Development of Multiple Linguistic Equation Models with Takagi-Sugeno Type Fuzzy Models

Esko K. Juuso
Control Engineering Laboratory
Department of Process and Environmental Engineering
University of Oulu, Finland
Email: esko.juuso@oulu.fi

Abstract— *Multimodel approaches are widely used with linear submodels, but border areas around submodels are problematic. Special cases of fuzzy linguistic equation models, which can be understood as linguistic Takagi-Sugeno (LTS) type fuzzy models, can be used to solve these problems in many cases. These models use a special nonlinear scaling approach for both inputs and outputs. The LTS models are robust solutions for applications where the same variables can be used for defining operating areas and also in the submodels. No special smoothing algorithms are needed.*

Keywords— linguistic equations, nonlinear systems, multimodels, Takagi-Sugeno fuzzy models.

1 Introduction

Fuzzy set systems enable the use of expert system techniques in uncertain and vague systems [1]. The traditions of physical modelling on the basis of understanding system behaviour are maintained with fuzzy rules and membership functions, which can represent not only gradually changing nonlinear mappings but also abrupt changes [2]. Various approaches using either expertise or data are used in constructing these mappings, but as the complexity of the application increases more and more combined approaches are required. These approaches are presented in the upper right-hand corner of Fig. 1. Heuristic knowledge and know-how can be introduced to fuzzy set systems with a trial and error based approach. Data-based approaches rely usually on automatic generation of rules from predefined simple sets of membership functions. Both knowledge and data need to be used together to develop practical applications.

Linguistic fuzzy models [3] are mainly used in the knowledge-based approach, whereas Takagi-Sugeno (TS) type fuzzy models [4] and fuzzy relational models [5] are mainly used for data-driven methods. TS models are constructed by combining supervised and unsupervised learning. The antecedent membership functions can be initialised by grid partioning, iterative search, or fuzzy clustering [6]. The neuro-fuzzy ANFIS method [7] is widely used in tuning. The structure and the parameters can also be updated recursively when new data become available [8]. The trade-off between global model accuracy and interpretability of local models as the linearisations of a nonlinear system is important in the development of a TS model. To restrict the freedom of the parameters a multiobjective identification for dynamic TS models is presented in [9].

This paper presents a set of fuzzy linguistic equation models, which combine nonlinear scaling and Takagi-Sugeno type fuzzy models.

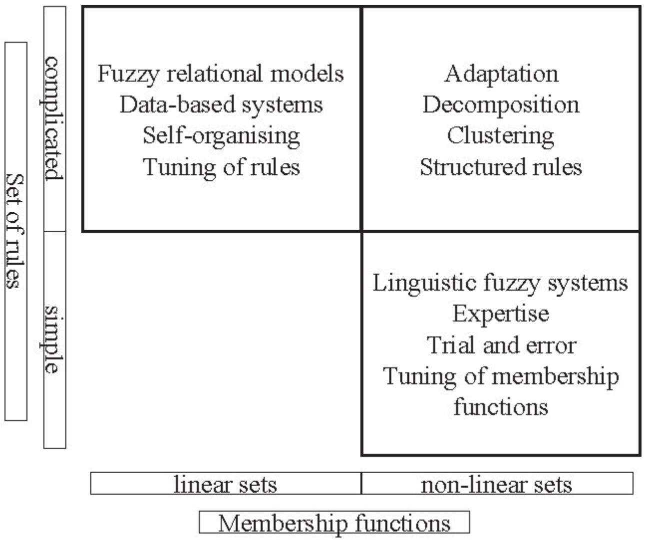


Figure 1: Classification of fuzzy set systems [2].

2 Methodologies

2.1 Linguistic equations

The linguistic equation (LE) approach originates from fuzzy set systems: rule sets are replaced with equations, and the effects of membership functions are handled with scaling [10]. For nonlinear models, the scaling technique must be nonlinear as the model equations are linear. The scaling functions are called membership definitions as they are closely connected to the membership functions used in fuzzy set systems. [2]

Nonlinear scaling. Nonlinear scaling is carried out using functions which denote membership definitions. The mapping function is performed by converting variable values from the variable range into a range [-2, 2], known as the *linguistic range*. The new range describes the distribution of variable values over the original range fairly accurately. Membership definitions are presented by a variable specific function

$$x_j = f(X_j)\,\forall\,\min(x_j) \leq x_j \leq \max(x_j),\ X_j \in [-2, 2], \tag{1}$$

where x_j is the value of variable j, and X_j is the corresponding value within the range $[-2,\ 2]$, which includes the normal operation range $[-1,\ 1]$ and areas for warnings and alarms. The values X_j are called *linguistic values* because the scaling function is based on the membership functions of fuzzy set systems: values -2, -1, 0, 1 and 2 can be associated to the

ISBN: 978-989-95079-6-8

linguistic labels, e.g.

$$\{very\,low,\; low,\; normal,\; high,\; very\,high\} \quad (2)$$

are defined with membership functions (Fig. 2). The number of membership functions is not limited to five: the values between these integers correspond to finer partitions of the fuzzy set system. Early applications of linguistic equations only used integer values [10].

In the case of polynomial membership definitions,

$$\begin{array}{ll} f_j^- = a_j^- X_j^2 + b_j^- X_j + c_j, & X_j \in [-2, 0), \\ f_j^+ = a_j^+ X_j^2 + b_j^+ X_j + c_j, & X_j \in [0, 2], \end{array} \quad (3)$$

the linguistic level of the input variable j is calculated according to the equation

$$X_j = \begin{cases} 2 & \text{with } x_j \geq \max x_j \\ \frac{-b_j^+ + \sqrt{{b_j^+}^2 - 4a_j^+ (c_j - x_j)}}{2a_j^+} & \text{with } c_j \leq x_j \leq \max(x_j) \\ \frac{-b_j^- + \sqrt{{b_j^-}^2 - 4a_j^- (c_j - x_j)}}{2a_j^-} & \text{with } \min(x_j) \leq x_j \leq c_j \\ -2 & \text{with } x_j \leq \min(x_j). \end{cases} \quad (4)$$

where a_j^-, b_j^-, a_j^+, and b_j^+ are coefficients of the polynomials (3), c_j is the real value corresponding to the linguistic value 0, and x_j is the real value. $\min(x_j)$ and $\max(x_j)$ are the minimum and maximum values of the real data corresponding to the linguistic values -2 and 2.

After the linguistic level of the model output, X_{out}, is calculated using the linguistic equation model, it is converted into a real value of output, x_{out}, using the following equation:

$$x_{out} = \begin{cases} a_{out}^- {X_{out}}^2 + b_{out}^- X_{out} + c_{out} \\ \qquad \text{with } X_{out} < 0 \\ a_{out}^+ {X_{out}}^2 + b_{out}^+ X_{out} + c_{out} \\ \qquad \text{with } X_{out} \geq 0 \end{cases} \quad (5)$$

where a_{out}^-, b_{out}^-, a_{out}^+ and b_{out}^+ are coefficients of the polynomials (3), and c_{out} is the real value corresponding to the linguistic value 0.

The coefficients of the polynomials can be represented by

$$\begin{array}{lcl} a_j^- & = & \frac{1}{2}(1 - \alpha_j^-)\, \Delta c_j^-, \\ b_j^- & = & \frac{1}{2}(3 - \alpha_j^-)\, \Delta c_j^-, \\ a_j^+ & = & \frac{1}{2}(\alpha_j^+ - 1)\, \Delta c_j^+, \\ b_j^+ & = & \frac{1}{2}(3 - \alpha_j^+)\, \Delta c_j^+, \end{array} \quad (6)$$

where $\Delta c_j^- = c_j - (c_l)_j$ and $\Delta c_j^+ = (c_h)_j - c_j$. Membership definitions may contain linear parts if some coefficients α_j^- or α_j^+ equal one.

Membership definitions are determined by the centre point c_j, and the core and support areas, which guarantee that the resulting membership definitions are monotonously increasing functions. An easier way is to define the centre point, the core $[(c_l)_j,\ (c_h)_j]$ and the ratios

$$\begin{array}{l} \alpha_j^- = \frac{(c_l)_j - \min(x_j)}{c_j - (c_l)_j}, \\ \alpha_j^+ = \frac{\max(x_j) - (c_h)_j}{(c_h)_j - c_j}, \end{array} \quad (7)$$

from the range $\frac{1}{3} \ldots 3$, and calculate the support $[\min(x_j),\ \max(x_j)]$. The membership definitions of each variable are configured with five parameters, which can presented in three consistent sets. The working point (centre point) c_j belongs to all these sets, and the others are:

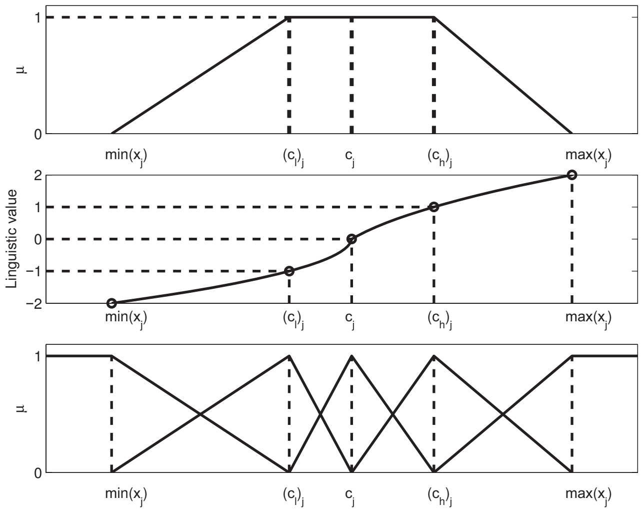


Figure 2: Feasible range, membership definitions and membership functions [2].

- the corner points $\{\min(x_j),\ (c_l)_j,\ (c_h)_j,\ \max(x_j)\}$ are good for visualisation;
- the parameters $\{\alpha_j^-,\ \Delta c_j^-,\ \alpha_j^+,\ \Delta c_j^+\}$ are suitable for tuning;
- the coefficients $\{a_j^-,\ b_j^-,\ a_j^+,\ b_j^+\}$ are used in calculations.

The upper and the lower parts of the scaling functions can be convex or concave, independent of each other. Simplified functions can also be used: a linear membership definition only requires two parameters: c_j and $b_j = b_j^+ = b_j^-$ or $\Delta c_j = \Delta c_j^+ = \Delta c_j^-$, since $\alpha_j^+ = \alpha_j^- = 1$ and $a_j^+ = a_j^- = 0$; an asymmetrical linear definition has $\Delta c_j^+ \neq \Delta c_j^-$ and $b_j^+ \neq b_j^-$.

Interactions. The basic element of a linguistic equation (LE) model is a compact equation

$$\sum_{j=1}^{m} A_{ij} X_j + B_i = 0, \quad (8)$$

where X_j is a linguistic level of the variable j, $j = 1...m$. The direction of the interaction is represented by the interaction coefficients A_{ij}. The bias term B_i was introduced for fault diagnosis systems. A LE model with several equations is represented as a matrix equation

$$AX + B = 0, \quad (9)$$

where the interaction matrix A contains all the coefficients A_{ij} and the bias vector B all the bias terms B_i.

The model is represented by

$$x_{out} = f_{out}\left(-\frac{1}{A_{i\,out}} \Big(\sum_{j=1, j\neq out}^{m} A_{ij}\, f_j^{-1}(x_j) + B_i \Big) \right), \quad (10)$$

ISBN: 978-989-95079-6-8

where the functions f_j and f_{out} are membership definitions. In the general case, the weight factors

$$w_{ij} = -\frac{A_{ij}}{A_{i\ out}}, \tag{11}$$

and the bias term

$$B_i = -\frac{B_i}{A_{i\ out}}, \tag{12}$$

Altogether, there are $n_i + 1$ parameters. As the scaling functions of each variable require three additional parameters for every two parameters needed for normalisation, the total number of additional parameters is $4 + 4n_i$ for n_i input variables.

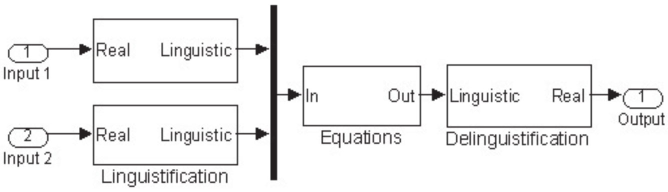


Figure 3: A LE model.

Nonlinear steady-state models can be constructed with linguistic equations, and then extended to dynamic systems using dynamic structures. Case-based systems can include both steady state and dynamic models.

2.2 Takagi-Sugeno type fuzzy systems

Takagi-Sugeno (TS) fuzzy model [4], in which the consequent is a crisp function of the antecedent variables, can be interpreted in terms of local models, see Fig. 4(a). A TS model with a common consequent structure can be understood as a global linear model with input-dependent parameters. For widely used linear functions, the standard weighted mean inference is

$$y = \frac{\sum_{i=1}^{K} \beta_i(\mathbf{x}) y_i}{\sum_{i=1}^{K} \beta_i(\mathbf{x})}, \tag{13}$$

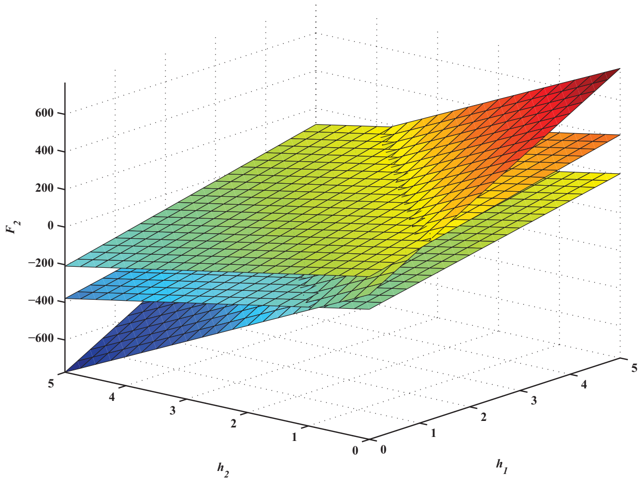
where the degree of fulfillment, $\beta_i(\mathbf{x})$, is the membership degree of the input vector $\mathbf{x}$ in the antecedent of the rule i, see Fig. 4(b). The models surface (Fig. 4(c)) is constructed from the values y_i which are calculated from the local models

$$y_i = \mathbf{a}_i^T \mathbf{x},\ i = 1, 2, \ldots, K. \tag{14}$$

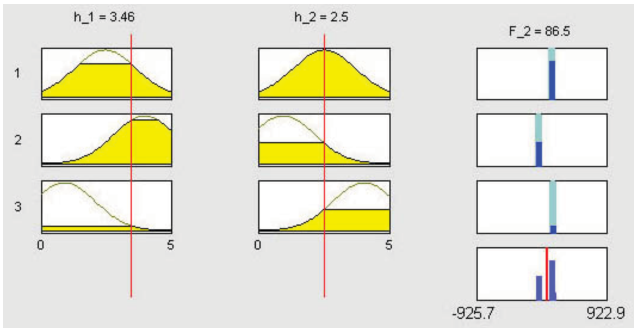
The inference method (13) introduces some undesirable properties in the border areas of the local models if the local models intersect within the border area. These drawbacks can be partly removed by using crisp transitions, $y = \max(y_1, y_2)$ and $y = -\max(-y_1, y_2)$ for a convex and a concave case, respectively. However, the slopes of the local models should differ drastically, and anyway the fuzzy inference is lost and the result is a piece-wise linear approximation. The inference (13) results are much better if the local models intersect outside the border area.

The smoothness of the model is directly dependent on the smoothness of the antecedent membership functions, e.g. the frequently used trapezoidal functions result in nonsmooth ouputs [3].

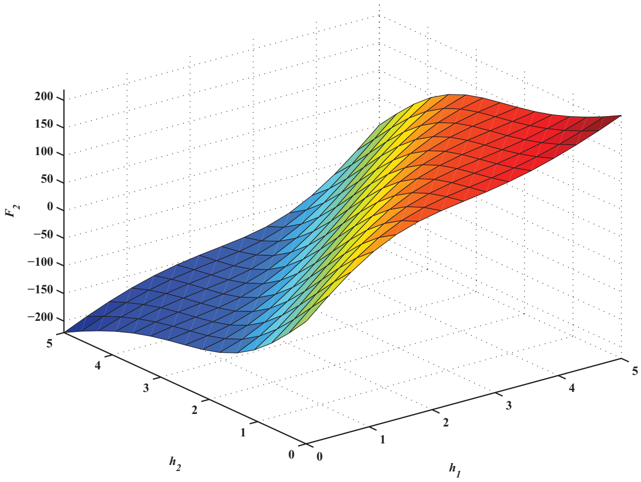
Takagi-Sugeno (TS) type fuzzy models are widely used for the identification of nonlinear systems, since the neuro-fuzzy ANFIS method [7] provides an efficient tuning method for these models. Fitting results are very good with strongly overlapping models, but process insight is lost as the individual models do not have any meaning. The ANFIS tuning increases the overlap of clusters and destroys the meanings of the individual linear models, e.g. the role of some submodels may transform into a part of a smoothing algorithm.



(a) Consequent surfaces.



(b) Fuzzy reasoning.



(c) Model surface.

Figure 4: A Takagi-Sugeno type fuzzy model.

The strong overlap may also result in a steep increase or decrease in the borders of the operating area as can be seen in Fig. 4(c). In dynamic simulation, these steep changes can be alleviated by limiting the range of the output variable, in

ISBN: 978-989-95079-6-8

addition, the operating area of rule 1 should be narrowed down from the solution provided by ANFIS tuning.

Steady-state TS models created with subtractive clustering [11] and ANFIS have proven to be very accurate in the fed-batch enzyme fermentation process [12], however dynamic simulation turned out to be too demanding [13]. Dimension reduction with clustering is necessary when the number of inputs is large. TS models based on grid partitioning require a large number of membership functions and rules which are difficult to tune in practice, e.g. models of the Kappa number in a continuous digester already resulted eight local models with three input variables when two membership functions were used for each variable [14].

The interpolation properties of the TS models can be improved by replacing the weighted mean by mechanisms which result in a piece-wise convex or concave interpolation surface, in which [3]: the gradients are bound by the gradients of the rule consequent functions, and the surface is smooth with continuous derivatives of a sufficient order. Then the model approximates the function more accurately than the surface generated by (13).

For example, the smoothing maximum function [15] connects two consequent hyperplanes, y_1 and y_2, by a smooth convex or concave surface, i.e.

$$y = y_1 + s_\gamma(y_2 - y_1), \qquad (15)$$

where $s_\gamma(z)$ is a piece-wise polynomial

$$s_\gamma = \begin{cases} 0, & z \leq -\gamma, \\ \frac{z+\gamma)^2}{4\gamma} & -\gamma < z < \gamma, \\ z, & z \geq \gamma \end{cases} \qquad (16)$$

The interpolant starts to deviate from the consequent y_1 when $y_2 - y_1 = \gamma$. The smoothing parameter γ, which is a positive real number, is determined for each pair of adjacent rules.

Sharper borders require nonlinear consequent models, i.e. smoothing should be a part of each individual local model.

3 Multimodel LE system

A *multimodel approach* based on fuzzy LE models has been developed for combining specialised submodels [16]. The approach is aimed for systems that cannot be sufficiently described with a single set of membership definitions due to very strong nonlinearities. Additional properties can be achieved because equations and delays can also vary between different submodels. In the multimodel approach, the working area is defined by a separate working point model. The submodels are developed using the case-based modelling approach.

A multimodel system contains several submodels and a fuzzy decision system for selecting a suitable model for each situation using several working point variables. If several inputs are combined into a single working point index, the fuzzy set system is reduced to a fuzzification block (Fig. 5). Linguistic equation (LE) models have been used in several applications [17, 18, 19, 20, 21].

Linguistic Takagi-Sugeno fuzzy models (LTS) belong to this class of models, however with one limitation: the fuzzy partition is defined with same variables as the models. As *LE* models are nonlinear, the local models are also nonlinear.

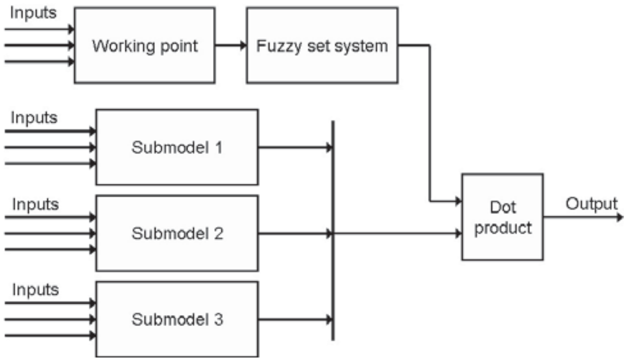


Figure 5: Multimodel LE system with fuzzy decision module.

LTS models can be developed and tuned with the same methods as the normal TS models. The only difference is that the variable values are scaled with the nonlinear scaling functions presented above.

The only difference to the normal LE model is that the equation part is handled with a fuzzy set system (Fig. 6). Nonlinear scaling is done with the same variable specific functions in each local model.

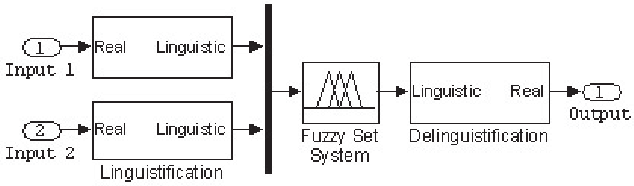


Figure 6: A fuzzy LE simulator.

4 Example: a tank system

The training environment consists of tank systems which include interactive parts where the inflow is divided between two tanks and the flow between the tanks depends on the levels of these tanks, h_1 and h_2, respectively (Fig. 7). The flow F_2 depends on the level of the tanks 1 and 2 (h_1 and h_2):

$$F_2 = c_1\sqrt{h_1 - h_2}, \qquad (17)$$

$$F_3 = c_2\sqrt{h_2}. \qquad (18)$$

The working point is set with the valve properties presented by coefficients c_1 and c_2.

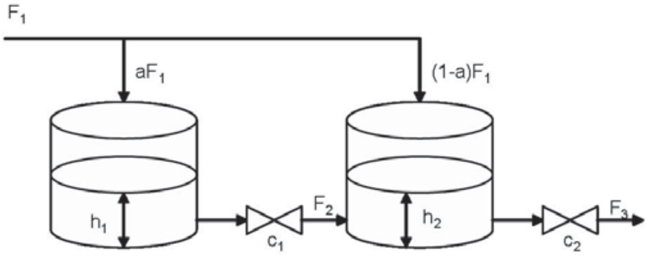
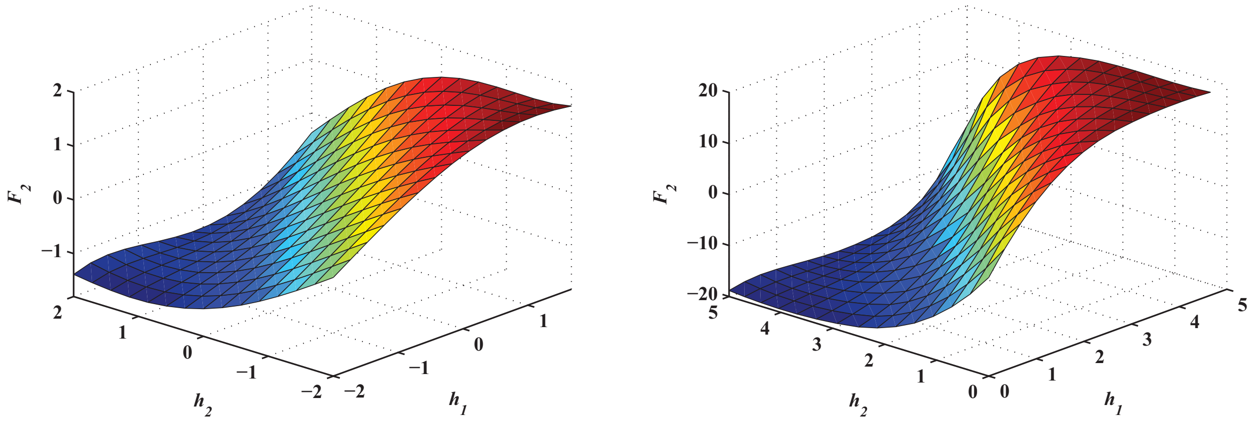


Figure 7: Interactive tank system.

ISBN: 978-989-95079-6-8



(a) The model in the linguistic range.

(b) The model in the real scale.

Figure 8: The model surface of the fuzzy LE model.

TS model. A good fitting can be achieved with a TS model

$$y_1 = 154.2h_1 - 153.9h_2 - 1.98, \quad (19)$$
$$y_2 = 57.61h_1 - 59.86h_2 - 80.08, \quad (20)$$
$$y_3 = 64.35h_1 - 60.18h_2 + 93.15. \quad (21)$$

This model shown in Fig. 4 was tuned with ANFIS resulting in a fairly extensive overlap between the submodels. Strong differences between the submodels can still be observed, but the detection of the models on the basis of their performance is not very clear. These problems were discussed in Section 2.2.

LTS model. A good fitting can be achieved with a LTS model

$$\widetilde{y_1} = 1.704\widetilde{h_1} + 1.785\widetilde{h_2}, \quad (22)$$
$$\widetilde{y_2} = 1.775\widetilde{h_1} + 3.144\widetilde{h_2}, \quad (23)$$
$$\widetilde{y_3} = 1.93\widetilde{h_1} + 5.013\widetilde{h_2}, \quad (24)$$

where the variables h_1 and h_2 are scaled with the nonlinear functions: $\widetilde{h_1} = f_1^{-1}(h_1)$ and $\widetilde{h_2} = f_2^{-1}(h_2)$. The output is first constructed in the linguistic range by using (13) and the submodels $\widetilde{y_1}$, $\widetilde{y_2}$ and $\widetilde{y_3}$. The result is then converted into a real value F_2 by (5) with the scaling parameters of the output variable: a^-_{out}, b^-_{out}, a^+_{out}, b^+_{out} and c_{out}.

The operating areas are clearly observed, and the operation is smooth (Fig. 8(a)). The first consequent model is rather steep and operates when the levels of two tanks are close to each other (Fig. 8(b)). The second model corresponds to negative flow, i.e. $h_2 > h_1$, and the third one to positive flow, $h_1 > h_2$. No smoothing algorithms are needed, and there are no problems in the boundaries of the operating area (Fig. 8(b)). Performance evaluation also provides a clear indication of the active submodels.

5 Applications

LTS models provide good results if the same variables can be used for defining the operating areas and in the submodels. There are clear differences between the applications in this sense.

5.1 Single LE models

A single LE model proved to be a reliable solution for forecasting the Kappa number in continuous cooking with much less parameters than TS models [14]. Also, the steady-state LE model developed in an early lime kiln application provided better results than a TS model since the weaknesses of the data could be clearly observed and corrected in the LE approach [22].

A single LE model can capture nonlinear behaviour so well that the multimodel structure is not needed. The gas furnace data [23] was modelled using a single LE model, which provided very accurate results even in dynamic simulation [24]. The dynamic model of a flotation unit calculates the outlet turbidity from the properties of incoming water, chemical dosages, and previously calculated turbidity. This model is also used as an indicator of water quality [25]. The dynamic model of fluidised bed granulation consists of three interactive models: temperature, humidity, and granule size. All submodels are based on single LE models. [20]

5.2 Multimodels

Dynamic LE multimodels are used for the control design of two application: a lime kiln [10] and a solar collector field [26]. Smooth changes between the submodels were controlled by a fuzzy decision module as in Fig. 5. The lime kiln model is based on six operating areas defined by the production level and the trend of the fuel feed [16]. The interaction matrices are similar, and the differences between the submodels are introduced by scaling functions. The model has been used in a fuel quality indicator, which had an essential part in the development of a successful control system [10]. The model of a solar collector field consists of four specialised LE models, which are specified with interaction coefficients and scaling function parameters. The fuzzy decision module contains a working point model, which is based partly on different variables [24]. The simulator represents the field operation very accurately, even oscillatory conditions are handled correctly [26].

There are forecasting applications for batch cooking [19] and fed-batch fermentation [21]. In batch cooking, specific submodels are needed due to variations in the quality of the

ISBN: 978-989-95079-6-8

chips and the properties of the incoming cooking liquor. Multimodel aspects are taken into account by adapting the model to different operating conditions by selecting an appropriate speed factor depending on the H-factor and alkali level [19]. The dynamic model of fed-batch fermentation consists of three interactive models: carbon dioxide concentration, oxygen transfer rate, and dissolved oxygen concentration [21]. Transitions between the three phases, lag, exponential growth, and steady state, are defined on the basis of time, oxygen transfer rate, and glucose feed rate. The primary aim of the simulator is to detect fluctuations in the process control.

6 Conclusions

Special cases of fuzzy linguistic equation models can be understood as linguistic Takagi-Sugeno (LTS) type fuzzy models, in which nonlinear scaling is used for both inputs and outputs. The LTS models are robust solutions to applications where the same variables can be used for defining the operating areas and also in the submodels. No special smoothing algorithms are needed near the borders of the submodels.

References

[1] L. A. Zadeh. Knowledge Representation in Fuzzy Logic. *IEEE Transactions on Knowledge and Data Engineering*, 1(1):89–100, 1989.

[2] E. K. Juuso. Integration of intelligent systems in development of smart adaptive systems. *International Journal of Approximate Reasoning*, 35:307–337, 2004.

[3] R. Babuška. *Fuzzy Modeling and Identification*. Kluwer Academic Publisher, Boston, 1998.

[4] T. Takagi and M. Sugeno. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics*, 15(1):116–132, 1985.

[5] W. Pedrycz. An identification algorithm in fuzzy relational systems. *Fuzzy Sets and Systems*, 13:153–167, 1984.

[6] R. Babuška and H. Verbruggen. Neuro-fuzzy methods for nonlinear system identification. *Annual Reviews in Control*, 27:73–85, 2003.

[7] J.-S. R. Jang. Anfis: Adaptive-network-based fuzzy inference systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 23(3):665–685, 1993.

[8] P. P. Angelov and D. P. Filev. An approach to online identification of Takagi-Sugeno fuzzy models. *IEEE Transactions on Systems, Man, & Cybernetics*, 34(1):484–498, 2004.

[9] T. A. Johansen and R. Babuška. Multiobjective identification of Takagi-Sugeno fuzzy models. *IEEE Transactions on Fuzzy Systems*, 11(6):847–860, 2003.

[10] E. K. Juuso. Fuzzy control in process industry: The linguistic equation approach. In H. B. Verbruggen, H.-J. Zimmermann, and R. Babuska, editors, *Fuzzy Algorithms for Control, International Series in Intelligent Technologies*, pp. 243–300. Kluwer, Boston, 1999.

[11] S. Chiu. Fuzzy Model Identification Based on Clustering Estimation. *Journal of Intelligent & Fuzzy Systems*, 2(3):Sept., 1994.

[12] U. Saarela, K. Leiviskä, E. Juuso, and A. Kosola. Modelling of a fed-batch enzyme fermentation process. In *IFAC International Conference on Intelligent Control Systems and Signal Processing. Faro, Portugal*. IFAC, 2003.

[13] E. K. Juuso. Dynamic simulation of a fed-batch enzyme fermentation process. In *Proceedings of SIMS 2005, 46th Conference on Simulation and Modeling, Trondheim, Norway*, pp. 117–124. Tapir Academic Press, Trondheim, 2005.

[14] K. Leiviskä, E. Juuso, and A. Isokangas. Intelligent modelling of continuous pulp cooking. In K. Leiviskä, editor, *Industrial Applications of Soft Computing*, pp. 147–158. Springer, Heidelberg, 2001.

[15] I. Zhang. A smoothing-out technique for min-max optimization. *Math. Prog.*, 19:61–77, 1980.

[16] E. K. Juuso. Intelligent dynamic simulation of a lime kiln with linguistic equations. In *ESM'99: Modelling and Simulation: A tool for the Next Millenium, 13th European Simulation Multiconference, Warsaw, Poland*, pp. 395–400, Delft, The Netherlands, 1999. SCS.

[17] E. K. Juuso. Modelling and simulation with intelligent methods. In *White paper of the Virtual Institute for Simulation (Sim-Serv): www.sim-serv.com*. Sim-Serv, 2004. 17 pp.

[18] E. K. Juuso. Applications of smart adaptive system in pulp and paper industry. In *Proceedings of Eunite 2004 - European Symposium on Intelligent Technologies, Hybrid Systems and their implementation on Smart Adaptive Systems, Aachen, Germany*, pp. 21–33. Wissenschaftsverlag Mainz, Aachen, 2004.

[19] E. K. Juuso. Forecasting batch cooking results with intelligent dynamic simulation. In *Proceedings of the 6th EUROSIM Congress on Modelling and Simulation, Ljubljana, Slovenia*, volume 2, page 8 pp. University of Ljubljana, Ljubljana, Slovenia, 2007.

[20] E. K. Juuso. Intelligent modelling of a fluidised bed granulator used in production of pharmaceuticals. In *Conference Proceedings of SIMS 2007 - The 48th Scandinavian Conference on Simulation and Modeling, Göteborg (Särö)*, pp. 101–108. Linkping University Electronic Press, Linkping, Sweden, 2007.

[21] E. K. Juuso. Intelligent dynamic simulation of a fed-batch enzyme fermentation process. In *Tenth International Conference on Computer Modelling and Simulation, EUROSIM/UKSim, Cambridge, UK*, pp. 301–306. The Institute of Electrical and Electronics Engineers IEEE, 2008.

[22] E. K Juuso, T. Ahola, and K. Leiviskä. Fuzzy modelling of a rotary lime kiln using linguistic equations and neuro-fuzzy methods. In *Proceedings of the 3rd IFAC Symposium on Intelligent Components and Instruments for Control Applications - SICICA'97, Annecy, France*, pp. 579–584, 1997.

[23] G. E. P. Box and G. M. Jenkins. *Time Series Analysis, Forecasting and Control*. Holden Day, San Francisco, 1970.

[24] E. K. Juuso. Intelligent systems design with linguistic equations. In *9th Workshop Fuzzy Control des GMA-FA 5.22 am 4/5.11.1999, Dortmund, Deutschland. Forschungsberict Nr. 0449*, pp. 177–196, Dortmund, 1999. Universitt Dortmund, Facultt fur Electrotechnik.

[25] I. Joensuu, M. Piironen, and E. Juuso. Dynamic simulator for dosing of water treatment chemicals. In *Proceedings of European Symposium on Computer Aided Process Engineering-15 (Escape-15), Barcelona, Spain, Computer-aided chemical engineering, 20A*, pp. 301–306. Elsevier, Amsterdam, 2005.

[26] E. K. Juuso. Intelligent dynamic simulation of a solar collector field. In *Simulation in Industry, 15th European Simulation Symposium ESS 2003*, pp. 443–449. SCS, Gruner Druck, Erlangen, Germany, 2003.

ISBN: 978-989-95079-6-8