Optimization of the Fuzzy Partition of a Zero-order Takagi-Sugeno Model

Felipe Fernández
DTF-FI-UPM,
Campus de Montegancedo
Madrid, 28660, Spain
Felipe.Fernandez@es.bosch.com

Abraham Duarte
ESCET-URJC
Campus de Móstoles
Móstoles, 28933, Spain
abraham.duarte@urjc.es

Ángel Sánchez
ESCET-URJC
Campus de Móstoles
Móstoles, 28933, Spain
angel.sanchez@urjc.es

Abstract

An extension of Shannon's expansion theorem is introduced in this paper to simplify and normalize the local description of triangular product-sum zero-order Takagi-Sugeno fuzzy inference systems. Also, an antecedent's parameter optimization procedure is established on the base of a multi-start randomized local search metaheuristic algorithm. In the involved multi-objective optimization problem, the antecedent parameters are adjusted by a nonuniform sampling procedure. This approach provides a set of words or linguistic labels suitable to describe the target function. In particular, the proposed learning method shows good performance and robustness in practical cases analyzed.

Keywords: Takagi-Sugeno systems, antecedent learning, partition of unity, metaheuristics

1 Introduction

System identification is the process of specifying a model of an unknown system. Due to the inherent complexity of many real structures, conventional modeling techniques have proved to be too restrictive in many real situations. In these cases a more intelligent modeling is required. Trainable fuzzy systems have been developed as a gray box modeling technique suitable for the task of fuzzy system identification. These systems have been successfully applied to various fields of science and engineering. The corresponding models include a parameter learning method combined with the flexibility and intuitive interpretation of fuzzy systems.

A variety of optimization and learning algorithms have been introduced to train fuzzy inference systems (FIS) [1, 5]. Takagi and Sugeno presented an adaptive algorithm for their fuzzy inference system [7]. ANFIS (Adaptive network based fuzzy inference system) [4] is one of the most popular models that has been used for different purposes, such as system identification, control, prediction and signal processing. This trainable model has a hybrid learning method based on gradient descent and least square estimation. Some other methods include adaptive B-spline modeling [3, 9, 10], adaptive network-based FIS [2] and fuzzy rule extraction by an evolutionary algorithm [6, 8].

Many of these methods are constructed over Takagi Sugeno fuzzy inference systems. They try to fulfill the Aristotelian principle of network parsimony: the best models are obtained using the simplest acceptable structures, containing the smallest number of adjustable parameters.

The gradient based learning procedures, evolutionary strategies, genetic algorithms and simulated annealing are well known due to their high performance and rather successful convergence to global optima. However, in many practical problems dealing with real world sampled data obtained from unknown complex systems, these algorithms have a high computational complexity. The gradient-based learning methods also have the additional problem of differentiation. Evolutionary strategies, genetic algorithms, and simulated annealing do not always converge in a short time.

More recent metaheuristics [13, 14] such as Memetic Algorithms, Tabu Search, GRASP, Path Relinking and Scatter Search, are also capable of escaping local optima in order to discover global solutions. They define general iterative master processes that guide and modify the operations of subordinate heuristics to efficiently produce high-quality solutions.

In multiobjective decision making [11], the learning algorithms search the satisfaction of different goals and they can show good performance in many practical tasks of complex real world problems. In this paper, a multiobjective metaheuristic is designed to optimize the local performances of the corresponding identification function, which is more rational and precise than the corresponding single global optimization method.

The paper consists of five sections. The main characteristics of Takagi-Sugeno system are shortly analyzed in Section 2. The third section deals with the normalized piecewise multilinear Takagi-Sugeno model considered. Section 4 describes the antecedent optimization method. In Section 5 some experimental results are given. The last section presents some concluding remarks.

2 Zero-order Takagi-Sugeno Model

Takagi-Sugeno (TS) fuzzy models were originally introduced by Takagi and Sugeno (1985) as the first systematic method for fuzzy system identification [7]. The basic concept of the TS method is the separation of the data space into fuzzy local regions. Each region is associated with a functional sub-model, which is valid to a certain degree. The global nonlinear system is obtained by a fuzzy weighted combination of local functional models. Zero-order TS systems are a group of rule-based models with fuzzy antecedents and crisp consequents:

Ri: **IF** x_1 is $N_{i,1}$ … **AND** x_n is $N_{i,n}$ **THEN** $z = z_i$

where Ri denotes the ith fuzzy rule; $i = 1..R$ (R is the number of fuzzy rules); $x = [x_1,.., x_n]$ is the input vector; $N_{i,j}$ denotes the antecedent fuzzy set of i-th rule for input j; z is the output of the MISO system and z_i is a constant output value.

The output of this model can be calculated by

$$z = \frac{\sum_{i=1}^{R} z_i \prod_{j=1}^{n} N_{ij}\left(x_j\right)}{\sum_{i=1}^{R} \prod_{j=1}^{n} N_{ij}\left(x_j\right)}$$

This computation is greatly simplified if the M_i linguistic terms of each variable x_i form a *fuzzy partition* or *partition of unity*:

$$\sum_{j=1}^{M_i} N_{ij}\left(\underline{x_i}\right) = 1$$

The output function computation is reduced to

$$z = \sum_{i=1}^{R} z_i \prod_{j=1}^{n} N_{ij}\left(x_j\right)$$

since the tensor product of fuzzy partition (or partition of unity) is also a fuzzy partition.

The corresponding trainable fuzzy inference system can be considered a learning network that includes the parameters of the considered TS model. It can be easily simplified in different ways to produce a parsimonious model, which can be optimized by various learning strategies. In general, zero-order TS models have two main sets of adjustable parameters: the antecedent parameters of the input membership functions and the rule consequent values. In this paper the consequent parameters are considered as known and only the problem of finding the optimal antecedent fuzzy partition is studied.

Supervised learning methods in fuzzy modeling are usually aimed to minimize the following MSE function (mean square error of estimation):

$$J = \frac{1}{Ns} \sum_{i=1}^{\mathbf{Ns}} \left(z(i) - z_r(i)\right)^2$$

where Ns is the number of data samples and z_r is the corresponding reference function value. In this paper a more complex objective is considered, which is a weighted addition of MSE and maximum square error of TS estimation function. The corresponding optimization method is described in Section 4

3 Piecewise multilinear TS Model

In this paper the partitions of unity considered for each variable x_i are triangular ones or second order B-splines $\{N_{ij}\}$. Each triangular fuzzy partition is defined by a knots sequence $T_i = (x_{i0}, x_{i1},..., x_{ij}, x_{imi})$. The set of knot sequences also defines a multidimensional tensor product crisp partition composed of n-dimensional intervals (Figure 1).

To simplify the analysis of the output function, in each unidimensional interval (x_{ij}, x_{ij+1}) of a variable x_i, a local variable u_{ij} is defined:

$$u_{ij}= (x_i-x_{ij}) / (x_{ij+1}-x_{ij}); \quad u_{ij}\in [0, 1]$$

In each unidimensional transformed interval, the corresponding local triangular antecedent terms are defined on unitary intervals [0, 1], and are characterised by the basics function

$$N^0(u)= 1-u = \text{Not}\ (u) = /u \quad \text{and}$$

$$N^1(u)= u$$

In each transformed multidimensional interval or fuzzy hypercubes [(0,1) …(0,1)], the corresponding local output $z = f(u_1,u_2,\ldots,u_n)$ is reduced to the canonical sum-of-products form of 2^n terms:

$$f(u_1,u_2,\ldots,u_n)=$$

$$/u_1/u_2\ldots/u_n f(0,0,\ldots,0) + /u_1/u_2\ldots u_n f(0,0,\ldots,1) + \ldots + u_1u_2\ldots/u_n f(1,1,\ldots,0) + u_1u_2\ldots u_n f(1,1,\ldots,1)$$

where $f(0,0,\ldots,0,0)$; $f(0,0, \ldots,0,1)$; … ; $f(1,1,\ldots,1,1)$ correspond to the 2^n function values in the corners of the local hypercube.

Output z, using a suitable factorization method, can also be computed by means of a multilinear recurrence relation. A particular calculus schema is

$$f(u_1,u_2,\ldots,u_n) = u_1f(1,u_2,\ldots,u_n)+/u_1f(0,u_2,\ldots,u_n)$$

$$f(0,u_2,\ldots,u_n) = u_2f(0,1,\ldots,u_n)+/u_2f(0,0,\ldots,u_n)$$

$$f(1,u_2,\ldots,u_n) = u_2f(1,1,\ldots,u_n)+/u_2f(1,0,\ldots,u_n)$$

…

$$f(0,0, \ldots,0,u_n) = u_nf(0,0, \ldots,0,1)+/u_nf(0,0, \ldots,0,0)$$

…

$$f(1,1, \ldots,1,u_n) = u_nf(1,1, \ldots,1,1)+/u_nf(1,1, \ldots,1,0)$$

The resultant global output function is therefore a piecewise multilinear C^0 hypersurface, and in each local multidimensional interval of the domain, an n-ruled hypersurface is generated by n distinct meshes of lines.

This local multilinear interpolation schema in normal form, composed of 2^n-1 interpolations, can be considered a generalisation of Shannon's expansion theorem to fuzzy functions, where the

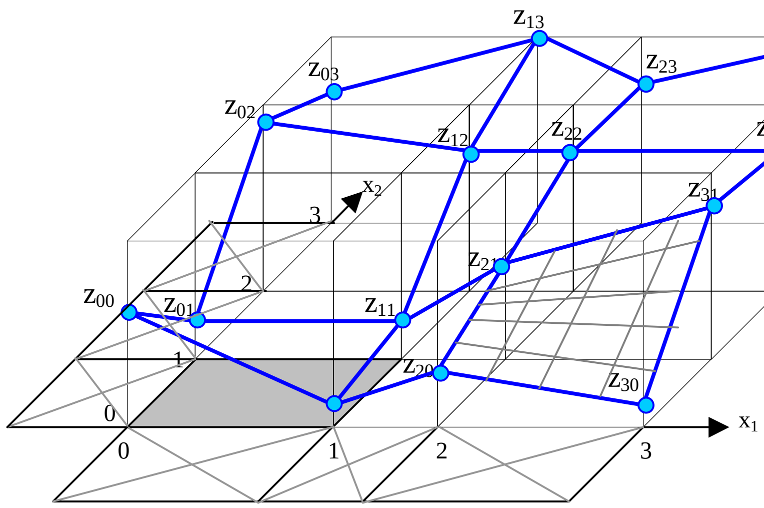

Figure 1: Example of a zero-order TS fuzzy system computed by a bivariate C^0 piecewise output function composed by bilinear cells.

fuzzy functions are defined on a unitary hypercube.

For a bivariate TS system (n=2), the following 3-steps bilinear interpolation schema follows, in the involved bidimensional cell (Figure 2):

$$f(u_1,0) = /u_1\, f(0,0) \; + u_1\, f(1,0)$$

$$f(u_1,1) = /u_1\, f(0,1) \; + u_1\, f(1,1)$$

$$f(u_1, u_2) = /u_2\, f(u_1,0) + u_2\, f(u_1,1)$$

This bivariate function determines a hyperbolic paraboloid surface [12] in each local patch.

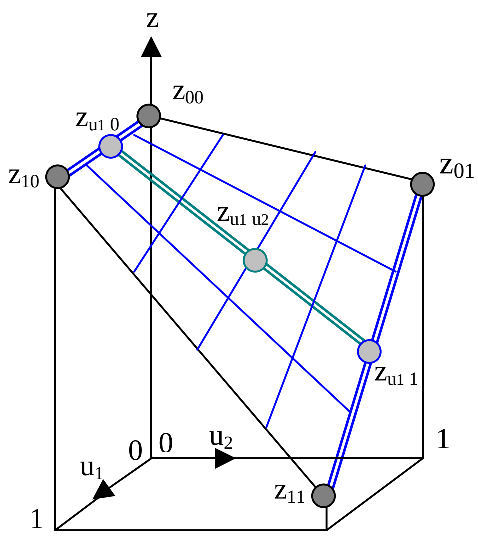

Figure 2: Example of bilinear TS output computation in a local normalized cell.

In general, the global output of the referred triangular zero-order TS system is generated by a tensor product of linear B-splines, and it is obtained by the union of the corresponding multilinear patches. The corresponding output function has the following main properties:

1. **Local multilinear form**: the considered output function is a C^0 piecewise mulilinear function formed by n-ruled patches.

2. **Local linear form**: in each local cell, it is linear in each input variable when the other are held fixed. Therefore, it is linear in each edge of the corresponding local hypercubes, which is obtained by linear interpolation along the boundaries of the grid cells.

3. **Interpolation property**: output function interpolates the original output data set.

4. **Affine invariance**: involved linear B-splines tensor product is invariant under affine transformations.

5. **Convex hull**: In each local cell, the output function always lies within the convex hull of the output control points

6. **Affine precision:** The output function preserves any affine or linear map.

Sketch of proof. These properties are directly deduced from the blending function used: tensor product of linear B-splines.

Next sections describe the multiobjective local learning method developed.

4 Fuzzy Partition Optimization

The initial tensor product triangular fuzzy partition given by the expert also establishes an orthogonal sampling grid for the corresponding target or reference function considered. The optimization of this sampling grid, to reduce the error of the approximate piecewise multilinear TS function, is the main objective of the learning method presented in this section.

Figure 3 shows a particular simple example of an orthogonal sampling grid for a bivariate

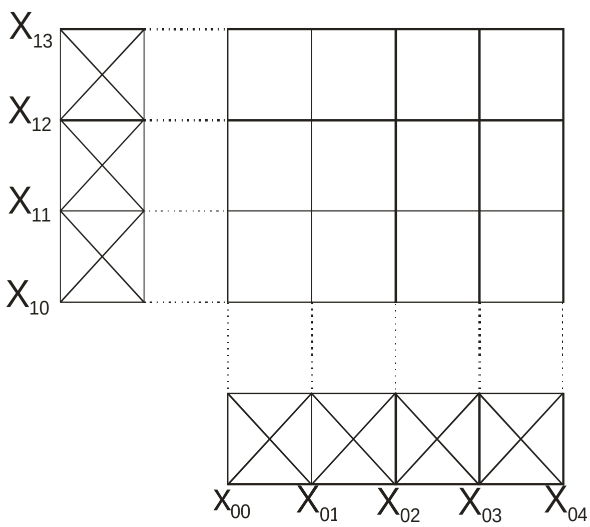

Figure 3: Sampling grid of a bivariate tensor product triangular fuzzy partition.

reference function $z = f_r(x_0, x_1)$.

Moreover, this Cartesian sampling grid is also the starting point for the TS function that we are trying to optimize. If we suppose that the target system is modeled by a reference function $f_r(x_1, x_2, \ldots, x_n)$, in each point of the sampling grid the defined TS function matches with the value of the reference function in that point. The rest of function values are obtained by means of local multilinear interpolations.

Figure 4 shows a non-optimized bivariate TS approximation example, where the reference function looks approximately like a sinusoidal cylindrical function. In this case, five triangular labels were used, in the significant input variable, to describe the corresponding nonlinearity.

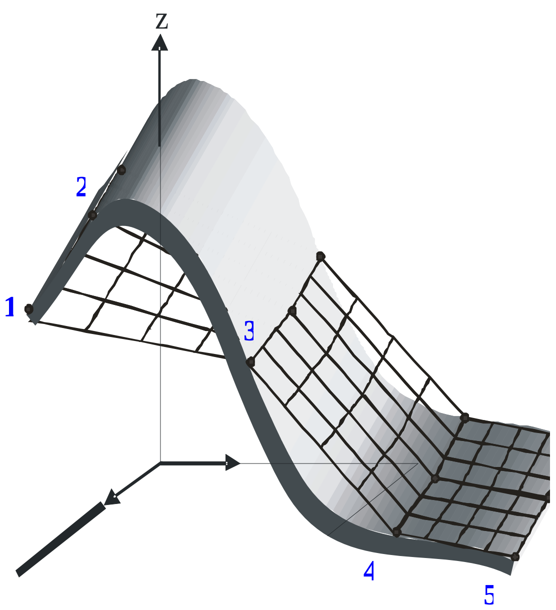

Figure 4: Raw initial piecewise multilinear approximation generated by the TS model

In order to minimize the difference between the piecewise multilinear approximation and the reference function, an error function must be defined. In this paper we use the following weighted sum of mean square error (N_S: number of samples) and maximum square error.

$$Error = 0{,}7 * mse + 0{,}3 * \max se$$

where

$$mse = \frac{1}{Ns} \sum_{i=1}^{Ns} \left(z(i) - z_r(i)\right)^2$$

$$maxse = \max_{i=1}^{Ns} \left(z(i) - z_r(i)\right)^2$$

This weighted sum is the overall objective function considered. For getting an optimal approximation of the reference function with a restricted description complexity, it is necessary to find a suitable sampling grid that minimizes the objective function. For achieving this task, a multi-start procedure with an embedded randomized local search algorithm is used.

Basic local search algorithms use an iterative improvement procedure (hill-climbing method): start from some initial solution and iteratively try to replace (move) the current solution by a better solution in a given neighborhood of that current solution [13].

Since we are considering tensor product partitions, one movement in a sampling point necessarily forces the movement of the linked points. In Figure 5 is highlighted a sampling point, its influence area (row or column) and its neighborhood, in a bivariate tensor product partition.

In each nonuniform univariate partition, to reduce the search space size, we consider between two consecutive sampling points (labels) a fixed number of possible sub-sampling points. For simplicity, is our practical experiments this parameter was set to 10. For each sampling point, the Manhattan normed neighborhood structure was therefore limited between 0 and ±5 sub-points, in each dimension of the corresponding local cell. This limitation avoids local cells crossing.

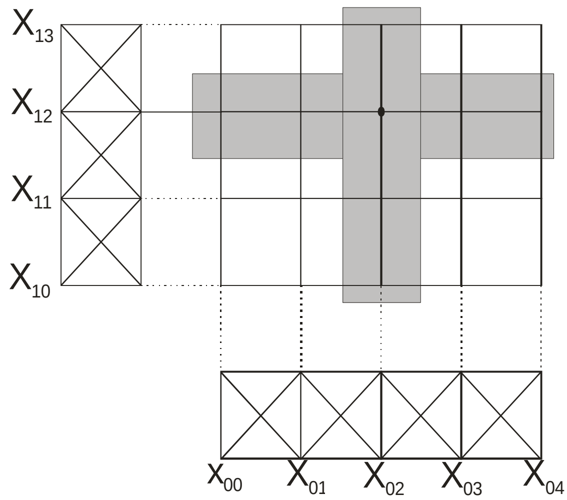

Figure 5: Example of a neighborhood structure in a sampling grid of a bivariate tensor product

With this neighborhood structure, the size of a movement in whatever direction does not have to be equal, because the movement is performed in a subsampling grid that is not uniform in all directions since some labels are closer than the others. This characteristic is quite interesting for functions that have high change rates (high spatial frequency components) in some regions and low change rates (only low spatial frequency components) in other regions.

To be able of escaping local optima, a multi-start procedure is defined, and in each iteration the possible movements sequence is randomized in order to better explore the search space.

The local search is guided by the overall error SE_{ij} in each slice of the domain (row, column, etc.), which are based on the *n* sequences of knots: $((X_{11},\ldots,X_{1m1}),\ldots,(X_{n1},\ldots,X_{nm1}))$. For example for the grid structure of Figure 3: $((X_{00},\ldots,X_{04}),(X_{10},\ldots,X_{13}))$, seven slice errors $((SE_{00},\ldots,SE_{03}),(SE_{10},\ldots,SE_{12}))$ are defined. The four column slice errors are:

$$SE_{0i} = \underset{j=0..2}{Error\ in\ cells}(i,j); \quad i = 0..3$$

and analogously, the three row slice errors are:

$$SE_{1j} = \underset{i=0..3}{Error\ in\ cells}(i,j); \ j = 0..2$$

A high level pseudo-code of the multi-start randomized local search algorithm (Multi-Start_RLS) is given in Figure 6

```
Multi-Start_RLS(T1,T2,…,Tn: knots sequences)
/* Knots sequence Ti = (Xi1, Xi2,…, Xi mi)  i=1..n */
For k = 1 to Number_of_Iterations
 While improved
  Compute overall error SEij in each partition slice
  Order the set of partition slices according to their
    errors SEij
  Place well-ranked elements in a restricted
    candidate list (RCL) of slice errors
  Randomly select a knot Xi from the RCL
  Compute movement size:  step = (Xi+1 - Xi)/10
  Make the better movement to reduce the error:
     Xi '= Move_0 (Xi , step) = Xi + step   Or
     Xi+1 '= Move_1 (Xi+1 , step) = Xi+1 - step
  If SE(Xi ) < SE(Xi) improved = true
  else if (RCL is not empty) try another movement
  else improved = false
```

Figure 6: High level pseudo-code of the multi-start local search algorithm

Figure 7 shows an optimized bivariate piecewise multilinear TS approximation from the previous example of Figure 4.

As it can be seen in Figure 7, the third label or sampling point has been drastically modified, now it is closer to the second label. With this new antecedent label definition (sampling) the overall error has been considerably reduced.

5 Experimental Results

To practically analyze the previous optimization algorithm, two simple reference functions were considered. The fist one is a univariate function: $z_r = \sin(x)$ in the domain (0=x=10), which is depicted in Figure 8. In the left image is presented the function that models the system and the 6-labeling (sampling) given by an expert. In the right image is presented the same initial function and the optimized labeling. As it can be observed, the difference between the TS approximation and the reference model has been drastically reduced.

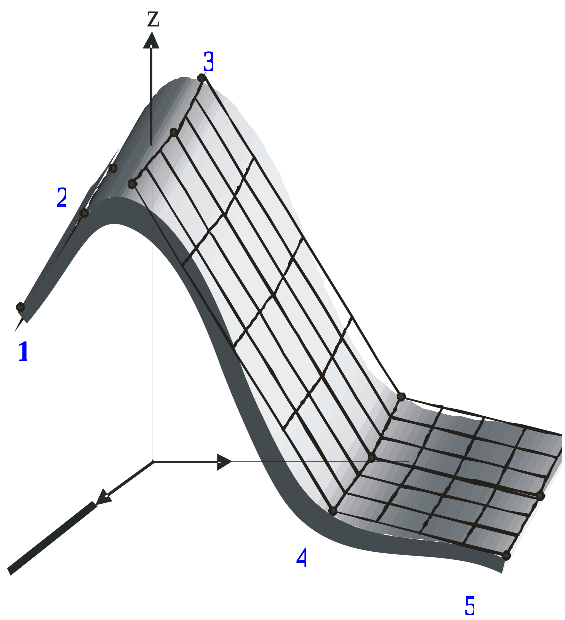

Figure 7: Optimized piecewise multilinear approximation generated by the TS model

The second function that we have studied is a bivariate function $z_r = sin\,(x^2+y^2)$ in the domain ((0=x=3),(0=y=3)). This function is shown in the first row of Figure 9. In the left column is the original function. The right column shown the same function rotated 90º, to better observe some details of this function. In the second row is presented the result of the 6⊗6-labeling proposed by an expert and the same image rotated 90º. Finally, in the last row appears the optimized labeling obtained by the presented algorithm. As in the previous case, the labeling adjustment greatly reduces the overall approximation error. Note that these results are not as good as in the previous case because this function has a radial symmetry and the modeling (sampling) has been carried out by means of a Cartesian grid. This approximation is obviously in this case much more complex than the approach with radial coordinates.

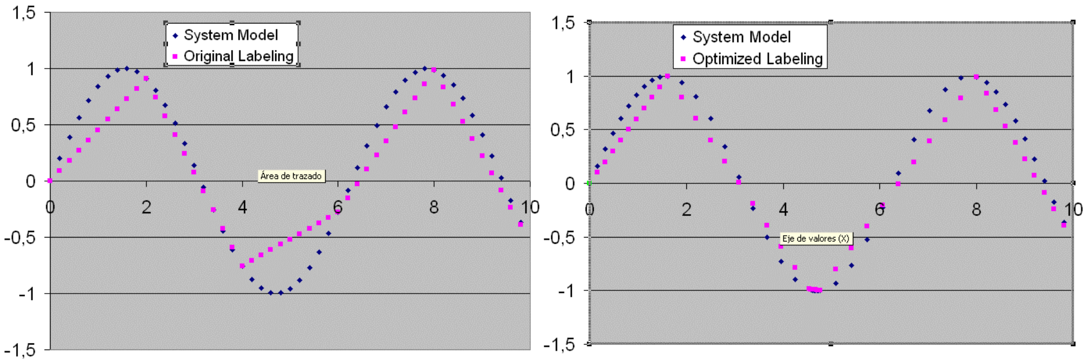

Figure 8: Example of labeling optimization of a univariate TS model.

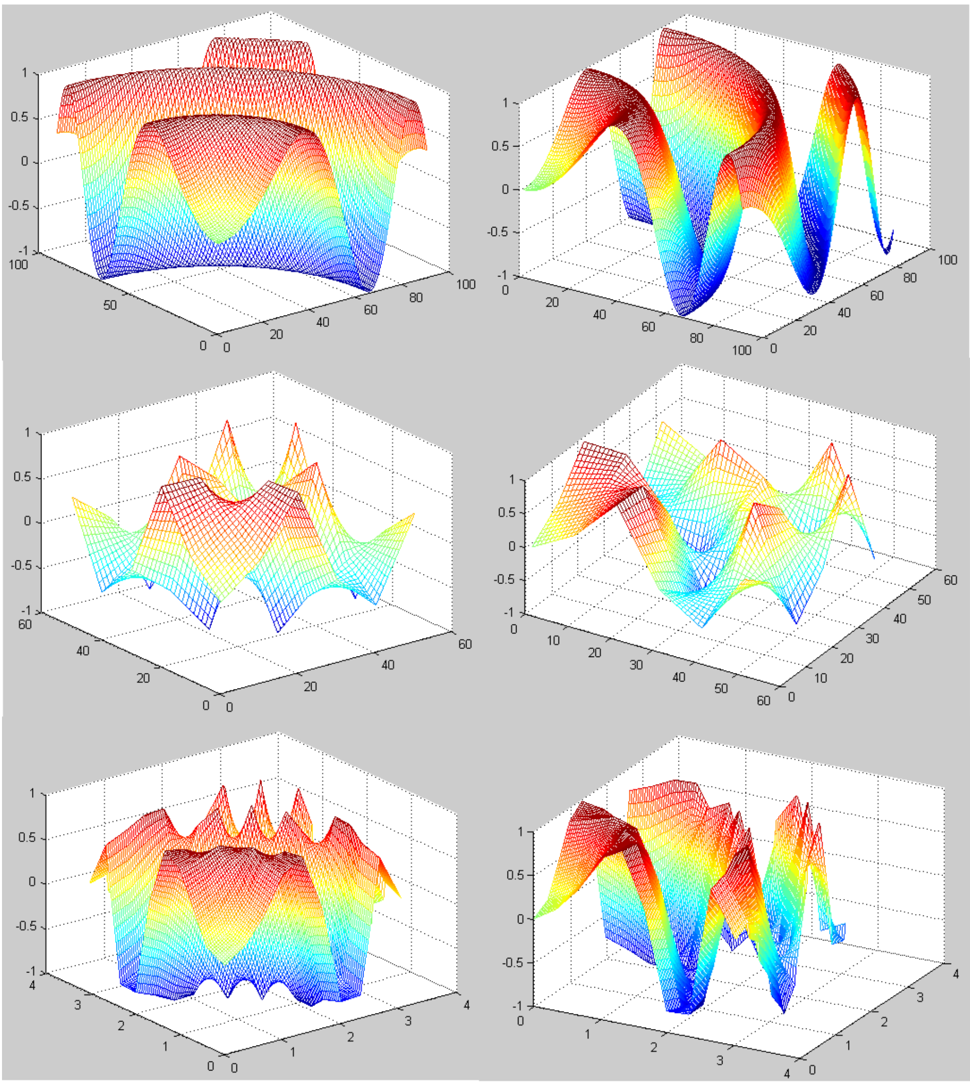

Figure 9: Example of labeling optimization of a bivariate TS model.

6 Conclusions

In this paper, we have first described an extension of Shannon's expansion theorem of logic functions to the recursive local computation of a TS model on a local fuzzy hypercube. This calculus schema normalizes and simplifies the corresponding expressions of the piecewise multilinear output function generated by a product-sum zero-order TS system defined on a tensor product triangular fuzzy partition.

Second, the antecedent adaptive TS model has been solved as a nonuniform sampling optimization schema, which was implemented by means of a multiobjective metaheuristic algorithm. The obtained results show that the presented algorithm provides a suitable functional approximation with a low computational load. A systematic comparison with other more classical methods such as ANFIS and adaptive B-splines, it is left as a future research work.

References

[1] M. Brown, C.J. Harris, *Neuro fuzzy adaptive modeling and control*, Prentice Hall, New York, 1994.

[2] Jang JSR, ANFIS: Adaptive-network-based fuzzy inference systems, *IEEE Trans Systems Man Cybernet*, vol 23, pp. 665–685, 1993.

[3] T. Kavli, ASMOD: An algorithm for adaptive spline modeling of observation data, *Int. J. of Control*, Vol. 58, No.4, pp. 947-967, 1993

[4] Keller JM, Tahani H, Bakpropagation neural networks for fuzzy logic, *Inform Sci*, 62, pp. 205–221, 1992.

[5] Kosko B, Fuzzy systems as universal approximators. *Proc IEEE Int Conf on Fuzzy Systems*, San Diego, CA, pp 1153–1162, 1992.

[6] O. Nelles, *Nonlinear system identification*, Springer Verlag, Berlin, 2001.

[7] T. Takagi, M. Sugeno, Fuzzy identification of systems and its applications to modeling and control, *IEEE Tran. On systems, Man and Cybernetics*, vol. 15, pp. 116-132, 1985.

[8] Victor, J. and A. Dourado (2003). Evolving Takagi-Sugeno Fuzzy Models. Technical Report,CISUC, 2003.

[9] J. Zhang and A. Knoll, Constructing Fuzzy Controllers with B-spline Models, *IEEE International Conference on Fuzzy Systems*, 1996.

[10] J. Zhang and A. Knoll, Unsupervised Learning of Control Surfaces Based on B-spline Models, *IEEE International Conference on Fuzzy Systems*, pp. 1725-1730, 1997.

[11] Gandibleux, X., Sevaux, M., Sörensen, K., T'kindt, V. (Eds.) *Metaheuristics for Multiobjective Optimisation*, Lecture Notes in Economics and Mathematical Systems, Vol. 535 , 2004.

[12] G. Farin, Curves and Surfaces for Computer Aided Geometric Design. Morgan Kaufmann Publishers, 5th edition, 2002.

[13] S. Voss, Metaheuristics: State of the Art. *Lecture Notes in Computer Science*, Springer-Verlag, 2001.

[14] F. Glover and G. Kochenberger, Handbook of Metaheuristics. Morgan Kaufmann Publishers, 5th edition, 2003.