

## FUZZY ASSOCIATION RULE MINING FOR DATA DRIVEN ANALYSIS OF DYNAMICAL SYSTEMS

F.P. PACH, F. SZEIFERT, S. NEMETH, P. ARVA AND J. ABONYI\*

Department of Process Engineering, University of Veszprém,  
Veszprém, Egyetem u. 10, H-8200, HUNGARY,  
www.fmt.vein.hu/softcomp, abonyij@fmt.vein.hu

In system identification a key step is to find a suitable model structure. The utilizations of prior knowledge and physical insight about the system are very important when selecting the model structure. In nonlinear black-box modeling no physical insight is available we have “only” observed inputs and outputs from the dynamical system. Association rule mining is one of the widely used data mining tools. It finds interesting association or correlation relationships among a large data set. The aim of this paper is to demonstrate that this data mining tool can be effectively applied for the data-driven modeling and analysis of dynamical systems. The detected association rules can be interpreted as simple local input-output models of the modeled process. Hence, the analysis of the mined association rules (models) can provide useful information about the structure and the order of the model that can adequately describe the dynamical behavior of the process. In this paper a fuzzy association rule mining algorithm is introduced and a rule-based simplification algorithm is presented for the generation of a set of “rule-based models” that can be directly used as a qualitative model of the system. The general applicability of the developed tool is illustrated by the analysis of the input-output data of a continuously stirred styrene polymerization reactor. The detected association rules is used for the selection of the structure of a linear and nonlinear (neural network) models for this process and determine the most relevant process variables.

**Keywords:** process modeling, model structure selection, association rules, rule base systems, polymerization

### Introduction

In process modeling the a priori knowledge, experimental data and experiments are crucial. The process of modeling from experimental data is known as system identification by Ljung [1]. The main steps of the system identification process are summarized well by Petrick and Wigdorowitz [2]:

1. Design an experiment to obtain the physical process input/output experimental data sets pertinent to the model application.
2. Examine the measured data. Remove trends and outliers. Apply filtering to remove measurement and process noise.
3. Construct a set of candidate models based on information from the experimental data sets. This step is the model structure identification.
4. Select a particular model from the set of candidate models in step 3 and estimate the

model parameter values using the experimental data sets.

5. Evaluate how good the model is, using an objective function. If the model is not satisfactory then repeat step 4 until all the candidate models have been evaluated.
6. If a satisfactory model is still not obtained in step 5 then repeat the procedure either from step 1 or step 3, depending on the problem.

A key step (step 3) is to find a suitable model structure which is capable of representing the dynamical behavior of the system. Therefore effective methods for structure selection are necessary.

Consider the main aspects influencing the choice of a model structure:

- What type of model is needed, nonlinear or linear, static or dynamic, distributed or lumped?

---

\*Correspondence concerning this article should be addressed to J. Abonyi (abonyij@fmt.vein.hu)

- How large must the model set be? This question includes the issue of expected model orders and types of nonlinearities.
- How must the model be parameterized? This involves selecting a criterion to enable measuring the closeness of the model dynamic behavior to the physical process dynamic behavior as model parameters are varied.

Large number of model structure selection methods has been introduced. For linear models, for example correlation analysis and multivariate structure selection techniques [3] such as principal component analysis (PCA) are proposed.

Several information-theoretical criteria have been also proposed for the structure selection of linear dynamic input-output models. These methods are based on the minimization of a criterion function which involves the estimation of the one-step-prediction error plus some penalty function. The classical criteria are the Final Prediction Error (FPE), the Akaike Information Criterion (AIC) [4], the Minimum Description Length (MDL) criterion [5], the Schwarz criterion (BIC) [6] and the Hannan-Quinn Criteria (HIC) [7]. They only differ on the employed penalty function, but in [8] a new criterion function is introduced based on the decomposition of the variance of the innovations of the model in terms of their frequency components. The information criteria have been used in a context of regression models [9, 10], in distributed lag regression models [11], or in selection of the order an autoregressive and autoregressive moving average model [12, 13, 14]. In paper [15] the effects of the model selection problem, and in paper [16] the variable selection problem are studied.

Determining the structure of linear systems is a rather straightforward task with these tools, but for nonlinear systems other structure selection methods are need. Aguirre and Billings [17] defined the concepts of term clusters and cluster coefficients and used in the context of system identification. This approach is used for the structure selection of polynomial models in the paper of Aguirre and Mendes [18]. In [19] an alternative solution is introduced by initially conducting a forward search through the many possible candidate model terms before performing an exhaustive all-subset model selection on the resulting model. A backward search approach based on orthogonal parameter estimation is also applied to structure selection [20, 21]. The paper [22] discusses several model structures selection methods and nonlinear input-output models that are suitable for implementation of feed-forward neural networks. A systematic method for the selection of model order and time delay is presented in [23]. The method is applied to the neural network modeling of a multivariable chemical process rig. A deterministic suitability measure is introduced in [24] that quantifies the capability of a particular model class to capture the control relevant I/O-behavior of a nonlinear system. This suitability measure can be used for the purpose of

model structure selection prior to the actual parameter identification. The fast bootstrap (FB) methodology to select the best model structure is presented in [25]. The methodology is applied to a regression task. In [26] a methodology for model structure selection based on a genetic algorithm was introduced and applied to nonlinear discrete-time dynamic systems. A modified genetic programming approach for model structure selection is introduced in [27]. It is combined with a classical technique for parameter estimation. Hong and Harris [28] introduced a learning algorithm for model subset selection which based on a new composite cost function that simultaneously optimizes the model approximation ability and model adequacy. In [29] a cost functional is evaluated for each identified model and the model with minimum cost is preferred. Suboptimal search strategies are adopted, forward and stepwise strategies are considered.

We introduce in this paper a new data-driven structure selection method. The new method is based on fuzzy association rule mining and it is called MOSSFARM (Model Structure Selection by Fuzzy Association Rule Mining). Association rule mining finds interesting association or correlation relationships among a large data set. The problem of mining association rules was introduced over supermarket basket data in [30]. It helps to learn more about the buying habits of the customer. It gets information and answers for the market questions. But the market basket analysis is just one application of association rule mining this paper presents a new application area, the model structure selection.

This paper is organized as follows. The first section introduces the system identification problem in nonlinear black box modeling. Association rule mining theory is presented in the second section. In third section the fuzzy association rule mining is detailed. Our new method based on fuzzy association rule mining is introduced in the fourth section. The last fifth section illustrates how the MOSSFARM method select the most important model structures of a linear (least square method) and a non-linear (neural network) model of a styrene polymerization CSTR.

### **System Identification in Nonlinear Black Box Modeling**

To be successful the entire modeling process should be given as much information about the system as is practical. The utilization of prior knowledge and physical insight about the system are very important, but in nonlinear black-box situation no physical insight is available, we have “only” observed inputs and outputs from the system. This paper concentrates on structure selection task in case of the black-box modeling.

In a system identification problem in case of black-box modeling [31] we have only input,  $\mathbf{u}_k$  and output,  $\mathbf{y}_k$  data from the process (system),

$$\mathbf{u}_k = [u_1 \ u_2, \dots, u_k] \quad (1)$$

$$\mathbf{y}_k = [y_1 \ y_2, \dots, y_k] \quad (2)$$

We are looking for a relationship between past observations  $[\mathbf{u}_{k-1}, \mathbf{y}_{k-1}]$  and future outputs,

$$y_k = f(\mathbf{u}_{k-1}, \mathbf{y}_{k-1}) + e_k, \quad (3)$$

where  $e_k$  represents an error value, because  $y_k$  will not be an exact function of past data. However, a goal must be that  $e_k$  is small, so that we may think of  $f(\cdot)$  function as a good prediction of  $y_k$ .

Eq. 3 models general discrete time dynamic systems, but nonlinear static processes can be also represented by the following regression model:

$$y_k = f(\mathbf{x}_k) \quad (4)$$

The  $f(\cdot)$  is a non-linear function,  $\mathbf{x}_k$  represents its input vector and  $k = 1, \dots, N$  denotes the  $k$ -th input-output data. The model regression vector  $\mathbf{x}_k$  in a NARX (Non-linear Auto-Regressive models with eXogeneous Inputs) model contains the past values of the process outputs  $y_k$  and the process inputs  $u_k$  as regressors:

$$\mathbf{x}_k = [y_{k-1}, y_{k-2}, \dots, y_{k-m}, u_{k-1}, u_{k-2}, \dots, u_{k-n}]^T \quad (5)$$

The  $m$  determines the number of past outputs and  $n$  represents the past inputs (model order). While the output of the regression model,  $y_k$  is the one-step-ahead prediction of the process. This SISO form of the NARX models could be extended to the MIMO case.

### Association Rule Mining

One of the widely used research tasks in data mining is the discovery of frequent item sets and association rules. The problem originates in market basket analysis which aims at understanding the behavior of retail customers, or in other words, finding associations among the items purchased together. A famous example of an association rule in such a database is “Diapers => Beer”, i.e. young fathers being sent off to the store to buy diapers, reward themselves for their trouble. Because of the practical usefulness of association rule discovery, this approach can be applied in various research areas.

How can we search association rules? The association rule mining is based on frequent item set searching. An item could be for example one product in the

supermarket example, e.g. {beer}, and an item set is a set of items (products), e.g. {milk, beer, diapers}. The occurrences of an item (item sets) in a data set are called support. The support value of an item (item set) could be seen as a probability value. The support gives in how many percent of the transactions is the item (are the items of an item set together)? Be the  $X$  an item set, the support value of  $X$  is calculated as follows:

$$\text{supp}(X) = P(X) = \frac{\text{\#of transactions with } X}{\text{\#of transactions}} \quad (6)$$

An item  $x$  (or the item set  $X$ ) is called frequent item (item set) if its support is higher than a given (user defined) threshold, namely the minimal support ( $\sigma$ ).

See Table 1 which includes an example supermarket transaction data set, where each row represents a transaction. The first column contains the transaction number (Tid - transaction identifier) and in the second column the purchased products are listed in the transaction.

Table 1 Example transaction data set

Tid	Items
1	Bread, Milk
2	Beer, Bread, Diaper, Eggs
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk

The frequent item set searching is a very easy task for this example data set, because the number of transactions is only four. If the minimum support is equal to 50 percent ( $\sigma =$  two occurrences), we can search all the frequent items and item sets, e.g. a frequent item is the {Milk} (with 75 % support), or a frequent item set is the {Diaper, Beer} (with 75 % support). But if we have a large data set (database) with many transactions and several items the frequent item set searching demands to use an efficient algorithm.

A widely used frequent item set searching algorithm is the *Apriori* algorithm (was introduced in [30]). The name of algorithm is based on the fact that the *Apriori* uses prior knowledge of frequent item sets already determined. It is an iterative, breadth-first search algorithm, based on generating stepwise longer *candidate* item sets, and clever pruning of non-frequent item sets. Pruning takes advantage of the so-called *apriori* (or *upward closure*) *property* of frequent item sets: all subsets of a frequent item set must also be frequent. Each candidate generation step is followed by a counting step where the supports of candidates are checked and non-frequent ones deleted. Generation and counting alternate, until at some step all generated candidates turn out to be non-frequent.

If we searched all the frequent item sets by the *Apriori* algorithm, we can generate association rules from them. An association rule has two parts: the rule antecedent

(denoted by  $X$ ) and the rule consequent (denoted by  $Y$ ), and both of them contain items. Therefore an association rule is represented by the form  $X \Rightarrow Y$ . From a frequent item set we can generate all the possible rules. Each item and sub item set could be placed in both parts of a rule. In the previous example, we can generate six rules from the frequent item sets  $\{\text{Beer, Bread, Diaper}\}$  (see the possible rules in Figure 1).

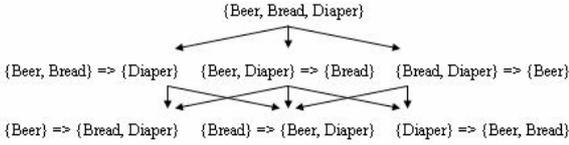


Fig.1 Example for association rule generating

It is very important to know which association rules are well usable, namely which gives the most information about the data. Two basic measures are used to calculate how “important” an association rule. First one is the previously defined support measure. The support of an association rule is the support of the set of its items. For example, the support of the rule “ $\{\text{Bread, Diaper}\} \Rightarrow \{\text{Beer}\}$ ” is equal to the support of the item set  $\{\text{Beer, Bread, Diaper}\}$ .

The second one, the confidence measure of a rule is calculated as follows:

$$\text{conf}(X \Rightarrow Y) = \frac{\text{supp}(X \cup Y)}{\text{supp}(X)} \quad (7)$$

Because the confidence measure is a conditional probability (the quotient of the rule - and antecedent supports) it serves information about the relationship of the antecedent and consequent parts of a rule. A rule  $X \Rightarrow Y$  is called *important*, or *strong rule* if its support and confidence are higher than the minimum support ( $\sigma$ ) and the minimum confidence thresholds ( $\gamma$ ). The confidence is a basic rule interestingness measure, but many other measures are also can be used to determine the importance and ordering the mined rules (e.g. RI – Rule Interesting, Lift, Correlation, Jaccard, Piatetsky-Shapiro, etc. measures).

The number of the possible association rules is very high in case of an item set with many elements (e.g. if ten items are in an item set, the number of possible rules is  $2^{10} - 2$ ). Therefore it is well worth using the *anti-monotonic feature*. For generating of the  $(n-1)$ -large rules we can use in the previous step selected frequent  $(n)$ -large rule. For example be the  $Z$  a frequent item set. The following two association rules are generated from  $Z$ : 1)  $X \Rightarrow Z \setminus X$  and 2)  $x \Rightarrow Z \setminus x$ , where  $x \in X$ . If the first rule does not fulfil the support criteria, the second rule is also can not to be frequent.

The importance of an association rule can be determined not only by objective way. The determination can be also subjective. The users can add the “right” form of the rules. Suppose that a user wants to search only rules with a distinguished item in consequent part. For example be this item a product-family, books. In this case, only the rules where a book is placed in the consequent part will be strong rules.

To increase the usability of the association rule, fuzzy association rules are proposed. In the next section, the basic definitions of fuzzy association rule theory are presented.

## Fuzzy Association Rule Mining

The fuzzy association rules can be discovered in also two steps as we showed in crisp case: 1) *mining frequent item sets*, and 2) *generating fuzzy association rules* from the discovered set of frequent item sets.

A dataset (database) includes records (data rows) via the data fields (columns). The fields are frequent called as attributes.

Because in fuzzy association rule theory the items are fuzzy sets, a partition method is necessary which transforms the crisp data set into fuzzy data set for all attributes. For the numerical attributes (as for example the temperature, pressure, etc.) of the data set, fuzzy sets can be defined with Gaussian, sigmoid, or piecewise linear fuzzy dichotomies. Therefore triangular, trapezoidal type of fuzzy sets (intervals) can be used for data partition.

See the Figure 2 for an example, where the attributes  $z_1$  and  $z_2$  are partitioned by two-two trapezoidal fuzzy sets.

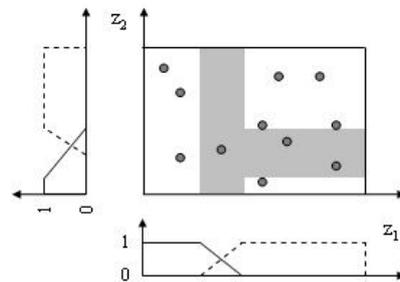


Fig.2 A fuzzy partition of data space ( $z_1, z_2$ )

Let  $D = \{t_1, t_2, \dots, t_N\}$  be a transformed (partitioned) fuzzy dataset of  $N$  tuples (data records ~ data points) with a set of attributes  $Z = \{z_1, z_2, \dots, z_q\}$  and let  $c_{i,j}$  be an arbitrary fuzzy interval (fuzzy set) associated with attribute  $z_i$  in  $Z$  where  $q$  denotes the number of attributes. From this point, we use the notation  $\langle z_i : c_{i,j} \rangle$  for an *attribute-fuzzy interval pair*, or simply

*fuzzy item*. An example could be  $\langle \text{Age} : \text{young} \rangle$ . For *fuzzy item sets*, we use expressions like  $\langle Z : C \rangle$  to denote an ordered set  $Z \subseteq \mathcal{Z}$  of attributes ( $\mathcal{Z}$  denotes the set of the all possible attributes) and a corresponding set  $C$  of some fuzzy intervals, one per attribute, i.e

$$\langle Z : C \rangle = [\langle z_{i_1} : c_{i_1,j} \rangle \cup \langle z_{i_2} : c_{i_2,j} \rangle \cup \dots \cup \langle z_{i_q} : c_{i_q,j} \rangle].$$

In the literature, the fuzzy support value has been defined in different ways. Some researchers suggest the minimum operator as in fuzzy intersection, others prefer the product operator. They can be defined formally as follows: value  $t_k(z_i)$  for attribute  $z_i$ , then the *fuzzy support* of  $\langle Z : C \rangle^2$  with respect to  $D$  is defined as

$$FS(Z : C) = \frac{\sum_{k=1}^N \Lambda_{\langle z_i : c_{i,j} \rangle \in \langle Z : C \rangle} t_k(z_i)}{N} \quad (8)$$

where the  $\Lambda = \{\min, \Pi, \dots\}$  is an operator set. In this paper we prefer the product form. A fuzzy support reflects how the record of the identification data set support the item set. A fuzzy item set  $\langle Z : C \rangle$  is called *frequent* if its fuzzy support value is higher than or equal to a user-defined minimum support ( $\sigma$ ). The following example illustrates the calculation of the fuzzy support value. Let  $\langle X : A \rangle = [\langle \text{Balance} : \text{medium} \rangle \cup \langle \text{Income} : \text{high} \rangle]$  be a fuzzy item set and the example dataset is shown in Table 2.

Table 2 Example database containing memberships

Balance:	Credit:	Income:
med.	high	high
0.5	0.6	0.4
0.8	0.9	0.4
0.7	0.8	0.7

The fuzzy support of  $\langle X : A \rangle$  is calculated as follows:

$$FS(X : A) = \frac{0.5 \cdot 0.4 + 0.8 \cdot 0.4 + 0.7 \cdot 0.7}{3} = \mathbf{0.3367}$$

Since the rules are generated from the frequent item sets, the generation of fuzzy association rules becomes relatively straightforward. More precisely, each frequent item set  $\langle Z : C \rangle$  is divided into the consequent  $\langle Y : B \rangle$  and antecedent  $\langle X : A \rangle$ , where  $X \subset Z$ ,  $Y = Z - X$ ,  $A \subset C$  and  $B = C - A$ . With the use of this notation a *fuzzy association rule* can be represented in the form of

$$\text{If } X \text{ is } A, \text{ then } Y \text{ is } B, \quad (9)$$

or in more compact form,

$$\langle X : A \rangle \Rightarrow \langle Y : B \rangle. \quad (10)$$

A fuzzy association rule is considered *strong* if its support and confidence exceeds the given minimum support ( $\sigma$ ) and minimum confidence ( $\gamma$ ). Since the rules are generated from frequent item sets, they satisfy the minimum support automatically. The *fuzzy confidence* of a fuzzy association rule  $\langle X : A \rangle \Rightarrow \langle Y : B \rangle$  is defined as

$$FC(X : A \Rightarrow Y : B) = \frac{FS(\langle X : A \rangle \Rightarrow \langle Y : B \rangle)}{FS(X : A)} \quad (11)$$

and it is a conditional probability of the parts of the rule:  $P(\langle Y : B \rangle | \langle X : A \rangle)$ .

Fuzzy association rules mined using the above fuzzy support-confidence framework are useful for many applications. However, a rule might be identified as interesting when, in fact, the occurrence  $\langle X : A \rangle$  does not imply the occurrence of  $\langle Y : B \rangle$ . The occurrence of a fuzzy item set  $\langle X : A \rangle$  is independent of the item set  $\langle Y : B \rangle$  if  $FS(Z : C) = FS(X : A) \cdot FS(Y : B)$ , otherwise item sets  $\langle X : A \rangle$  and  $\langle Y : B \rangle$  are dependent and correlated as events. The correlation between the occurrence of  $\langle X : A \rangle$  and  $\langle Y : B \rangle$  can be measured by computing the interestingness of a given rule:

$$Fcorr(\langle X : A \rangle, \langle Y : B \rangle) = \frac{FS(Z : C)}{FS(X : A) \cdot FS(Y : B)} \quad (12)$$

If the resulting value of Eq. 12 is less than one, the occurrence of  $\langle X : A \rangle$  is negatively correlated with the occurrence of  $\langle Y : B \rangle$ . If the resulting value is greater than one,  $\langle X : A \rangle$  and  $\langle Y : B \rangle$  are positively correlated. It means the occurrence of one implies the other. If the resulting value is near to one, then  $\langle X : A \rangle$  and  $\langle Y : B \rangle$  are independent and there is no correlation between them.

After the review of basics of association rule mining, the next section presents how use the fuzzy association rules for model structure selection.

### MOSSFARM - Model Structure Selection by Fuzzy Association Rule Mining

Since in the previous section all of the necessary definitions and methods to mine fuzzy association rules were considered, this section will focus on the main steps of our method that are needed to solve the studied model structure (order) selection problem in case of NARX model.

Suppose that we have only measured input-output data from a SISO process. In a NARX model of the process, from this I/O data set we can construct a regression vector from each  $k = 1, \dots, N$  input-output data pairs by the following way, similar for Eq. 5:

$$\mathbf{x}_k = [y_{k-1}, y_{k-2}, \dots, y_{k-m}, u_{k-1}, u_{k-2}, \dots, u_{k-n}]^T$$

where the past values of the process outputs  $y_k$  and the process inputs  $u_k$  are the regressors. The number of past inputs ( $n$ ) and past outputs ( $m$ ) are often referred to as model order.

The question is how to select the right model order? An answer could be our method, the MOSSFARM.

The method consists of the following five steps:

- 1) Generate a fuzzy database
- 2) Mine frequent fuzzy item sets
- 3) Generate fuzzy association rules
- 4) Prune the fuzzy rule base
- 5) Aggregate the mined rules, select the model structure

*Step 1)* Observed (measured) input-output data are general crisp values. In the first step the “attributes” (regressors in the regression vector) need to partition to get fuzzy valued data set. The fuzzy Gustafson-Kessel (GK) [33] clustering algorithm partitions the initial data on every attribute (dimension of data ~ all the candidate regressors). The resulted membership functions are transformed into trapezoidal membership functions (see an example in Fig. 4 where on two attributes are four and three fuzzy sets, respectively).

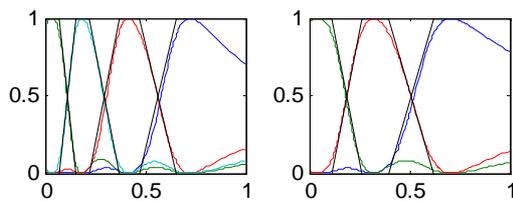


Fig.4 Trapezoidal membership functions

*Step 2)* The resulted fuzzy data set includes the membership function values of each data points on each attributes, and the index of fuzzy set which give the highest membership value for the data point in a given attribute. These indices can be the items and the set of them are the item sets. The frequent item set searching is based on a fuzzy implementation of the *Apriori* algorithm. The fuzzy support values are calculated as the Eq. 8 shows.

*Step 3)* The mined frequent item sets are the base of fuzzy rule generation step. Every fuzzy rules are generated, but only the rules with high support ( $FS >$

$\sigma$ ) and confidence ( $FC > \gamma$ ) values are the relevant rules. The fuzzy support and confidence values are calculated as the Eq. 8 and 11 show. The interesting value of the rules are determined by the correlation factor ( $F_{corr}$ , calculated by Eq.12).

*Step 4)* The advantage of the application of the correlation measure for the analysis of the quality of the rules is that it is *upward closed*. Based on this, a rule based pruning algorithm has been developed that removes the unnecessarily complex rules. Such rules contain input variables that do not significantly improve the correlation of rules.

*Step 5)* We have to analyze the mined association rules to determine model structures. The rules where the first indices of the fuzzy sets are equal in the antecedent parts, give identical model structures. Therefore it is necessary to aggregate the support, the confidence, and the correlation measures of these individual rules. After the aggregation the given model structures must be ordered by the correlation (calculated by Eq. 12) measure (or by other rule interesting measures) and accordingly the first structures will be the most interesting structures of the models.

In the next section an application study is showed to illustrate how this method determines the structures of the models for a dynamic system and selects the most relevant process variables.

## Application Study

### Model of styrene polymerization CSTR

The data-driven modeling of styrene polymerization in a continuously stirred tank reactor is considered as a case study to demonstrate the applicability of the proposed method. The schematic diagram of the polymerization process is shown in Fig. 5.

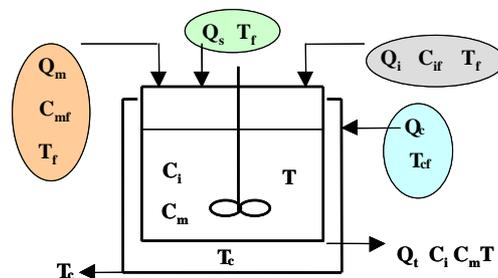


Fig.5 Scheme of the styrene polymerization CSTR with a cooling jacket

On Figure 5,  $Q_m$ ,  $C_{mf}$ ,  $T_f$ ,  $C_m$  denote the monomer flowrate, the monomer feed concentration, the feed

temperature, and the monomer concentration, respectively. The solvent flowrate is represented by  $Q_s$ , while  $Q_i$ ,  $C_{if}$ ,  $C_i$  are the initiator flowrate, the initiator feed concentration, and the concentration of the initiator in the reactor, respectively.  $T_{cf}$ ,  $Q_c$  and  $T_c$  are the coolant feed temperature, the coolant flowrate, and the coolant temperature. The total flowrate is denoted by  $Q_T$  and  $T$  is the reactor temperature. The initiator is azobisisobutyronitrile (AIBN) dissolved in benzene, while the monomer is styrene and the solvent is benzene.

For the simulation of this system the model of Hidalgo and Brosilow [33] is applied:

$$\frac{dC_i}{dt} = \frac{Q_i C_{if} - (Q_i + Q_m + Q_s) C_i}{V} - k_d C_i \quad (13)$$

$$\frac{dC_m}{dt} = \frac{Q_m C_{mf} - (Q_i + Q_m + Q_s) C_m}{V} - k_p C_m C_{gp} \quad (14)$$

$$\frac{dT}{dt} = \frac{(Q_i + Q_m + Q_s)(T_f - T)}{V} + \frac{(-Hr)}{\rho C_p} k_p C_m C_{gp} - \frac{U A}{\rho C_p V} (T - T_c) \quad (15)$$

$$\frac{dT_c}{dt} = \frac{Q_c (T_{cf} - T_c)}{V_c} + \frac{U A}{\rho C_p V} (T - T_c) \quad (16)$$

$$C_{gp} = \sqrt{\left[ \frac{2fk_d C_i}{k_t} \right]} \quad (17)$$

where  $C_{gp}$  represents the concentration of the growing polymer.

The dimensionless model and its nominal parameter values are detailed in [34]. The dynamical behavior of this dimensionless model is illustrated in Figure 6, where the coolant flowrate ( $Q_c$ ) is considered as an input variable. The steady-state input-output relationship of  $Q_c$  and the reactor temperature ( $T$ ) confirms the nonlinear behaviour of the model (see Figure 7).

Suppose we have only input output simulated data taken from the above CSTR model, and we want to identify a linear ARX or a neural network based NARX model with some model structure. To determine (select) the structure of these models the proposed fuzzy association rule based method will be followed in the next session.

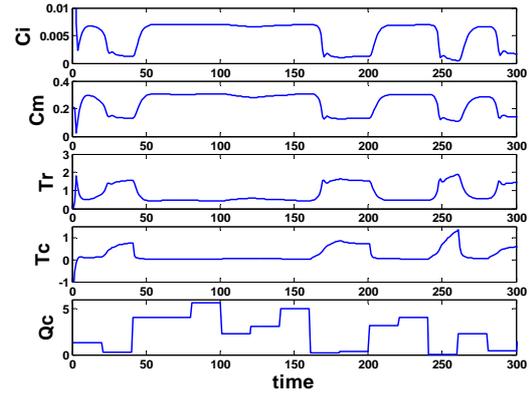


Fig.6 Dynamical behavior of the dimensionless model of the styrene polymerization reactor

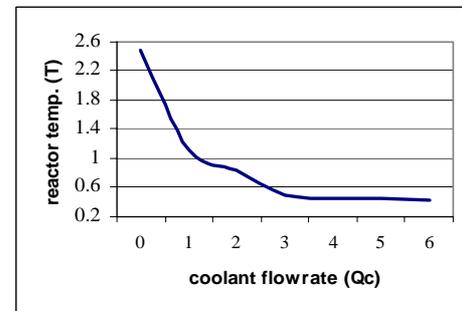


Fig.7 Steady-state input-output relationship of the coolant flowrate and the reactor temperature

#### Model structure selection by MOSSFARM

A SISO NARX model of the previous system is considered where the output of model ( $y$ ) is the reactor temperature, and the input ( $u$ ) is the coolant flowrate. The maximal number of lagged inputs and outputs is four-four, therefore the more complex model is:

$$y_{k+1} = f([y_k, y_{k-1}, \dots, y_{k-4}, u_k, u_{k-1}, \dots, u_{k-4}]) \quad (18)$$

If the original (the full, see Eq. 18) model structure is used for a linear and a neural network (NN) based model, the mean square error (MSE) values are 0.00047 and 0.00008, respectively. In the linear model Least Square Method is used. At the NN model the number of regressors in the structure is used for set the number of neurons in input, hidden and output layers (e.g. for the structure  $[y_k, y_{k-1}, y_{k-2}, y_{k-3}, u_k]$  input: 5, output: 1, hidden: 3). The applied learning method was the back-propagation method. The structures with highest correlation factor (selected by MOSSFARM with  $\sigma = 1\%$ ,  $\gamma = 95\%$ ) are listed in Table 3.

Table 3 The selected model structures

#	Structure	FS	FC	Fcorr
1.	$y_k, y_{k-1}, y_{k-2}, y_{k-3}, u_k$	8.1	96.3	822
2.	$y_k, y_{k-1}, y_{k-3}, u_k$	8.2	95.9	819
3.	$y_k, y_{k-2}, y_{k-3}, u_k$	8.2	95.7	817
4.	$y_k, y_{k-1}, y_{k-2}, y_{k-3}, u_{k-1}$	8.7	95.3	814
5.	$y_k, y_{k-3}, u_k, u_{k-2}, u_{k-3}$	8.3	95	811

Table 4 shows the selected model structures for several minimal support and confidence conditions. The selected model structures (e.g. in Table 3) can be used to identify the linear and the NN model.

Table 4 The selected model structures for several searching condition (support, confidence)

$\sigma$	$\gamma$	Best structure
1	95	$y_k, y_{k-1}, y_{k-2}, y_{k-3}, u_k$
1	80	$y_{k-3}, u_k$
1	70	$y_k, y_{k-1}, y_{k-3}, u_k, u_{k-2}$
1	60	$y_k, y_{k-1}, y_{k-2}, y_{k-3}, u_k, u_{k-1}$
1	50	$y_{k-1}, y_{k-3}, u_k, u_{k-1}, u_{k-2}, u_{k-3}$
5	50	$y_k, y_{k-3}, u_k$
8	60	$y_k, y_{k-3}, u_k$
3	70	$y_k, y_{k-3}, u_k$

The results are showed in Figures 8-11. We can see that the NN model gives lower MSE values in all cases. The MSE values are lower at the using of first structures as the other structures in both model (linear and NN). The resulted MSE values are higher than the original (the full) structures, but the differences are not considerable and these structures are smaller than the original model structure. Therefore MOSSFARM can be an efficient method for determining the model structure for input-output models.

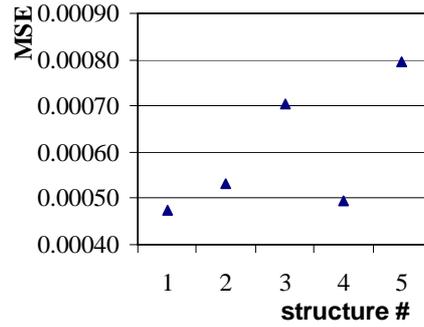


Fig.8 The MSE values for the linear model with the selected model structures

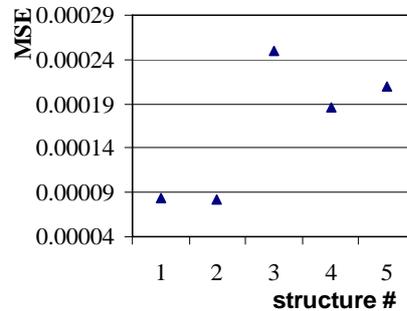


Fig.9 The MSE values for the NN model with the selected model structures

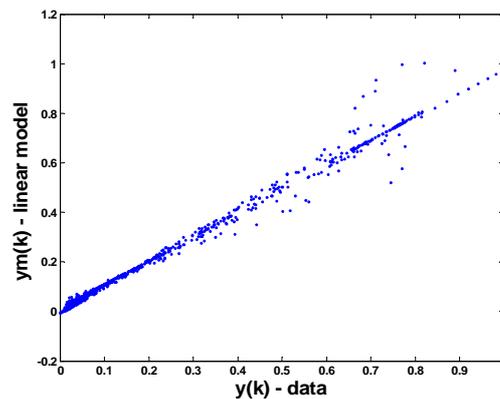


Fig.10 The linear model output values -  $y_m(k)$  - in function of the simulated output values -  $y(k)$  -

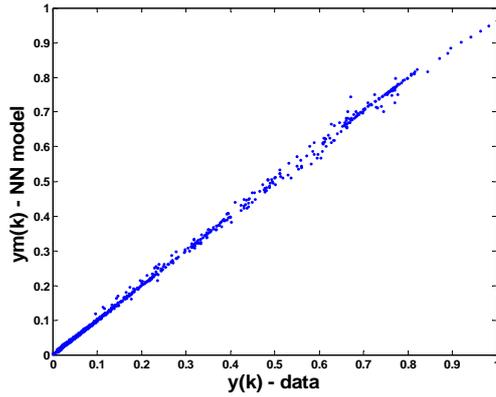


Fig.11 The NN model output values -  $y_m(k)$  - in function of the simulated output values -  $y(k)$  -

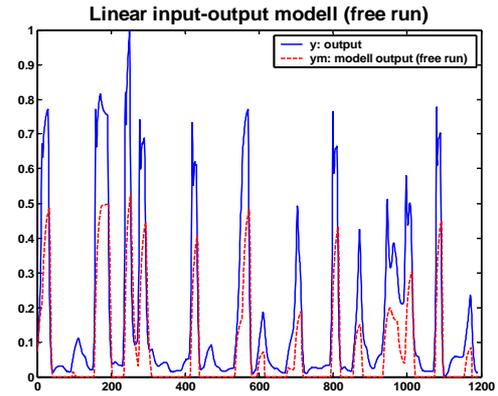


Fig.12 Results of the free run of linear model for  $y_k, y_{k-1}, y_{k-2}, u_{k-1}, u_{k-2}$  structure

Table 5 The effects of the rule pruning step

First structure	Original r. base	Pruned r. base
	$y_k, u_k, u_{k-1}, u_{k-2}, u_{k-3}$	$y_k, y_{k-3}, u_k$
Rules	714	27
Conditions	3157	77
MSE of Lin.	0.001	0.0008025
MSE of Lin. free	0.0352	0.0141
MSE of NN	0.00039197	0.00030913
MSE of NN free	0.03	0.0093

Table 5 shows the results of the rule base complexity analysis. The searching conditions were followings:  $\sigma = 3\%$ ,  $\gamma = 90\%$  for the original simulated (with the regressor in Eq. 18) data. The rule pruning step give smaller, but well usable model structures both in linear and NN models. The pruned rule base has 97.56 percent less complexity than the original rule base and lower MSE values.

If the structures are ordered by the number of aggregated rules, the first structures are the  $y_k, y_{k-1}, y_{k-2}, u_{k-1}, u_{k-2}$  (with the original rule base), and the  $y_k, y_{k-3}, u_k$  (with pruned rule base, all other searching parameters are equal as was in case of Table 5). For the structure  $y_k, y_{k-1}, y_{k-2}, u_{k-1}, u_{k-2}$  the MSE values of the linear and the NN model in the case of free running are 0.0223, 0.0192 respectively (the results are depicted in Fig. 12 and Fig. 13). For structure  $y_k, y_{k-3}, u_k$  lower MSE values are resulted: 0.0141 at free running of linear model and 0.0174 at free running of NN model.

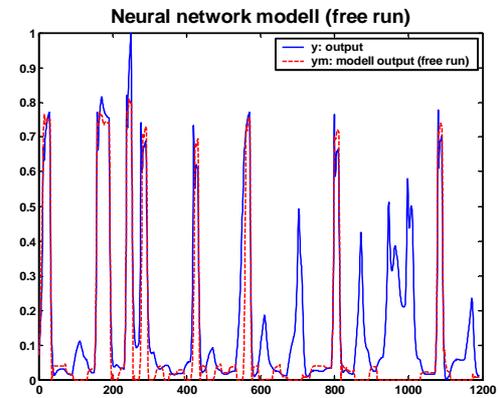


Fig.13 Results of the free run of neural model  $y_k, y_{k-1}, y_{k-2}, u_{k-1}, u_{k-2}$  structure

Fig

#### Feature (variable) selection by MOSSFARM

The proposed method also can be used for select the most relevant variable which determine the output variable. The four state dimensionless form of the model Hidalgo and Brosilow could be extended with the dimensionless moment equations. A new dataset is based on the simulation of this extended dimensionless model with six state variables. The dimensionless number-average molecular weight (NAMW) is the ratio of the last two state variables of the model [35]. The relevant variables which determine the value of the NAMW are selected by the MOSSFARM method. The new initial model structure is:

$$y_{k+1} = f([x_{1k}, x_{2k}, \dots, x_{7k}]) \quad (19)$$

where the dimensionless variables are the following:  $y$ : model output NAMW,  $x_1$ : the initiator concentration,  $x_2$ : monomer concentration,  $x_3$ : reactor temperature,  $x_4$ : jacket temperature,  $x_5$ : the first variable in the moment equations,  $x_6$ : the second variable in the moment equations and  $x_7$ : cooling jacket flowrate. If the searching conditions were the followings: number of

partitions (~ clusters) for the output variable: 5, the number of partitions in the regressor variables: 3,  $\sigma = 10\%$ ,  $\gamma = 50\%$ , the MOSSFARM selects the structure  $x_{1k}, x_{3k}$  ("aggregated" from one rule) as first (ordered by the correlation). This result says that the initiator concentration and the reactor temperature determine the number-average molecular weight. If the ordering is based on the number of aggregated rules, the first selected structure is the  $x_{6k}, x_{7k}$ . (aggregated from two rules). The results are summarized in Table 6.

Table 6 Linear and NN model for estimate number-average molecular weight

First structure	Ordering by correlation	Ordering by rule number
	$x_{1k}, x_{3k}$	$x_{6k}, x_{7k}$
MSE of Lin.	0.0103	0.0049
MSE of Lin. free	0.0106	0.576
MSE of NN	0.0001723	0.000416
MSE of NN free	0.2080	0.576

### Conclusions

This paper showed a new model-free, fuzzy association rule mining based method for model structure selection for input-output data-driven models. The results show that the developed tool provides an efficient method for determining the model structure of both linear and neural network based input-output models. Moreover this method is also can be used to selection of the most relevant process variables (feature selection problem). The proposed approach has been implemented as a MATLAB program called MOSSFARM (Model Structures Selection by Association Rules Mining), it will be free available from:

[www.fmt.vein.hu/softcomp](http://www.fmt.vein.hu/softcomp).

### Acknowledgement

This project has been financially supported in part by the Hungarian National Science Foundation OTKA (No. T037600, No. T049534).

### REFERENCES

- LJUNG L.: System Identification. 1987, Prentice Hall
- PETRICK M. H., WIGDOROWITZ B.: A priori nonlinear model structure selection for system identification. Control Eng. Practise, 1997, 5(8), 1053-1062
- WINKLER P.: Optimized Multivariate Lag Structure Selection. Computational Economics, Springer, 2000, 16 (1/2), 87-103
- AKAIKE H.: A new look at the statistical model identification. IEEE Trans. Autom. Control, 1974, 19, 716-723
- LIANG G., WILKES D. and CADZOW J.: Arma model order estimation based on the eigenvalues of the covariance matrix. IEEE Trans. Signal Process, 1993, 41 (10), 3003-3009
- SCHWARZ G.: Estimating the dimension of a model. Annals of Statistics, 1978, 6, 461-464
- HANNAN E. J., QUINN B. G.: The determination of the order of an autoregression. Journal of the Royal Statistical Society Series B 1979, 41, 190-195
- HIDALGO J.: Consistent order selection with strongly dependent data and its application to efficient estimation. Journal of Econometrics, 2002, 110, 213-239
- SHIBATA R.: An optimal selection of regression variables. Biometrika, 1981, 68, 45-54
- PÖTSCHER B. M.: Model selection under nonstationarity: autoregressive models and stochastic linear regression models. Annals Statistics, 1989, 17, 1257-1274
- GEWEKE J., MEESE R.: Estimating regression models of finite but unknown order. International Economic Review, 1981, 22, 55-70
- SHIBATA R.: Selection of the order of an autoregressive model by Akaike's information criterion. Biometrika, 1976, 63, 117-126
- SHIBATA R.: Asymptotic efficiency selection of the order of the model for estimating parameters of a linear process. Annals of Statistics, 1980, 8, 147-164
- HANNAN E. J.: The estimation of the order of an ARMA process. Annals of Statistics, 1980, 1071-1081
- PÖTSCHER B. M.: Effects of model selection on inference. Econometric Theory, 1991, 7, 163-185
- GEORGE E. I.: The variable selection problem. Journal of the American Statistical Association, 2000, 95, 1304-1308
- AGUIRRE L. A., BILLINGS S. A.: Improved structure selection for nonlinear models based on term clustering. Int. J. Control, 1995, 62, 569-587
- AGUIRRE L. A., MENDES E. M. A. M.: Global nonlinear polynomial models: Structure, term clusters and fixed points. Int. J. Bifurcation Chaos, 1996, 6, 279-294
- MENDES E. M. A. M., BILLINGS S. A.: An alternative solution to the model structure selection problem. IEEE Trans. Syst. Man Cybernetics, Part A: Syst. Humans, 2001, 31 (6), 597-608
- KORENBERG M., BILLINGS S. A., LIU Y., MCOILROY P.: algorithm for nonlinear stochastic systems. Int. J. Control, 1988, 48, 193-210
- ABONYI J.: Fuzzy Model Identification for Control, 2001, Birkhauser, Boston

22. PETROVIC I., BAOTIC M., PERIC N.: Model structure selection for nonlinear system identification using feedforward neural network. International Joint Conference on Neural Networks (IJCNN'00), 2000, 1, 53-57
23. YU D. L., GOMM J. B., WILLIAMS D.: Neural model input selection for a MIMO chemical process. Eng. App. Of Artificial Intelligence, 2000, 13, 15-23
24. MENOLD P. H., ALLGÖWER F., PEARSON R. K.: Nonlinear structure identification of chemical processes. Computers chem. Engng., 1997, 21, 137-147
25. LENDASSE A., SIMON G., WERTZ V., VERLEYSSEN M.: Fast bootstrap methodology for regression model selection. Neurocomputing, 2005, 64, 161-181
26. AHMAD R., JAMALUDDIN H., HUSSIAN M. A.: Model structure selection for a discrete-time non-linear system using a genetic algorithm, in Proceedings of the I MECH E Part I Journal of Systems & Control Engineering, 2004, 85-98
27. METENIDIS M. F., WITCZAK M., KORBICZ J. X.: A novel genetic programming approach to nonlinear system modeling: application to the DAMADICS benchmark problem. Eng. App. Of Artificial Intelligence, 2004, 17, 363-370
28. HONG X., HARRIS C. J.: Nonlinear model structure detection using optimum experimental design and orthogonal least squares. IEEE Trans Neural Networks, 2001, 12 (2) , 435-439
29. BASSO M., GIARRÉ L., GROPPI S., ZAPPA G.: NARX models of an industrial power plant gas turbine. IEEE Trans. on Control Systems Technology , 2005, 13 (4)
30. AGRAWAL R., IMIELINSKI T. and SWAMI A.: Database mining: A performance perspective. IEEE Transactions on Knowledge and Data Engineering, December 1993, 5(6):914-925, Special Issue on Learning and Discovery in Knowledge-Based Databases
31. SJÖBERG J., ZHANG Q., LJUNG L., BENVENISTE A., DEYLON B., GLORENNEC P-Y., HJALMARSSON H., JUDITSKY A.: Nonlinear Black.box modeling in system identification: a unified overview. Automatica 1995, 31(12), 1691-1724
32. GUSTAFSON D. E. and KESSEL W. C.: Fuzzy clustering with fuzzy covariance matrix, In Proceedings of the IEEE CDC, San Diego, 1979, pages 761-766
33. HIDALGO P. M. and BROSILOV C. B.: Nonlinear model predictive control of styrene polymerization at unstable operating points, Comp. Chem. Eng., 1990, 14, 481-494
34. RUSSO L. P. and BEQUETTE B. W.: Operability of chemical reactors: multiplicity behavior of a jacketed styrene polymerization reactor, Chem. Eng. Science, 1998, 53 (1), 27-45