots i_n)} x' + \dots + k_n^{(i_1 \dots i_n)} x^{(n-1)})] \end{aligned} \quad (61)$$

Firstly, the affine term of the control action is used to eliminate the affine term of the system:

$$\begin{aligned} a_o^{(i_1 \dots i_n)} + b^{(i_1 \dots i_n)} k_0^{(i_1 \dots i_n)} &= 0 \\ k_0^{(i_1 \dots i_n)} &= \frac{-a_o^{(i_1 \dots i_n)}}{b^{(i_1 \dots i_n)}} \end{aligned}$$

and the feedback system is rewritten as follows:

$$\begin{aligned} SC^{(i_1 \dots i_n)} : \text{ If } x \text{ is } M_l^{i_l} \text{ and } x' \text{ is } M_2^{i_2} \text{ and } \dots \text{ and } x^{(n-1)} \text{ is } M_n^{i_n}, \\ \text{then } x^{(n)} &= a_1^{(i_1 \dots i_n)} x + a_2^{(i_1 \dots i_n)} x' + \dots + a_n^{(i_1 \dots i_n)} x^{(n-1)} \\ &+ b^{(i_1 \dots i_n)} [r - k_1^{(i_1 \dots i_n)} x + k_2^{(i_1 \dots i_n)} x' + \dots + k_n^{(i_1 \dots i_n)} x^{(n-1)}] \end{aligned} \quad (62)$$

6.3. State space feedback control based on the proposed estimation approach

In order to calculate the rest of the control coefficients, any control design methodology through state feedback control can be applied. By using the pole assignment method for example, the desired characteristics and the coincidence of all the subsystems (see Equation 62) can be achieved, ensuring the stability of the resulting feedback system, except for the errors discussed in Section (6.1) that can be considered negligible in this case. However, this methodology can lead to systems where the control action exceeds the allowed environment limits. Selecting closed loop pole with great negative real parts makes the dynamic response of the system to be quick, whereas the control effort to be greater than permissible levels. If selection of the closed loop poles makes saturation of control signals, dynamic behavior of the system will not be as same as the desired behavior even it may become unstable. Therefore, optimal selection of closed loop poles will lead to a trade-off between speed of dynamic response and control effort. Together with the proposed estimation method, the well-known linear quadratic regulator (LQR) method might be an appropriate choice. With this method, we lose the guarantee of stability, which needs to be analyzed a posteriori, gaining in return a balance between static and dynamic behavior of the system with admissible control actions. Q is a non-negative definite matrix which is called weighting matrix. On the other hand, R is a positive definite weighting matrix. Selection of values of these matrices determines the dynamic speed of the controller as well as amplitudes of state variables and control signals. For example, if R is selected small, whereas Q is selected large, more stability will be attained with large control efforts. The system represented in state space form is:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$

$$x \in \mathbb{R}^n, u \in \mathbb{R}^m, A \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m}$$

The objective is to find the control action $u(t)$ to transfer the system from any initial state $x(t_0)$ to some final state $x(\infty) = 0$ in an infinite time interval, minimizing a quadratic performance index of the form:

$$J = \int_{t_0}^{\infty} (x^t Q x + u^t R u) dt$$

where $Q \in \mathbb{R}^{n \times n}$ is a symmetric matrix, at least positive semidefinite one and $R \in \mathbb{R}^{m \times m}$ is also a symmetric positive definite matrix, and K is referred to as the state feedback gain matrix. The optimal control law is then computed as follows:

$$u(t) = -Kx(t) = -R^{-1}B^t Lx(t) \quad (63)$$

where the matrix $L \in \mathbb{R}^{n \times n}$ is a solution of the Riccati equation:

$$0 = -Q + LBR^{-1}B^t L - LA - A^t L$$

The design algorithm includes firstly, the cancelation of the affine term as explained in Section (6.2). Thus, each subsystem will be as follows:

$$x^{(n)} = a_1^{(i_1 \dots i_n)} x + a_2^{(i_1 \dots i_n)} x' + \dots + a_n^{(i_1 \dots i_n)} x^{(n-1)} + b^{(i_1 \dots i_n)} u \quad (64)$$

The system is then represented in a state space form as:

$$\mathbf{A}^{(i_1 \dots i_n)} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \dots & 0 & 1 \\ a_1^{(i_1 \dots i_n)} & a_2^{(i_1 \dots i_n)} & a_3^{(i_1 \dots i_n)} & \dots & a_n^{(i_1 \dots i_n)} \end{bmatrix}, \mathbf{B}^{(i_1 \dots i_n)} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ b^{(i_1 \dots i_n)} \end{bmatrix},$$

$$\mathbf{x}^T = [x \quad x' \quad \dots \quad x^{n-1}]$$

Secondly, the LQR methodology is applied for each subsystem using a common state weighting matrix Q and input matrix R for all the rules. Thus, the Riccati equation is solved for each subsystem as follows:

$$0 = -Q + L^{(i_1 \dots i_n)} B^{(i_1 \dots i_n)} R^{-1} B^{(i_1 \dots i_n)^t} L^{(i_1 \dots i_n)} - L^{(i_1 \dots i_n)} A^{(i_1 \dots i_n)} - A^{(i_1 \dots i_n)^t} L^{(i_1 \dots i_n)}$$

Then the the state feedback gain vector can be obtained from (63):

$$K^{(i_1 \dots i_n)} = \begin{bmatrix} k_1^{(i_1 \dots i_n)} & k_2^{(i_1 \dots i_n)} & \dots & k_n^{(i_1 \dots i_n)} \end{bmatrix} = R^{-1} B^{(i_1 \dots i_n)^t} L^{(i_1 \dots i_n)}$$

As the distance between two adjacent subsystems is small, it is expected that the distance between the corresponding matrices $L^{(i_1 \dots i_n)}$ will be small as well, and therefore, the same will occur for the distance between the corresponding state feedback matrices $K^{(i_1 \dots i_n)}$ of the adjacent subsystems.

7. ILLUSTRATIVE EXAMPLE

In this section, the proposed estimation method and its application to the control design of the FC-LQR is illustrated by an example of an inverted pendulum.

Consider the problem of stabilizing and balancing of swing up of an inverted pendulum (see Figure 2). The control of this system is a widely used performance measure of a controller, because this system is unstable and highly nonlinear. The objective is to maintain the inverted pendulum upright with θ despite small disturbances because of wind or system noises.

The inverted pendulum can be represented as follows:

$$\ddot{\theta} = \frac{g \sin \theta - \cos \theta \left(\frac{u + m l \dot{\theta}^2 \sin \theta}{M + m} \right)}{l \left(\frac{4}{3} - \frac{m \cos^2 \theta}{M + m} \right)} \quad (65)$$

where θ denotes the angular position (in radians) deviated from the equilibrium position (vertical axis) of the pendulum and $\dot{\theta}$ is the angular velocity, g (gravity acceleration) = $9.8 \frac{m}{sec^2}$, M (mass) of the cart = 1 kg, m (mass) of the pole = 0.1 kg, l is the distance from the center of the mass (m) of the pole to the cart = 0.5 m.

Figure 2. Inverted pendulum system.

Assuming that $x_1 = \theta$ and $x_2 = \dot{\theta}$, then (65) can be rewritten in state space form as follows:

$$\begin{aligned} x_1 &= \theta \\ x_1' &= x_2 \\ x_2 &= \dot{\theta} \\ x_2' &= \frac{g \sin(x_1) - \cos(x_1) \left(\frac{u + m l x_2^2 \sin(x_1)}{M+m} \right)}{l \left(\frac{4}{3} - \frac{m \cos^2(x_1)}{M+m} \right)} \end{aligned} \quad (66)$$

The aim is to move the pendulum to its unstable equilibrium position, that is, $x_1 = x_2 = u = 0$

Firstly, the model of the inverted pendulum is estimated in three operation points for both the angle and its derivative. The universe of discourse of the angle is $[-\frac{\pi}{4}, \frac{\pi}{4}] \text{ rad}$, and one of the angular velocity is $[-5, 5] \frac{\text{rad}}{\text{seg}}$. Both membership functions for the angle x and its derivative $\dot{x}$ are shown in Figures 3–4, respectively.

If we apply the T-S method directly to this example, then the condition number of the matrix X is $3.4148e + 015$, which shows clearly a nonreliable result. Using the parameters' weighting method with weighting factor $\gamma = 0.01$, the inverted pendulum fuzzy model can be represented as follows:

$$\begin{aligned} S^{11} : & \text{If } (x_1 \text{ is } M_1^1) \text{ and } (x_2 \text{ is } M_2^1), \text{ then } \dot{x}_2 = -8.1994 + 3.4151x_1 - 0.2006x_2 - 1.0443u \\ S^{12} : & \text{If } (x_1 \text{ is } M_1^1) \text{ and } (x_2 \text{ is } M_2^2), \text{ then } \dot{x}_2 = -8.3766 + 3.0426x_1 + 0.0000x_2 - 1.0443u \\ S^{13} : & \text{If } (x_1 \text{ is } M_1^1) \text{ and } (x_2 \text{ is } M_2^3), \text{ then } \dot{x}_2 = -8.1994 + 3.4151x_1 + 0.2006x_2 - 1.0443u \\ S^{21} : & \text{If } (x_1 \text{ is } M_1^2) \text{ and } (x_2 \text{ is } M_2^1), \text{ then } \dot{x}_2 = -0.0251 + 5.5416x_1 - 0.0085x_2 - 1.5332u \\ S^{22} : & \text{If } (x_1 \text{ is } M_1^2) \text{ and } (x_2 \text{ is } M_2^2), \text{ then } \dot{x}_2 = 0.0225 + 6.1796x_1 - 0.0000x_2 - 1.5332u \\ S^{23} : & \text{If } (x_1 \text{ is } M_1^2) \text{ and } (x_2 \text{ is } M_2^3), \text{ then } \dot{x}_2 = -0.0251 + 5.5416x_1 + 0.0085x_2 - 1.5332u \\ S^{31} : & \text{If } (x_1 \text{ is } M_1^3) \text{ and } (x_2 \text{ is } M_2^1), \text{ then } \dot{x}_2 = 8.1236 + 3.8634x_1 + 0.2363x_2 - 1.0903u \\ S^{32} : & \text{If } (x_1 \text{ is } M_1^3) \text{ and } (x_2 \text{ is } M_2^2), \text{ then } \dot{x}_2 = 8.0313 + 3.6645x_1 - 0.0000x_2 - 1.0903u \\ S^{33} : & \text{If } (x_1 \text{ is } M_1^3) \text{ and } (x_2 \text{ is } M_2^3), \text{ then } \dot{x}_2 = 8.1236 + 3.8634x_1 - 0.2363x_2 - 1.0903u \end{aligned}$$

The resultant mean square error from this approximation is 0.0013. In this case, the condition number of the extended matrix X_a becomes $1.4569e + 004$, thus improving the reliability of results.

By using the identification with the classical minimum square method in an interval close to the equilibrium point,

$$x_1 \in \left[-\frac{\pi}{4}, \frac{\pi}{4}\right] \quad x_2 \in [-2.5, 2.5]$$

Figure 3. Membership functions for the angle x of the inverted pendulum.

Figure 4. Membership functions for the angular velocity of the inverted pendulum.

The resultant linear model of the system in this interval is:

$$\dot{x}_2' = 0.0092 + 15.2665x_1 - 0.0000x_2 - 1.4187u$$

This approximation gives us an idea of the quantitative relation among various variables. Comparing these parameters with those obtained in the previous fuzzy model, it is evident that its values are, in general, far enough from its physical ones. Moreover, if we compare the parameters of various adjacent subsystems, it can clearly verify that they are quite different.

By applying the tuning of parameters method, taking as a reference, the parameters obtained through minimum square method with a weighting factor $\gamma = 0.001 * np$, where np is the number of samples used in the identification, we obtain:

$$\begin{aligned} S^{11} : & \text{If } (x_1 \text{ is } M_1^1) \text{ and } (x_2 \text{ is } M_2^1), \text{ then } \dot{x}_2 = 0.1642 + 15.0162x_1 - 0.3272x_2 - 1.2458u \\ S^{12} : & \text{If } (x_1 \text{ is } M_1^1) \text{ and } (x_2 \text{ is } M_2^2), \text{ then } \dot{x}_2 = 0.4847 + 14.6365x_1 + 0.0000x_2 - 1.1546u \\ S^{13} : & \text{If } (x_1 \text{ is } M_1^1) \text{ and } (x_2 \text{ is } M_2^3), \text{ then } \dot{x}_2 = 0.1642 + 15.0162x_1 + 0.3272x_2 - 1.2458u \\ S^{21} : & \text{If } (x_1 \text{ is } M_1^2) \text{ and } (x_2 \text{ is } M_2^1), \text{ then } \dot{x}_2 = -0.0064 + 15.4291x_1 + 0.0171x_2 - 1.4291u \\ S^{22} : & \text{If } (x_1 \text{ is } M_1^2) \text{ and } (x_2 \text{ is } M_2^2), \text{ then } \dot{x}_2 = -0.0796 + 15.5791x_1 - 0.0000x_2 - 1.4536u \\ S^{23} : & \text{If } (x_1 \text{ is } M_1^2) \text{ and } (x_2 \text{ is } M_2^3), \text{ then } \dot{x}_2 = -0.0064 + 15.4291x_1 - 0.0171x_2 - 1.4291u \\ S^{31} : & \text{If } (x_1 \text{ is } M_1^3) \text{ and } (x_2 \text{ is } M_2^1), \text{ then } \dot{x}_2 = -0.0927 + 15.1524x_1 + 0.2813x_2 - 1.3232u \\ S^{32} : & \text{If } (x_1 \text{ is } M_1^3) \text{ and } (x_2 \text{ is } M_2^2), \text{ then } \dot{x}_2 = -0.2593 + 14.9994x_1 + 0.0000x_2 - 1.2568u \\ S^{33} : & \text{If } (x_1 \text{ is } M_1^3) \text{ and } (x_2 \text{ is } M_2^3), \text{ then } \dot{x}_2 = -0.0927 + 15.1524x_1 - 0.2813x_2 - 1.3232u \end{aligned}$$

which has a mean square error of 0.0332 greater than the previous one but with more physical relationship between variables. Nevertheless, as it can be seen that in this model $a_0^{22} = -0.0796$, which does not fulfil an important characteristic of the existence of an unstable equilibrium point which is also the objective of control $x_1 = x_2 = u = 0$. In order that this point becomes an equilibrium in the fuzzy model, a_0^{22} should be zero.

To achieve this target, we can either fix the value of this parameter or assign it a very large weighting factor as explained in Section (4.2). In this case, a weighting factor of 100γ has been assigned to this parameter and finally the fuzzy model becomes:

$$\begin{aligned} S^{11} : & \text{If } (x_1 \text{ is } M_1^1) \text{ and } (x_2 \text{ is } M_2^1), \text{ then } \dot{x}_2 = 0.1642 + 15.0164x_1 - 0.3271x_2 - 1.2458u \\ S^{12} : & \text{If } (x_1 \text{ is } M_1^1) \text{ and } (x_2 \text{ is } M_2^2), \text{ then } \dot{x}_2 = 0.4848 + 14.6366x_1 + 0.0002x_2 - 1.1546u \\ S^{13} : & \text{If } (x_1 \text{ is } M_1^1) \text{ and } (x_2 \text{ is } M_2^3), \text{ then } \dot{x}_2 = 0.1642 + 15.0162x_1 + 0.3272x_2 - 1.2458u \\ S^{21} : & \text{If } (x_1 \text{ is } M_1^2) \text{ and } (x_2 \text{ is } M_2^1), \text{ then } \dot{x}_2 = -0.0073 + 15.4272x_1 + 0.0172x_2 - 1.4291u \\ S^{22} : & \text{If } (x_1 \text{ is } M_1^2) \text{ and } (x_2 \text{ is } M_2^2), \text{ then } \dot{x}_2 = 15.5778x_1 - 0.0003x_2 - 1.4536u \\ S^{23} : & \text{If } (x_1 \text{ is } M_1^2) \text{ and } (x_2 \text{ is } M_2^3), \text{ then } \dot{x}_2 = -0.0072 + 15.4287x_1 - 0.0170x_2 - 1.4291u \\ S^{31} : & \text{If } (x_1 \text{ is } M_1^3) \text{ and } (x_2 \text{ is } M_2^1), \text{ then } \dot{x}_2 = -0.0001 + 15.1478x_1 + 0.3000x_2 - 1.3232u \\ S^{32} : & \text{If } (x_1 \text{ is } M_1^3) \text{ and } (x_2 \text{ is } M_2^2), \text{ then } \dot{x}_2 = -0.2646 + 14.9965x_1 + 0.0080x_2 - 1.2568u \\ S^{33} : & \text{If } (x_1 \text{ is } M_1^3) \text{ and } (x_2 \text{ is } M_2^3), \text{ then } \dot{x}_2 = -0.0942 + 15.1516x_1 - 0.2821x_2 - 1.3232u \end{aligned}$$

The minimum square error is 0.0333 which is slightly greater than that of the previous case but it represents the equilibrium point of the physical system, that is, $a_0^{22} = 0$. It can be clearly noticed in this model that the distance between adjacent subsystem is relatively small.

For each subsystem defined in the previous rules,

$$\dot{x}_2' = a_0^{(i_1 i_2)} + a_1^{(i_1 i_2)} x_1 + a_2^{(i_1 i_2)} x_2 + b^{(i_1 i_2)} u$$

A feedback state LQR controller has been designed with the affine term:

$$u = k_0^{(i_1 i_2)} + k_1^{(i_1 i_2)} x_1 + k_2^{(i_1 i_2)} x_2 + b^{(i_1 i_2)} u$$

Firstly, the affine term of the controller is used to eliminate the affine term of the system

$$a_0^{(i_1 i_2)} + b^{(i_1 i_2)} k_0^{(i_1 i_2)} = 0 \implies k_0^{(i_1 i_2)} = \frac{-a_0^{(i_1 i_2)}}{b^{(i_1 i_2)}}$$

The other terms are calculated by the LQR minimizing the following performance index:

$$J = \int_{t_0}^{\infty} (100x_1^2 + 10x_2^2 + u^2) dt$$

Thus, the resultant fuzzy optimal LQR is:

- R^{11} : If $(x_1 \text{ is } M_1^1)$ and $(x_2 \text{ is } M_2^1)$, then $u = 0.1318 + 27.7146x_1 + 7.1239x_2$
- R^{12} : If $(x_1 \text{ is } M_1^1)$ and $(x_2 \text{ is } M_2^2)$, then $u = 0.4199 + 28.8232x_1 + 7.6493x_2$
- R^{13} : If $(x_1 \text{ is } M_1^1)$ and $(x_2 \text{ is } M_2^3)$, then $u = 0.1318 + 27.7153x_1 + 7.6493x_2$
- R^{21} : If $(x_1 \text{ is } M_1^2)$ and $(x_2 \text{ is } M_2^1)$, then $u = -0.0051 + 25.5103x_1 + 6.7723x_2$
- R^{22} : If $(x_1 \text{ is } M_1^2)$ and $(x_2 \text{ is } M_2^2)$, then $u = 25.3745x_1 + 6.7019x_2$
- R^{23} : If $(x_1 \text{ is } M_1^2)$ and $(x_2 \text{ is } M_2^3)$, then $u = -0.0050 + 25.5120x_1 + 6.7486x_2$
- R^{31} : If $(x_1 \text{ is } M_1^3)$ and $(x_2 \text{ is } M_2^1)$, then $u = 0.0001 + 26.6491x_1 + 7.3211x_2$
- R^{32} : If $(x_1 \text{ is } M_1^3)$ and $(x_2 \text{ is } M_2^2)$, then $u = -0.2105 + 27.5014x_1 + 7.3389x_2$
- R^{33} : If $(x_1 \text{ is } M_1^3)$ and $(x_2 \text{ is } M_2^3)$, then $u = -0.0712 + 26.6529x_1 + 6.8812x_2$

It can also be noticed that the distance between the parameters of various adjacent control rules is relatively small. This in turn, strengthens the hypothesis of designing a local control algorithm for each subsystem.

Figure 5 shows the controlled system behavior from an initial condition $[\frac{\pi}{4} \ 1]^t$. To examine the robustness of the proposed controller, it has been subjected to successive disturbance effects of 5° , -5° , and 10° with respect to its equilibrium state as shown in Figure 6. It is clearly shown that the inverted pendulum returns rapidly and smoothly to its equilibrium condition after applying each disturbance. Moreover, we have proven that the LQR is robust and invariant against measurement noise. If we suppose that the angle sensor introduces a noise of $\sigma = 1^\circ$, it can be observed that the output is only affected by $\sigma = 0.04^\circ$ as shown in Figure 7. Figure 8 shows several trajectories in state space form of the system for several initial conditions.

Figure 5. Transient response of the inverted pendulum controlled by the fuzzy linear quadratic regulator.

Figure 6. Transient response of the inverted pendulum controlled by the fuzzy linear quadratic regulator subjected to successive disturbances.

Figure 7. Transient response of the inverted pendulum controlled by the fuzzy linear quadratic regulator subjected to measurement noise.

Figure 8. Trajectories in state space form of the system for several initial conditions.

8. CONCLUSIONS

New efficient approaches have been presented to improve the local and global approximation of the T-S fuzzy model. The main problem of the T-S identification method is that it cannot be applied when the membership functions are overlapped by pairs. This restricts the use of the T-S method because this type of membership function has been widely used during the last 2 decades in the stability, controller design, and are popular in industrial control applications. The approaches developed here can be considered as a generalized version of the T-S method with optimized performance in approximating nonlinear functions. We propose a noniterative method through weighting of parameters approach and an iterative algorithm by applying the extended Kalman filter based on the same idea of parameters' weighting. We show that the Kalman filter is an effective tool in the identification of the T-S fuzzy model. An FC based LQR has been proposed in order to show the effectiveness of the estimation method developed here in control applications. An illustrative example of an inverted pendulum has been chosen to evaluate the robustness and remarkable performance of the proposed method and the high accuracy obtained in approximating nonlinear and unstable systems locally and globally in comparison with the original T-S model. Simulation results have shown the potential, simplicity, and generality of the algorithm.

REFERENCES

1. Takagi T, Sugeno M. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics* 1985; **15**(1):116–132.
2. Gang F. A survey on analysis and design of model-based fuzzy control systems. *IEEE Transactions on Fuzzy Systems* 2006; **14**(5):676–697.
3. Guerra TM, Vermeiren L. LMI-based relaxed nonquadratic stabilization conditions for nonlinear systems in the Takagi–Sugeno's form. *Automatica* 2004; **40**:823–829.
4. Lian K-Y, Su C-H, Huang C-S. Performance Enhancement for T-S Fuzzy Control Using Neural Networks. *IEEE Transactions on Fuzzy Systems* 2006; **14**(5):619–627.
5. Hseng T, Li S, Tsai S-H. Fuzzy bilinear model and fuzzy controller design for a class of nonlinear systems. *IEEE Transactions on Fuzzy Systems* 2007; **15**(3):494–506.
6. Chen B, Liu X, Tong S. Adaptive fuzzy output tracking control of MIMO nonlinear uncertain systems. *IEEE Transactions on Fuzzy Systems* 2007; **15**(2):287–300.
7. Jae-Hun K, Hyun C-H, Kim E, Park M. Adaptive synchronization of uncertain chaotic systems based on T-S fuzzy model. *IEEE Transactions on Fuzzy Systems* 2007; **15**(3):359–369.
8. Zeng K, Zhang NY, Xu WL. A comparative study on sufficient conditions for Takagi–Sugeno fuzzy systems as universal approximators. *IEEE Transactions Fuzzy Systems* 2000; **8**(6):773–780.
9. Johansen TA, Shorten R, Murray-Smith R. On the interpretation and identification of dynamic Takagi–Sugeno models. *IEEE Transactions on Fuzzy Systems* 2000; **8**(3):297–313.
10. Tanaka K, Wang HO. *Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach*. Wiley: New York, 2001.
11. Kóczy LT, Hirota K. Size reduction by interpolation in fuzzy rule bases. *IEEE Transactions on Systems, Man, and Cybernetics, Part B* 1997; **27**(1):14–25.
12. Tikk D, Baranyi P, Patton RJ. Polytopic and TS models are nowhere dense in the approximation model space. *Proceedings of the International Conference on Systems, Man and Cybernetics (SMC'02)*, Hammamet, Tunisia, 2002.
13. Baranyi P, Korondi P, Hashimoto H. Global asymptotic stabilisation of the prototypical aeroelastic wing section via TP model transformation. *Asian Journal of Control* 2005; **7**(2):99–111.
14. Tikk D, Baranyi P, Patton RJ. Approximation properties of TP forms and its consequences to TPDC design framework. *Asian Journal of Control* 2007; **9**(3):221–231.
15. Baranyi P, Szeidl L, Várlaki P, Yam Y. Definition of the HOSVD based canonical form of polytopic dynamic models. *Proceedings of 3rd International Conference on Mechatronics (ICM 2006)*, Budapest, Hungary, 3–5 July 2006; 660–665.
16. Szeidl L, Várlaki P. HOSVD based canonical form for polytopic models of dynamic systems. *Journal of Advanced Computational Intelligence and Intelligent Informatics* 2009; **13**(1):52–60.
17. Baranyi P. SVD-based reduction to MISO TS models. *IEEE Transactions on Industrial Electronics* 2003; **51**(1):232–242.
18. Matía F, Jiménez A, Al-Hadithi BM. An affine model with decoupled fuzzy dynamics, 22–27 June 2008; 713–720.
19. Cao SG, Rees NW, Feng G. Analysis and design for a class of complex control systems- Part I: fuzzy modeling and identification. *Automatica* 1997; **33**:1017–1028.

20. Molloy S, Babuska R, Abonyi J, Verbruggen HB. Effective optimization for fuzzy model predictive control. *IEEE Transactions on Fuzzy Systems* 2004; **12**(5):661–675.
21. Skrjanc I, Blazic S, Agamennoni O. Interval fuzzy model identification using l-Norm. *IEEE Transactions on Fuzzy Systems* 2005; **13**(5):561–568.
22. Klawonn F, Kruse R. Constructing a fuzzy controller from data. *Fuzzy Sets and Systems* 1997; **85**(2):177–193.
23. Hong TP, Lee CY. Induction of fuzzy rules and membership functions from training examples. *Fuzzy Sets and Systems* 1996; **84**(1):33–47.
24. Kim J, Suga Y, Won S. A new approach to fuzzy modeling of nonlinear dynamic systems with noise: relevance vector learning mechanism. *IEEE Transactions on Fuzzy Systems* 2006; **14**(2):222–231.
25. Baranyi P, Korondi P, Patton RJ, Hashimoto H. Trade-off between approximation accuracy and complexity for T-S fuzzy models. *Asian Journal of Control* 2004; **6**(1):21–33.
26. Tikk D, Baranyi P. Comprehensive analysis of a new fuzzy rule interpolation method. *IEEE Transactions on Fuzzy Systems* 2000; **8**(3):281–296.
27. Baranyi P, Kóczy LT, Gedeon TD. A generalized concept for fuzzy rule interpolation. *IEEE Transactions on Fuzzy Systems* 2004; **12**(6):820–837.
28. Yam Y, Wong ML, Baranyi P. Interpolation with function space representation of membership functions. *IEEE Transactions on Fuzzy Systems* 2006; **14**(3):398–411.
29. Yam Y, Kóczy LT. Representing membership functions as points in high dimensional spaces for fuzzy interpolation and extrapolation. *IEEE Transactions on Fuzzy Systems* 2000; **8**(6):761–772.
30. Kumar M, Stoll R, Stoll N. A min-max approach to fuzzy clustering, estimation, and identification. *IEEE Transactions on Fuzzy Systems* 2006; **14**(2):248–262.
31. Tao C, Taur J. Design of fuzzy controllers with adaptive rule insertion. *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics* 1999; **29**:389–397.
32. Simon D. Design and rule base reduction of a fuzzy filter for the estimation of motor currents. *International Journal of Approximate Reasoning* 2000; **25**:145–167.
33. Wang L, Mendel J. Back-propagation of fuzzy systems as nonlinear dynamic system identifiers. *IEEE International Conference on Fuzzy Systems*, San Diego, California, 1992; 1409–1418.
34. Simon D. Training fuzzy systems with the extended Kalman filter. *Fuzzy Sets and Systems* 2002; **132**:189–199.
35. Simon D. Sum normal optimization of fuzzy membership functions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 2002; **10**(4):363–384.
36. Wang L, Yen J. Extracting fuzzy rules for system modelling using a hybrid of genetic algorithms and Kalman filter. *Fuzzy Sets and Systems* 1998; **101**:353–362.
37. Hsiao C. Modified gain fuzzy Kalman filtering algorithm. *JSME International Journal. Series C* 1999; **42**:363–368.
38. Kobayashi K, Cheok K, Watanabe K. Estimation of the absolute vehicle speed using fuzzy logic rule-based Kalman filter. *Proceedings of the American Control Conference*, Seattle, WA, 21–23 Jun 1995; 3086–3090.
39. Jiménez A, Al-Hadithi BM, Matía F. An optimal T-S Model for the estimation and identification of nonlinear functions. *WSEAS Transactions on Systems and Control* 2008; **3**(10):897–906.
40. Kalman RE. A new approach to linear filtering and prediction problems. *Transactions of the ASME. Series D, Journal of Basic Engineering* 1960; **82**:35–45.
41. Gelb A. *Applied Optimal Estimation*. MIT Press: Cambridge, MA, 1974.
42. Lee TC, Yang DR, Lee KS, Yoon TW. Indirect adaptive backstepping control of a pH neutralization process based on recursive prediction error method for combined state and parameter estimation. *Industrial and Engineering Chemistry Research* 2001; **40**:4102–4110.
43. Puskorius G, Feldkamp L. Neurocontrol of nonlinear dynamical systems with Kalman filter trained recurrent networks. *IEEE Transactions Neural Networks* 1994; **5**:279–297.
44. Jiang T, Li Y. Generalized defuzzification strategies and their parameter learning procedures. *IEEE Transactions on Fuzzy Systems* 1996; **4**:64–71.