

Modeling High Interest Areas in Descriptive TS Fuzzy Rule Based Systems

Ingo Renners

University of Applied Sciences
Soest Campus, Dept. Mathematics
Center of Computational
Intelligence & Cognitive Systems
Lübecker Ring 2, D-59494 Soest / Germany
renners@fh-swf.de

Adolf Grauel

University of Applied Sciences
Soest Campus, Dept. Mathematics
Center of Computational
Intelligence & Cognitive Systems
Lübecker Ring 2, D-59494 Soest / Germany
grauel@fh-swf.de

Abstract

A descriptive Takagi-Sugeno fuzzy rule based system suffers under the curse of dimensionality since the number of rules is equal to a fuzzy system with a fully filled up decision table. By defining high interest areas as areas where the input space is covered with many membership functions to achieve an acceptable error, the consequence is an often inappropriate fine grid of membership functions in orthogonal arranged low interest areas and an exponential increased number of rules. A possible solution is that the fine grid of membership functions, originated by the projection of a high interest area on the concerning single input variables, is only used if all elements of the input vector are situated in a high interest area. To guarantee the interpretability of a descriptive fuzzy rule based system the additional membership functions caused by high interest areas should fit in the coarser grid by retaining the interpretability of all linguistic variables. This paper propose an approach using evolutionary computation to identify high interest areas and using b-splines as membership functions to maintain a maximum of system interpretability.

Keywords: Descriptive Takagi-Sugeno Fuzzy System, High Interest Area, Evolutionary Identification, B-spline, Grammar based Genotype-Template

1 Introduction

Section 2 is divided in three subsections which introduce the concepts and nomenclature used in section 3 to 6. Section 3 clarifies the concept of high interest areas (HIAs). Section 4 presents the evolutionary method used to find (sub)optimal Takagi-Sugeno fuzzy rule based systems (TS-FRBSs) with HIAs. Section 5 presents some results achieved with exemplary artificial data and the paper concludes with a brief discussion in section 6.

2 Used Concepts

This sections provides the main concepts and the nomenclature which will be used in section 3 to 6.

2.1 Fuzzy Rule Based Systems

Fuzzy logic is an extension to the classical two-valued logic by concepts of fuzzy set theory, introduced by Zadeh in 1965 [9]. A fuzzy rule based system (FRBS) makes use of fuzzy logic concepts to represent a knowledge base and/or to model interactions and relations of system variables. Knowledge in FRBSs is represented by linguistic variables with an associated set of linguistic values. Linguistic values are defined by fuzzy sets, where a fuzzy set A in X is a set $A = \{x, \mu_A(x) | x \in X\}$ of ordered pairs which are defined by a membership function (MF) $\mu_A(x) \in [0, 1]$ and a linguistic term for labeling.

A FRBS is called *descriptive* if all rules share the same linguistic terms provided by a global database. Thus, the global semantic of rules is assured and the interpretability of the linguistic values of a linguistic variable can be provided by observing and constrain-

ing each set of MFs stored in the global database.

Takagi and Sugeno proposed in 1985 [7] a FRBS with following rule structure: IF x_1 is A_1 AND \dots AND x_n is A_n THEN $f(x)$, thus each rule output is a function of the input vector and the overall output of a TS-FRBS is a weighted sum of all rule outputs.

By interpreting univariate basis functions (Fig. 1) as membership functions (MFs), and n -variate basis functions, built up by multiplying n univariate basis functions, as an antecedent of a FRBS rule, radial basis function networks (RBFNs) has shown to be functional equivalent to TS-FRBS [3]. Furthermore, if the univariate basis functions of a RBFN are defined globally for each possible input n and if the n -variate basis function of all RBFN-neurons are built up by permutating all possible univariate basis function combinations, then an RBFN becomes equivalent to a descriptive TS-FRBS [1, 4]. As consequence, supervised learning of TS-FRBS can be performed very efficiently by solving an overdetermined linear system of equations [5].

2.2 Evolutionary Computation

By concerning a population as a set of solutions, the term evolutionary computation (EC) subsumes all population based approaches which perform random combination, alteration of potentially useful structures to generate new solutions and a selection mechanism to increase the proportion of better solutions. Thus, EC comprises techniques such as genetic algorithms, evolutionary strategies, evolutionary programming, genetic programming and artificial immune systems or hybrid methodologies resulting from the combination of such techniques. If EC concepts are used in algorithms, those algorithms are referred as evolutionary algorithms (EAs).

2.3 B-Splines

Because of some special characteristics we use b-splines as MFs. B-splines are recursively defined over a knotvector λ , consisting of at least order $k + 1$ knots, by

$$B_j^{k+1}(x) = \frac{x - \lambda_j}{\lambda_{j+k-1} - \lambda_j} B_j^k(x) + \frac{\lambda_{j+k} - x}{\lambda_{j+k} - \lambda_{j+1}} B_{j+1}^k(x)$$

$$B_j^1(x) = 1, \text{ if } x \in [\lambda_j, \lambda_{j+1})$$

$$= 0, \text{ otherwise}$$

with x as input value and $B_j^k(x)$ as activation value of the j^{th} b-spline defined over the knots λ_j to λ_{j+k} . The concept of a knotvector and the characteristics of b-splines meets our demand in four ways.

- It is compatible with the construction demands of a descriptive FRBSs. Each single knot vector defines a whole set of MFs for one linguistic variable.
- It is easy to achieve a finer grid of linguistic terms on each one-dimensional projection of a HIA by simply inserting some knots in the concerning knotvector.
- B-splines can form extremely different shapes simply by changing their order or their knot-positions.
- B-splines show some for FRBSs essential characteristics such as positivity and local support and furthermore they form a partition of unity (activation of all b-splines defined by one knotvector sum up to one) which simplifies the interpretation of a FRBS and improves its learning capabilities [8].

Thus b-splines, which have been employed and were successfully deployed in surface-fitting algorithms for computer aided design tasks [2], are first class candidates to form MFs in descriptive FRBSs.

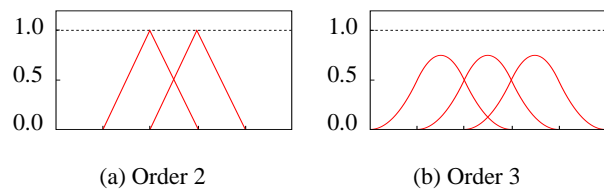


Figure 1: Univariate b-splines of order 2 and 3.

3 Embedding HIAs in Descriptive TS-FRBSs

By covering the input variables with more MFs to represent the data in a HIA leads often to an excessively complex modeling in other areas (Fig. 2). Not only memory requirements rises exponentially, but also the danger of over-fitting the data in inappropriate fine

gridded areas. A possible solution is to use an additional separate fine partitioning for some HIAs. This reduces the number of rules for an exemplary FRBS, as depicted in Fig. 2, from $6 * 6 = 36$ (fine grid is integrated in rough grid) to $4 * 4 + 2 * 2 = 20$ (use fine grid in addition to rough grid), with an associated improvement of generalization in the no longer overly complex modeled areas. Thus, the overall FRBS output is a weighted sum of the activation values of all n -variate activation functions (rule-antecedents) of the rough grid plus the weighted sum of the activation values of all n -variate activation functions (rule-antecedents) of each HIA grid. The HIA n -variate activation functions can be seen as rule-antecedents which refine the output of the more general output based on the rough grid. By using b-splines as MFs, two adjacent knots of each rough grid dimension are limiting the HIAs and by utilizing those limiting knots as outermost knots in each HIA-knotvector, we assure (if order > 1) that there is no gap in the overall FRBS output.

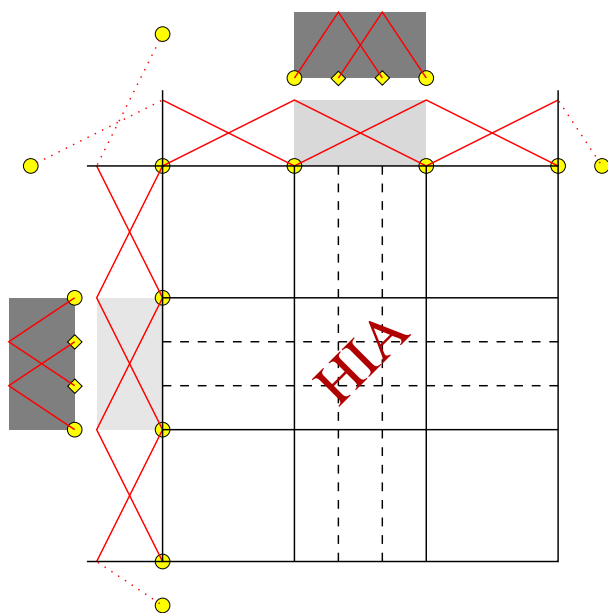


Figure 2: Rough and fine input partitioning.

4 Evolutionary Identification of HIAs

To find optimal knotvectors for all grids (rough grid and HIA grids) we used an EA which utilizes an n -ary tree (see Fig. 3) to represent genotypes [6]. Because the overall task can be classified as fuzzy modeling or system identification, variable length coding of the genotype is desirable to preserve generalization ability to other datasets and demanded constraints. There-

fore the genotype template is grammar based and easy to configure for other datasets with an arbitrary number of possible inputs and phenotype TS-FRBSs with an arbitrary number of claimed rules and HIAs.

For sake of simplicity we restricted the TS-FRBS to use only one HIA. The usage of more HIAs is straightforward. By regarding a TS-FRBS as a functional entity (FE) and each component (i.e. features, MFs, etc.) of this main FE as subordinate FEs, we can assign FEs to parts of the genotype tree (see Fig. 4). For example the whole model (the FE TS-FRBS) is represented by the complete genotype n -ary tree, thus, the root node of the genotype n -ary tree is labeled “TS-FRBS”. A feature is only a part of the whole model and consequently it is represented by a subtree of the genotype n -ary tree and the concerning root node of the subtree is labeled as “feature”. The FE “feature” has in example two child-nodes, namely a FE called “general knotvector” and a FE called “HIA knotvector”. Both of these FEs have as much child-nodes as the phenotype (the TS-FRBS) will have knots to define the MFs. As described in Fig. 2 all knots belonging to a “HIA knotvector” are placed between two knots of a “general knotvector”.

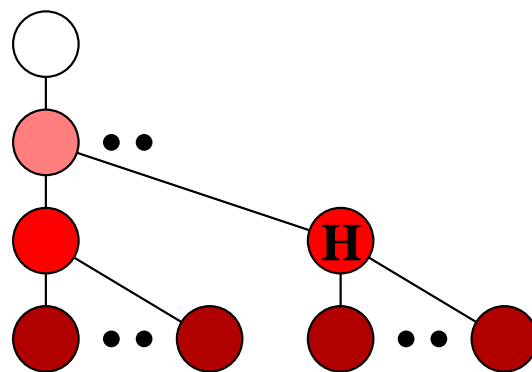


Figure 3: Exemplary n -ary tree based genotype.

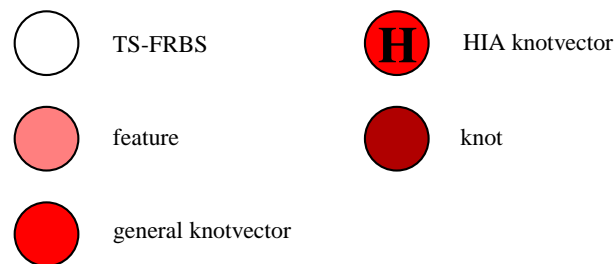


Figure 4: The functional entity label of the genotype (sub)trees with node as root node.

Each node can act as container for variables (see Fig. 5). So called decision variables are subject to evolutionary changes. Real valued decision variables are coupled to a strategy parameter and mutation is done by concepts as used in evolutionary strategies. Integer valued decision variables are mutated by increasing/decreasing or replacing the concerning value of the decision variable. Furthermore some functions to support simple constraint handling were implemented. An example is that all children of the FE “general knotvector” should contain variables (representing the knots in the phenotype) with values of increasing order. Crossover is performed by randomly choosing two nodes of each parent. The concerning subtrees are swapped if the genotype-grammar indicates legality, otherwise crossover is not performed.

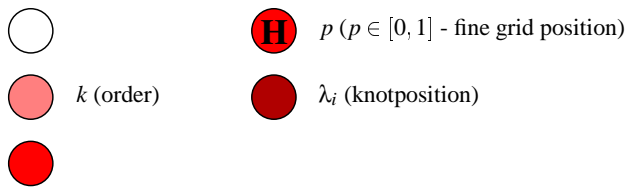


Figure 5: Decision variables assigned to nodes.

5 Results

To test our approach we used a mathematical function with two inputs and one output. The function is given

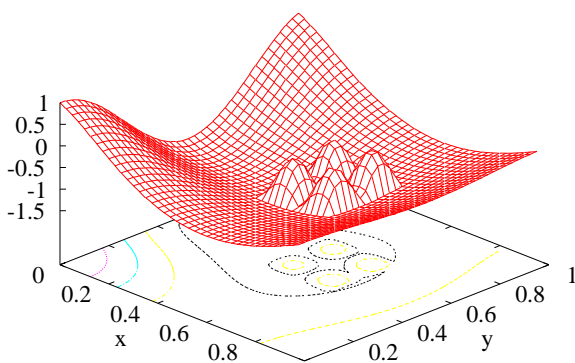


Figure 6: Artificial training dataset.

by $f(x,y) = \cos(6y)/(1 + (10x^2)) - \sin(3x)$. Only in the region $x \in [0.4, 0.7] \wedge y \in [0.4, 0.7]$ we added to the

output $abs(\sin(6\hat{x})\sin(6\hat{y}))$ where \hat{x} and \hat{y} are rescaled (to the interval $[0, 1]$) versions of x and y . We created 1681 uniformly distributed samples leading to a dataset as depicted in Fig. 6.

The parameter settings of the EA were set to 50 generations, number of individuals 80, competitors in tournament selection equal to 4, one elite individual and the mutation probability of each integer valued decision variable was set to 0.001. Furthermore we restricted the EA to use for each “general knotvector” a number of MFs in the range of $[6, 10]$ and for each “HIA knotvector” a number of MFs in the range of $[3, 10]$. The TS-FRBS should also use less than 100 receptive fields (rules). Fig. 7 illustrates the best found phenotype.

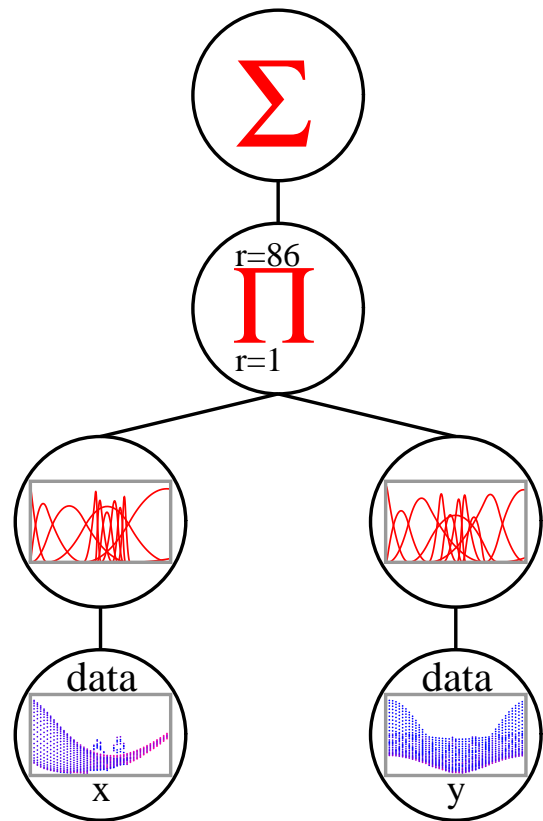


Figure 7: Evolutionary optimized TS-FRBS using one HIA and 86 receptive fields.

As expected the TS-FRBS used b-splines of order three to approximate the smooth original function. The general rule-set consist of $8 * 7 = 56$ n -variate functions and The HIA (covered by $6 * 5 = 30$ n -variate HIA functions) is placed correctly. The overall mean square error (MSE) of the model is computed to 0.0005875, which is approximately 25 times smaller as with the best evolutionary identified descriptive TS-FRBS restricted to 100 rules. Fig. 8 shows the TS-FRBS output using the input values of the original dataset.

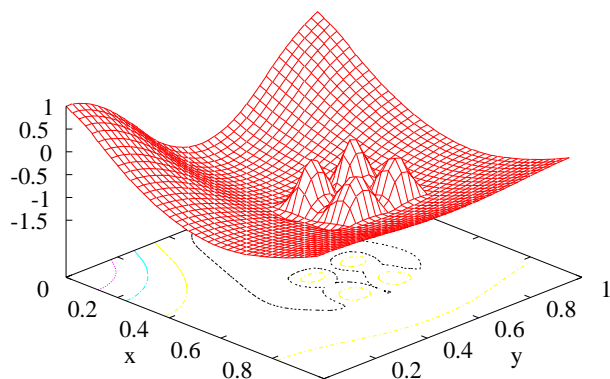


Figure 8: Dataset computed by the evolutionary optimized TS-FRBS using one HIA.

6 Discussion

TS-FRBSs with HIAs are closely related to descriptive TS-FRBS, but their performance by using the same number of rules, is superior. The interpretability of TS-FRBS is still given. In fact TS-FRBSs with HIAs can be extended to real hierarchical fuzzy systems by decoupling the learning of the weights of the n -variate functions (rule-antecedents) into a general rule learning part and a HIA rule learning part. Thus, the above described approach could be a starting point for the creation of hierarchical TS-FRBS in that way, that first the HIA identification is performed, secondly the HIA MFs are decoupled and hierarchically arranged, and finally the weight of each n -variate function (rule-antecedent) layer is trained by back-propagation. Next work will be done by using more than one HIA and by considering the effect of different partitioning of the general rule grid.

Acknowledgment

This work has been supported by the Ministry of Science and Research, North-Rhine Westphalia.

References

- [1] Cordon, O., Herrera, F., Hoffmann, F. and Magdalena, L. (2001) Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases. *Advances in Fuzzy Systems - Application and Theory Vol. 19*, World Scientific Publishing Co. Pte. Ltd., ISBN 981-02-4016-3, p. 337.
- [2] Cox, M.G. (1990) Algorithms for Spline Curves and Surfaces. *NPL Report DITC 166/90*, 36 pages.
- [3] Jang, J.R. and Sun, C.T. (1993) Functional equivalence between radial basis function networks and fuzzy inference systems. *IEEE Trans. Neural Networks*, vol. 4, no.1, pp. 156–159.
- [4] Nelles, O. (2001) Nonlinear System Identification - From Classical Approaches to Neural Networks and Fuzzy Models. *ISBN 3-540-67369-5*, Springer.
- [5] Renners, I. and Grauel, A. (2001) Optimizing lattice-based associative memory networks by evolutionary algorithms. *Information Sciences*, vol. 136, pp. 69–84.
- [6] Renners, I. and Grauel, A. (2003) Variable length coding of genotypes using n -ary trees. submitted to 9th Int. Conf. on Soft Computing – Mendel 2003.
- [7] Takagi, T. and Sugeno, M. (1985) Fuzzy identification of systems and its application to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 15(1), pp. 116–132.
- [8] Werntges, H.W. (1993) Partitions of unity improve neural function approximators. *IEEE Int. Conference on Neural Networks (ICNN)*, vol. 2 pp. 914–918, San Francisco, USA.
- [9] Zadeh, L.A. (1975) The Concept of a Linguistic Variable and its Application to Approximate Reasoning *Information Sciences - Application to Approximate Reasoning I - III*