

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.

For more information visit www.intechopen.com



Parametric Type-2 Fuzzy Logic Systems

Arturo Tellez, Heron Molina, Luis Villa,
Elsa Rubio and Ildar Batyrshin
*IPN, CIC, Mexico City,
Mexico*

1. Introduction

The use of Fuzzy Logic Systems (FLS) for control applications has increased since they became popular from 80's. After Mendel in 90's showed how uncertainty can be computed in order to achieve more robust systems, Type-2 Fuzzy Logic Systems (T2FLS) are in the focus of researchers and recently they became a new research topic.

At same time, Batyrshin et al demonstrated that parametric conjunctions can be useful for tuning a FLS in order to achieve better performance beyond the set parameter tuning. In signal processing and system identification, this fact let the designer to add freedom degrees to adjust a general FLS.

This chapter presents the parametric T2FLS and shows that this new FLS is a very useful option for sharper approximations in control. In order to verify the advantages of the parametric T2FLS, it is used the Ball and Plate System as a testbench. This study case helps us to understand how a parametric conjunction affects the controller behavior in measures like response time or overshoot. Also, this application let us observe how the controller works in noise presence.

2. Parametric T2FLS

A Parametric Type-2 Fuzzy Logic Systems (PT2FLS) is a general FLS which can be fully adjusted through a single or multiple parameters in order to achieve a benefit in its general performance. It means that a PT2FLS has several options to adjust set parameters (i.e. membership function parameters), rule parameters and output set parameters. Fig. 1 shows the structure of a PT2FLS which it is almost equal to a general T2FLS.

In this figure the Defuzzification stage comprises the Output Processing Block and the Defuzzifier as Mendel stated in (Karnik, Mendel et al. 1999). For Interval Type-2 Fuzzy Logic Systems (IT2FLS) this block represents only the centroid calculation for example considering the WM Algorithm (Wu and Mendel 2002). As it can be seen, a dashed arrow crosses every stage; this means that every stage is tunable for optimization purposes.

A general Fuzzy System is a function where all input variables are mapped to the output variables according to the knowledge base defined by rules. Rule Set represents the configuration of the T2FLS.

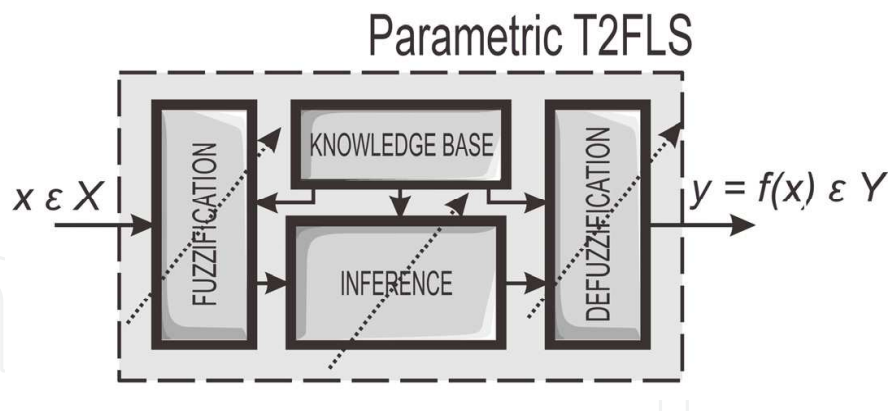


Fig. 1. Parametric Type-2 Fuzzy Logic Systems

Every input variable (where X is the input vector) for a T2FLS has associated a single or multiple Fuzzy Sets (FS), in this case a Type-2 Fuzzy Set (T2FS). Those T2FS express the uncertainty associated with ideas or linguistic expressions of the people. A T2FS is characterized by a complex Membership Function (MF) (defined by their outer MFs, which has several parameters that define it) called Footprint of Uncertainty (FOU). In FLS, those equivalent parameters help the expert to improve the entire system performance when performing adaptation. In case of T2FS, additional parameters are needed.

In the other hand, every output variable (where Y is the output vector) has associated also a FOU. This FOU has its own parameters which can be also tuned. Adaptation for output T2FS or FS when defuzzifying implies in adjusting their output centroids. This stage is not very used for adaptation, but it can be realized.

Adaptation in inference stage is not used, because of the complexity of the parametric operation. Adding more complexity in a system which is by its own very complex is not suitable. For this reason, it is introduced an operation which is simpler, the *Parametric Conjunction* (Batyrrshin and Kaynak 1999; Batyrrshin, Rudas et al. 2009; Prometeo Cortes, Ildar Z. Batyrrshin et al. 2010).

2.1 Overview

In a T2FLS the inference step combines every rule and maps input sets to output sets (premises to consequents). Each premise that is related with another premise (implied sets) is related by a rule using a Conjunction Operation. This conjunction operation normally is performed with a t-norm operation. Suppose a rule l in the rule set with M rules of a given MISO T2FLS with m inputs ($x_1 \in X_1, x_2 \in X_2, \dots, x_p \in X_m$) and $n = 1$ output ($y_1 \in Y_1$), so that $R^l: IF x_1 \text{ is } \tilde{A}_1^j \wedge x_2 \text{ is } \tilde{A}_2^j \dots x_p \text{ is } \tilde{A}_m^j \rightarrow y \text{ is } \tilde{B}^j$, where \tilde{A}_i^j denotes a specific set that belongs to a specific input variable. Symbol " \wedge " represents a conjunction operation performed with a basic t-norm, typically a minimum, which can be replaced for a parametric operation.

This rule represents the relation between the input space of every variable $X_1 \times X_2 \times \dots \times X_m$ (where $i = 1, 2, \dots, m$) and the output space Y and the relation of those variables are expressed

as $\mu_{\tilde{A}_1 \times \tilde{A}_2 \times \dots \times \tilde{A}_m \rightarrow \tilde{B}}(x, y)$. Suppose any input variable with a T2FS defined as $\tilde{F}_i^j(x_i; \mathbb{P}_s^j) = FOU(x_i; \mathbb{P}_s^j)$, where $1 \leq j \leq m$, $1 \leq j \leq k$ and function $FOU(\cdot)$ is the characterization of the T2FS defined between its Upper Membership Function (UMF) and its Lower Membership Function (LMF), i.e. the FOU; \mathbb{P}_s represents the set of parameter of the T2FS which defines its basic FOU shape (triangular, trapezoidal, Gaussian, etc.). Such parameters let the expert or an extern intelligent system to modify its behavior.

A parametric fuzzy conjunction operation represents the variable intersection of two premises related by a parameter, i.e. that two premises are implied in a measured way in order to take a specific decision. Those premises are T2FS (\tilde{A}_i^j).

For a specific rule R^l , a firing strength F^l for the implication of two or more premises is expressed as

$$F^l = [\bar{F}^l; \underline{F}^l] = [\bar{\mu}_{\tilde{A}_i^j}(x_i); \underline{\mu}_{\tilde{A}_i^j}(x_i)]$$

$$\bar{F}^l = \bar{\mu}_{\tilde{A}_1^j}(x_1) \wedge \bar{\mu}_{\tilde{A}_2^j}(x_2) \wedge \dots \wedge \bar{\mu}_{\tilde{A}_m^j}(x_m)$$

$$\underline{F}^l = \underline{\mu}_{\tilde{A}_1^j}(x_1) \wedge \underline{\mu}_{\tilde{A}_2^j}(x_2) \wedge \dots \wedge \underline{\mu}_{\tilde{A}_m^j}(x_m)$$

Until here, upper and lower firing strengths are defined using non-parametric conjunctions for operator " \wedge ". Once it is considered a parametric conjunction operation for performing implication of the premises, every firing strength can be controlled by a parameter, arising to a parametric inference process, as it is used for FLS in (Batyrshin, Rudas et al. 2009). So a parametric firing strength is expressed as

$$\bar{F}^l = \mathbf{T}(\bar{\mu}_{\tilde{A}_1^j}(x_1), \bar{\mu}_{\tilde{A}_2^j}(x_2), \dots, \bar{\mu}_{\tilde{A}_m^j}(x_m); \mathbb{P}_r) = \mathbf{T}(\bar{\mu}_{\tilde{A}_i^j}(x_i); \mathbb{P}_r^l) \tag{1}$$

$$\underline{F}^l = \mathbf{T}(\underline{\mu}_{\tilde{A}_1^j}(x_1), \underline{\mu}_{\tilde{A}_2^j}(x_2), \dots, \underline{\mu}_{\tilde{A}_m^j}(x_m); \mathbb{P}_r) = \mathbf{T}(\underline{\mu}_{\tilde{A}_i^j}(x_i); \mathbb{P}_r^l) \tag{2}$$

where \mathbf{T} is a parametric conjunction and \mathbb{P}_r^l is the set of parameters used to manipulate the implication of the premises related in l th rule.

Finally, every firing strength must be aggregated by a disjunction operator or t-conorn operator in order to complete the composition.

$$\bar{\mu}_{\tilde{B}_1^j}(y) = \sqcup_{x \in X} \left(\mathbf{T}(\bar{\mu}_{\tilde{A}_i^j}(x_i); \mathbb{P}_r) \right) \tag{3}$$

$$\underline{\mu}_{\tilde{B}_1^j}(y) = \sqcup_{x \in X} \left(\mathbf{T}(\underline{\mu}_{\tilde{A}_i^j}(x_i); \mathbb{P}_r) \right) \tag{4}$$

For the defuzzification stage in a T2FLS, the corresponding centroids of an output T2FS can be parametric also. However, if an expert tries to calculate them all using the KM algorithm, it can be a complex task. Instead of calculation of output centroids it is suggested the use heuristically techniques. Next section, explains two suitable parametric conjunctions used in FLS and T2FLS.

2.2 Parametric conjunctions

A fuzzy conjunction is the operation between two values of membership degrees that relate two fuzzy sets considered as premises in an inference scheme. Those premises let the system to decide for a specific decision in a given moment. This process is called Implication. Every rule describes an implication as shown in Fig. 2.

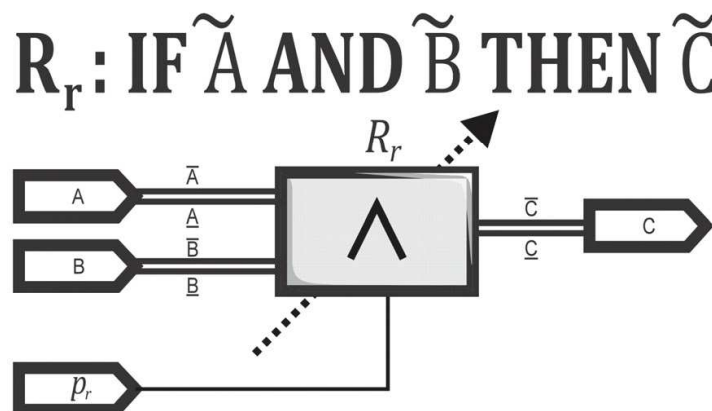


Fig. 2. Implication of premises of a single rule using (p)-Monotone Sum of Conjunctions

Most popular conjunction and disjunction operations are t-norm T and t-conorm S (also called s-norm), respectively. They are defined as functions $T, S: [0,1] \times [0,1] \rightarrow [0,1]$ satisfying the following axioms of commutativity, associativity, monotonicity and boundary conditions (E. P. Klement, R. Mesiar et al. 2000):

$$\begin{array}{lll}
 T(a, b) = T(b, a) & S(a, b) = S(b, a) & \text{commutativity} \\
 T(T(a, b), c) = T(a, T(b, c)) & S(S(a, b), c) = S(a, S(b, c)) & \text{associativity} \\
 T(a, b) \leq T(c, d) & S(a, b) \leq S(c, d) & a \leq c, b \leq d \quad \text{monotonicity} \\
 T(a, 1) = a & S(a, 0) = a & \text{boundary conditions}
 \end{array}$$

A parametric fuzzy conjunction uses some parameters to control the way the inference will be done. In order to simplify the complexity of a traditional parametric conjunction, it was proposed in (Batyrrshin and Kaynak 1999) to use non-associative conjunction operations, due to the lack of use of this property and the usage of only two operands in applied fuzzy systems. For this reason in definition of conjunctions T further we use only axioms of commutativity, monotonicity and boundary conditions.

A single parametric conjunction may behave in different ways depending of a parameter value. There are some works today about the parametric conjunctions (Batyrrshin and Kaynak 1999; Batyrrshin, Rudas et al. 2009; Rudas, Batyrrshin et al. 2009; Prometeo Cortes, Ildar Z. Batyrrshin et al. 2010). Next subsections describe briefly some parametric conjunctions suitable for software and hardware implementations, which share the usage of basic t-norms and other simple functions.

2.3 Monotone sum of conjunctions

This parametric conjunction is characterized depending of a parameter value and also fulfill with properties of monotonicity, boundary conditions and commutativity (Batyrrshin, Rudas et al. 2009). Other suitable parametric conjunctions can be found in (Batyrrshin and Kaynak 1999).

Suppose $G = \{1,2, \dots, n\}; n \geq 2$ be an index set and H is a *partition* of $[0,1]$ on pairwise disjoint intervals $\{H_1, H_2, \dots, H_n\}$ such that if $i < j$ then $a < b$ for all $a \in H_i$ and $b \in H_j$. Denote a *section* as $D_{ij} = H_i \times H_j$ and suppose $G = \{1,2, \dots, n\}; n \geq 2$ be an index set and H is a partition of $[0,1]$ on pairwise disjoint intervals $\{H_1, H_2, \dots, H_n\}$ such that if $i < j$ then $a < b$ for all $a \in H_i$ and $b \in H_j$. Suppose Q is some index set and $(T_q, \leq)_{q \in Q}$ is a partially ordered set of fuzzy conjunctions, e.g. a set of all basic t-norms. Then assign to each section $D_{ij} = H_i \times H_j$ in $[0,1] \times [0,1]$ some $T_{ij} = T_q$ from this set such that $T_{ij}(a, b) \leq T_{st}(u, v)$ if $i \leq s, j \leq t$ and $a \leq u, b \leq v$ where $(a, b) \in D_{ij}$ and $(u, v) \in D_{st}$. Define a function T on $[0,1] \times [0,1]$ by

$$T(a, b) = T_{ij}(a, b) \text{ if } (a, b) \in D_{ij}; i, j \in G \tag{5}$$

Then T is a conjunction called a monotone sum (Batyrrshin, Rudas et al. 2009) of $(D_{ij}, T_{ij})_{i,j \in G}$ or monotone sum of fuzzy conjunctions $T_{ij}; i, j \in G$. If it is desirable to construct commutative conjunctions then it should be considered:

$$T_{i,j} = T_{j,i} \tag{6}$$

Next subsections describe two types of monotone sums using a single parameter.

2.3.1 (p) – Monotone sum

Suppose a partition on two intervals is defined by some parameter $0 \leq p \leq 1$ as $H_1 = [0, p]$ and $H_2 = (p, 1]$. Assign to each $D_{ij}, i = [1,2]$, fuzzy conjunctions T_{11}, T_{21}, T_{12} and T_{22} ordered as follows: $T_{11} \leq T_{12} \leq T_{22}, T_{11} \leq T_{21} \leq T_{22}$. Then define the (p) –monotone sum of fuzzy conjunctions from (5) as follows:

$$T(a, b, p) = \begin{cases} T_{11}(a, b), & (a \leq p) \wedge (b \leq p) \\ T_{21}(a, b), & (a > p) \wedge (b \leq p) \\ T_{12}(a, b), & (a \leq p) \wedge (b > p) \\ T_{22}(a, b), & (a > p) \wedge (b > p) \end{cases} \tag{7}$$

As it can be seen all four sections are defined by parameter p , then a monotone sum of conjunctions is able to behave in different ways depending of this parameter. For example, if $p = 0$ then its behavior will be T_{22} as stated in (6).

2.3.2 (p, 1 – p) – Monotone sum

Suppose three partitions defined by some parameter p as $H_1 = [1, p], H_2 = (p, 1 - p]$ and $H_3 = (1 - p, 1]$. Assign to each section $D_{i,j}, i = [1,2,3]$ fuzzy conjunctions $T_{11}, T_{12}, T_{13}, T_{21}, T_{22}, T_{23}, T_{31}, T_{32}$ and T_{33} ordered as follows: $T_{11} \leq T_{12} \leq T_{13} \leq T_{23} \leq T_{33}, T_{11} \leq T_{21} \leq T_{31} \leq T_{32} \leq T_{33}, T_{12} \leq T_{22} \leq T_{32}, T_{21} \leq T_{22} \leq T_{23}$. Then define the $(p, 1 - p)$ – monotone sum of fuzzy conjunctions from (5) as follows:

$$T(a, b, p) = \begin{cases} T_{11}(a, b), & (a \leq p) \wedge (b \leq p) \\ T_{21}(a, b), & (a > p) \wedge (a \leq 1 - p) \wedge (b \leq p) \\ T_{31}(a, b), & (a > 1 - p) \wedge (b \leq p) \\ T_{12}(a, b), & (a \leq p) \wedge (b > p) \wedge (b \leq 1 - p) \\ T_{22}(a, b), & (a > p) \wedge (a \leq 1 - p) \wedge (b > p) \wedge (b \leq 1 - p) \\ T_{32}(a, b), & (a > 1 - p) \wedge (b > p) \wedge (b \leq 1 - p) \\ T_{13}(a, b), & (a \leq p) \wedge (b > p) \\ T_{23}(a, b), & (a > p) \wedge (a \leq 1 - p) \wedge (b > p) \\ T_{33}(a, b), & (a > p) \wedge (b > p) \end{cases} \quad (8)$$

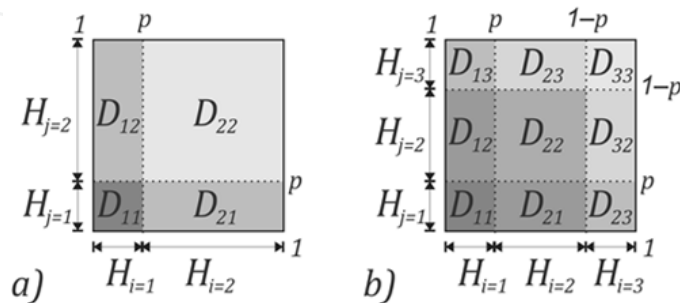


Fig. 3. Monotone sum of conjunctions a) (p) and b) (p,1-p)

Fig. 3 shows both monotone sums described here and its construction is very similar between them. Next section describes a case of study for PT2FLS application: the Ball and Plate System.

fig. 4. Mechanical Model of B&P System

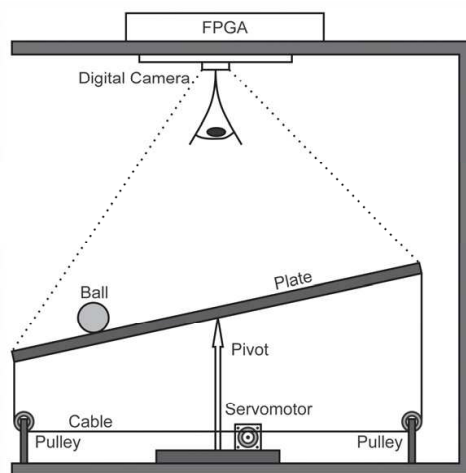


Fig. 4. The tilt of plate let the ball to move from one point to another over its surface. The position of ball is captured from a digital camera that is mounted over the plate on a specific and convenient distance in order to scan the plate surface completely.

3. A case of study: The ball and plate system

As reported in (Moreno-Armendariz, Rubio-Espino et al. 2010), it was built a prototype of B&P mechanism that can be used as a testbench for control implementations. This model

consists of a plate mounted over a pivot that let the plate to tilt along any of its axes using two servomotors. Fig. 4 shows this description only for a single axis.

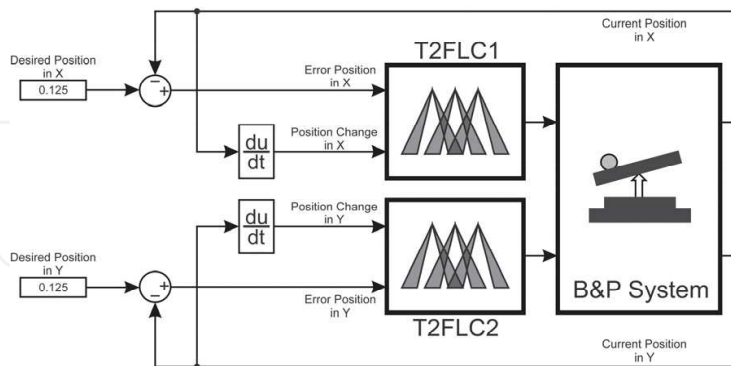


Fig. 5. PT2FLC for B&P System

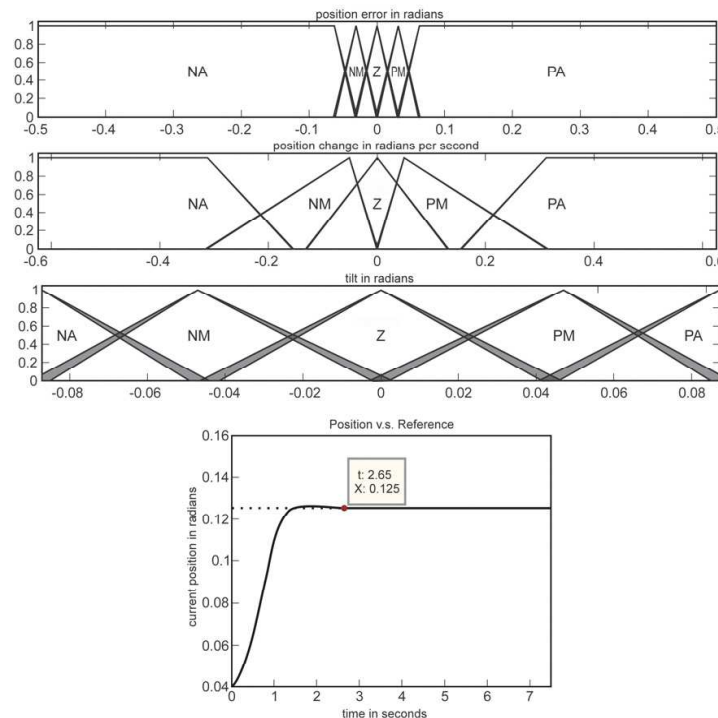


Fig. 6. Initial set distribution for input and output variables for every PT2FLC

The computer vision is implemented in a Field Programmable Gate Array (FPGA) using a development kit, manufactured by Terasic (DE2 Development Kit). This kit has several interfaces to test a digital system and let the usage of an embedded vision system in the same chip. The vision system calculates with (10) the centroid of the ball and determines its position (coordinates). It was implemented the T1FLC that controls the B&P system also, embedding it in the same chip using just the 15% of the Cyclone II EP2C35F672C6N.

In this work, B&P System is controlled using PT2FLC. This system is shown in Fig. 5 and describes a control system that establishes a desired position in axis X and a desired position

in axis Y. Servomotors perform the adequate tilt over both axes. Every tilt value is calculated by its corresponding PT2FLC using the error position and the position change. Position change is the differential of the feedback of the plant, i.e. the current position.

It is noteworthy that PT2FLC hardware has not been implemented and tested for this application. Only simulations are performed in order to show all advantages of the use of PT2FLC in control applications. Mechanical model proposed in (Moreno-Armendariz, Rubio-Espino et al. 2010) has the characteristic of designing and testing new and improved controllers, which it is a suitable future work, because of the flexibility of FPGA.

$$\begin{aligned}
 A &= \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -9.81 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ -6.1313 \times 10^4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -9.81 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -6.1313 \times 10^4 & 0 & 0 & 0 \end{bmatrix} \\
 B &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 5.6818 \times 10^4 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 5.6818 \times 10^4 \end{bmatrix} \\
 C &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \\
 D &= [0] \\
 x' &= Ax + Bu \\
 y &= Cx
 \end{aligned} \tag{9}$$

The characteristics of this B&P System (9) is a linearized state-space model, the same as described in (Moreno-Armendariz, Rubio-Espino et al. 2010). With (10), it can be calculated the current velocity, acceleration and position in axis x.

$$\begin{aligned}
 v(k) &= \frac{(x(k) - x(k-1))}{T} \\
 a(k) &= v(k) - v(k-1) \\
 x_e(k+1) &= x(k) + \frac{v(k)T}{1} + \frac{a(k)T^2}{2} \\
 e(k+1) &= x_d - x_e(k+1)
 \end{aligned} \tag{10}$$

The vision system described in (Moreno-Armendariz, Rubio-Espino et al. 2010) uses a sampling time T (50ms), which captures and processes a single image in that period. After the vision system process the image, FPGA calculates the current position of the ball in axis X, $x(k)$, where k is the current sample. Once it is known the position, it is possible to find the current velocity component $v(k)$ and the current acceleration component $a(k)$ of the

ball, over the axis X. If it is assumed that the velocity and the acceleration of the ball are constant at shorter values of T , it is possible to estimate the next position of ball with $x_e(k + 1)$ and finally the desired position x_d and the estimated error $e(k + 1)$.

Tilt		Change				
		NA	NM	Z	PM	PA
Error	NA	NM	Z	PM	PA	PA
	NM	NM	NM	PA	PA	PA
	Z	NA	NA	Z	PA	PA
	PM	NA	NA	NA	PM	PM
	PA	NA	NA	NM	Z	PM

Table 1. Optimal Rule Set of B&P System with T1FLC described in (Moreno-Armendariz, Rubio-Espino et al. 2010) used for the PT2FLC purposes.

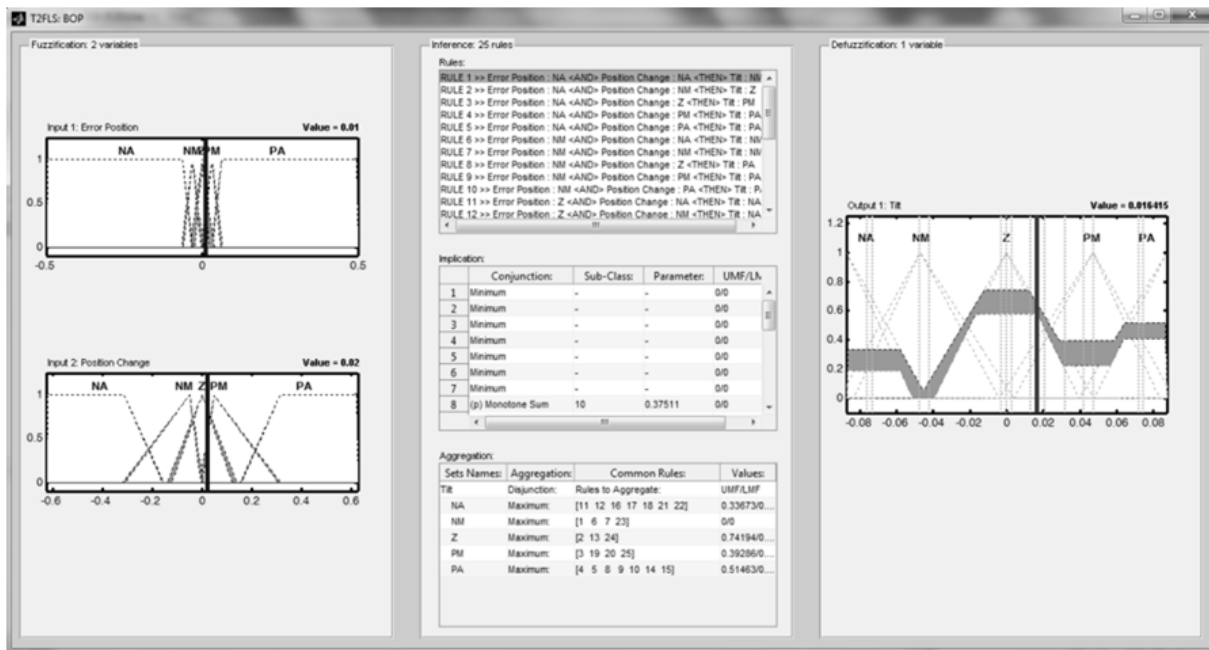


Fig. 7. IT2FLS Simulator for B&P System

It is proved that B&P System is a decoupled system over its two axes (Moreno-Armendariz, Rubio-Espino et al. 2010). So, (10) are similar for the axis Y. Fig 5 shows that B&P system block has two inputs and two outputs for our control purposes; so, every in-out pair corresponds to every axis.

T1FLC proposed by (Moreno-Armendariz, Rubio-Espino et al. 2010) has two inputs and one output. Every variable has 5 FS (Fig. 6) associated to linguistic variables “high positive” (PA), “medium positive” (PM), “zero or null” (Z), “medium negative” (NM), and “high negative” (NA). Rule set is described in Table 1.

The T1FLC controls the tilt of plate using the information that the FPGA takes form the camera and calculates the current position using (10) under perfect environment conditions. But what happens when some external forces (e.g. weather) complicate the system stability? Some equivalent phenomena may be introduced to the plate. For example, the illuminating

variation due to light incidence over the plate, an unbalanced motor tied to the plate, a low quality image sensor or some interference noise added to the processed image, may be introduced as external disturbances.

Experiment	Overshoot	SSE	Ripple
When all sets in every variable are T1, except the variable which set FOUs are increasing from zero.	Yes	No	No
When all sets in input variable error are T2 and their FOU are decreasing until they become T1. All other variable sets are wide as much as it can be possible.	No	No	Yes
When all sets in input variable change are T2 and their FOU are decreasing until they become T1. All other variable sets are wide as much as it can be possible.	Yes	No	No
When all sets in output variable tilt are T2 and their FOU are decreasing until they become T1. All other variable sets are wide as much as it can be possible.	No	Yes	No

Table 2. Phenomena associated with the FOU of every set in system

In initial experiments, noise-free optimization is performed and similar results are achieved in order to compare it with T1FLC. For noise tests it is only considered an unbalanced motor tied to the plate that makes it tremble while a sine trajectory is performed, analyzing a single axis. This experiment helps us to verify the noise-proof ability of the T2FLC.

4. Experimental results

FS distribution, i.e., FS shape parameters may arise several characteristic phenomena that expert must take into account when designing applied-to-control fuzzy systems, so-called Fuzzy Logic Controller (FLC). As described in (Moreno-Armendariz, Rubio-Espino et al. 2010), authors found an optimal FS distribution where FLC shows a great performance in 3.8 seconds. However, when it is used this same configuration some phenomena arises when it is introduced T2FS.

Starting from the initial optimal set distribution and without considering any possible noise influence, it was tested several configurations modifying every set FOU, starting from a T1FS (without FOU) and increasing it as much as possible; or starting from a very wide FOU and collapsing it until it becomes a T1FS. Some phenomena are related to them as described in Table 2, but in general, when it is introduced a T2FS a certain level of overshoot is found, no matter which variable was modified; so, if every variable has a T2FS, then the expert has to deal with the influence of nonlinear aggregation of overshoot, steady-state error or offset (SSE) and ripple, when tuning a PT2FLC, which might be a complicated task.

For every experiment it was used an implemented simulator for IT2FLS. With some instructions it can be constructed any parametric IT2FLS and expert may choose from set shape, several parametric conjunctions and defuzzification options (Fig. 7).

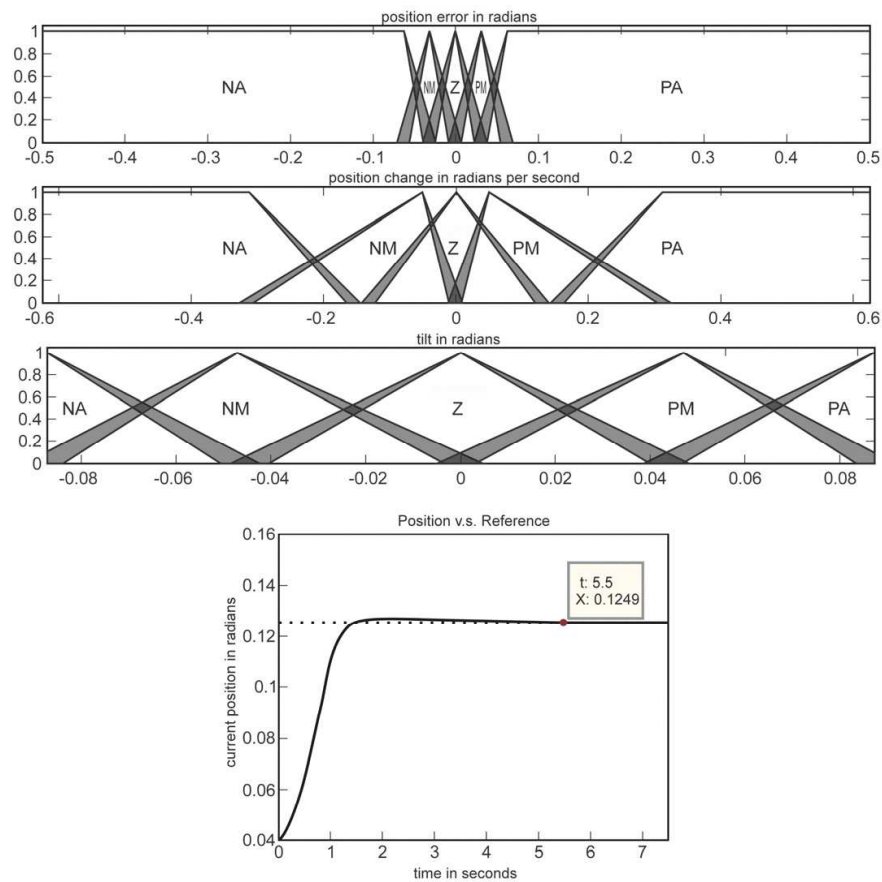


Fig. 8. Second approximation of PT2FLC modifying the FOU of sets

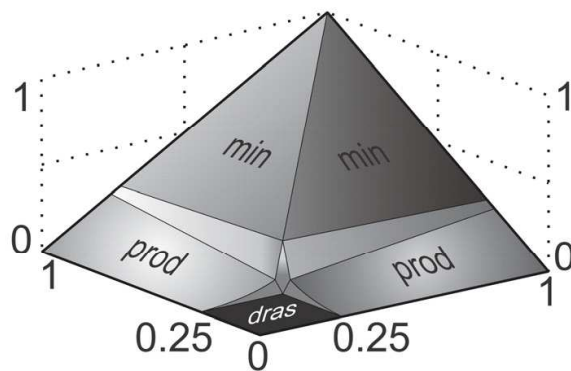


Fig. 9. Parametric Fuzzy Conjunction using (p) –Monotone Sum with parameter $p = 0.25$

In first experiments, (Fig. 6) it was re-adjusted the FOU of every set, leaving the set distribution intact, so it was found that only for a very thin FOU in every input set it is gotten a good convergence without overshoot and other phenomena. But, what is the sense of having a very short FOU like T1FS if they will not capture the associated uncertainties of the system? So, there should be a way of tuning the PT2FLC without changing this initial optimal set distribution.

In second experiments, it was moved the FOU of every set in every variable and found a very close approximation of time response as described in Fig. 8. This configuration has wider FOU in every input and output variable as much as necessary (with uniform spread) for supporting variations in error until 0.0075 radians, in change until 0.01 radians per second and in tilt until 0.004 radians, all around the mean of every point of its corresponding set and variable.

As it can be seen, every set exhibits a wider FOU and its time response has increased over 5 seconds. Also, some overshoot and ripple are present, but reference is reached, so SSE is eliminated. This is the first best approximation using the same optimal distribution of sets, although it does not mean that there could not be any other set distribution for this application.

As it is sated in (Batyrsin, Rudas et al. 2009), a parametric operator may help to tune a T1FLC through the inference step, so every rule of the knowledge base related with the implication of the premises might be a parametric conjunction. In third experiments, it is used commutative (p) –monotone sum of conjunctions (11), where it is assigned to every section the following conjunctions: $D_{11} = T_d$ is the drastic intersection, $D_{12} = D_{21} = T_p$ is the product and $D_{22} = T_m$ is the minimum, using (7) as follows:

$$T(x, y, p) = \begin{cases} T_d(x, y), & (x \leq p) \wedge (y \leq p) \\ T_p(x, y), & [(x > p) \wedge (y \leq p)] \vee [(x \leq p) \wedge (y > p)] \\ T_m(x, y), & (x > p) \wedge (y > p) \end{cases} \quad (11)$$

In (10), it is possible to assure that when parameter $p = 0$ then the conjunction in (11) will have a minimum t-norm behavior, but when parameter $p = 1$, it will be a drastic product t-norm behavior as it can be seen in Fig. 9. If p has any other value between the interval (0,1), then it will have a drastic, product or minimum t-norm behavior depending on the membership values of operands. Resulting behavior of this monotone sum might help to diminish the fuzzy implication between two membership degrees of premises and therefore to reduce the resulting overshoot of system and then reach the reference faster. Now another task is to choose the values of every parameter of conjunctions.

Moreover the optimal FS distribution, it is used the same rule set of (Moreno-Armendariz, Rubio-Espino et al. 2010) as shown in Table 1 in order to show that any T1FLC can be extended to a PT2FLC. So, $M = 25$ rules define the T1FLC configuration, it means that there are 25 parametric conjunctions and therefore 25 parameters. When searching for an optimal value of every p , it is recommended to use an optimization algorithm in order to obtain optimal values and the resulting waste of time when calculating them manually.

According to (11), the initial values of \mathbb{p}_r , make the conjunctions to behave like min, i.e.

$$\mathbb{p}_r = [0,0]$$

It is proposed some values when optimization was performed with heuristics to get optimal rule parameters, i.e.

$$\mathbb{p}_r = [0,0.25,0.25,0,0,0,0,0.2,0,0,1,0,1,0,1,0,0,0.2,0,0,0,0,0.25,0.25,0] \quad (12)$$

Rule Parameter	Description	Over shoot	SSE	Ripple
1, 2, 5, 6, 10, 16, 17, 20, 21, 22, 25	These rules have no influence with the final response, so their parameter values might be any. These rules may be quantified using non-parametric conjunctions.	No	No	No
3, 4, 9, 11, 12, 15	These rules have a very slight influence with the final response. Some of them reduce the ripple, but they are negligible. These rules may be quantified using non-parametric conjunctions also.	No	No	Yes
13b, 14c, 18a, 23b, 24b	These rules have a very positive influence with the final response, especially the parameter value of rule 18.	Yes	No	No
7b, 19c	These rules increase or decrease the offset of the final response, but could add some overshoot.	No	Yes	No
8b	This rule help to stretch the ripple slightly but also might be useful to reduce small ripple.	No	No	Yes

Table 3. Phenomena associated with the rule operator of every implication in inference

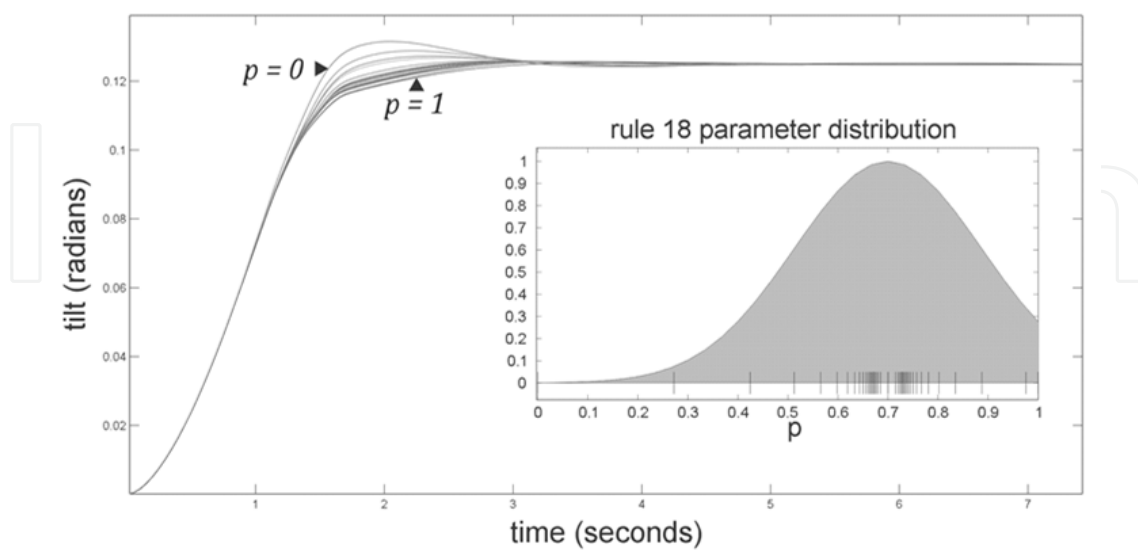


Fig. 10. Rule 18 parameter distribution for 43 experiments

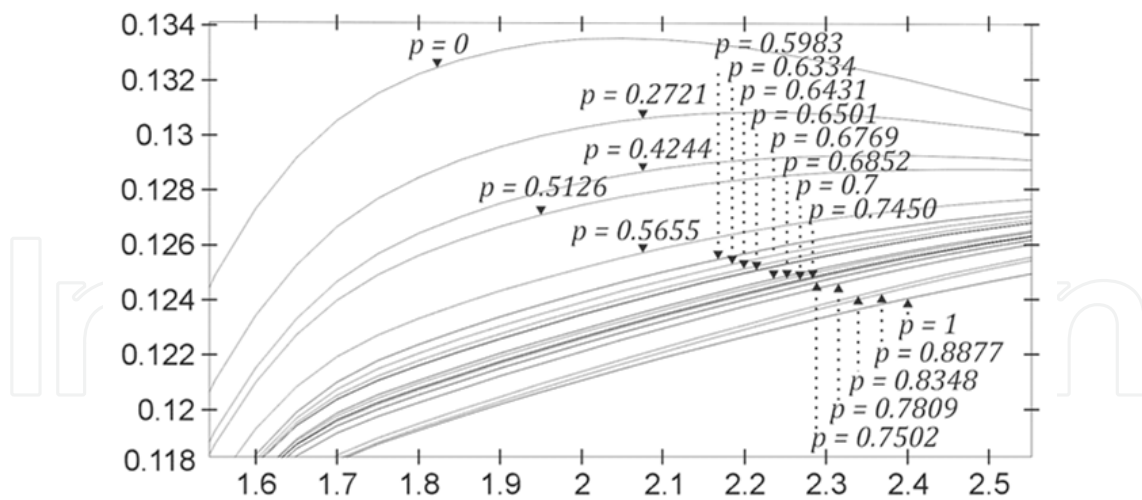


Fig. 11. Transient response for several values of parameter of rule 18

Transient Values	Min	Max	μ	σ^2
Overshoot (rads)	0.0038	0.0529	0.0086	6.5267e-05
Delay Time (s)	0.95	0.95	0.95	0
Rise Time (s)	1.05	1.2	1.1465	8.5992e-04
Peak Time (s)	2.05	3.95	3.2162	0.1249
Settling Time (s)	2.35	3.95	3.2465	0.0774

Table 4. Transient characteristics for parameter variation of rule 18

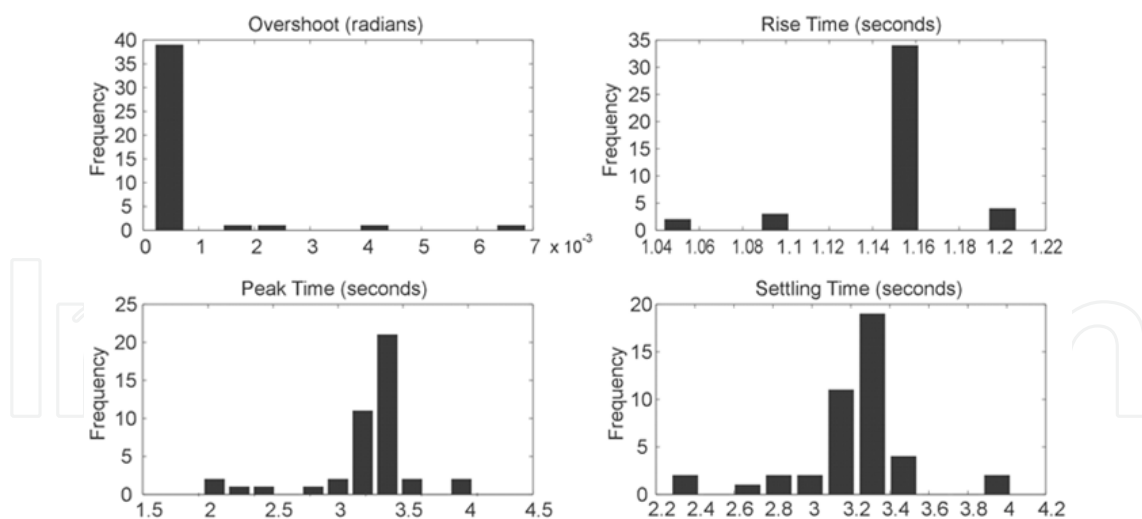


Fig. 12. Histograms for transient measures (overshoot, rise time, peak time and settling time) for rule parameter 18

Also, it was found that every rule parameter has a full, medium or null influence with final response. Table 3 shows the analysis made with every implication. For example, with rule 18 it can be diminished the overshoot when PT2FLC is just trying to control the system to reach a specific tilt of plate.

Suppose a PT2FLC where it is only modified the parameter value of rule 18 and a set of parameters that can be spread randomly around the mean of its value $p_r(18) = 0.7$. For this experiment, it was performed 43 iterations in order to show how the variation of $p_r(18)$ affects the overshoot attenuation and also other phenomena (Fig. 10-11).

Table 4 shows some results about the transient when trying to reach a tilt = 0.125 rads. Other phenomena can be analyzed for all 43 iterations. Also, in Fig.12 it can be seen that overshoot is attenuated drastically when $p_r(18) \rightarrow 1$, if it is only modified this rule. Time response (rise time, peak time and settling time) is also compromised due to parametric conjunctions. It can be seen also that drastic attenuation of overshoot occurs for $p_r(18) \lesssim 0.7$. Greater values do not affect it meaningfully. As it can be seen in (12), rule parameter proposed as the optimal for rule 18 is near to 1, which might be different with other configurations. This is because of the influence of the rest of rule parameters. However, this optimal configuration does not compromise the response time but it does eliminate the overshoot completely.

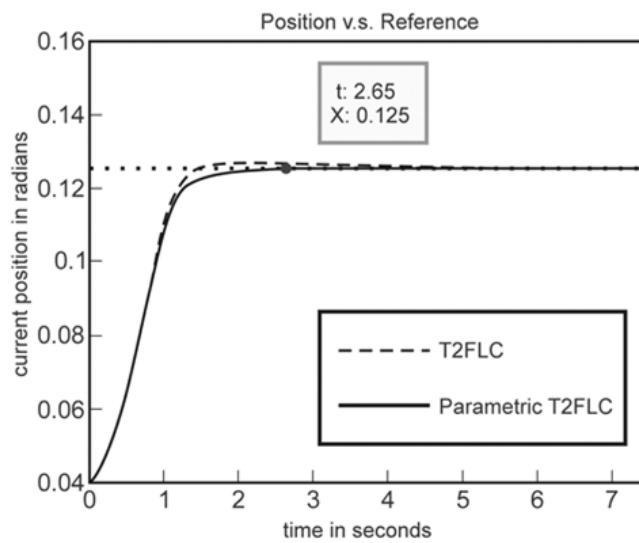


Fig. 13. Final approximation of T2FLC modifying rule parameters

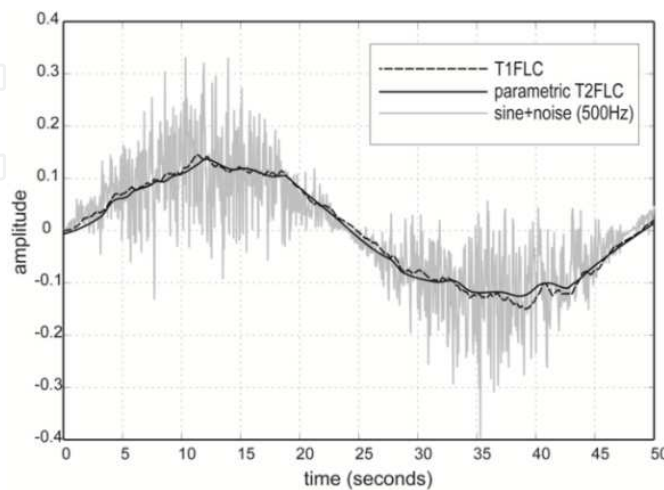


Fig. 14. Comparison of response between T1FLC and parametric T2FLC when reference is a noisy sine signal

Once it has been chosen the right parameter values of every rule it is possible to see that the influence of premises over a consequent may be regulated using a parametric conjunction. Then, overshoot and ripple have been completely removed and time response has been improved also as it can be seen in Fig. 13.

Finally Fig. 14 depicts this response of T1FLC and PT2FLC using the optimal set and rule parameters when reference cannot be determined in presence of noise. In this last experiment, signal to follow is a noisy sine signal with noise frequency equal to 500 Hz (applied to a single axis of plate). PT2FLC follows this shape very similar to T1FLC. It can be seen that PT2FLC filters all drastic changes of this noisy signal unlike T1FLC.

5. Discussion

Some of encountered problems and solutions are listed below.

5.1 Overshoot

The best results were obtained when it was reduced the FOU of every set, but reducing their FOU to zero converts the T2FLC into a T1FLC, so, this system could not deal with the uncertainties that could exist in feedback of control system (e.g. noise in sensor or noise due to illumination of room). The use of parametric conjunction operators instead the common t-norm operators, e.g. min, is the best solution to reduce the reminding overshoot after considering to modify the FOU of the sets. Due to overshoot is present when the ball is nearby the reference, inertia pulls the ball over the reference and no suitable control action could be applied. In order to smooth this action it is possible to decrease its effect diminishing the influence of premises using a parametric conjunction. A suitable value of parameter p of certain rule let drop that excessive control action, and therefore decrease the overshoot. Parameters of rules 8 and 18 have the major influence on overshoot.

5.2 Steady-State Error

There is not a precise solution to decrease the SSE. But expert can play with FOU widths of variables. For example, reducing the SSE having a big FOU in sets of variable error and decreasing all FOUs of variable change is a good option to reduce all SSE. Also it is possible to reduce it modifying the centroids of output variable tilt. Unfortunately those actions could generate additional nonlinearities so an expert must evaluate this situation.

5.3 Ripple

Ripple can be controlled considering the FOU width of the variable error. Having a big FOU in sets of variable change can help to reduce the ripple.

5.4 Response time

A simpler approximation is possible considering the values of parameters of rules 8 and 18. If $p_8 = 1$ then all reminding ripple is cleared and if $p_{18} = 1$ then almost all overshoot is eliminated, but time response is increased. Hence, if the expert has not any timing constraints then the usage of those rule parameters might help to reduce the undesired phenomenon considering this compromise.

6. Conclusion

It is introduced a PT2FLC suitable for control system implementation using a new set of parametric conjunction called (p) –monotone sum of conjunctions.

Some phenomena are present when trying to tune a fuzzy system. Original B&P T1FLC was tuned to obtain the best results as in (Moreno-Armendariz, Rubio-Espino et al. 2010). When it was implemented a B&P PT2FLC with same set distribution in input and output with same rule set, as its counterpart, it was found that some phenomenon appears again. Final system response is related with all their variables, like set distribution, FOU width or conjunction parameters and they all have an implicit phenomenon which might be controlled, depending on the characteristics of the plant and the proposed rule set for a particular solution.

A parametric conjunction to perform the implication can be applied to any fuzzy system, no matter if it is type1 or type 2. The usage of parametric conjunctions in inference help to weight the influence of premises and therefore it can be forced to obtain a certain crisp value desired. Finally it was obtained an optimal result when trying to control the B&P system, reaching the reference without overshoot, SSE nor ripple in 2.65 seconds.

When the PT2FLC is subjected to external perturbations, i.e. an extra level of uncertainty is aggregated to the system; the PT2FLC exhibits a better response over its T1 counterpart. Therefore, uncertain variations in inputs of a general FLC require sets with an appropriated FOU that can capture and support them.

Therefore, the usage of PT2FLS for control purposes gives additional options for improving control precision and the usage of Monotone Sum of Conjunctions gives an opportunity to implement PT2FLC in hardware for real time applications.

Future research needs to examine the use of other parametric classes of conjunctions using simple functions. Moreover, this work can be extended using optimization techniques for calculating both better rule parameter selection and other parameters like set distribution and rule set. A hardware implementation is convenient in order to validate its behavior in real time applications.

7. Acknowledgements

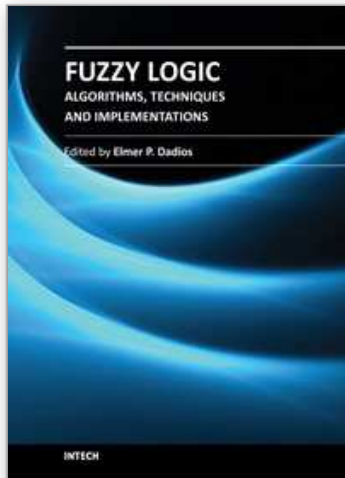
This work was supported by the Instituto de Ciencia y Tecnologia del Distrito Federal (ICyTDF) under project number PICCT08-22. We also thank the support of the Secretaria de Investigacion y Posgrado of Instituto Politecnico Nacional (SIP-IPN) under project number SIP-20113813 and project number SIP-20113709, COFFA-IPN and PIFI-IPN. Any opinions, findings, conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the sponsoring agency.

8. References

- Batyrshin, I. Z. and O. Kaynak (1999). "Parametric Classes of Generalized Conjunction and Disjunction Operations for Fuzzy Modeling." *IEEE Transactions on Fuzzy Systems* 7(5): 586-596.

- Batyrshin, I. Z., I. J. Rudas, et al. (2009). "On Generation of Digital Fuzzy Parametric Conjunctions." *Studies in Computational Intelligence* **243**: 79-89.
- E. P. Klement, R. Mesiar, et al. (2000). *Triangular norms*. Dordrecht, Kluwer.
- Karnik, N. N., J. M. Mendel, et al. (1999). "Type-2 Fuzzy Logic Systems." *IEEE Transactions on Fuzzy Systems* **7**(6): 643-658
- Moreno-Armendariz, M. A., E. Rubio-Espino, et al. (2010). Design and Implementation of a Visual Fuzzy Control in FPGA for the Ball and Plate System. *IEEE International Conference on Reconfigurable Computing and FPGAs, Cancun, Mexico*.
- Prometeo Cortes, Ildar Z. Batyrshin, et al. (2010). FPGA Implementation of (p)-Monotone Sum of Basic t-norms. *International Conference on Fuzzy Systems*.
- Rudas, I. J., I. Z. Batyrshin, et al. (2009). Digital Fuzzy Parametric Conjunctions for Hardware Implementation of Fuzzy Systems. *IEEE International Conference on Computational Cybernetics, Budapest, Hungary*.
- Wu, H. and J. M. Mendel (2002). "Uncertainty Bounds and their Use in the Design of Interval Type-2 Fuzzy Logic Systems." *IEEE Transactions on Fuzzy Systems* **10**(5): 622-639.

IntechOpen



Fuzzy Logic - Algorithms, Techniques and Implementations

Edited by Prof. Elmer Dadios

ISBN 978-953-51-0393-6

Hard cover, 294 pages

Publisher InTech

Published online 28, March, 2012

Published in print edition March, 2012

Fuzzy Logic is becoming an essential method of solving problems in all domains. It gives tremendous impact on the design of autonomous intelligent systems. The purpose of this book is to introduce Hybrid Algorithms, Techniques, and Implementations of Fuzzy Logic. The book consists of thirteen chapters highlighting models and principles of fuzzy logic and issues on its techniques and implementations. The intended readers of this book are engineers, researchers, and graduate students interested in fuzzy logic systems.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Arturo Tellez, Heron Molina, Luis Villa, Elsa Rubio and Ildar Batyrshin (2012). Parametric Type-2 Fuzzy Logic Systems, Fuzzy Logic - Algorithms, Techniques and Implementations, Prof. Elmer Dadios (Ed.), ISBN: 978-953-51-0393-6, InTech, Available from: <http://www.intechopen.com/books/fuzzy-logic-algorithms-techniques-and-implementations/parametric-type-2-fuzzy-systems>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2012 The Author(s). Licensee IntechOpen. This is an open access article distributed under the terms of the [Creative Commons Attribution 3.0 License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

IntechOpen

IntechOpen