Available online at www.sciencedirect.com
ScienceDirect
Fuzzy Sets and Systems 157 (2006) 2876–2896
FUZZY sets and systems
www.elsevier.com/locate/fss

Comparison of clustering algorithms in the identification of Takagi–Sugeno models: A hydrological case study

H. Vernieuwe[a], B. De Baets[a,*], N.E.C. Verhoest[b]

[a]*Department of Applied Mathematics, Biometrics and Process Control, Ghent University, Coupure links 653, 9000 Gent, Belgium*
[b]*Laboratory of Hydrology and Water Management, Ghent University, Coupure links 653, 9000 Gent, Belgium*

Received 22 April 2005; received in revised form 28 February 2006; accepted 10 April 2006
Available online 17 May 2006

Abstract

In this paper different clustering algorithms are used to identify Takagi–Sugeno models in a data-driven manner. All but one of these clustering algorithms are based on the minimization of an objective function; the other one is the subtractive clustering algorithm. To guide the objective function-based clustering algorithms, an algorithm called *ClusterFinder* is developed in order to determine the optimal number of clusters as a compromise between model complexity and model accuracy. The hydrological case study considered concerns the modelling of unsaturated groundwater flow. The Takagi–Sugeno models are identified on the basis of an artificially generated training data set for a specific soil type, and can be incorporated into a fuzzy rule-based groundwater model.

Keywords: Fuzzy clustering; Takagi–Sugeno models; Physics; Groundwater flow

1. Introduction

The one-dimensional groundwater flow in the unsaturated zone of the soil is generally modelled by means of a partial nonlinear differential equation: the one-dimensional Richards equation. This equation can only be solved analytically in some special cases [9]. It is then natural to resort to numerical algorithms for approximating the solution of the unsaturated flow equation. Bárdossy et al. [5–7] proposed an alternative solution to the Richards equation in the form of a fuzzy rule-based model. Their goal was to introduce a simpler methodology to calculate the water movement in the soil. Bárdossy et al. [5] used fuzzy rules to model the one-, two- and three-dimensional unsaturated flow. The main advantage is the simplicity of their algorithm and its "small computational requirement". They also state that numerical two- and three-dimensional solutions to the Richards equation are scarce and have a high CPU requirement. Although their fuzzy rule-based solution gives promising results, several disadvantages can be recognized. The method is restricted to triangular fuzzy sets and requires an explicit definition of the support of the antecedent fuzzy sets, indicating that knowledge about the system to model is indispensable, certainly in order to obtain a representative rule base. This need for expert knowledge could, for instance, be bypassed through the use of optimization techniques such as genetic algorithms, demanding an extra effort. Furthermore, the method requires a predetermined number of fuzzy

* Corresponding author. Tel.: +32 9 264 59 41; fax: +32 9 264 62 20.
E-mail addresses: Hilde.Vernieuwe@UGent.be (H. Vernieuwe), Bernard.DeBaets@UGent.be (B. De Baets), Niko.Verhoest@UGent.be (N.E.C. Verhoest).

0165-0114/$ - see front matter
doi:10.1016/j.fss.2006.04.007

sets and hence number of rules. Alternatively, an exhaustive search on the number of rules that yield an accurate fuzzy rule base could be performed.

In this paper, a fuzzy rule-based model with similar rules as those described by Bárdossy et al. [5–7] is used to simulate the one-dimensional unsaturated groundwater flow. The kernel of our fuzzy rule-based groundwater model consists of a first order Takagi–Sugeno model: a flexible fuzzy model that can be identified with a data-driven method, suitable to model nonlinear systems. The antecedent parameters of the Takagi–Sugeno model are identified using fuzzy clustering methods and grid partitioning, whereas the consequent parameters are identified using the global least squares method [3]. A comparison between several clustering methods is made, as well as a comparison between the performance of Takagi–Sugeno models using multi- and one-dimensional (projected) membership functions.

This paper is organized as follows. Section 2 describes the one-dimensional groundwater flow and the structure of the fuzzy rule-based groundwater model. Section 3 explains how the training data set is generated and describes the identification methods used to identify the antecedent and the consequent parameters of the Takagi–Sugeno models. The performance of the models is evaluated by means of two performance indices given in Section 4. Section 5 explains the algorithm *ClusterFinder* developed in order to determine the number of rules of the Takagi–Sugeno models identified with the objective function-based clustering algorithms. The results obtained with the different Takagi–Sugeno models are discussed in Section 6. Section 7 briefly addresses the identification of Takagi–Sugeno models on a gradient-sensitive training data set in order to improve the models. Conclusions and suggestions for further research are formulated in Section 8.

2. Fuzzy rule-based groundwater models

2.1. One-dimensional unsaturated groundwater flow

The water movement in a one-dimensional, isotropic soil matrix is described by the one-dimensional Richards equation [23,9]

$$\frac{\partial\theta}{\partial t} = -\frac{\partial}{\partial z}\left(K\frac{\partial\psi}{\partial\theta}\frac{\partial\theta}{\partial z} + K\right), \tag{1}$$

with θ is the soil moisture content (–), t the time [T], z the gravity head which expresses the elevation [L] of a point w.r.t. the soil surface for which $z = 0$, and is defined positive upward, K is the hydraulic conductivity of the soil [L/T] and ψ is referred to as the matric head [L] which varies with the soil moisture content. The one-dimensional Richards equation (1) is a combination of the law of Darcy describing the water movement in a porous medium

$$q = K\frac{\partial h}{\partial z}, \tag{2}$$

and the continuity or mass balance equation

$$\frac{\partial\theta}{\partial t} + \frac{\partial q}{\partial z} = 0, \tag{3}$$

where q is a volumetric flux or the *Darcy flux* [L/T] defined positive in the downward direction expressing the amount of water flowing downward through a unit area per unit of time and $h = z + \psi$ defines the total head [L].

The matric head ψ and the soil moisture content θ are linked through the van Genuchten equation [13]

$$\theta(\psi) = \theta_r + (\theta_s - \theta_r)\left(\frac{1}{1 + (\alpha\psi)^n}\right)^m, \tag{4}$$

where α, n, m are soil-related parameters (with $m = 1 - 1/n$), θ_s is the saturated moisture content and θ_r is the residual moisture content of the soil.

2.2. Structure of the fuzzy rule-based groundwater model

The fuzzy rule-based model of Bárdossy et al. [5–7] consists of a zero-order Takagi–Sugeno model, built on expert knowledge. However, this type of Takagi–Sugeno models is rarely used. Far more popular are the first-order Takagi–Sugeno models [26], because of their high ability to model nonlinear systems [3,1,14,25]. Several techniques exist to

identify the parameters of a Takagi–Sugeno model without the need for expert knowledge given a training data set. The kernel of our fuzzy rule-based groundwater model consists of a first-order Takagi–Sugeno model with rules of the following type:

$$R_i : \text{IF } \left[\left(\frac{\theta}{\theta_s}\right)_u, \left(\frac{\theta}{\theta_s}\right)_l\right] \text{ IS } A_i \text{ THEN } q_v = a_i \left(\frac{\theta}{\theta_s}\right)_u + b_i \left(\frac{\theta}{\theta_s}\right)_l + c_i \,, \tag{5}$$

with A_i a multi-dimensional membership function and a_i, b_i and c_i the consequent parameters. The subscripts u and l refer to the upper and lower soil cell. If one-dimensional membership functions are used, A_i represents the cartesian product $A_{1,i} \times A_{2,i}$. The Takagi–Sugeno model, either with one- or multi-dimensional membership functions, calculates the flux q_v between two adjacent soil cells with relative moisture contents $(\theta/\theta_s)_u$ and $(\theta/\theta_s)_l$.

The fuzzy rule-based groundwater model, incorporating the Takagi–Sugeno model, updates the moisture content with the simulated fluxes. Given an initial soil profile of moisture contents, the Takagi–Sugeno model calculates the incoming $q_{v,in}$ and outgoing $q_{v,out}$ fluxes for each soil cell with thickness Δz. Using the initial moisture contents θ_{old} and the newly calculated fluxes at time t, the moisture content θ_{new} at time $t + \Delta t$ is calculated using the continuity equation

$$\theta_{new} = \left(\theta_{old}\Delta z + q_{v,in}\Delta t - q_{v,out}\Delta t\right)/\Delta z \,. \tag{6}$$

The newly obtained moisture contents are given as input to the Takagi–Sugeno model to calculate new fluxes and the process restarts.

3. Identification of the Takagi–Sugeno models

3.1. Generation of the training data set

The training data set was synthetically generated using a discretization of the equation of Darcy:

$$q = \sqrt{K_u K_l}\left(\frac{h_u - h_l + \Delta z}{\Delta z}\right) , \tag{7}$$

with K_u and K_l the hydraulic conductivity [L/T], in accordance with the h-value, of the upper and lower soil cell [L], respectively, and Δz, the discretization step [L]. With this equation, only the vertical water movement is considered and the z-axis is chosen positive upward and the positive flux points downward. With the variation of the moisture content, and hence the matric heads (see Eq. (4)), and the calculation of the corresponding flux, the whole domain can be covered with training data points. A discretization step, expressed in relative moisture content θ/θ_s, of 0.0025 was chosen for the variation of the moisture contents. With this discretization step, a data set containing 154 449 data points was obtained. In view of the CPU time that would be required to cluster this data set, the number of data points had to be reduced, and a training data set of 10 000 data points was randomly selected out of the larger data set (see Fig. 1). This training data set was generated for a sandy loam soil used in an experimental set-up by the nonvegetated terrain (NVT) workgroup of the European Microwave Signature Laboratory (EMSL), Joint Research Centre of the European Community, Ispra (Italy), which aimed at the investigation of the use of active microwave observations to estimate volumetric soil moisture content. A detailed description of the experiment and its results is given by Mancini et al. [19] and Hoeben and Troch [16]. The saturated and residual moisture content for this soil θ_s and θ_r are, respectively, 0.55 (–) and 0.01 (–), α is 0.048/cm, n is 1.5632 (–) and m is 0.3603 (–). The saturated hydraulic conductivity K_s of this soil is 5 cm/h.

3.2. Identification of the antecedent parameters

3.2.1. Fuzzy clustering algorithms

Several clustering algorithms are used to identify the antecedent parameters. The popular fuzzy c-means (FCM) and the Gustafson–Kessel (GK) clustering algorithms [15], the simplified Gustafson–Kessel (SGK) clustering algorithm [17], the Gath–Geva (GG) clustering algorithm [12], the simplified Gath–Geva (SGG) clustering algorithm [17], the modified Gath–Geva (MGG) clustering algorithm [1] and the subtractive clustering (SC) algorithm [10] are applied. The latter clustering algorithm is implemented using ANFIS. All but the subtractive clustering algorithm can be categorized

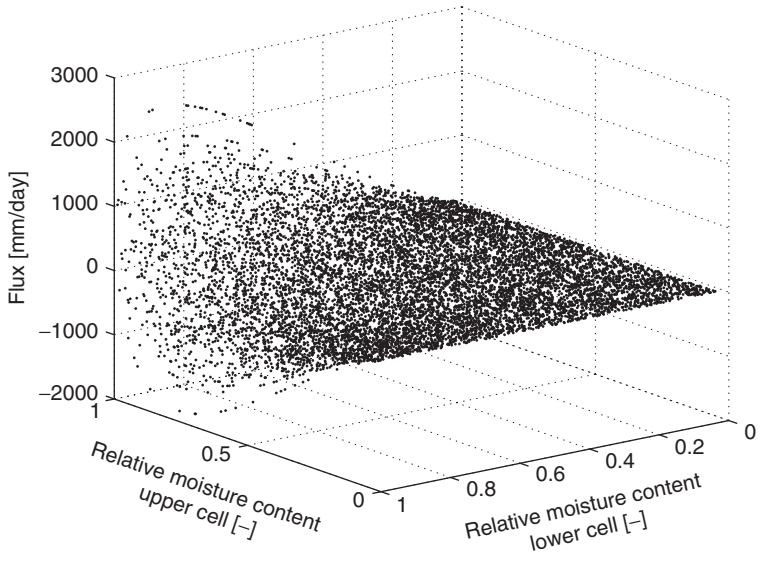

Fig. 1. Training data set.

into the group of the objective function-based clustering algorithms, since they all try to minimize the same objective function:

$$J = \sum_{k=1}^{N} \sum_{i=1}^{c} u_i(\mathbf{z}_k)^m \cdot d(\mathbf{z}_k, \mathbf{c}_i)^2 \,. \tag{8}$$

This objective function can be seen as a generalization of the squared error with N the number of data points, $d(\mathbf{z}_k, \mathbf{c}_i)$ a measure for the distance between data vector $\mathbf{z}_k$ and cluster centre $\mathbf{c}_i$ and $u_i(\mathbf{z}_k) \in [0, 1]$ the membership degree of data vector $\mathbf{z}$ to cluster C_i. The exponent $m \in \mathbb{R}_{>1}$, called *fuzzifier*, represents an additional parameter. A common choice of the fuzzifier is $m = 2$ [17,3] and this value will be used throughout this paper. The objective function given in Eq. (8) can be augmented as follows:

$$J = \sum_{k=1}^{N} \sum_{i=1}^{c} u_i(\mathbf{z}_k)^m \cdot d(\mathbf{z}_k, \mathbf{c}_i)^2 + \sum_{i=1}^{c} \eta_i \sum_{k=1}^{N} (1 - u_i(\mathbf{z}_k)^m) \,, \tag{9}$$

with η_i a positive cluster-dependent parameter. Clustering algorithms that try to minimize this extended objective function are called possibilistic clustering algorithms [18] as they do not result in a partition matrix that satisfies $\sum_{i=1}^{c} u_i(\mathbf{z}_k) = 1$. Minimizing Eq. (8), however, implies an accurate estimation of the parameters η_i. Moreover, considering this more general objective function for each of the clustering algorithms mentioned above would be infeasible and render the exposition ill-digested.

All these objective function-based clustering algorithms have an iterative outline as shown in Algorithm 1. The iteration ends when the maximal difference between the partition matrices, which store the membership degree of the data points to the clusters, of the previous and the current iteration becomes smaller than a predetermined tolerance value ε:

$$\|U^{(l)} - U^{(l-1)}\| < \varepsilon. \tag{10}$$

The major difference between these clustering algorithms lies in the calculation of the distance of a data point to a cluster centre. The fuzzy c-means clustering algorithm uses the Euclidean distance, indicating that data points with the same distance to the cluster centre are situated on a sphere. The Gustafson–Kessel clustering algorithm uses covariance matrices in calculating the distance between a data point and a cluster centre. Data points with the same distance to the cluster centre are then situated on an ellipsoid. In the Gath–Geva clustering algorithm, the data are considered to be realizations of normally distributed random variables. This clustering algorithm is equivalent to a Gaussian mixture

model that expands a probability density function into a sum over c clusters [1]. The distance of a data point to a cluster centre is then calculated as the reciprocal of an unnormalized Gaussian function. The simplified Gustafson–Kessel and the simplified Gath–Geva clustering algorithms are simplifications of the original GK and GG clustering algorithms in the sense that they only permit axes-parallel clusters, i.e. the covariance matrices have off-diagonal zero elements. As the modified Gath–Geva algorithm has been specifically developed to identify Takagi–Sugeno models, it is not surprising that the identification of the consequent parameters has been incorporated into it.

Algorithm 1: Outline of an objective function-based clustering algorithm.

Data : Data set Z
Number of clusters $1 < c < N$
The weighting exponent $m > 1$
The termination tolerance $\varepsilon > 0$

Result : Cluster centres $\mathbf{c}_i$
Other cluster parameters: covariance matrices, a priori probabilities, consequent parameters (for the MGG algorithm), *etc.*

Initialize partition matrix $U^{(0)}$
while $\|U^{(l)} - U^{(l-1)}\| \geqslant \varepsilon$ **do**
  Compute cluster prototypes $\mathbf{c}_i$
  Compute other cluster parameters
  Compute the distance between data points $\mathbf{z}_k$ and cluster centres $\mathbf{c}_i$
  Update the partition matrix
end

The subtractive clustering algorithm is a modification of the Mountain Method of Yager and Filev [28] and was introduced by Chiu [10]. While the Mountain Method constructs a grid on the data space and considers each grid point as a possible cluster centre, the subtractive clustering algorithm regards each data point as a possible cluster centre. The advantage of the subtractive clustering algorithm over the above described clustering algorithms is the fact that the number of clusters does not need to be specified in advance. However, four other parameters, the cluster radius r_a, the squash factor η and the accept and reject ratio $\bar{\varepsilon}$ and $\underline{\varepsilon}$, need to be set. For a given data set, the best parameters to be used are not known, therefore, a parameter search is usually required. In this paper, the subtractive clustering algorithm is carried out in the framework of the Adaptive Network-based Fuzzy Inference System or ANFIS in MATLAB, in which Gaussian membership functions are assigned to the input dimensions, and an optimization on the model parameters is performed.

With the objective function-based clustering algorithms, multi-dimensional membership functions are obtained. The membership degree of a data point to the multi-dimensional membership function is calculated as is done in the corresponding clustering algorithm, as a function of the distance to the cluster centre. Since the corresponding output needs to be predicted, and is therefore unknown, only the distance and the membership degree with respect to the input dimensions of the cluster centre can be calculated, i.e. the dimensions of the corresponding matrices, in particular the covariance matrix, reduce with one.

In order to obtain one-dimensional membership functions, the clusters obtained with the objective function-based clustering algorithms are projected onto the variable axes. The projection is mainly performed as it is written in the source code of the Fuzzy Modelling and Identification Toolbox for MATLAB [4] through projection of the partition matrix obtained by each clustering algorithm onto the input variable axes. The resulting projection is filtered by means of a low-pass filter and exponential membership functions are fitted. In the toolbox, a factor 7 is used in the expression for the exponential membership function. This constant factor is replaced by a variable factor β which was, for our applications, altered from 1 to 10, with steps equal to 1. The exponential membership function is then given by

$$A(x) = \begin{cases} \mathrm{e}^{-\beta((x-a_2)^2/(a_2-a_1)^2)} & \text{if } x < a_2, \\ \mathrm{e}^{-\beta((x-a_3)^2/(a_4-a_3)^2)} & \text{if } x > a_3, \\ 1 & \text{otherwise.} \end{cases} \tag{11}$$

The interval $[a_2, a_3]$ is the kernel of the membership function, and a_1 and a_4 are the parameters for which the membership degree equals $e^{-\beta}$.

3.2.2. Grid partitioning

In the grid partitioning method, the domain of each antecedent variable is partitioned into equidistant and identically shaped membership functions. Using the available input–output data, the parameters of the membership functions can be optimized. A major drawback is that the membership functions for each variable are obtained independently of one another, causing the neglect of the relationship between the variables. Another disadvantage lies in the fact that all combinations of antecedent membership functions need to be covered into rules, leading to the so-called curse of dimensionality. The models identified in this paper on the basis of this grid partitioning of antecedent variables use triangular, trapezoidal, symmetric and asymmetric Gaussian, generalized bell shaped, Π-shaped and the difference and the product of two sigmoidal membership functions. The grid partitioning method is implemented using ANFIS in MATLAB. The initial partition results in a so-called Ruspini-partition [24]. However, during the optimization, it is not guaranteed that the Ruspini-partition is preserved.

3.3. Identification of the consequent parameters

A Takagi–Sugeno model consisting of n rules can be seen as a weighting of n 'regression models'. Therefore, its consequent parameters can be estimated by means of a least squares method taking into account the respective weighting parameters β, which are function of the degrees of fulfillment. The calculation of the output values by means of a Takagi–Sugeno model can be expressed in matrix notation:

$$O = I \times P\,, \tag{12}$$

with

$$O = [y_1 \cdots y_N]^{\mathrm{T}} = \begin{bmatrix} \beta_{1,1}x_{1,1} & \cdots & \beta_{1,1}x_{1,p} & \beta_{1,1} & \cdots & \beta_{1,n}x_{1,1} & \cdots & \beta_{1,n}x_{1,p} & \beta_{1,n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \beta_{N,1}x_{N,1} & \cdots & \beta_{N,1}x_{N,p} & \beta_{N,1} & \cdots & \beta_{N,n}x_{N,1} & \cdots & \beta_{N,n}x_{N,p} & \beta_{N,n} \end{bmatrix} \times \begin{bmatrix} a_{1,1} \\ \vdots \\ a_{1,p} \\ b_1 \\ \vdots \\ a_{n,1} \\ \vdots \\ a_{n,p} \\ b_n \end{bmatrix}, \tag{13}$$

where $\beta_{k,j} = w_j(\mathbf{x}_k)/\sum_{i=1}^{n} w_i(\mathbf{x}_k)$ are the normalized membership degrees, with $k = 1, \ldots, N$; $j = 1, \ldots, n$ with N the number of data points and n the number of rules. Using $\beta_{k,j}$ in the matrix I, the weighting of the rules is taken into account, and a minimal global prediction error is obtained. The parameters P are then estimated by

$$P = [I^{\mathrm{T}} I]^{-1} I^{\mathrm{T}} O\,. \tag{14}$$

This method is also referred to as the global least squares method [3]. Alternatively, the recursive least squares method [2] which bypasses the problem of singularity of $[I^{\mathrm{T}} I]$ can be used. The recursive least squares method is standardly used by ANFIS.

4. Performance indices for Takagi–Sugeno models

The accuracy of the Takagi–Sugeno models is evaluated using two performance indices. The first index is the Nash and Sutcliffe (NS) performance index [20], widely used in hydrology:

$$\mathrm{NS} = 1 - \frac{\sum_{k=1}^{N}(y_m(k) - y(k))^2}{\sum_{k=1}^{N}(y(k) - \overline{y})^2}, \tag{15}$$

with N the number of data points, y_m and y the modelled and observed output, and $\overline{y}$ denotes the mean of the observed values. The optimal value of NS is 1, meaning a perfect match of data and model. A value of zero indicates that the model predictions are as good as that of a 'no-knowledge' model continuously simulating the mean of the observed signal. Negative values indicate that the model is performing worse than this 'no-knowledge' model [8]. The second performance index is the root mean square error (RMSE):

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{k=1}^{N}(y(k) - y_m(k))^2}{N}}, \tag{16}$$

with N the number of data points and y_m and y the modelled and observed output. The RMSE can have values $\geqslant 0$ with an optimal value of 0, indicating a perfect match of data and model.

5. Determination of the number of clusters: algorithm *ClusterFinder*

5.1. Preliminary analysis

The number of clusters was initially varied between 2 and 64 for the two popular objective function-based clustering algorithms: the fuzzy c-means and the Gustafson–Kessel clustering algorithms. Since these clustering algorithms depend on a random initialization, 30 models corresponding to different initializations were identified for each number of clusters. The behaviour of the performance on the training data set, with respect to an increasing number of rules or complexity, is given in Fig. 2 for the models identified with the Gustafson–Kessel clustering algorithm. A similar behaviour is obtained for the models identified with the fuzzy c-means clustering algorithm.

This figure shows that an obvious relationship between the accuracy and the complexity of the models exists. The complexer the model, the more accurate its performance. Furthermore, adding an extra cluster strongly improves the value of the NS- or RMSE-index for a low number of clusters, whereas the improvement due to an additional cluster becomes less apparent for a higher number of clusters. Determining the number of clusters with an acceptable accuracy and complexity signifies that a compromise between an increasing number of clusters, i.e. a more complex model, and the accuracy of the model needs to be sought. This is performed by our algorithm *ClusterFinder*. Given the fact that all applied clustering algorithms requiring a predetermined number of clusters try to minimize the same objective function and furthermore have the same algorithmic structure, a similar behaviour with respect to the number of rules is expected.

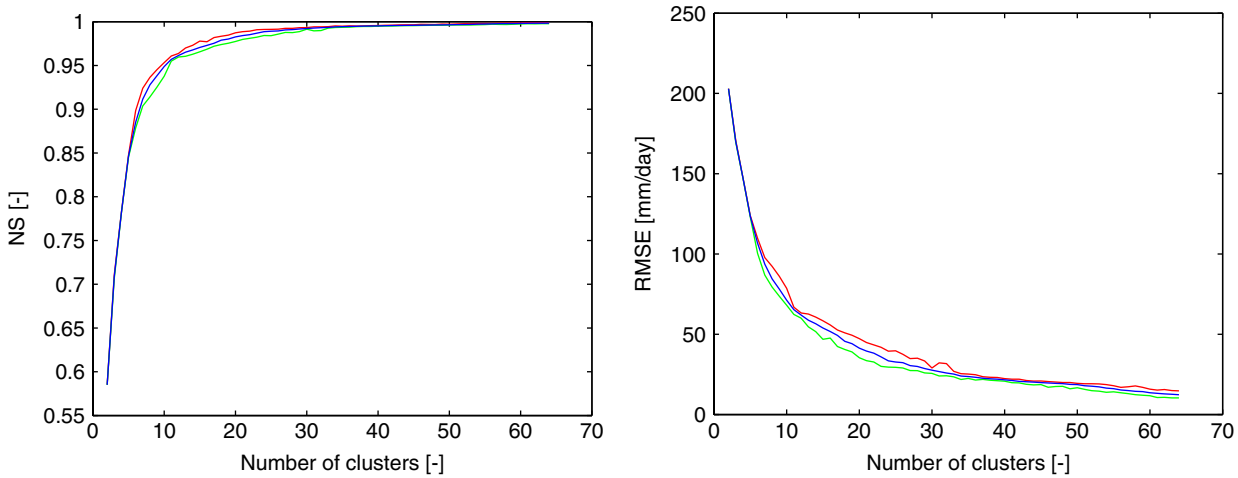

Fig. 2. Plots of the maximal, mean and minimal values of the performance indices for the models with multi-dimensional membership functions based on the Gustafson–Kessel clustering algorithm.

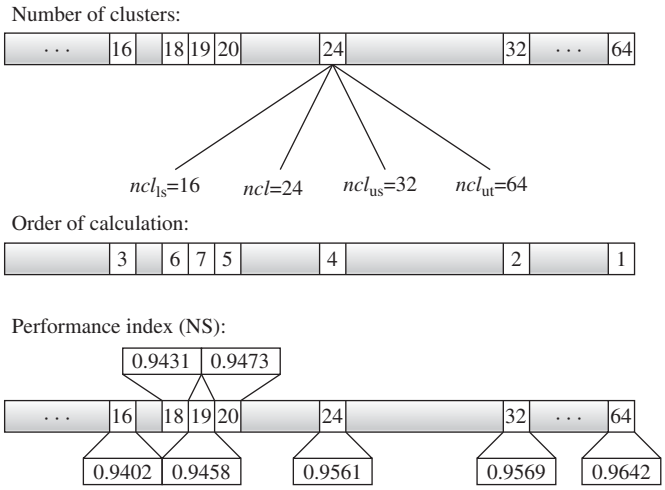

Fig. 3. Example of a pathway followed by *ClusterFinder*.

5.2. Description of the algorithm

A schematic overview of a pathway followed by the algorithm is shown in Fig. 3. This figure shows the number of clusters for which models were identified, together with the corresponding performance index obtained by these models. The order in which models were identified is shown as well. In the case of the model identification for 24 clusters, the corresponding boundaries of the search interval (see further) are shown together with the values of *ncl* and ncl_{ut}. The algorithm itself is presented in Algorithm 2. This heuristic algorithm searches for the optimal number of clusters, and is applicable to all of the fuzzy clustering algorithms. *ClusterFinder* starts, in this particular case, with 64 clusters (denoted as *nclmax*). A margin α by which the obtained maximal accuracy of the model with 64 clusters can be decreased and for which the accuracy can still be seen as acceptable has to be determined first. This margin can be seen as the price we are ready to pay (in terms of the considered performance index) for not doubling the complexity of the model. The algorithm then establishes the boundary values of the total interval and the search interval. The boundary values of the total interval are determined by identifying Takagi–Sugeno models for *nclmax* and $ncl = nclmax/2$ clusters. Normally, the Takagi–Sugeno models for *nclmax* yield the best performance and hence *nclmax* is denoted as the upper boundary of the total interval ncl_{ut}. Still, if a Takagi–Sugeno model with a higher performance can be found for *ncl* clusters, the upper boundary of the total interval is changed to *ncl* and the new number of clusters *ncl* then becomes $ncl/2$. The search for the upper boundary of the total interval is then continued following the same method until no better Takagi–Sugeno models are found. The number of clusters for which this occurs is set to ncl_{ut}. In this way, ncl_{ut} results in a power of two. Therefore, it is important that the initially chosen *nclmax* is an integer power of two. The lower boundary of the total interval is permanently set at 0. The boundaries of the search interval ncl_{us} and ncl_{ls} initially equal the boundaries of the total interval ncl_{ut} and ncl_{lt}, respectively. *ncl* is situated in the middle of the search interval.

The remaining part of the algorithm narrows the boundaries of the search interval according to the difference in maximal performance between the models with *ncl* and ncl_{ut} number of clusters. If a difference in performance of at least α is obtained, i.e. the model with *ncl* number of clusters has a lower accuracy than the minimal allowed value, the lower boundary of the search interval is changed to $ncl_{ls} := ncl$. The considered number of clusters *ncl* is changed to the centre of the new search interval. If a difference smaller than α is obtained, i.e. the model with *ncl* number of clusters has a higher accuracy than the minimal allowed value, a model with lower complexity and similar accuracy can possibly be present. Therefore, the upper boundary of the search interval ncl_{us} is changed to *ncl*. The new current number of clusters *ncl* is then situated in the middle of the new search interval. These steps are iterated until the length of the search interval equals one.

The proposed algorithm acts on the basis of the NS index. An algorithm based on the RMSE index differs in this way that the number of clusters yielding a model with a minimal value of the RMSE index fitting within the margin

α is sought. The algorithm will be based on minimum values of the performance index instead of maximum values. Apart from that, the algorithm proceeds analogously.

Algorithm 2: *ClusterFinder*

Input : data set Z
initial number of clusters *nclmax*
number of repetitions *nrep*
margin α

Output: fuzzy rule base
number of clusters (*ncl*)
matrix containing the values of the performance indices (NS)

for $i \rightarrow 1$ ***to*** *nrep* **do**
  Build the fuzzy rule base for *nclmax* and simulate the data set Z
  Calculate NS$(i, nclmax)$
  Build the fuzzy rule base for $ncl \leftarrow nclmax/2$ and simulate the data set Z
  Calculate NS$(i, nclmax/2)$
end
$diff \leftarrow \max(\text{NS}(., nclmax)) - \max(\text{NS}(., nclmax/2))$
while $diff < 0$ **do**
  for $i \leftarrow 1$ ***to*** *nrep* **do**
    Build the fuzzy rule base for $ncl \leftarrow ncl/2$ and simulate the data set Z
    Calculate NS(i, ncl)
  end
  $diff \leftarrow \max(\text{NS}(., ncl * 2)) - \max(\text{NS}(., ncl))$
end
$ncl_{\text{ut}} \leftarrow 2 * ncl$
$ncl_{\text{us}} \leftarrow 2 * ncl$
$ncl_{\text{ls}} \leftarrow 0$
while $(ncl_{\text{us}} - ncl) > 1$ **do**
  if $diff \geqslant \alpha$ **then**
    $ncl_{\text{ls}} \leftarrow ncl$
    $ncl \leftarrow ncl + (ncl_{\text{us}} - ncl)/2$
  else
    $ncl_{\text{us}} \leftarrow ncl$
    $ncl \leftarrow ncl - (ncl - ncl_{\text{ls}})/2$
  end
  for $i \leftarrow 1$ ***to*** *nrep* **do**
    Build the fuzzy rule base for *ncl* and simulate the data set Z
    Calculate NS(i, ncl)
  end
  $diff \leftarrow \max(\text{NS}(., ncl_{\text{ut}})) - \max(\text{NS}(., ncl))$
end
if $diff \geqslant \alpha$ **then**
  $ncl \leftarrow ncl + 1$
end
if $diff < \alpha$ **then**
  $ncl \leftarrow ncl$
end

6. Results

6.1. Clustering algorithms based on an objective function

The fuzzy *c*-means (FCM), Gustafson–Kessel (GK), Gath–Geva (GG), simplified Gustafson–Kessel (SGK), simplified Gath–Geva (SGG) and modified Gath–Geva (MGG) clustering algorithms are employed together with the global least squares method to identify Takagi–Sugeno models on the training data set. These fuzzy models are then incorporated into a fuzzy rule-based groundwater model. Babuška [3] and Höppner et al. [17] refer to the fact that the data scale may have an influence on the performance of the clustering algorithms. Therefore, the training data set is additionally standardized:

$$x^* = \frac{x - \bar{x}}{s}, \tag{17}$$

with $\bar{x}$ and s the mean and standard deviation of the training data set. Hence, two training data sets are used, referred to as the original and standardized set. The *ClusterFinder* algorithm is used to determine the optimal number of clusters. Based on the results obtained with the FCM and GK models, a margin α of 0.02 was chosen, since the complexity of the models is sufficiently reduced, while the accuracy compared to the model with 64 rules can still be seen as acceptable. Models with the obtained clusters, resulting in multi-dimensional membership functions, and models with the projections of these clusters onto the variable axes, resulting in one-dimensional membership functions, are identified. For the objective function-based clustering algorithms the fuzziness exponent $m = 2$ and the tolerance value $\varepsilon = 0.001$ are used.

6.1.1. Multi-dimensional membership functions

The obtained number of clusters and the performance of the corresponding models with 64 rules on the training data set, indicated by NS_{64} and $RMSE_{64}$, are listed in Table 1. The subscript s refers to the results obtained with models identified on the standardized set. The values in this table show that the worst values both in terms of NS and RMSE are obtained with the fuzzy *c*-means (FCM) clustering algorithm. The NS- and RMSE-values corresponding to the other clustering algorithms are more or less similar. The best obtained values for the optimal number of clusters are given by the simplified Gath–Geva algorithm (SGG_s). The Gustafson–Kessel (GK_s) algorithm yields the best values for the model with 64 clusters.

Fig. 4 illustrates the error surface on the training data set for the SGG_s models with 33 clusters and for the GK_s model with 64 clusters. The error surfaces obtained with the model with 33 clusters exhibit large errors at the borders corresponding to high moisture contents. The presence of those large errors is due to the steepness of the slope of the training surface in these areas. The error surface of the model with 64 clusters still shows some larger errors at those borders, compared to the errors at the remaining parts of the surface. However, these errors are considerably

Table 1
Number of clusters obtained with *ClusterFinder* for the different clustering algorithms and the performance of the corresponding models.

Clustering method	#Clusters	NS (–)	RMSE (mm/day)	NS_64 (–)	$RMSE_{64}$ (mm/day)
FCM	19	0.9458	73.3523	0.9642	59.5813
FCM_s	18	0.9600	62.9959	0.9796	44.9775
GK	16	0.9794	45.1877	0.9989	10.4480
GK_s	17	0.9814	42.9116	**0.9990**	**10.0465**
SGK	17	0.9735	51.2496	0.9921	27.9158
SGK_s	17	0.9729	51.8817	0.9920	28.2636
GG	22	0.9809	43.5188	0.9987	11.3245
GG_s	22	0.9819	42.3681	0.9985	12.0260
SGG	33	0.9791	45.5201	0.9964	18.9872
SGG_s	33	**0.9833**	**40.6644**	0.9960	20.0416
MGG	37	0.9768	48.0146	0.9953	21.6053
MGG_s	36	0.9765	48.3396	0.9950	22.3502

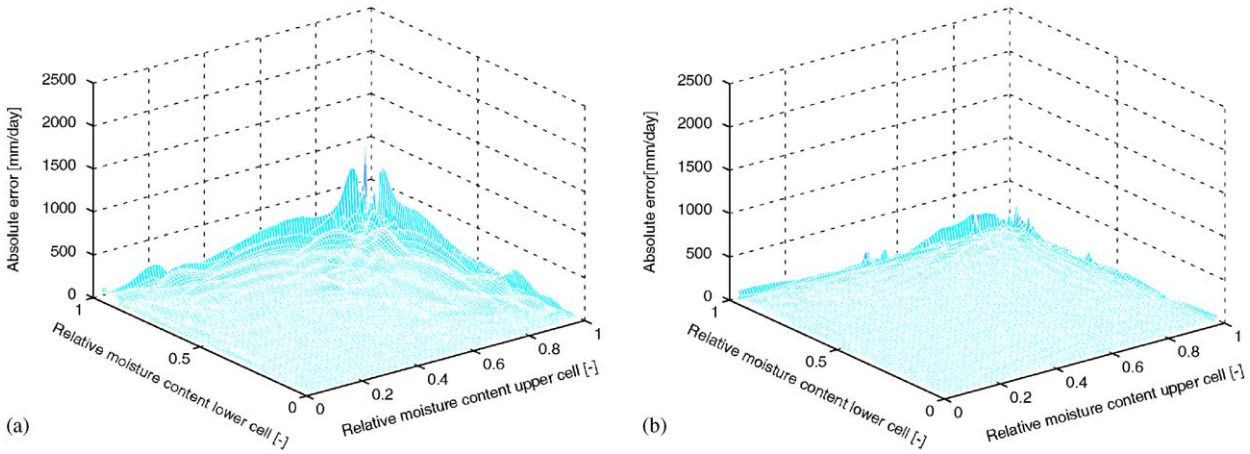

Fig. 4. Errors on the training data set for the model with 33 clusters identified with the simplified Gath–Geva clustering algorithm (a) and the model with 64 clusters identified with the Gustafson–Kessel clustering algorithm (b).

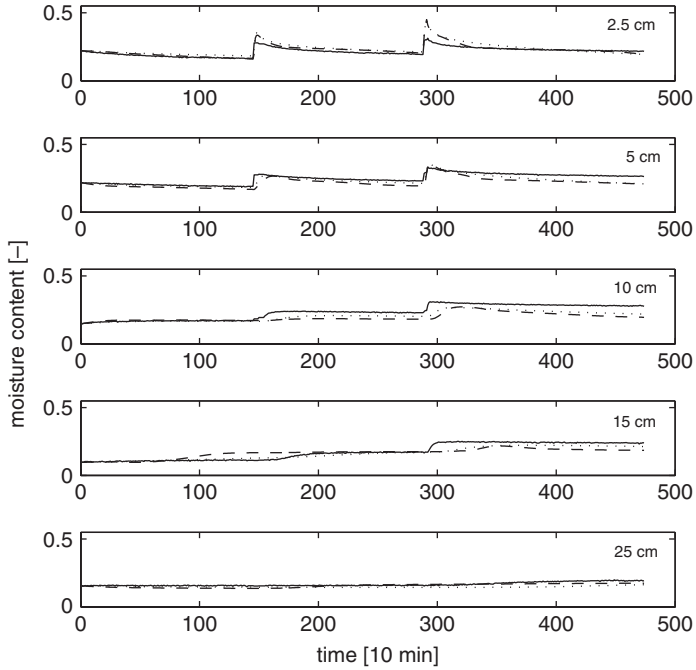

Fig. 5. Measurements (solid lines) and simulation results of the SGG_S (dashed lines) and the numerical model (dotted lines).

reduced compared to those of the optimal model. Furthermore, the differences in NS- and RMSE-values between the model with 33 and 64 rules do not reflect the difference in the corresponding error surfaces. A more complex model smooths the error surface and is able to reduce the errors at the borders of high moisture content. Still, $\alpha = 0.02$ seems a reasonable choice to obtain a less complex model with a sufficiently high performance since the transition between moisture contents in a soil profile is mainly gradual. The location of the moisture content combinations on the training surface, during time series simulation, will mainly be around the diagonal from the (0,0) corner to the (1,1) corner. For all models, low (or almost no) errors are found at the regions with lower or moderate moisture contents.

Fig. 5 shows simulated moisture contents at several depths throughout a soil profile, obtained with the SGG_s model with 33 rules. The measurements of these moisture contents were obtained on a sandy loam soil during an experiment set up by the nonvegetated terrain (NVT) workgroup of the European Microwave Signature Laboratory (EMSL), Joint

Research centre of the European Community in Ispra (Italy) [19,16]. Results obtained by the numerical model of Hoeben and Troch [16] using the same initial and boundary conditions are plotted as well. This figure shows that a good agreement between the numerical and fuzzy rule-based model is obtained, and that both models simulate the measurements quite well. Similar results as shown for the SGG_s model were obtained with the other fuzzy models discussed in the remainder of the paper.

Additionally, contour plots of the multi-dimensional membership functions and corresponding contours of error surfaces for the optimal models obtained on the original training data set are given in Figs. 6 and 7, respectively. The contour plots of models obtained on the standardized training data are similar except for models obtained with the fuzzy *c*-means algorithm (Fig. 8). The cluster centra of these models obtained on the original training data set are located on two curves, whereas the clusters for models obtained with the standardized training data set are more equally spread over the original domain. This indicates that the location of the clusters with the fuzzy *c*-means algorithm is affected by the data scale, whereas the other clustering algorithms are not or less influenced by the scale. A connection between the positioning of the clusters and the contour lines in the error surfaces exists: generally higher errors are found in between two clusters. Furthermore, Fig. 6 clearly shows that the clusters obtained with the Gath–Geva, the simplified Gath–Geva and the modified Gath–Geva algorithm are more closely connected, i.e. only a few places in the domain exist for which the maximum membership value is lower than 0.5, whereas the other clustering algorithms exhibit more areas with a membership value of 0.5 and less (i.e. the areas outside the cluster contours).

6.1.2. Projected membership functions

A second way of identifying Takagi–Sugeno models is performed by projecting the multi-dimensional clusters, obtained by the different clustering methods. In this way, one-dimensional membership functions were obtained. The degree of fulfilment of a rule is then obtained by multiplying the membership degrees of the considered data point to the antecedent fuzzy sets.

ClusterFinder was employed with a margin $\alpha = 0.02$ to obtain the number of rules. Table 2 lists the number of rules and the corresponding factor β yielding the best values of the performance indices. The values of the performance indices obtained for the corresponding models with 64 rules for the same β-values are listed as well. Models having their origin in the fuzzy *c*-means clustering algorithm show a remarkably better performance than the corresponding models with multi-dimensional membership functions. The optimal model obtained with the modified Gath–Geva algorithm (MGG) shows the best results for both NS and RMSE values. The model with 64 rules, identified with the Gustafson–Kessel algorithm, shows the best results in terms of NS and RMSE (values indicated in bold).

Similar error surfaces as for the models with multi-dimensional membership functions are obtained for the models identified with the modified Gath–Geva algorithm (MGG) and the Gustafson–Kessel algorithm (GK) for the optimal number of rules and 64 rules.

6.1.3. Subtractive clustering

As described in Section 3.2.1, an advantage of the subtractive clustering algorithm is that the algorithm itself determines the number of clusters. However, four other parameters (η, r_a, $\bar{\varepsilon}$ and $\underline{\varepsilon}$), influencing the number of rules and hence the accuracy, have to be determined. Chiu [10] suggested a value $\eta = 1.5$. Often, $0.15 \leqslant r_a \leqslant 0.3$ [11], $\bar{\varepsilon} = 0.5$, and $\underline{\varepsilon} = 0.15$ [10,22] and [21] are used. Depending on the data, other parameter combinations can lead to better models, therefore a parameter search is performed to identify the values of the parameters yielding a fuzzy model with desired accuracy and complexity. The subtractive clustering algorithm is carried out with the ANFIS tool in MATLAB. The parameters of the membership functions obtained with the subtractive clustering algorithm are then further optimized by means of a neural network, whereas the consequent parameters are calculated with the recursive least squares method.

Initially, the values of each of the four parameters were varied with steps of 0.1 between their imposed minimal and maximal values ([0.1, 1] for r_a, $\bar{\varepsilon}$ and $\underline{\varepsilon}$ and [0.5, 2] for η) while the values of the other three parameters were fixed at their minimal value. The number of training epochs was set at 150, since the decrease of the training error at this number of epochs became negligible. The number of rules obtained for the respective parameter values are given in Fig. 9. As can be seen from this figure, a large influence is observed for η and r_a, whereas a lower influence is observed for the accept ratio $\bar{\varepsilon}$ and the reject ratio $\underline{\varepsilon}$ apparently has no influence. Low values of all parameters result in models with a very high (ca. 1000) number of rules. Higher values of η and r_a are able to decrease this number considerably. As a high number of rules is not desirable with respect to computing and evaluation time, and as it is convenient to

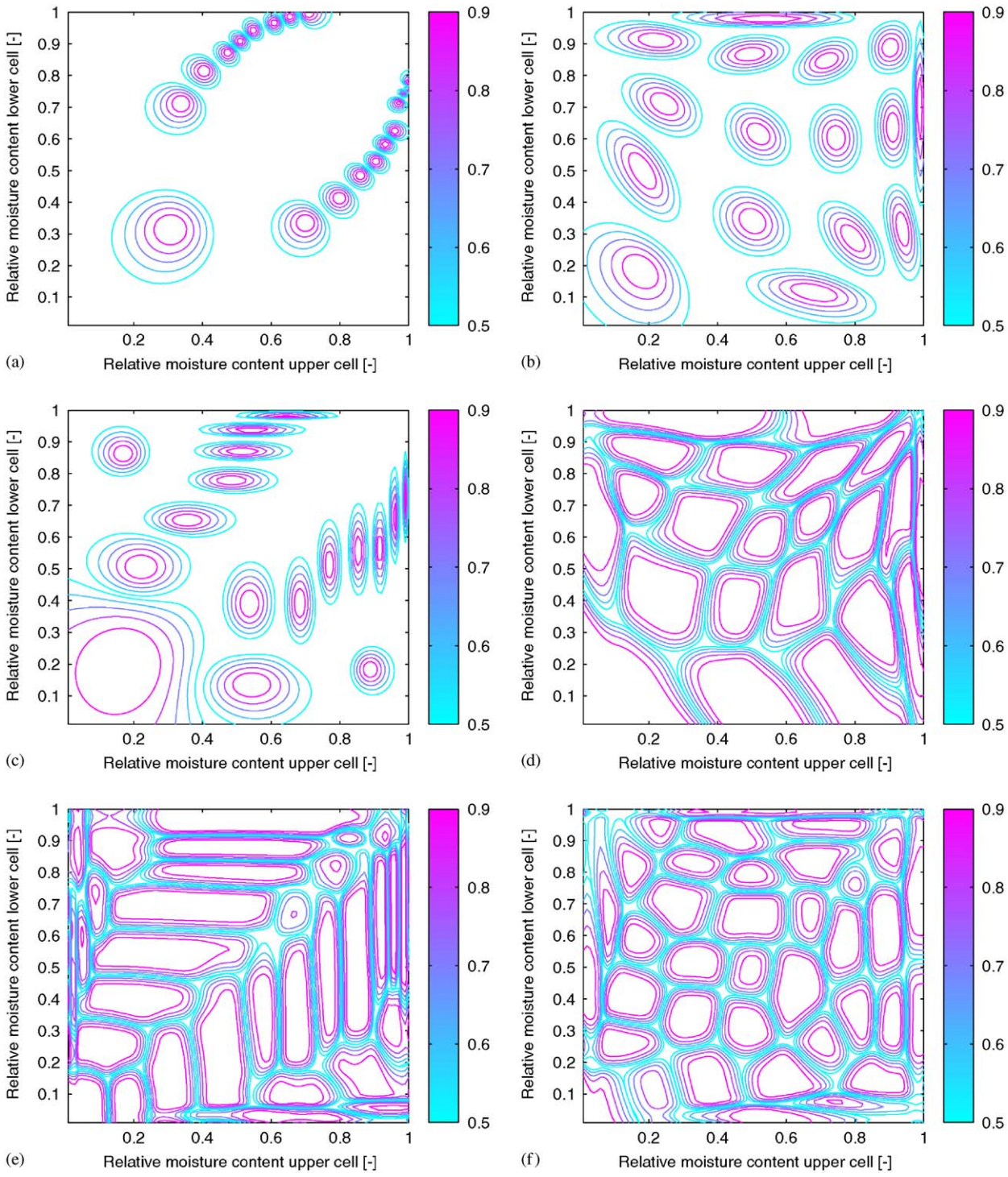

Fig. 6. Contour plots of the multi-dimensional membership functions obtained with (a) FCM, (b) GK, (c) SGK, (d) GG, (e) SGG and the (f) MGG clustering algorithms. Contour lines are plotted for membership degrees from 0.5 to 0.9.

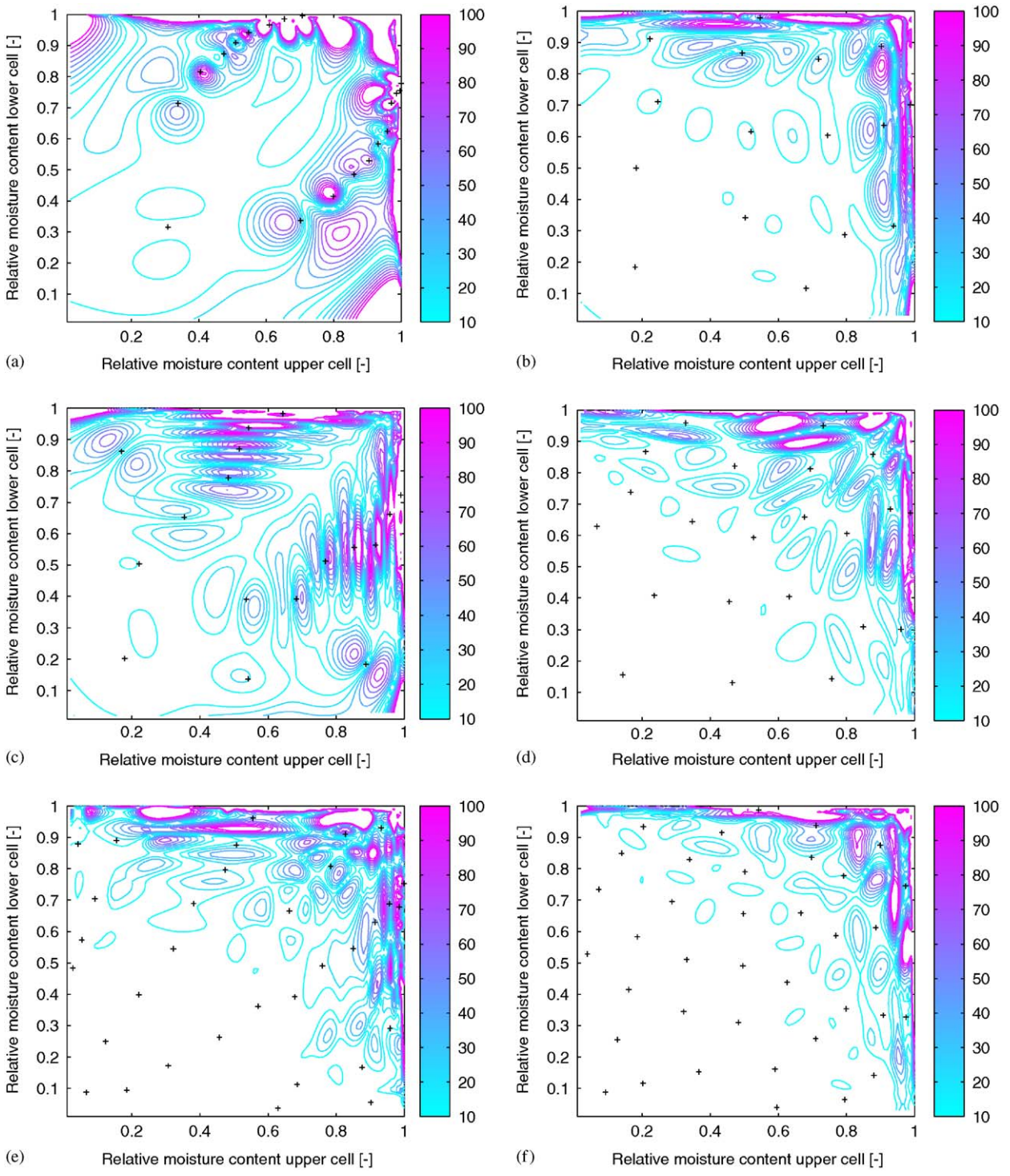

Fig. 7. Contour plots of the error surfaces with contours lines indicated in mm/day and cluster centra (+) corresponding to the models obtained with (a) FCM, (b) GK, (c) SGK, (d) GG, (e) SGG and (f) MGG clustering algorithms.

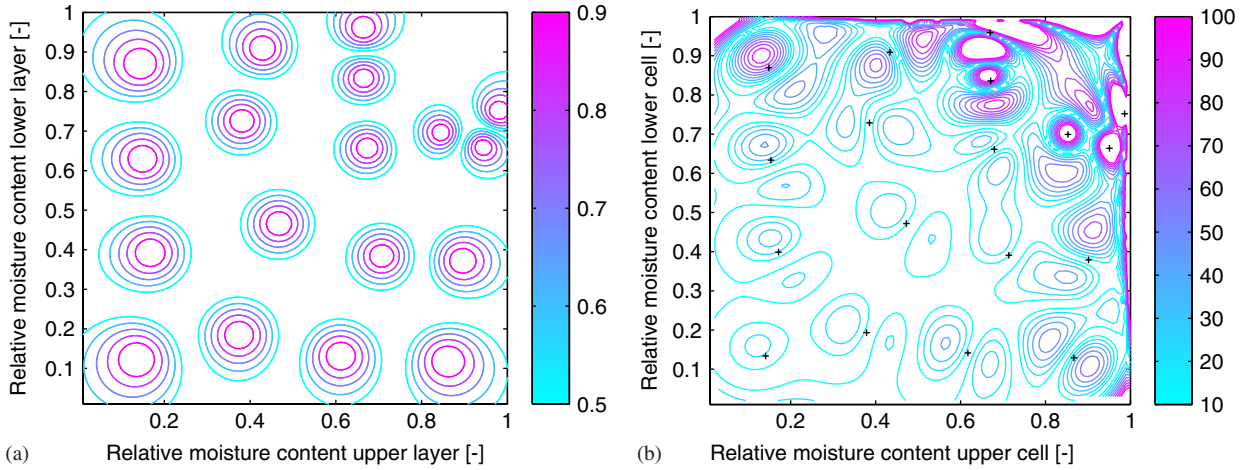

Fig. 8. Contour plots of the multi-dimensional membership functions obtained with the FCM_S algorithm (a) and the corresponding error surface with contour lines indicated in mm/day (b) with indication of the cluster centra (+).

Table 2
Number of rules and β-values obtained with the projection of the different clustering methods and the performance of the corresponding models

Clustering method	#Clusters	β	NS (–)	RMSE (mm/day)	NS_{64} (–)	$RMSE_{64}$ (mm/day)
FCM	7	9	0.9824	41.7798	0.9960	19.9008
FCM_s	11	1	0.9780	46.7696	0.9949	22.0542
GK	12	3	0.9799	44.7054	**0.9983**	**13.0154**
GK_s	15	8	0.9819	42.3764	0.9981	13.8519
SGK	12	9	0.9773	47.4773	0.9913	29.4199
SGK_s	12	2	0.9766	48.1666	0.9907	30.4038
GG	21	7	0.9786	46.0812	0.9978	14.8396
GG_s	18	1	0.9790	45.7057	0.9978	14.9234
SGG	31	7	0.9774	47.4109	0.9955	21.1998
SGG_s	31	4	0.9765	48.3143	0.9956	20.8502
MGG	26	2	**0.9813**	**43.1025**	0.9978	14.7846
MGG_s	25	5	0.9808	43.3853	0.9976	15.4197

reduce the search space, η and r_a were varied between [0.5, 2] and [0.1, 1], respectively, while $\bar{\varepsilon}$ and $\underline{\varepsilon}$ were imposed a value of 0.1. Contour lines indicating the number of clusters obtained with these variations are indicated in Fig. 10.

In order to allow for a fair comparison with the models described in the previous sections (see Sections 6.1.1 and 6.1.2), the maximal allowed complexity is set to be more or less equal to the complexity set for the models identified with the objective function-based clustering algorithms. Therefore, models with a number of rules lower or equal to 65 were retained (64 could not be used since no models with 64 rules were found). A checking data set consisting of 10 000 randomly selected points, different from the training data set, was used to prevent overfitting. Since the training finished after 150 training epochs, without a premature stop due to an increase of the error obtained on the checking data set, the models obtained after 150 training epochs are used for evaluation. Fig. 11 shows the obtained values for the NS index with respect to the number of clusters. Fig. 11 shows that for five clusters one of the models yields a performance between 0.99 and 1 (the highest class of performance values). It was obtained with $r_a = 0.6$ and $\eta = 1.5$ and results in NS- and RMSE-values of 0.9902 and 31.1767 mm/day, respectively. The error surfaces obtained on the training data set for the optimal model and the model with 65 rules have a similar behaviour of those obtained with the objective-function-based clustering algorithms. Fig. 12 additionally shows the membership functions obtained for the model with 5 rules. Noticeable is that all membership functions show a lower membership degree at the border of high moisture contents, indicating that no rule specifically operates in this region.

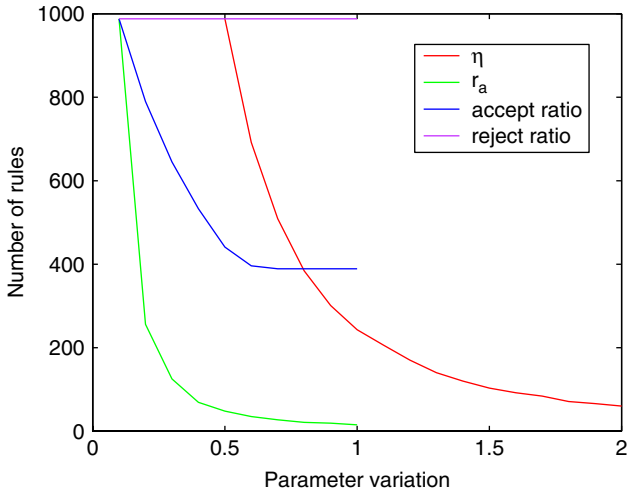

Fig. 9. Number of rules obtained with different values of the respective parameters.

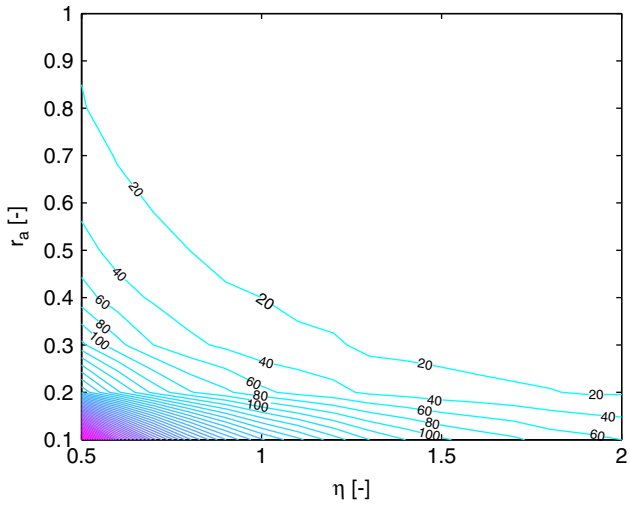

Fig. 10. Contour lines of obtained number of clusters for variations of η and r_a, with $\overline{\varepsilon} = \underline{\varepsilon} = 0.1$.

6.1.4. Grid partitioning

In order to allow a similar complexity as for the fuzzy models described in Sections 6.1.1 and 6.1.2, the number of membership functions for models identified by grid partitioning was varied for each input variable between 2 and 8. In this way, the minimal and maximal complexity of the models is set at 4 and 64 rules, respectively. The same checking data set as used for the models identified with subtractive clustering was used to prevent overfitting. For the models identified with grid partitioning the training stopped after 150 training epochs without a premature stop due to an increase of the error on the checking data set. The models obtained after 150 training epochs were retained for evaluation. The performance results, given in terms of NS, for the obtained models with generalized bell-shaped membership functions are given in Fig. 13.

Fig. 13 shows that a simple model consisting of $3 \times 3 = 9$ rules already yields a high performance value, $0.99 < \mathrm{NS} \leqslant 1$. The same observation was made for the other types of membership functions. Models with less than two membership functions in one of the input variables are generally not used: if one would like to give a linguistic interpretation to the membership functions, at least three terms would be used in a natural language: “low”, “medium” and “high”. Therefore, our attention is focused to models with minimal 3×3 rules. The obtained values of the

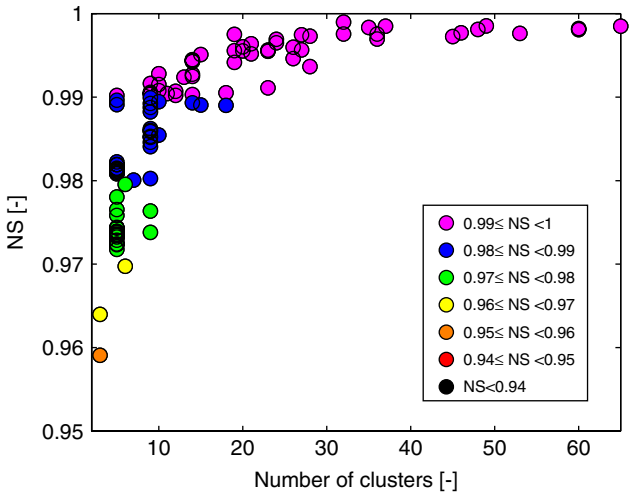

Fig. 11. Performance in terms of values of the NS-index corresponding to the obtained number of clusters.

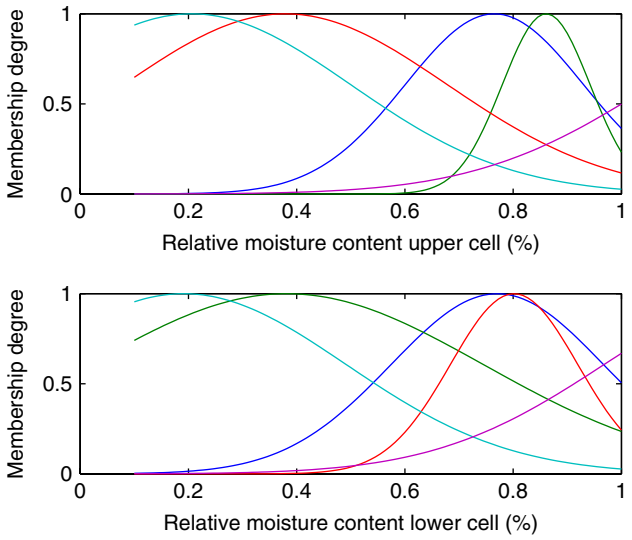

Fig. 12. Membership functions obtained through subtractive clustering and further optimization for the model with 5 rules.

performance indices corresponding to these models are listed in Table 3. The models obtained with the generalized bell-shaped membership functions perform best for 9 and 64 rules (values are indicated in bold). The error surfaces on the training data, obtained with these two models, are shown in Fig. 14. These error surfaces show a flatter surface with lower errors at the borders of higher moisture content than those of the models obtained through clustering which are shown in the previous Sections 6.1.1–6.1.3.

The obtained membership functions for the generalized bell-shaped membership functions are given in Fig. 15. One of the obtained membership functions gives a very low maximal membership degree, a value lower than 0.1 and almost 0 is obtained, which would not be allowed by a modeller aiming at interpretable models.

7. Takagi–Sugeno models identified on a gradient-sensitive training set

An attempt was made to reduce the higher errors without increasing the complexity of the model. Models with the same number of rules and the same identification methods were identified on a training data set that consisted of the

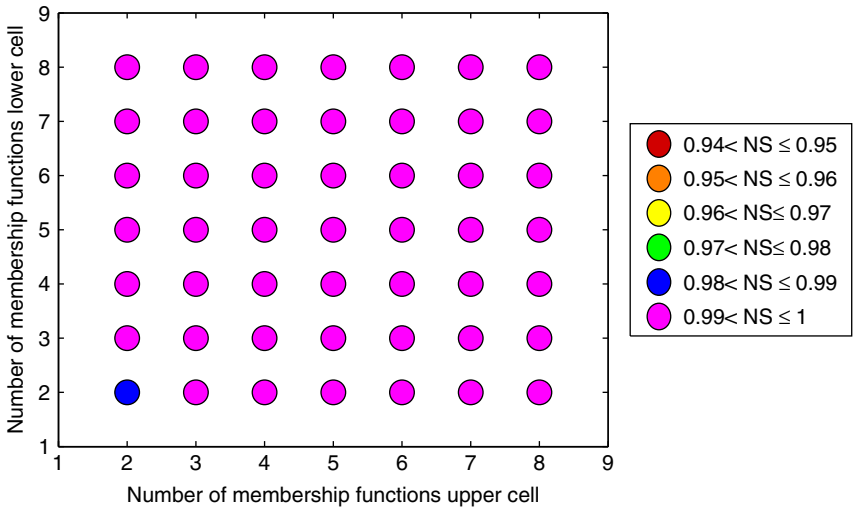

Fig. 13. Performance indices for different numbers of membership functions for the models with generalized bell-shaped membership functions.

Table 3
Performances of fuzzy models obtained with different types of membership functions

Membership function	NS (–)	RMSE (mm/day)	NS_{64} (–)	$RMSE_{64}$ (mm/day)
Triangular	0.9900	31.4878	0.9975	15.8504
Trapezoidal	0.9918	28.5936	0.9970	17.1543
Sym. Gaussian	0.9908	30.1860	0.9989	10.3920
Asym. Gaussian	0.9915	29.0042	0.9985	12.0984
Generalized bell	**0.9962**	**19.4058**	**0.9997**	**5.0878**
p-sigmoidal	0.9943	23.7762	0.9994	7.3904
Π-shaped	0.9924	27.4287	–	–
d-sigmoidal	0.9943	23.7766	0.9994	7.3927

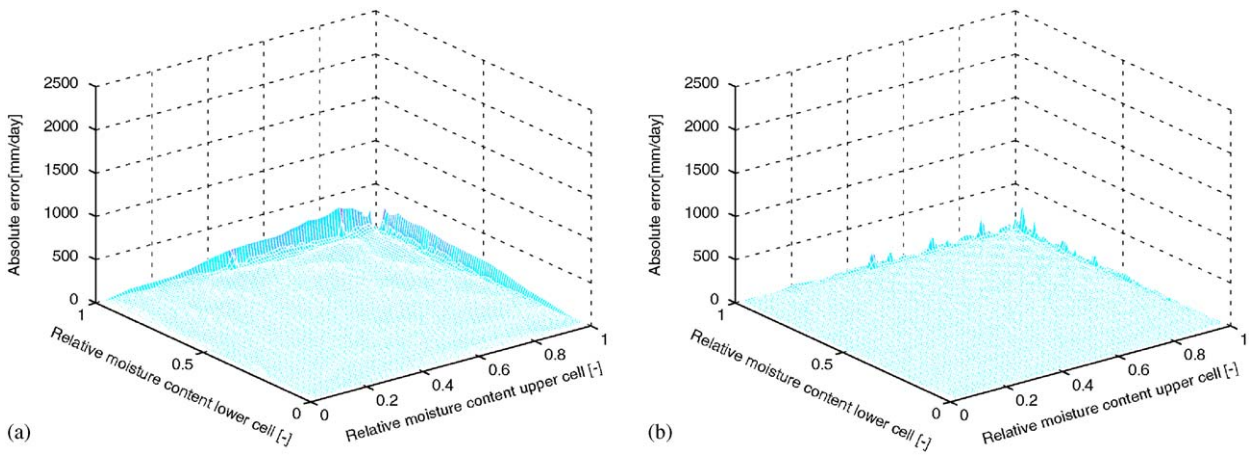

Fig. 14. Errors on the training data set for models with generalized bell-shaped membership functions and with (a) 9 and (b) 64 rules.

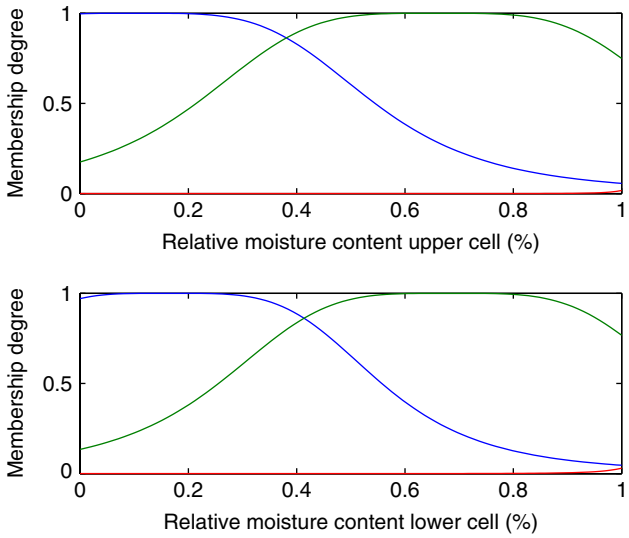

Fig. 15. Generalized bell-shaped membership functions obtained with grid partitioning.

Table 4
Number of clusters for the listed clustering algorithms and the performance of the corresponding models

Clustering algorithm	#Clusters	β	NS (–)	RMSE (mm/day)
SGG_s	33	–	0.9951	34.0054
MGG	26	4	0.9932	39.9016

random training data set and a refinement of the larger data set of 154 449 data points. The refinement is performed depending on the gradient of the data set in the following way: the data points in this larger data set are positioned on a grid in the plane determined by the moisture contents. The grid points are formed by the moisture content values. A kernel consisting of one grid cell is moved along the grid to detect differences between the fluxes of two of the nodes which form the corners of the kernel. Differences between adjacent nodes in the horizontal, vertical and diagonal direction are determined. For differences larger than 5 mm/day, denoted as q_{tol} or for data points for which the two fluxes have opposite signs, intermediate data points are generated and the newly obtained flux is verified again until the obtained difference is equal to or is smaller than 5 mm/day or no opposite signs are obtained. From this refinement, 5000 data points are randomly selected and added to the random training data set. In this way, a data set consisting of 15 000 data points for which a higher concentration of data points can be found at the steep slopes was obtained.

Takagi–Sugeno models with the same number of rules as those obtained on the random training data set were identified on this gradient-sensitive training data set. Table 4 lists the obtained values of the performance indices on this gradient-sensitive training set for the SGG_s model with multi-dimensional membership functions and for the MGG model with one-dimensional membership functions. The values of the performance indices show that better results are obtained for both models. Concerning the error surfaces, a better performance in the regions of higher moisture contents was observed. The higher errors were reduced, however, they did not disappear completely.

8. Conclusions and future research

In this paper, different methods were used to identify the antecedent parameters of the Takagi–Sugeno models on a artificially generated training data set for a sandy loam soil used in the EMSL experiment. In order to determine the number of rules, needed in the fuzzy rule-based model, an algorithm *ClusterFinder* was developed. This algorithm, with a user-defined parameter α, tries to find an optimal number of clusters as a golden mean between very complex accurate models, in this case models with 64 rules, and simpler, less accurate models that can give an impetus towards interpretability.

With α set at 0.02, indicating the allowed reduction for the NS-values, it was observed that the error surfaces on the training data set showed higher errors at two borders, corresponding to a higher or saturated moisture content in one of the two soil cells. These errors were the lowest for the models identified with grid partitioning. This behaviour was also reflected in the obtained values of the performance indices. For all identification methods, these errors were strongly reduced in the obtained error surfaces for models with 64 rules.

An attempt to reduce these higher errors through identification of Takagi–Sugeno models on a gradient-sensitive training set was only partially successful. In general, better values for the performance indices were obtained; however, these higher errors did not disappear completely.

In order to be able to use Takagi–Sugeno models incorporated in a fuzzy rule-based groundwater model, the higher errors in the error surfaces should be reduced. Additional research could be carried out in order to improve the accuracy of the Takagi–Sugeno groundwater models in these regions, and thus to obtain a flatter error surface, without unnecessarily increasing the complexity of the models. Furthermore, the artificial generation of the training data set incorporates soil-specific parameters, which depend on the soil type. Applying a fuzzy rule-based groundwater model for a different soil type involves a regeneration of the training data and re-identification of the Takagi–Sugeno models. In order to avoid this repeatedly generating of training data sets and identifying of fuzzy models, it could be investigated, for instance, whether an umbrella fuzzy model could be identified in which different (fuzzy) models, simulating the unsaturated groundwater flow for different prototypes of soil classes, could be weighted. The antecedent parameters of this umbrella model could then be defined by e.g. fuzzifying the different soil classes represented by means of the texture triangle [27].

References

[1] J. Abonyi, R. Babuška, F. Szeifert, Modified Gath–Geva fuzzy clustering for identification of Takagi–Sugeno fuzzy models, IEEE Trans. Systems Man Cybernet. Part B—Cybern. 32 (5) (2002) 612–621.

[2] K. Astrom, B. Wittenmark, Computer Controlled Systems: Theory and Design, Prentice-Hall, Englewood Cliffs, NJ, 1984.

[3] R. Babuška, Fuzzy Modeling for Control, International Series in Intelligent Technologies, Kluwer Academic Publishers, Boston, USA, 1998.

[4] R. Babuška, Fuzzy Identification Toolbox for MATLAB, URL:⟨http://Icewww.et.tudelft.nl/∼babuska/⟩, 2004.

[5] A. Bárdossy, A. Bronstert, B. Merz, 1-, 2- and 3-dimensional modeling of water movement in the unsaturated soil matrix using a fuzzy approach, Adv. Water Resour. 18 (4) (1995) 237–251.

[6] A. Bárdossy, M. Disse, Fuzzy rule-based models for infiltration, Water Resour. Res. 29 (2) (1993) 373–382.

[7] A. Bárdossy, L. Duckstein, Fuzzy Rule-Based Modeling with Applications to Geophysical, Biological and Engineering Systems, CRC Press, New York, USA, 1995.

[8] K. Beven, Rainfall-Runoff Modelling, The Primer, Wiley, Chichester, UK, 2000.

[9] M. Celia, E. Bouloutas, R. Zarba, A general mass-conservative numerical solution for the unsaturated flow equation, Water Resour. Res. 26 (7) (1990) 1483–1496.

[10] S.L. Chiu, Fuzzy model identification based on cluster estimation, J. Intell. Fuzzy Systems 2 (3) (1994) 267–278.

[11] K. Demirli, S. Cheng, P. Muthukumaran, Subtractive clustering based modeling of job sequencing with parametric search, Fuzzy Sets and Systems 137 (2) (2003) 235–270.

[12] I. Gath, A. Geva, Unsupervised optimal fuzzy clustering, IEEE Trans. Pattern Anal. Mach. Intell. 11 (1989) 773–781.

[13] M. van Genuchten, A closed-form equation for predicting the hydraulic conductivity of unsaturated soils, Soil Sci. Soc. Amer. J. 44 (1980) 892–898.

[14] A. Gomez-Skarmeta, M. Delgado, M. Vila, About the use of fuzzy clustering techniques for fuzzy model identification, Fuzzy Sets Syst. 106 (1999) 179–188.

[15] D. Gustafson, W. Kessel, Fuzzy clustering with a fuzzy covariance matrix, in: Proc. of the IEEE CDC, San Diego, CA, USA, 1979, pp. 761–766.

[16] R. Hoeben, P. Troch, Assimilation of active microwave observation data for soil moisture profile estimation, Water Resour. Res. 36 (10) (2000) 2805–2819.

[17] F. Höppner, F. Klawonn, R. Kruse, T. Runkler, Fuzzy Cluster Analysis, Wiley, New York, 1999.

[18] R. Krishnapuram, J. Keller, A possibilistic approach to clustering, IEEE Trans. Fuzzy Systems 1 (2) (1993) 98–110.

[19] M. Mancini, R. Hoeben, P. Troch, Multifrequency radar observations of bare surface soil moisture content: a laboratory experiment, Water Resour. Res. 35 (6) (1999) 1827–1838.

[20] J. Nash, J. Sutcliffe, River flow forecasting through conceptual models part I—a discussion of principles, J. Hydrol. 10 (1970) 282–290.

[21] R. Paiva, A. Dourado, Structure and parameter learning of neuro-fuzzy systems: a methodology and a comparative study, J. Intell. Fuzzy Systems 11 (3–4) (2001) 147–161.

[22] R. Paiva, A. Dourado, B. Duarte, Quality prediction in pulp bleaching: application of a neuro-fuzzy system, Control Eng. Pract. 12 (5) (2004) 587–594.

[23] L. Richards, Capillary conduction of liquids through porous media, Physics 1 (1931) 318–333.

[24] E. Ruspini, A new approach to clustering, Inform. and Control 15 (1969) 22–32.

[25] J. Sousa, U. Kaymak, Fuzzy Decision Making in Modeling and Control, World Scientific Series in Robotics and Intelligent Systems, World Scientific, New Jersey, 2002.
[26] T. Takagi, M. Sugeno, Fuzzy identification of systems and its applications to modeling and control, IEEE Trans. Systems Man Cybernet. 15 (1) (1985) 116–131.
[27] W. Verheye, J. Ameryckx, Mineral fractions and classification of soil texture, Pedology 2 (1984) 215–225.
[28] R. Yager, D. Filev, Approximate clustering via the mountain method, IEEE Trans. Systems Man Cybernet. 24 (8) (1994) 1279–1284.