GENERATING HIERARCHICAL FUZZY SYSTEMS [1]

Zdeněk Vlček

ProTyS, Inc.
Czech Republic, Prague
e-mail: zvlcek@protys.cz

Abstract: The proposed type of hierarchical fuzzy systems is based on the rule base hierarchy. A quite new algorithm for designing a hierarchical fuzzy system from a conventional fuzzy system is developed and implemented. The important part is the unification procedure. Redundant intermediate labels are found and subsequently unified. Basic requirements are fulfilled – subsystems of a hierarchical system are simple and transparent. A single high-dimensional and large rule base is replaced by a collection of low-dimensional rule bases with small number of rules. The input-output relation remains unchanged and all properties and propositions of the original fuzzy system are preserved by the hierarchical fuzzy system.

Keywords: Fuzzy systems, fuzzy modelling, hierarchical systems, model reduction.

1. INTRODUCTION

A generated fuzzy system can be in some cases too large, typically for a higher number of inputs. The idea of a hierarchical fuzzy system is to put the input variables into a collection of low-dimensional fuzzy systems, instead of a conventional high-dimensional fuzzy system as in usual case. This approach is based on the rule base hierarchy (rule chaining) according to the figure 1. Firing strengths are in the interconnections. For example the output from the fuzzification block contains the membership degrees of the input sample to input reference fuzzy sets.

At first a conventional fuzzy system is designed. A conventional fuzzy system is designed by an arbitrary method. A fuzzy system is designed from measured data the algorithms based on the clustering analysis or the Wang's algorithm are used. But it can be designed for instance directly by experts. Then this system is decomposed into a hierarchical structure. The next part is the unification procedure. The redundant intermediate labels are found and subsequently unified (merged). It is supposed that a conventional fuzzy system is well designed and this large fuzzy system can be decomposed into a hierarchical fuzzy system. The large rule base is replaced by many smaller and simpler rule bases.

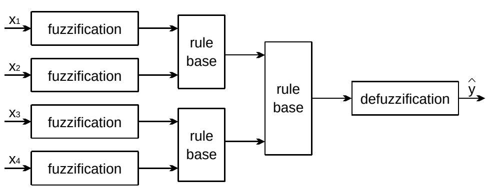

Fig. 1. hierarchical fuzzy system

[1] This work/result/product was supported by the Ministry of Education of the Czech Republic under Project LN00B096.

2. HIERARCHICAL FUZZY SYSTEM

A hierarchical system consists of blocks (subsystems) connected in a specific topology. The parts are fuzzification blocks, defuzzification blocks and blocks with the rule bases.

2.1 Hierarchical fuzzy system structuring

Fuzzification blocks perform a fuzzification. The crisp input enters into each block and the output is the membership degree of this input sample to each input reference fuzzy set. The output of the fuzzification block is a vector with firing strengths (membership degrees). The size of the output vector is equal to the number of input reference fuzzy sets of this input. I.e. the vector contains firing strengths of input to all membership functions of this input $[A_i^1(x_i), \ldots, A_i^n(x_n)]$.

Defuzzification blocks perform the defuzzification procedure. Thus the input into the defuzzification block is a vector of membership degrees – firing strengths of the output reference fuzzy sets. The output from this block is a crisp value computed as: $\hat{y} = \sum \phi^j b^j / \sum \phi^j$, where b^i is the parameter of ith output singleton.

Rule base blocks perform the logic part of the inference mechanism. A rule base block is often denoted as a subsystem. Simple rules in the form **If** x_1 *is* A_1^i**and**$\ldots$ **and** x_m *is* A_m^i**then** y *is* b^i are supposed. Hence the operator **and** in the antecedent part is used only. The product is used for the operator **and**.

Because fuzzified values already income into rule bases the product operation is performed only. The firing strength of the ith subrule is equal to $\phi^i = \prod_{k=1}^{m} \phi_k^i$, where m denotes the number of inputs in the subsystem. If more that one rules have the same conclusion (output reference fuzzy sets), the firing strength corresponding to this output reference fuzzy set is equal to the sum of firing strengths of these rules. Firing strengths of jth output reference fuzzy set is $\phi^j = \sum_i \phi^i$, where i denotes the index of the subrules with the same jth output reference fuzzy set. It results from the used defuzzification algorithm. The inference mechanism of the hierarchical system with this evaluation of the subrules is identical with the original inference mechanism of the conventional system. Hence the firing strength of jth output reference fuzzy set is equal to: $\phi^j = \sum_i \prod_{k=1}^{m} \phi_k^i$

2.2 Topology and notation

A simple topology of the hierarchical system is used. A topology can be in the form of a tree, the output of each subsystem can be used as an input of subsystem in next layers and the connections cannot branch.

A brief rule base format is established here instead of the standard notation. Rows are rules, reference fuzzy sets are in the columns and the numbers denotes the reference fuzzy sets. A consequent part is behind the pipe. For example there is the rule base in the standard notation:

R_1 : **if** x_1 is A_1 **and** x_2 is B_1 **and** x_3 is C_1 **then** y is b^1

R_2 : **if** x_1 is A_1 **and** x_2 is B_1 **and** x_3 is C_2 **then** y is b^2

R_3 : **if** x_1 is A_1 **and** x_2 is B_2 **and** x_3 is C_1 **then** y is b^1

R_4 : **if** x_1 is A_1 **and** x_2 is B_2 **and** x_3 is C_2 **then** y is b^2

R_5 : **if** x_1 is A_2 **and** x_2 is B_1 **and** x_3 is C_1 **then** y is b^2

R_6 : **if** x_1 is A_2 **and** x_2 is B_1 **and** x_3 is C_2 **then** y is b^3

R_7 : **if** x_1 is A_2 **and** x_2 is B_2 **and** x_3 is C_1 **then** y is b^1

R_8 : **if** x_1 is A_2 **and** x_2 is B_2 **and** x_3 is C_2 **then** y is b^2

This rule base is described as in the figure 2:

```
R1 : 1 1 1 | 1
R2 : 1 1 2 | 2
R3 : 1 2 1 | 1
R4 : 1 2 2 | 2
R5 : 2 1 1 | 2
R6 : 2 1 2 | 3
R7 : 2 2 1 | 1
R8 : 2 2 2 | 2
```

Fig. 2. rule base in the brief format

3. DECOMPOSITION

The decomposition should fit a given structure (topology). Hence the required topology must be known or it is estimated by a problem analysis. Parameters of the fuzzy system are kept and they are not used for decomposition and unification procedures. Only the rule base is decomposed.

Basic principles of the decomposition into a hierarchical fuzzy system is described here: Assume that a rule of the conventional fuzzy system has the following form:

R_1 : **if** x_1 is A_1 **and** x_2 is B_1 **and** x_3 is C_1 **then** y is b^1 (1)

Assume the required topology of the hierarchical rule base is the figure 3. Hence the two fuzzified inputs go into the first subsystem in the first layer. This fact can be denoted by the parenthesis:

if $\Big($ x_1 is A_1 **and** x_2 is B_1 $\Big)$ **and** x_3 is C_1 **then** y is b^1 (2)

The part of the fuzzy rule in the parentheses can be extracted as a antecedent part of the fuzzy rule of the subsystem in the first layer and the consequent part z is D_1 is added. This consequent part substitutes the extracted part in the original rule and this rule is located in the subsystem in the second layer.

$$\textbf{if } x_1 \text{ is } A_1 \textbf{ and } x_2 \text{ is } B_1 \textbf{ then} z \text{ is } D_1 \qquad (3)$$

$$\textbf{if} z \text{ is } D_1 \textbf{and } x_3 \text{ is } C_1 \textbf{ then } y \text{ is } b^1$$

The first subrule is contained in the subsystem in the first layer and the second subrule becomes the rule of the subsystem in the second layer. The new variable z corresponds to the interconnection between these subsystems (figure 3). The proposition z is D_1 substitutes the proposition x_1 is A_1 **and** x_2 is B_1 in the original rule. Hence the proposition z is D_1 occurs in the consequent part of the subrule of the subsystem in the first layer and in the antecedent part of the subrule of the subsystem in the second layer.

Labels located at the connections between subsystems are special intermediate fuzzy sets. But the membership functions of these reference fuzzy sets are not available and the reference fuzzy sets can be arbitrary, because they mark firing strengths of subrules only. Hence they are denoted as *intermediate labels*.

The disadvantage of the proposed decomposition is that the high number of the intermediate labels arises at the interconnection between the subsystem and consequently the high number of rules arises in the rule bases of the subsystems.

3.1 Subsystem extraction

A subsystem extraction is shown in the example, because mathematical formalism is not transparent.

The simple equation $y = x_1(1 - x_2) + x_3$; $x_1, x_2, x_3 \in (0, 1)$ was used for generating data and the fuzzy system was generated by the Wang's algorithm. The rule base in the brief notation is in the figure 2. The required topology of the hierarchical rule base is in figure 3.

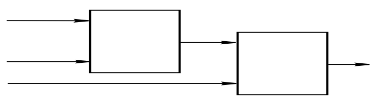

Fig. 3. required topology

There are two fuzzified inputs in the first subsystem and the first and second columns in the original rule base correspond to these inputs (see figure 3.). Thus these columns are extracted from the original rule base. It is depicted by the grey color in the figure 4. This extracted part creates the antecedent part of the rule base of the subsystem in the first layer.

The redundant parts of subrules (rows) are deleted. The n_r diverse antecedent parts of the rules are kept. No information about the consequent part is available, thus it can be generated arbitrary. For each rule one intermediate label is

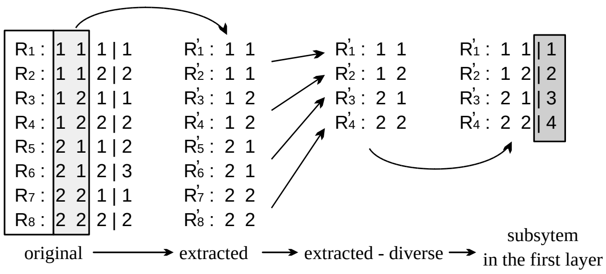

Fig. 4. decomposition procedure

generated, i.e. the output label D_i is generated for the ith rule R'_i. The generated consequent part of the subrule is depicted by the dark grey box in the figure 4. The rules of the subsystems are marked as R'_i.

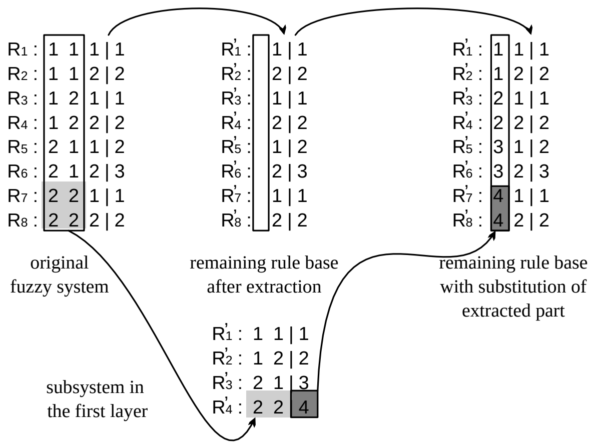

Fig. 5. subsystem extraction

The original rule base is changed too. Instead of columns, which were extracted, the column with labels corresponding to the extracted part is placed. In each row the removed part is substituted by the label from the consequent part of the rule from the newly generated subsystem with the same antecedent part as the removed part. This replacement is sketched in the figure 5. The result of the decomposition is displayed in the figure 6.

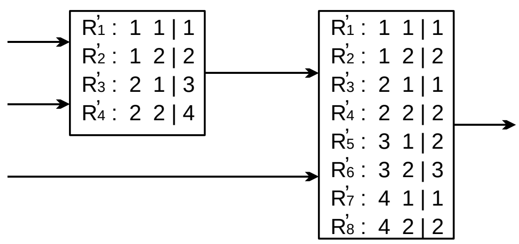

Fig. 6. decomposed fuzzy system

3.2 Decomposition algorithm

The decomposition procedure for the more extensive hierarchical systems is processed from the first layer. All the rule bases of the subsystems in the first layer are extracted from the original rule base. Then all rule bases of the subsystems in the next layer are extracted from the remaining rule base and finally the rule base of the subsystem in the last layer remains. The steps of the algorithm are described in detail in the previous section; the algorithm is as follows:

(1) Algorithm starts from the first layer $l = 1$.
(2) Algorithm starts from the first subsystem in the layer $s = 1$.
(3) Extract the sth subsystem in the lth layer.
    (a) Change the basic rule base. The columns corresponding to the sth subsystem are extracted.
    (b) Create the antecedent parts of the newly generated rule base from extracted columns.
    (c) Remove redundant rows from this incomplete rule base.
    (d) Create consequent part of this rule base. The label D_i is generated for the ith rule R_i'.
    (e) Placed new column instead of the first removed column of the remaining (basic) rule base. The column with labels, which substitute extracted part, is placed there.
(4) The extraction of the sth subsystem in the lth layer is done.
(5) Go to the step 3. to extract all subsystem in the lth layer.
(6) Go to the step 3. to perform all layers except the last layer.
(7) Remaining rule base is rule base of subsystem in the last layer.

4. UNIFICATION

The disadvantage of decomposed fuzzy systems is the high number of the intermediate labels at the interconnections between subsystems and consequently the high number of the rules in the rule bases. Different intermediate labels in the subsystems are generated for each rule. Rule bases of the subsystems in next layers are growing. It is supposed that the numbers of intermediate labels is too high and some redundant labels were generated on the interconnection between subsystems. These redundant labels can be unified into one label. The task is to find the redundant intermediate labels.

4.1 Redundancy

Redundancy of the intermediate labels is defined in this way. Two intermediate labels are *redundant* if this pair of the intermediate labels can be substituted by one intermediate label and the input-output relation does not change.

4.2 Redundancy test

The redundancy test of the two intermediate labels is as follows. The label A is *redundant* to the label B if A is replaced by B and no new rule is generated in the subsystem. Changed rules is already contained in the rule base of the subsystem before replacing.

The rule base of the composed system (input-output relation) does not change if the unified intermediate labels are fully redundant. Two intermediate labels A and B are *fully redundant*, if the intermediate label A is redundant to the label B and vice-versa the label B is redundant to the label A.

On the contrary two intermediate labels are *partially redundant*, if the intermediate label A is redundant to the label B but the label B is not redundant to the label A. This case is called the partial redundancy. If the term redundancy is used, the meaning is always full redundancy.

The test will be described by the example. The decomposed system, which will be simplified, is in the figure 6. The redundancy on the interconnection between subsystems is tested. Thus the redundancy of the first input in the subsystem located in the second layer is tested here.

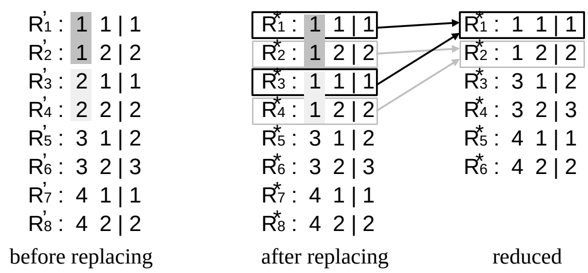

Fig. 7. redundancy of intermediate labels

For example the labels marked as 1 and 2 are tested, the label 2 is replaced by label 1. The rule bases of the subsystem in the second layer are in the figure 7. No new rule arises and hence the label 2 is redundant to the label 1. Further if the label 1 is replaced by the label 2, no new rule is again generated and the label 1 is redundant to the label 2. Thus the intermediate labels 1 and 2 are fully redundant.

If the label 4 is replaced by the label 3 the rule bases before and after replacing are in the figure 8. New rules appear. The rules R_7^* and R_8^* in the rule base after replacing are not contained in the original rule base of the subsystem. It means that the label 4 is not redundant regarding to the label 3. In the same way new rules arise after replacing the intermediate label 3 by 4. These intermediate labels are even not partially redundant.

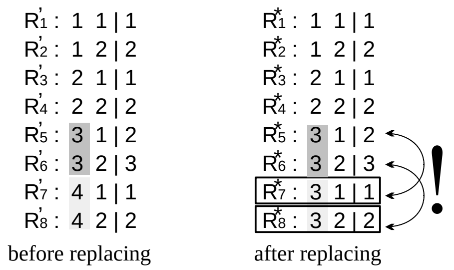

Fig. 8. redundancy test of intermediate labels

4.3 Non-commutability

In the pervious example it is important that the newly generated rules have the same antecedent part as some other rules, but they have different consequent part. The consistency of the rule base is corrupted. The changed rule R_7^* is inconsistent with the rule R_5^* and changed rule R_8^* is inconsistent with the rule R_6^*. The label 4 is strictly non-commutable regarding to the label 3. The new term strictly non-commutable is established here.

Two intermediate labels A and B are *strictly non-commutable*, if after replacing the label A by the label B or vice-versa the new rule is generated in the subsystem and the consistency of rule base is corrupted. Some new generated (changed) rule is inconsistent with any original rule. The strange case can occur if two intermediate labels are partially redundant and strictly non-commutable at the same time.

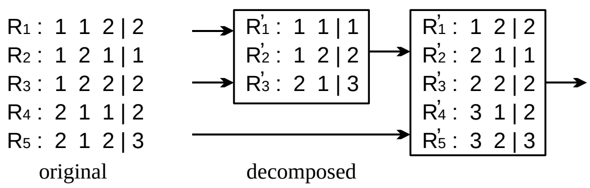

Fig. 9. example of the partial redundancy

4.4 Unification of intermediate labels

An *unification* of intermediate labels is possible if two intermediate labels are fully redundant. If two intermediate labels are partially redundant the input-output relation is changed after unification of these labels. But this change has a special quality and it is described in (Vlček, 2004).

If two intermediate labels A and B are unified (merged), the label B is replaced by the label A. It has to be performed in the both sides of the interconnections, i.e. in the subsystem, where the redundancy was tested, and in the subsystem in the previous layer (the consequent labels are unified here). In the subsystem, where the redundancy was tested, the same rules can arise. Only one of these identical rules is kept. The example of this procedure is depicted in the figure 10

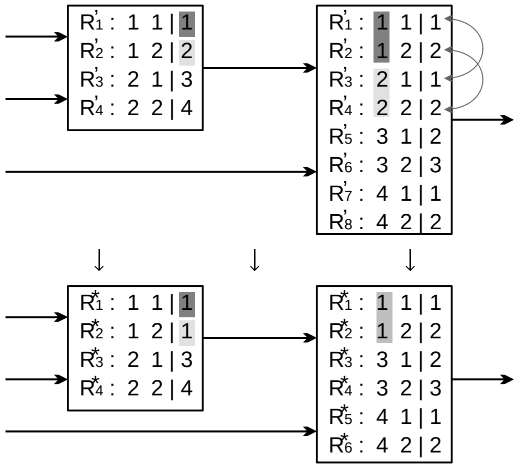

Fig. 10. unification of intermediate labels

4.5 Unification algorithm

A unification procedure has to try to unify (merge) all pairs of intermediate labels at all interconnections between the subsystems. The procedure of the unification is as follows:

(1) It starts from the subsystem in the last layer.
(2) The first input is chosen. The input connected with an output of another subsystem can be chosen only.
(3) The two intermediate labels are chosen and the test of the redundancy of these intermediate labels is performed.
(4) If the intermediate labels are fully redundant, the intermediate labels are unified.
(5) All pairs of intermediate labels are tested.
(6) All possible inputs of the performing subsystems are used for unification procedure.
(7) All subsystems in the layer are inspected.
(8) The procedure goes into previous layer to try unify intermediate labels in the subsystems in new layer.

It is suitable the shift the names (numbers) of the intermediate labels to remove the spaces (omitted numbers).

The result of the unification procedure for the example depicted in the figures 5 and 6 is shown in the figure 11. The number of intermediate labels decreases from 4 to 2 and the total number of rules decreases from 12 to 8.

The rule bases of this example are rewritten into the common form. The designed hierarchical fuzzy system is the model described by the simple equation $y = x_1(1 - x_2) + x_3$; $x_1, x_2, x_3 \in (0, 1)$. Thus the subsystem in the first layer should correspond to the equation $z = x_1(1 - x_2)$ and the subsystem

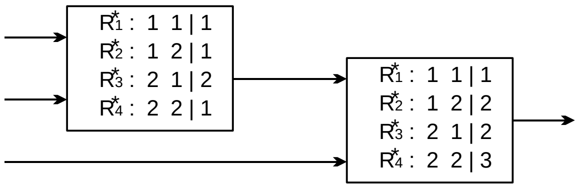

Fig. 11. result of the unification procedure

in the second layer to the equation $y = z + x_3$. The rule base of the subsystem in the first layer is:

R_1^* : **if** x_1 is A_1 **and** x_2 is B_1 **then** z is D_1

R_2^* : **if** x_1 is A_1 **and** x_2 is B_2 **then** z is D_1

R_3^* : **if** x_1 is A_2 **and** x_2 is B_1 **then** z is D_2

R_4^* : **if** x_1 is A_2 **and** x_2 is B_2 **then** z is D_1

And the rule base of the subsystem in the second layer is as follows:

R_1^* : **if** z is D_1 **and** x_3 is C_1 **then** y is b^1

R_2^* : **if** z is D_1 **and** x_3 is C_2 **then** y is b^2

R_3^* : **if** z is D_2 **and** x_3 is C_1 **then** y is b^2

R_4^* : **if** z is D_2 **and** x_3 is C_2 **then** y is b^3

The relation between the rule bases and equations is evident.

4.6 Relations between original and hierarchical system

The relations between the original and designed hierarchical fuzzy system can be summarized:

(1) Input-output relations of the original system and decomposed (without the unification procedure) system are the same.
(2) Input-output relations of the original system and decomposed system, where the unification of the fully redundant intermediate labels has been performed, are the same.
(3) Input-output relations of the original system and the decomposed system, where the unification of the partially redundant intermediate labels has been performed, are different. This difference is not usually significant, the new rules arise in the uncovered input space (Vlček, 2004).
(4) An unification of the non-commutable intermediate labels is not possible because of the consistency violation.

4.7 Interpretability and transparency

An interesting question is whether is possible to interpret a hierarchical fuzzy system and how to assign a physical meaning to the interconnection between subsystems. The next question, which is even more frequent, is whether the intermediate labels have the physical meaning and how the expert can understand them. The intermediate labels cannot be considered as ordinary reference fuzzy sets, they denote the firing strengths of some parts of the rule base. No information about the intermediate labels is available to set the intermediate reference fuzzy sets. Mainly the information about the mutual positions of the intermediate reference fuzzy sets should be helpful to assign the physical meaning of labels as “small”, “big” and so on. But this assignment depends only on an expert knowledge.

5. CONCLUSION

The algorithms for generating hierarchical fuzzy systems are newly proposed here. It is dealt with the hierarchical rule base only, not with the hierarchically connected complete fuzzy systems.

The algorithm is based on the decomposition of conventional fuzzy systems. The important part is the unification procedure. The redundant intermediate labels are found and unified. Hence the basic expectations and requirements are fulfilled – subsystems of the hierarchical system should be small, simple and transparent. A single high-dimensional with a large rule base is replaced by a collection of low-dimensional rule bases with a low number of rules.

The important fact is that the input-output relation is kept and all properties and propositions of the conventional fuzzy system are preserved also in the hierarchical fuzzy system. Only if the condition of the full redundancy is relaxed to the weaker condition of the partial redundancy in the unification procedure, the input-output relation can change. But the difference is not significant.

Some problems are still open. The first one is in using information about a topology for generating fuzzy systems. The second one is the problem of the interpretability of the intermediate labels.

REFERENCES

Chung, F.L. and J.C. Duan (2000). On multistage fuzzy neural network modeling. *IEEE Transactions on Fuzzy System* **78**, 125–142.

Nauck, D., F. Klawonn and R. Kruse (1997). *Foundations of Neuro-Fuzzy Systems*. Wiley. Chichester.

Vlček, Z. (2004). Clustering Methods and Designing Fuzzy Systems. PhD thesis. CTU Prague, Czech Republic.

Wang, L.X. (1999). Analysis and design of hierarchical fuzzy systems. *IEEE Transactions on Fuzzy System* **7**, 617–624.

Ying, Hao, Ed.) (2000). *Fuzzy Control and Modeling: Analytical Foundations and applications*. IEEE Press. New York.