View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Elsevier - Publisher Connector

ELSEVIER International Journal of Approximate Reasoning 26 (2001) 211–231
INTERNATIONAL JOURNAL OF APPROXIMATE REASONING
www.elsevier.com/locate/ijar

Function approximation using non-normalized SISO fuzzy systems

Marcello Salmeri [a], Arianna Mencattini [a], Riccardo Rovatti [b,*]

[a] *University of Rome Tor Vergata, DIE, Via di Tor Vergata 110, 00133 Rome, Italy*
[b] *University of Bologna, DEIS, viale Risorgimento 2, 40136 Bologna, Italy*

Received 1 November 2000; accepted 1 January 2001

Abstract

In this paper we propose an improvement in the field of fuzzy function approximation. It is well known that tuning the shape and the position of the membership functions, improves the approximation, but what about changing the heights of these functions? Usually the system is normalized so that the heights of the membership functions are set to 1, but an interesting result can be obtained if we make them variable, giving a further degree of freedom to the fuzzy system. We will use this feature in order to achieve a better function approximation, to build a second-order derivative approximation or to make the derivative of our approximation continuous. We will show also how to increase the spectral purity of the approximation function as in the case of sinusoidal functions. This approach will be analyzed under a theoretical point of view, comparing the results with those obtained with the classical approach. © 2001 Elsevier Science Inc. All rights reserved.

Keywords: Function approximation; Fuzzy logic

1. Introduction

Function approximation is an important issue both from the mathematical point of view, and in many applications in which we prefer a simplified

* Corresponding author. Tel.: +39-51-2093556; fax: +39-51-644-3073.
E-mail addresses: salmeri@ing.uniroma2.it (M. Salmeri), mencattini@ing.uniroma2.it (A. Mencattini), rrovatti@deis.unibo.it (R. Rovatti).

0888-613X/01/$ - see front matter © 2001 Elsevier Science Inc. All rights reserved.
PII: S0888-613X(01)00026-3

approach in order to simulate a more complex behavior. Also fuzzy systems have been used in literature [1–10] as function approximators because of their intrinsic simple implementation. In order to introduce the novel approach, let us recall some features of the original one.

A general fuzzy system can be regarded as a function $y = F(x)$ where x is the input vector and y is the output one. In [11] a universal fuzzy approximation theorem is proved; it states that, if f is a real, continuous function on a set $U \subset \mathbb{R}^n$, then for any $\epsilon > 0$ there exists a fuzzy function F such that

$$\sup_{x \in U} \|F(x) - f(x)\| < \epsilon,$$

where f is a target function.

Let us consider now the definition of the membership functions MFs of this fuzzy system. Consider some points in the range of approximation, named *characteristic points* ($P_1, \ldots, P_4$ in Fig. 1).

In the case shown we have chosen triangular MFs, but also different typologies may be used. If the target function f is growing monotone, the rules can be easily written as follows:

$$\begin{cases} \text{if } x \text{ is } A_0 \text{ then } y \text{ is } b_0, \\ \text{if } x \text{ is } A_1 \text{ then } y \text{ is } b_1, \\ \vdots \\ \text{if } x \text{ is } A_n \text{ then } y \text{ is } b_n. \end{cases}$$

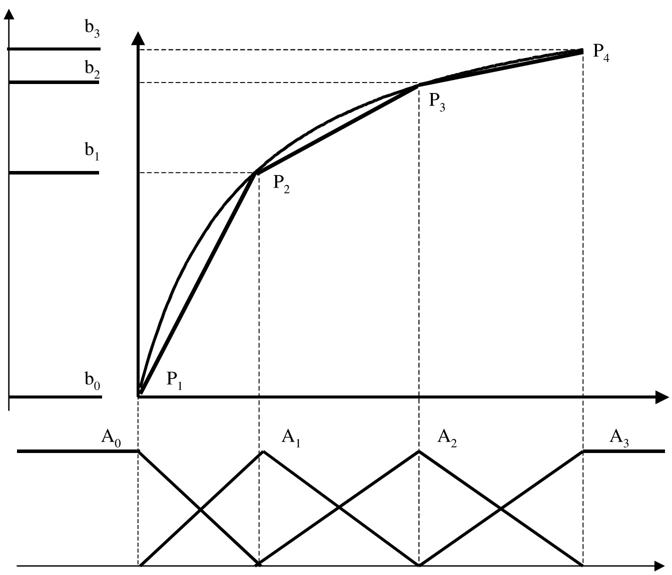

Fig. 1. Definition of the membership functions.

Note that in this case we consider an SISO Sugeno fuzzy system which produces a piecewise linear approximation. The shape of the MFs is responsible for the slope of the response and it is linked also to the local derivatives. Because the MFs overlap two by two, at most two rules are simultaneously activated. The fuzzy system described will be regarded as *normalized fuzzy system* since the heights of the MFs are all equal to 1. The best location of the characteristic points may be chosen, as we will show later, using various kinds of error parameters such as the MSE or others.

In this paper we want to use some features of the approach previously shown, such as the notion of characteristic points and the Sugeno inference method, but we want to change some other features such as the heights of the triangular-shaped MFs. We will show that changing the height of the MFs we will have a piecewise non-linear function instead of having a piecewise linear one.

Let us consider now, from a general point of view, what happens if we keep variable the MFs heights.

Let us consider a fuzzy system of the kind

$$F(x) = \frac{\sum_{i=1}^{n} c_i M_i(x)}{\sum_{i=1}^{n} M_i(x)}, \tag{1}$$

where the c_i are real consequences and the M_i are related to the MFs of the fuzzy system.

Let us consider the Sugeno inference method. We suppose to have the MFs defined as follows:

$$m_i(x) = \begin{cases} \dfrac{x - x_{i-1}}{x_i - x_{i-1}} & \text{if } i > 1 \text{ and } x \in [x_{i-1}, x_i], \\ \dfrac{x_{i+1} - x}{x_{i+1} - x_i} & \text{if } i < n \text{ and } x \in [x_i, x_{i+1}], \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

such that $M_i(x) = k_i m_i(x)$ for some constants $k_i \in \mathbb{R}^+$.

Let us consider now the Fig. 2.

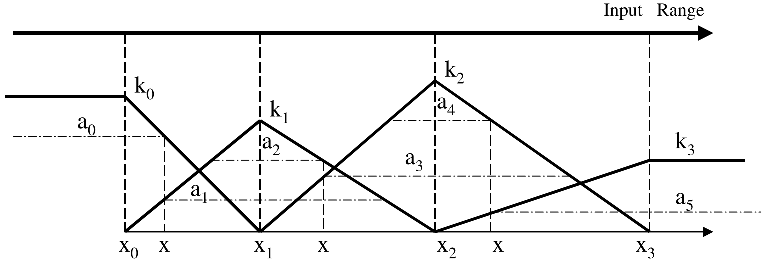

Fig. 2. Variable heights of the MFs.

For every input x in every interval $[x_{i-1}, x_i]$ with $i = 1, \dots, n$, we obtain the following relations:

$$k_i : a_i = (x_i - x_{i-1}) : (x - x_{i-1}),$$
$$k_i : a_{i+1} = (x_{i+1} - x_i) : (x_{i+1} - x),$$
$$k_{i+1} : a_{i+2} = (x_{i+1} - x_i) : (x - x_i).$$

So we have

$$a_i = \frac{k_i(x - x_{i-1})}{(x_i - x_{i-1})},$$
$$a_{i+1} = \frac{k_i(x_{i+1} - x)}{(x_{i+1} - x_i)},$$
$$a_{i+2} = \frac{k_{i+1}(x - x_i)}{(x_{i+1} - x_i)}.$$

Moreover we have that

$$y(x) = \frac{a_i c_i + a_{i+1} c_{i+1}}{a_i + a_{i+1}},$$

and so for every x such that

$$x_{i-1} \leqslant x \leqslant x_i$$

we have

$$y(x) = \frac{k_{i-1}(x_i - x)c_{i-1} + k_i(x - x_{i-1})c_i}{k_{i-1}(x_i - x) + k_i(x - x_{i-1})}.$$

Introducing the weight factor $W_i = k_{i-1}/k_i$, we can finally write

$$y(x) = \frac{W_i(x_i - x)c_{i-1} + (x - x_{i-1})c_i}{W_i(x_i - x) + (x - x_{i-1})}. \tag{3}$$

Note that if we set $W_i = 1$ for every $i = 1, \dots, n$, we have the classical output expression.

If we apply these considerations to the approximation of a target function $f : [a, b] \mapsto \mathbb{R}$ which is differentiable with continuous derivative, we can set $c_i = y_i$ in (3) where $y_i = f(x_i)$.

2. Non-normalized fuzzy systems: important features

Let us consider Fig. 3, where only the interval $[x_1, x_2]$ is considered. As we have already seen the output expression is the following:

$$y = \frac{y_1 W_2(x_2 - x) + y_2(x - x_1)}{W_2(x_2 - x) + (x - x_1)},$$

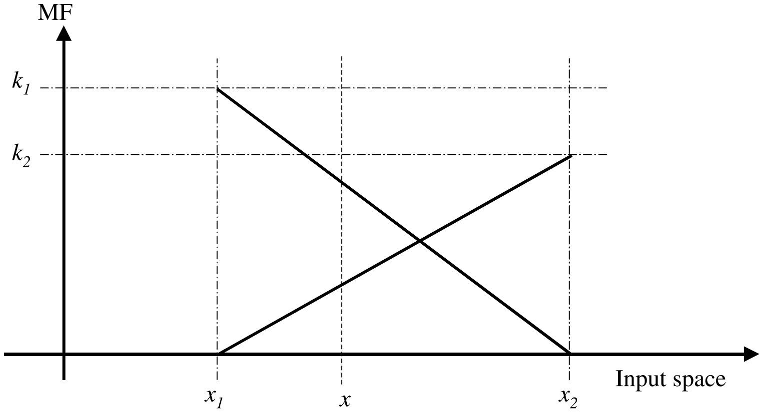

Fig. 3. Building the output.

where $W_2 = k_1/k_2$. Let us consider now the derivative of the output. We have

$$y' = \frac{\triangle x_{1,2} \triangle y_{1,2} W_2}{[(1 - W_2)x + (W_2 x_2 - x_1)]^2},$$

where we have set $\triangle x_{1,2} = x_2 - x_1$ and $\triangle y_{1,2} = y_2 - y_1$. Let us evaluate now $y'(x_1^+)$ and $y'(x_2^-)$. So we have that

$$y'|_{x_1^+} = \frac{\triangle y_{1,2}}{\triangle x_{1,2}} \frac{1}{W_2},$$
$$y'|_{x_2^-} = \frac{\triangle y_{1,2}}{\triangle x_{1,2}} W_2.$$

Let us consider now Fig. 4. The point P_1 is set to the origin of the axes. Let us consider now the tangents to the approximation function in the characteristic points P_1, P_2:

$$y = \frac{\triangle y_{1,2}}{\triangle x_{1,2}} \frac{1}{W_2} x P_1,$$
$$y = \frac{\triangle y_{1,2}}{\triangle x_{1,2}} W_2 (x - \triangle x_{1,2}) + \triangle y_{1,2} P_2.$$

We can evaluate now the set of the intersections of the two straight lines. So we have the following parameters' equations:

$$x = \frac{\triangle x_{1,2} W_2}{1 + W_2},$$
$$y = \frac{\triangle y_{1,2}}{(W_2)}.$$

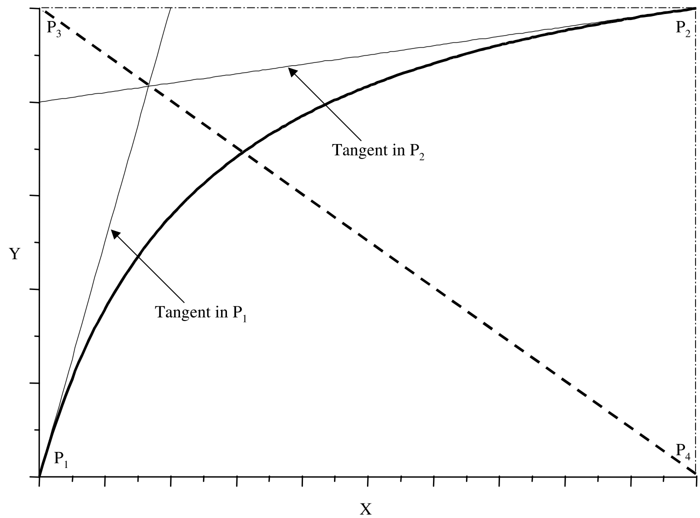

Fig. 4. The locus of the tangent intersections.

Finally we have

$$y = \triangle y_{1,2} - \frac{\triangle y_{1,2}}{\triangle x_{1,2}} x.$$

Note that this set is the sketched straight line through the points $P_3(x_1, y_2)$ and $P_4(x_2, y_1)$ shown in Fig. 4.

We will use all the considerations of this section in the following cases:

1. The approximation of a non-linear function minimizing some error parameters such as the MSE, the maximum error or others. In this case for every interval $[x_{i-1}, x_i]$ an optimum W_i has to be found to minimize this error.
2. The approximation of a non-linear function keeping the derivative of the approximation continuous in the characteristic points. In this case it is enough to fix the first value of W_i to evaluate the error parameters. The other W_is are then set too. We will show this case in Section 2.
3. The approximation of a function considering the behavior of the first and of the second derivatives, as in the following section.

3. Approximating second-order differential behavior

Let us consider the following particular case. Let $f : [a, b] \rightarrow \mathbb{R}$ be a doubly differentiable function with continuous second derivative that has been assigned as a target to be approximated. We are now interested in proving that, for any $\epsilon > 0$ there is an n, a sequence of points x_i, a sequence of weights k_i and a sequence of consequences c_i such that

$$E = \sum_{k=0}^{2} \sup_{x \in [0,1]} \left| f^{(k)}(x) - F^{(k)}(x) \right| \mathrm{d}x < \epsilon,$$

where $\cdot^{(k)}$ indicates the kth derivative so that $f^{(0)} = f$, $f^{(1)} = f'$ and $f^{(2)} = f''$. To do so, note first that defining the local error measure

$$E_i = \sum_{k=0}^{2} \sup_{x \in [a_i, a_{i+1}]} \left| f^{(k)}(x) - F^{(k)}(x) \right|$$

we get $E = \max_i E_i$.

To bound the local error measures we may consider the Taylor expansions of both f and F in the right neighborhood of x_i:

$$f(x) \simeq f(x_i) + f'(x_i)\triangle x_i + \frac{1}{2} f''(x_i)\triangle x_i^2 + g(x),$$

$$F(x) \simeq F(x_i) + F'(x_i)\triangle x_i + \frac{1}{2} F''(x_i)\triangle x_i^2 + G(x),$$

where $\triangle x_i = (x - x_i)$, both the remainders g and G are at least third-order infinitesimal for $x \to x_i$, and so we may reduce the contribution of these terms in the previous bound below any quantity simply reducing $(x_{i+1} - x_i)$.

Hence, for any $\epsilon < 0$ we can set $(x_{i+1} - x_i)$ small enough to have all those terms contributing for less than $\epsilon/2$ and, consequently,

$$E_i \leqslant \frac{\epsilon}{2} + \sum_{k=0}^{2} \sup_{x \in [x_i, x_{i+1}]} \left| f^{(k)}(x_i) - F^{(k)}(x_i) \right|.$$

We can finally show that also the magnitude of the residual term above can be reduced by shrinking $(x_{i+1} - x_i)$ if we additionally set

$$\begin{aligned} c_i &= f(x_i), \\ c_{i+1} &= f(x_{i+1}), \\ W_i &= \frac{f(x_{i+1}) - f(x_i)}{(x_{i+1} - x_i) f'(x_i)}. \end{aligned}$$

In fact, straightforward algebraic calculations reveal that with these values, we have $F(x_i) = f(x_i)$, $F'(x_i^+) = f'(x_i^+)$ and

$$F''(x_i) = 2\frac{f'(x_i)(f(x_{i+1}) - f(x_i) - f'(x_i)(x_{i+1} - x_i))}{(x_{i+1} - x_i)(f(x_{i+1}) - f(x_i))},$$

which can be easily verified to be such that

$$\lim_{x_{i+1} \to x_i} F''(x_i) = f''(x_i).$$

As a consequence, we may make $E_i < \epsilon$ by simply shrinking the overlap interval of two MFs $[x_i, x_{i+1}]$.

Since $E = \max_i E_i$, we may easily conclude that a rule set with appropriate consequences and MFs heights can be devised so that the input–output relationship F of the resulting system uniformly approximates simultaneously pointwise as well as first- and second-order differential features of any sufficiently smooth target function.

4. Approximating the target shape

In this section we want to show which may be the choice of the weights k_i in order to build a good approximation. As already said we may set the height of the MF, in every interval $[x_{i-1}, x_i]$, minimizing some error parameters such as:
1. Maximum absolute error.
2. Mean absolute error.
3. Mean relative error.
4. Mean square error.

For these cases the minimum error in the whole interval of approximation can be reached by the optimization of that error in each interval. Using this optimization method the speed of the searching is improved because not all the combinations of the k_is are considered. It has to be noted that we consider only strictly monotone target functions. If the function has one or more flexes, then it can be divided into sections in which that property is satisfied. The optimization can be performed in every section. Fig. 5 shows the best values, for each error, that can be achieved for a particular value of k in a specific interval.

Note that in these functions there are not any local minima. This means that a gradient algorithm can be used for the minimum search in order to find the best value of every k_i which minimizes a certain error in each interval. Similar graphs have been obtained for each interval, and different target functions.

5. The continuity of the derivative

In this section we want to show how the method works when the continuity of the derivative is considered. Note that this case is a particular case of the previous one. We have already seen that the derivative of the approximation function $y(x)$ with respect to x has the following expression:

$$y'(x) = \frac{W_i \triangle x_{i-1,i} \triangle y_{i-1,i}}{[W_i(x_i - x) + (x - x_{i-1})]^2}, \tag{4}$$

where we have set $\triangle y_{i-1,i} = y_i - y_{i-1}$ and likewise for $\triangle x_{i-1,i}$.

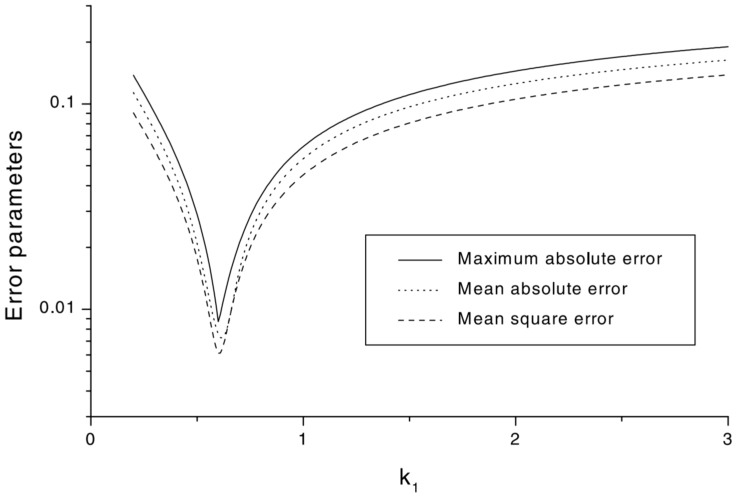

Fig. 5. Best errors vs. k.

Let us impose now the continuity of the derivative in the x_i's points. Evaluating the expressions of $y'(x)|_{x_i}$ from both the interval $[x_{i-1}, x_i]$ and the interval $[x_i, x_{i+1}]$ and making them equal we obtain

$$y'(x)|_{x_i^-} = y'(x)|_{x_i^+}$$

and then, using (4)

$$\frac{W_i \triangle y_{i-1,i}}{\triangle x_{i-1,i}} = \frac{\triangle y_{i,i+1}}{\triangle x_{i,i+1} W_{i+1}}.$$

So we obtain, for the weights W_i, the following iterative expression:

$$W_{i+1} = \frac{1}{W_i} \frac{\triangle y_{i,i+1}/\triangle x_{i,i+1}}{\triangle y_{i-1,i}/\triangle x_{i-1,i}}. \tag{5}$$

The initial condition is given by the following relation:

$$y'(x_0) = \frac{1}{W_1} \frac{\triangle y_{0,1}}{\triangle x_{0,1}}.$$

Remark 1. It is easy to see that simply setting the first weight W_1 we automatically have set the others. This initial choice is due to the error parameter used for the performances evaluation. In fact we have to know, for every chosen initial value W_1, which is the behavior of the fuzzy approximator. What we are saying is that it is not necessary to impose the initial value $y'(x_0)$ being equal to the real value $f'(x_0)$. In this way we may have many different cases of approximation as it is shown in Fig. 6.

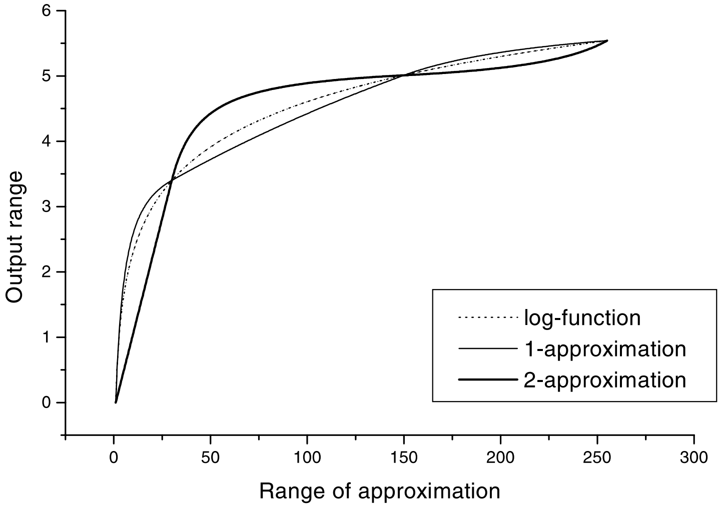

Fig. 6. The effect of changing the initial condition.

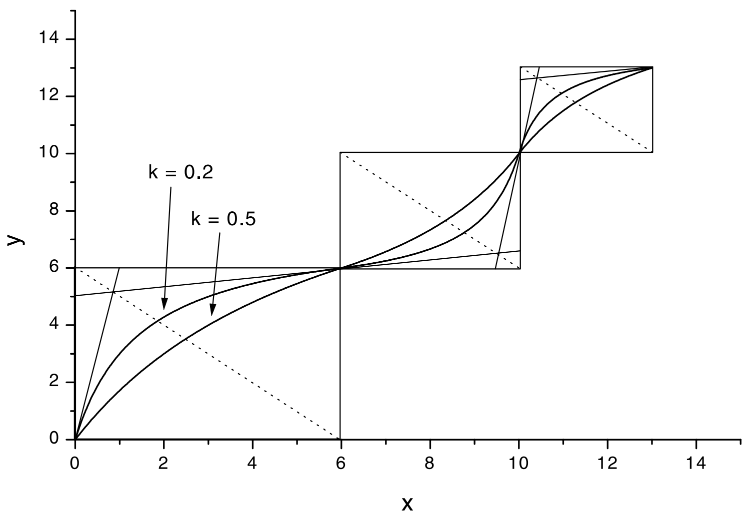

Fig. 7. Continuous derivative approximation.

Using different values of $y'(x_0)$, we obtain approximation function with change of concavity or not, but in every case we have a regular function. So it depends on what you want.

The remarks in Section 2 help us to explain some geometrical aspects of this kind of approximation.

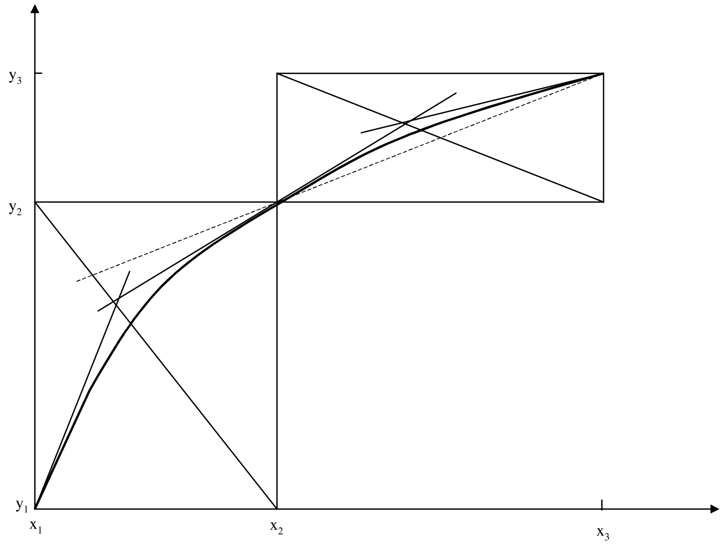

Fig. 8. Approximation without change of concavity.

As already seen, the constraint of having a continuous derivative leads to a particular shape of the output function, as the two represented in Fig. 7.

This figure shows that it is always possible to satisfy this constraint, providing that the change of concavity of the approximation function is acceptable. So, using the error parameters already described, we may set the value of W_1 and then set the others using the relation (5).

If you want an approximation without change of concavity, you have to bound the values k_i in some intervals. Let us consider Fig. 8.

If the angular coefficient of the tangent in x_2 (the solid line) is greater than the $\triangle y_{2,3}/\triangle x_{2,3}$ (of the sketched line) there is no change of concavity in x_2. This constraint can be generalized in the following way:

$$W_i \geqslant \frac{R_{i,i+1}}{R_{i-1,i}},$$

where $R_{i,i+1} = \triangle y_{i,i+1}/\triangle x_{i,i+1}$. Remember that in this case we have also the relation (5) to be satisfied and so these two constraints may limit also the choice of the x_is. Note that this kind of approximation may be interesting only if also the target function has no change of concavity.

6. Tuning of the characteristic points

As we have already said, another degree of freedom of this fuzzy system is the location of the characteristic points. The best position may be found

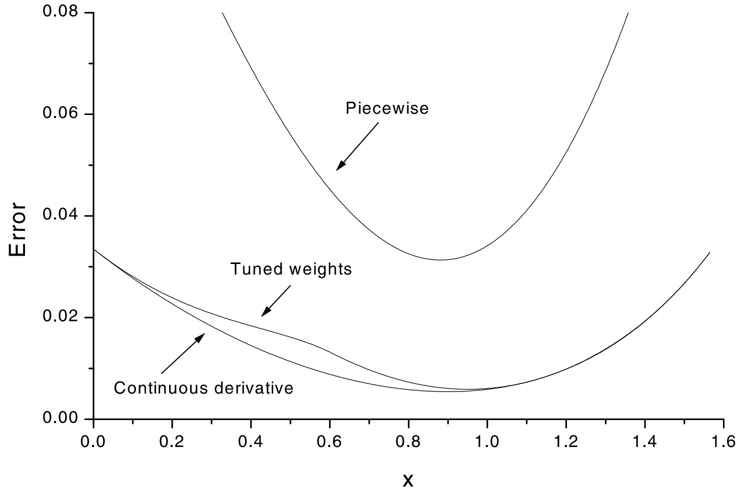

Fig. 9. Trend of the best error vs. the position of x_0.

minimizing some error parameter as we have done for the choice of the k_is. Joint optimization of all the free parameters may be unacceptably complex if the number of characteristic points increases. The graphs shown in Fig. 9 refers to the case of a function with one input variable and represent the trend of the best error versus the position of the characteristic point and are related to the different approximation methods.

The graphs shown in Figs. 10 and 11 represent the trend of the best error versus the position of the characteristic points and are related, respectively, to the tuned weights and the continuous derivative case of the same target function.

In the plane $(x, y, 0)$ there are the ranges of x_1 and x_2. The third dimension represents the weight k_0 using the mean absolute error for the optimization in every step. As you can see the graphs are very regular and there are not any local minima. This fact means that using the gradient method we may find the best location of the characteristic points; so even if the number of these points increase, this method allows us to simplify the searching.

7. Approximation results

In this section some results in the fuzzy function approximation will be shown. In order to compare this method with the original one, in which all the MFs weights are set to one, two sample functions have been considered. The first one is the function $y = f(x) = \log(x)$ and the second one is

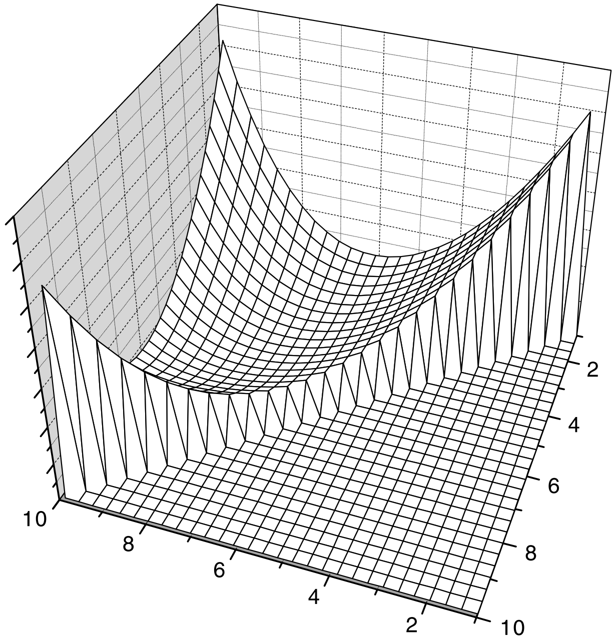

Fig. 10. Trend of the best error vs. the position of x_i's for tuned weights case.

$y = f(x) = \sin(x)$. The results are obtained optimizing the mean absolute error, but every other parameter can be used in order to achieve best results for that error. Moreover, in most cases, optimizing one of the errors above considered, also the others will be near the minimum value.

7.1. Approximating $\log(x)$ *function*

The first function we have analyzed is a logarithmic function defined in the x-range between 1 and 10. Fig. 12 shows the best approximation that can be obtained with a piecewise linear function with, respectively, one, two and three intermediate points. Fig. 13 shows the best approximation that can be obtained with a fuzzy system tuning the MF weights, with one intermediate point. We can notice how this approximation is better than one obtained in the classical piecewise linear case. The graphs showing the approximation achieved using this method with more intermediate points have been omitted because the approximation is too close to the target function and the difference cannot be appreciated.

Table 1 shows the results obtained and compares the two methods, while in Table 2 all results related to approximation with a 3-point fuzzy system with

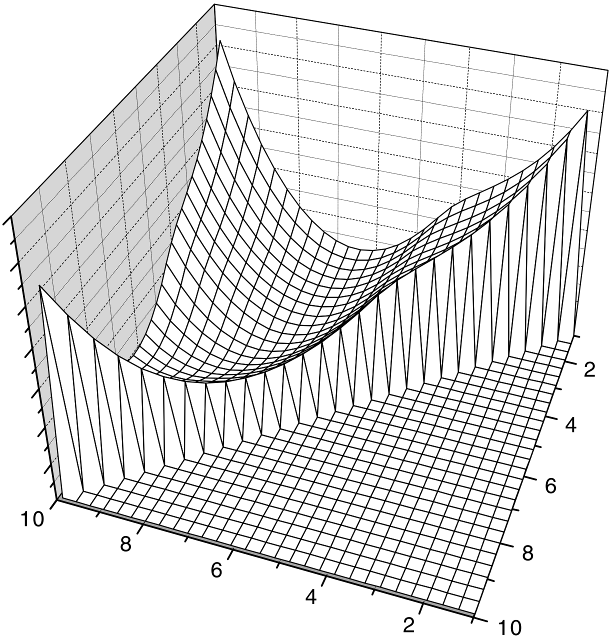

Fig. 11. Trend of the best error vs. the position of x_i's for continuous derivative case.

tuned weights are shown. These values are always related to the case of an approximation procedure with the mean absolute error optimization.

The position x_i of the characteristic points and the heights k_i of the corresponding MFs, in the case of 3 intermediate points are shown in Table 3.

7.2. *Approximating* sin(x) *function*

The second function we have analyzed is the sinusoidal function defined in the x-range between 0 and $\pi/2$. The value of the function for the other values of x can be easily derived from those in the considered range. This case of approximation is very important because many applications need to generate a sinusoidal signal with a very low spurious Fourier components. So, in addition to the previous considered errors, in this case we have taken into account also the SFDR, that show how the sinusoidal approximation is good with respect to the spectral analysis. The SFDR is defined as

$$\mathrm{SFDR} = 20\log\frac{C_i}{C_0}, \tag{6}$$

where C_0 is the module of the first Fourier coefficient and C_i is the module of the maximum Fourier coefficient except C_0.

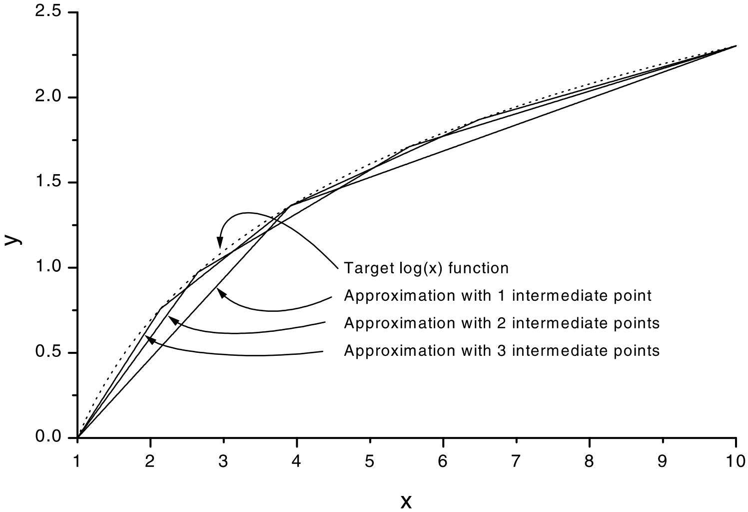

Fig. 12. Approximation of log(x) with a piecewise linear function.

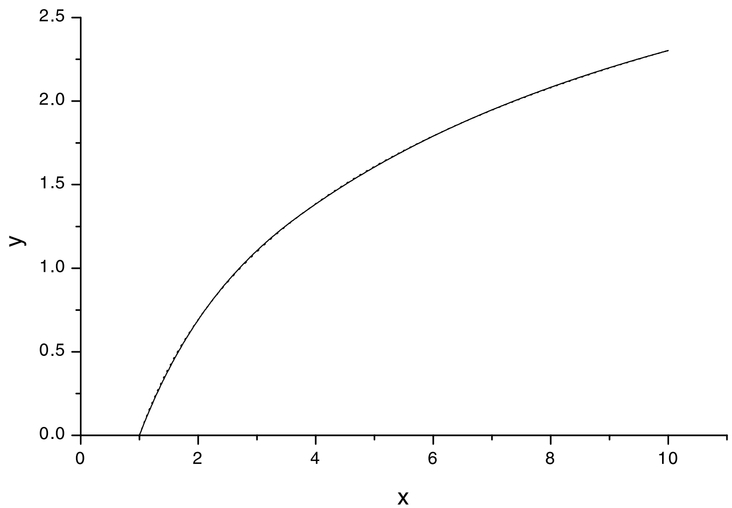

Fig. 13. Approximation of log(x) with 1-point tuned weights function: the two curves are nearly undistinguishable.

Fig. 14 shows a zoom of the graph showing the best approximation that can be obtained with a piecewise linear function with one, two and three intermediate points.

Table 1
Log(x) approximation results

	Piecewise linear	Tuned weights
1-point	0.0975197	0.00333864
2-point	0.0429848	0.000998084
3-point	0.0241075	0.000422435

Table 2
Log(x) approximation results for 3-point tuned weights approximation

	Error value
Maximum absolute	0.00168756
Mean absolute	0.000422435
Mean relative	0.076%
Mean square	0.000528096

Table 3
Log(x) 3-point fuzzy system configuration

	x_i	k_i
P_0	1	1
P_1	2.042	1.42618
P_2	3.73698	1.92717
P_3	6.30349	2.50117
P_4	10	3.14876

Fig. 15 shows the result of the sinusoidal function approximation obtained with a non-normalized fuzzy system with only one intermediate point. In Fig. 16 a zoom near the intermediate characteristic point is shown.

Table 4 shows the results comparing the two methods, while in Table 5 all results related to approximation with a 3-point fuzzy system with tuned weights are shown. These values are always related to the case of an approximation procedure with the mean absolute error optimization.

The position x_i of the characteristic points and the heights k_i of the corresponding MFs in the case of five intermediate points are shown in Table 6.

An interesting case in the fuzzy approximation is to drive the optimization in order to obtain a continuous derivative function. This objective can have many advantages either in the sinusoidal function approximation in other cases. It is true, for example, if we have to generate a control signal which has to be derived and this derivative must be continuous. The simulation results show that the errors we reach after the optimization phase are not much larger. In Table 7 the results obtained with 5-point fuzzy system are shown. If we compare these results with those in Table 6, we can notice that they are very similar.

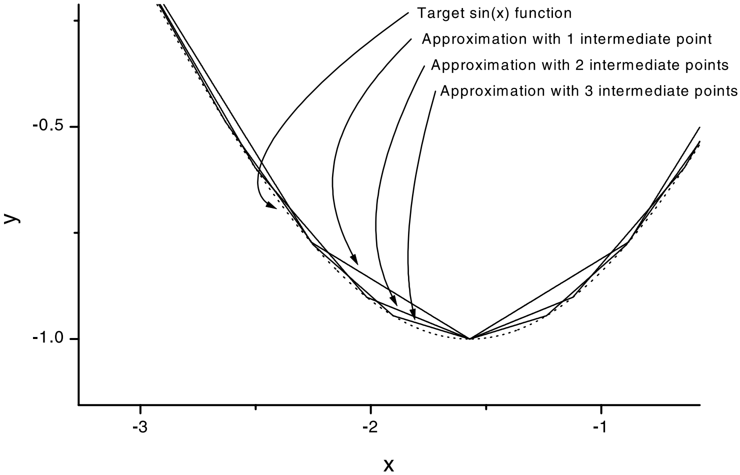

Fig. 14. Approximation of $\sin(x)$ with a piecewise linear function.

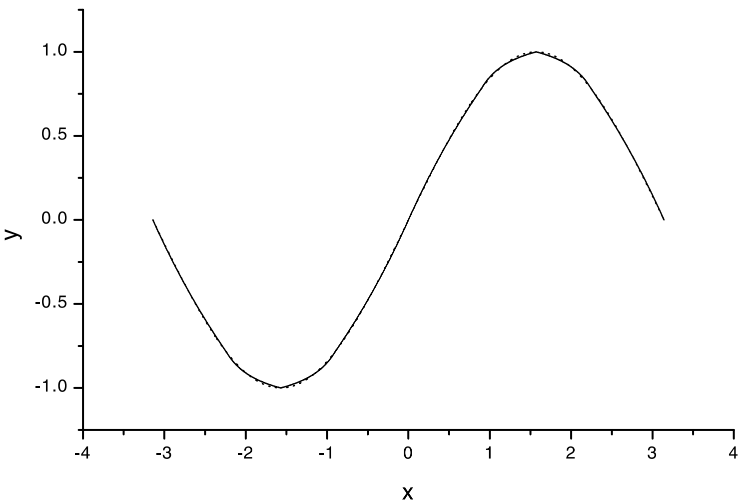

Fig. 15. Approximation of $\sin(x)$ with 1-point tuned weights function.

The fuzzy system obtained in this way is shown in Table 8.

Obviously, in this case, the derivate is not continuous in $x = \pi/2$. Moreover, in every target function that has a point in which the derivative is zero, it is not possible to obtain with this method an approximation function with the derivative continuous.

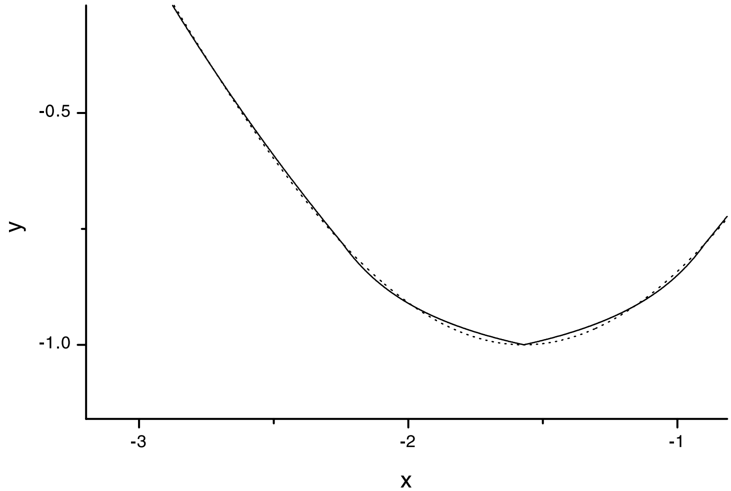

Fig. 16. Approximation of $\sin(x)$ with 1-point tuned weights function.

Table 4
Sin(x) approximation results

	Piecewise linear	Tuned weights
1-point	0.0313315	0.00542527
2-point	0.013524	0.00176777
3-point	0.00749607	0.000783107
4-point	0.00475514	0.00041307
5-point	0.00328262	0.000243867

Table 5
Sin(x) approximation results for 5-point tuned weights approximation

	Error value
Maximum absolute	0.000718411
Mean absolute	0.000243867
Mean relative	0.075%
Mean square	0.00028109
SFDR	85.95 dB

Another interesting result is derived from the analysis of the SFDR increasing the number of intermediate points. In Fig. 17 we can see the trend of SFDR in the case of a classical piecewise linear approximation and in the case of the approximation obtained with a non-normalized fuzzy system. The difference between these two values increases with the number of points. In Table 9 the numerical results are shown.

Table 6
Sin(x) 5-point fuzzy system configuration

	x_i	k_i
P_0	0	1
P_1	0.328296	1.02749
P_2	0.641233	1.11575
P_3	0.928313	1.2875
P_4	1.18438	1.61411
P_5	1.40277	2.36391
P_6	1.5708	6.17511

Table 7
Sin(x) approximation results for 5-point continuous derivative approximation

	Error value
Maximum absolute	0.000808013
Mean absolute	0.000258485
Mean relative	0.079%
Mean square	0.000307211
SFDR	81.59 dB

Table 8
Sin(x) 5-point fuzzy system configuration

	x_i	k_i
P_0	0	1
P_1	0.337067	1.02877
P_2	0.656593	1.12087
P_3	0.947243	1.29971
P_4	1.20358	1.63823
P_5	1.41659	2.39488
P_6	1.5708	5.47747

8. Conclusions

The paper shows the results reached in the field of the function approximation using non-normalized fuzzy systems. The errors we have in this case are very much better than those obtained with the classical approach. Moreover this system permits the synthesis of approximators that implements a function with a continuous derivative. This could be an important point in fuzzy control applications.

The equivalent system implementing the algorithm is not too complex. The only difference is due to adding a division in the inference calculus. In fact, as we can see in expression (1), the term to denominator is not always 1, as in the case of normalized fuzzy systems, and then we need to perform the division.

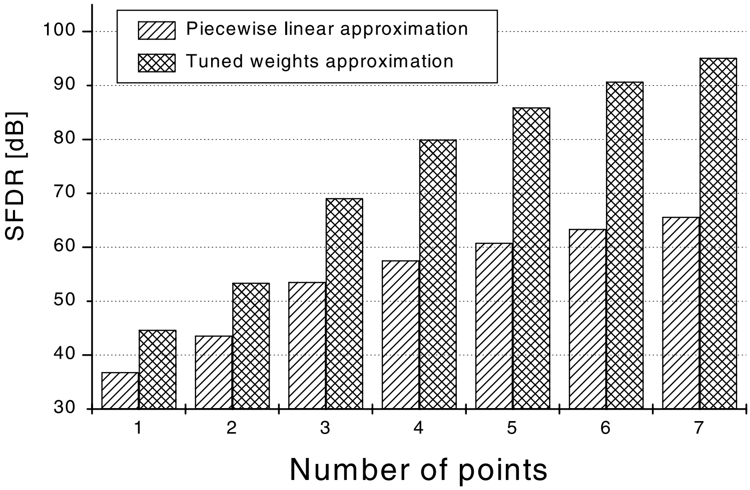

Fig. 17. Trend of SFDR with the number of points.

Table 9
SFDR values vs. the number of points

	Piecewise linear	Tuned weights
1-point	36.85	44.67
2-point	43.61	53.43
3-point	53.58	69.13
4-point	57.58	79.98
5-point	60.84	85.95
6-point	63.41	90.74
7-point	65.67	95.16

References

[1] F. Ashrafzadeh, E.P. Nowicki, M. Mohamadian, J.C. Salmon, Optimal approximation of non-linear functions by fuzzy systems, in: Proceedings of the IEEE 1997 Canadian Conference on Electrical and Computer Engineering, Engineering Innovation: Voyage of Discovery, 1997, vol. 2, pp. 781–784.

[2] L.X. Wang, J.M. Mendel, Fuzzy basis functions, universal approximation, and orthogonal least-squares learning, IEEE Trans. Neural Networks 3 (1992) 807–814.

[3] J.J. Buckley, Universal fuzzy controllers, Automatica 28 (1992) 1245–1248.

[4] B. Kosko, Fuzzy systems as universal approximators, IEEE Trans. Comput. 43 (1991) 1329–1333.

[5] H. Ying, Sufficient conditions on general fuzzy systems as function approximators, Automatica 30 (1994) 521–525.

[6] X.-J. Zeng, M.G. Singh, Approximation theory of fuzzy systems – siso case, IEEE Trans. Fuzzy Systems 2 (1994) 162–176.

[7] X.-J. Zeng, M.G. Singh, Approximation theory of fuzzy systems – miso case, IEEE Trans. Fuzzy Systems 3 (1995) 219–235.
[8] C. Fantuzzi, R. Rovatti, On the approximation capabilities of the homogeneous Takagi–Sugeno model, in: Proceedings of the Fifth IEEE International Conference on Fuzzy Systems, New Orleans, USA, 1996, pp. 1067–1073.
[9] G. Ferrari-Trecate, R. Rovatti, Sobolev approximation by means of fuzzy systems with overlapping gaussian concepts, in: Proceedings of IEEE International Fuzzy Systems Conference, Seoul, Korea, 22–25 August 1999.
[10] X.-J. Zeng, M.G. Singh, Approximation accuracy analysis of fuzzy systems as function approximators, IEEE Trans. Fuzzy Systems 4 (1996) 44–63.
[11] B. Kosko, Fuzzy systems as universal approximators, in: Proceedings of the IEEE International Conference on Fuzzy Systems, San Diego, USA, 8–12 March 1992.