On Fuzzy Rule Interpolation and Radial Basis Functions[1]

László T. Kóczy* **, Domonkos Tikk* ** and Tamás D. Gedeon**

* Budapest University of Technology and Economics, Budapest, Hungary

** Murdoch University, Perth, W.A., Australia

1. Introduction

One of the most important and intriguing goals of Artificial Intelligence is to construct methods, models and algorithms to cope with very complex systems, often lacking known analytical models. Such approaches have attempted to follow examples of natural intelligence and biological organisms. The classical AI approach was based on symbolic logic and its tautologies, such as *Modus Ponens*, *Hypothetical Syllogism*, etc. The former is the simple logical scheme that works when an implication and its antecedent part are known to be true; then, its consequent part is concluded to also be true:

$$\frac{\begin{array}{l} A \to B \\ A \end{array}}{B}$$

A great disadvantage of this approach is that knowledge stored in the form of implications does not relate directly to the structure of the state space of the problem in question. Most physical phenomena can be modeled in a state space where each dimension is a subset of R^k, usually a hyper interval $X = X_1 \times X_2 \times \ldots \times X_k = \prod_{i=1}^{k} [L_i, U_i]$. Consequently, there is an ordering $<_i \subset X_i \times X_i$ in every dimension of *X*, and a partial ordering in *X* itself, so that $x_1 \; \pi \; x_2 \Leftrightarrow x_{i1} <_i x_{i2}; \forall i; x_1 = \begin{bmatrix} x_{11} \\ x_{12} \\ \ldots \\ x_{1k} \end{bmatrix}, x_2 = \begin{bmatrix} x_{21} \\ x_{22} \\ \ldots \\ x_{2k} \end{bmatrix} \in X$. Further, there is a norm $\|\cdot\|$ in *X*, usually the Euclidean norm. When applying a symbolic logical approach to a system in the state space *X*, it is necessary to quantify this space, so that every “quantum” or interval is denoted by a symbol (see Figure 1). The symbols A_{ik} represent a whole interval in x_i, and there is no way to differentiate among values within the same quantum or interval. This fact results in a certain coarseness of the model that can be refined by applying smaller intervals and more symbols. It is a more essential problem, however,

---

[1] This research was supported by the Australian Research Council, the Hungarian Ministry of Education (MKM) FKFP 0235/1997 and FKFP 0422/1997 and the National Science Research Fund (OTKA) 019671 and 030655.

that the relative position of these symbols to each other cannot be expressed at this level, e.g. the fact that $A_{im} < A_{im+1} < A_{im+2}$. ("<" in the sense that the typical values or centres are ordered according to $<_i$). The same applies to relations like $\|A_{im} - A_{im+1}\| < \|A_{im} - A_{im+2}\|$, as the norm and the distance between two logical symbols are senseless without the additional information referring to the metrics of the space ("<" meaning here the usual inequality between the two norms).

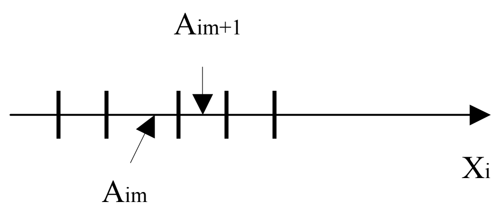

Figure 1. Quantification and symbol correspondence

The knowledge base consisting of symbolic rules has the form

$$R = \{R_i \mid i = 1...r\}, R_i = A_i \to B,$$

where $A_i = \prod_{i=1}^{k} A_{im}$, and each A_{im} is a logic symbol in X_i. A fact or observation in such a system has the form $A_j = \prod_{i=1}^{k} A_{jm}$ (Π denoting Cartesian product), and it must match exactly with the antecedent in one of R_i in order to make it possible to apply e.g. the *Modus Ponens* in order to obtain a conclusion (necessarily equal to one of the rule consequents).

It was an essential step towards establishing a means for describing the internal structure of the state space, when the use of fuzzy symbols was proposed [1]. By these, based on the information present in the membership functions attached to each symbol, the intervals close to each other were merging into single "fuzzy intervals" or "fuzzy numbers" By the possibility of these membership functions' partial overlapping, the partial semantic overlapping of the meaning of "close by" symbols could be formally expressed (see Figure 2).

A similar feature is inherent for artificial neural networks, where sub-symbolic information is contained in the activation functions and weights.

It is obvious that sub-symbolic models have a smaller number of symbols, and consequently, a smaller number of rules in the base, although in this case each symbol has a much more complicated description in the form of an additional membership function. Let us assume that there are T_i symbols used in dimension X_i. Then the number of rules in the base is $|R| = \prod_{i=1}^{k} T_i$. Let $T_i \leq T, \forall i$, then the number of rules in the base is at most $|R| \le T^k$. Let c be the rate of reduction of the number of symbols when fuzzy membership functions are introduced, and let e be the rate of extension of the amount of information belonging to a single symbol. Then, instead of approximately T^k pieces of information, only $\left(\frac{T}{c}\right)^k e$ will be needed. However, $\left(\frac{T}{c}\right)^k e < T^k$ holds, whenever $c > e^{\frac{1}{k}}$, i.e., when k is large enough ($k > \frac{\log e}{\log c}$). Obviously, the introduction of fuzzy symbols leads to the reduction of computational complexity when the number of state space dimensions is large enough.

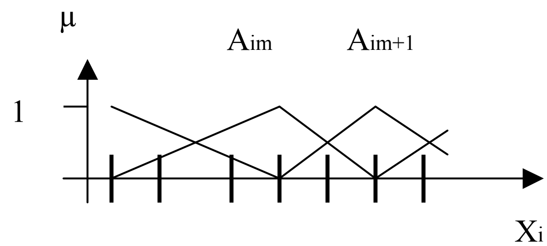

Figure 2. Symbols with fuzzy membership functions

The reason for this effective reduction is that the use of fuzzy membership functions utilizes the knowledge about the structure of the space and introduces a kind of interpolation.

Calculations about the explicit interpolation functions with triangular and trapezoidal membership functions in various fuzzy reasoning and control systems can be found in [2].

A common feature of such fuzzy rule bases is that the antecedents form a collection of fuzzy sets, where their kernels (either single points or intervals) represent typical values of *x*, and between these kernels the membership functions cover the space, so that for every $x \in X$ there is at least one such antecedent, which is true over *x* at least to a fixed degree $\alpha > 0$. (Usually $\alpha \ge 0.5$.) If this latter condition is satisfied, it is true that there is at least one rule for every *x*, which is at least as true as it is false. (If $\alpha > 0.5$ is satisfied then there is at least one rule which is more true than false.) If

$$\forall x \in X, i = 1...k, \exists j \; (\text{j} \in \{1...r\}) : A_{ji}(x_i) \ge \alpha > 0$$

then $\{A_{ji}\}(j = 1...r)$ form an α-cover of X_i. If also

$$\forall x \in X, \exists j \; (\text{j} \in \{1...r\}) : A_j(x) \ge \alpha > 0$$

is fulfilled, then $\{A_j\}(\text{j} = 1...\text{r})$ is an α*-cover of X*. In the original Mamdani-algorithm [3] e.g., the antecedents formed a cover for approximately α=0.75. In most current applications however $\alpha = 0.5$ is satisfactory.

If $\{R_j\}$ is a *full set of rules* in the sense that for arbitrary rules $R_{j1}, R_{j2},..., R_{jk} \in R$, if $A_{jl,i}$ denotes the *i*th projection of A_{jl}, the antecedent of rule R_{jl}, to the sub-space X_i (*l*=1…*k*), then

$$\exists m \in \{1...r\} : A_m = A_{j1,1} \times A_{j2,2} \times ... \times A_{jk,k} .$$

This means that the rule antecedents form a "grid" in *X* as illustrated in Figure 3. Full rule bases contain $|R| = \prod_{i=1}^{k} r_i$ rules, where r_i stands for the number of different antecedent projections appearing in dimension X_i. It is reasonable to use here *k* subscripts, where the *i*th one indicates that the antecedent of the given rule generates the *i*th projection in X_i if all projections are ordered along each component axis. In most practical applications the rule set is full, unless some rules in the original set are omitted in order to decrease the size of the rule base.

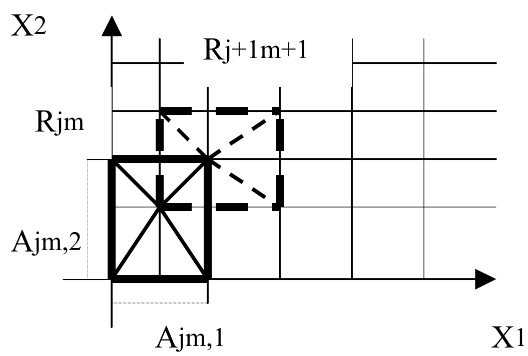

Figure 3. Rules represented by their supports in a full rule base in two-dimensional space

In most cases the antecedents also form a *Ruspini-partition of X*, meaning that

$$\forall x \in X : \sum_{j=1}^{r} A_j(x) = 1 .$$

Full rule bases with the property of Ruspini-partition are usually 0.5-covers.

2. Fuzzy Rule Based controllers

It is obvious that although the introduction of fuzzy membership functions for describing the sub-symbolic information behind the symbols leads to the reduction of the overall computational complexity, whenever *k* is large enough the reduction is not radical [8]. There are further possibilities to decrease the effective value of *T*, by applying rule bases where the antecedents do not form an α-cover of the input state space.

The original idea of reasoning and control within fuzzy rule bases was proposed by Zadeh [4], and was called the Compositional Rule of Inference (CRI) and had the disadvantage of running directly in the *k*-dimensional input variable space, while being able however to describe multi-dimensional membership function distributions of arbitrary shape. Its modified version, the Mamdani-algorithm [3] applied the projections of the antecedents and thus returned to the form of rules $R_i = A_i \rightarrow B$, where

$A_i = \prod_{i=1}^{k} A_{im}$, meaning that only such A_i could be used that were the cylindrical closures of some membership functions A_{ik}, each being of the type $A_{im} : X_m \to [0,1]$. This method offers much better computational speed.

The idea of the Mamdani-algorithm is to use the heights of the intersections of the observation with the antecedents and weighing the consequents with these, e.g. by simply "cutting them" at the given height:

$$B^* = \max_i \left\{ \min\left\{ height\left(A^*(x), A_i(x)\right), B_i(y) \right\}\right\},$$

where $height\left(S(x)\right) = \max_x S(x)$. In the Larsen-method, the modification is that the consequents are multiplied with these heights:

$$B^* = \max_i \left\{ height\left(A^*(x), A_i(x)\right) B_i(y) \right\}.$$

In both cases, y^* is obtained by defuzzification, e.g., in the most popular techniques, calculating the centre of gravity of the modified consequents. This latter means that instead of taking the maximum, in effect the sum of the membership functions will be considered:

$$B^* = \sum_{i=1}^{r} height\left(A^*(x), A_i(x)\right) B_i(y),$$

which is in case of a crisp singleton observation identical with

$$B^*_{eff} = \sum_{i=1}^{r} A_i(x^*) B_i(y).$$

This latter is the most popular formula used for fuzzy control applications.

3. Interpolation in sparse fuzzy rule bases

If for given α the antecedents form no α-cover, the rule base is α-sparse. If they do not form a cover for any $\alpha > 0$, the rule base is sparse. The CRI, the Mamdani-method and its variants, the Takagi-Sugeno method [5], which applies linear functions of x in the consequent part, and all other fuzzy control approaches based on some form of degree of matching between observation and antecedents cannot be used for sparse rule systems.

Instead, a more general notion of the degree of similarity was introduced, based on the family of *α-distances between two convex and normal fuzzy sets* (CNF sets), which is defined as

$$\tilde{d}(A_1, A_2) = \{d_{\alpha C}(A_1, A_2), \alpha \in (0,1], C \in \{U, L\}\} = \left\{ \left\| A_{1\alpha L} - A_{2\alpha L} \right\|, \left\| A_{1\alpha U} - A_{2\alpha U} \right\|, \alpha \in (0,1] \right\},$$

subscripts L and U denoting the minimum and maximum of the respective α-cuts. For simplicity, the distance belonging to fixed α and L or U will be denoted by $d_{\alpha C}(A_1, A_2)$. The conditions for the existence of the fuzzy distance set is that both fuzzy sets are CNF, and that they are comparable in the sense of π.

Using this notion of distance, the fuzzy similarity set can be defined as

$$\tilde{s}(A_1, A_2) = \{s_{\alpha C}(A_1, A_2), \alpha \in (0,1], C \in \{U, L\}\} = \left\{ \frac{1}{d_{\alpha C}(A_1, A_2)} \right\},$$

the elements of the similarity degree set being the reciprocals of the elements of the distance set.

The basic idea of the rule interpolation is formulated in the *Fundamental Equation of Rule Interpolation* (FERI):

$$D(A^*, A_1) : D(A^*, A_2) = D(B^*, B_1) : D(B^*, B_2) .$$

In this equation A^* and B^* denote the observation and the corresponding conclusion, while $R_1 = A_1 \to B_2, R_2 = A_2 \to B_2$ are the rules to be interpolated, such that $A_1 \pi A^* \pi A_2$ and $B_1 \pi B_2$. If D denotes the Euclidean distance between two symbols, the solution for B^* results in simple linear interpolation. If $D = \tilde{d}$ (the fuzzy distance family), linear interpolation between corresponding α-cuts is performed.

A more general form of FERI gives

$$B_\alpha^* = \sum_{i=1}^{r} s_\alpha (A_i, A^*) B_{i\alpha}^*, \alpha \in (0,1],$$

where s_α is some α-cut related similarity degree, e.g., the fuzzy similarity obtained from the reciprocal distances of the α-cuts. This similarity can be considered as an extended "degree of matching", and these similarity degrees replace the degrees $A_i(x^*)$ or $height(A_i \wedge A^*)$ which are used in the Mamdani-, Larsen-, (Takagi-) Sugeno- and other classical fuzzy reasoning algorithms. These methods can be substituted by fuzzy interpolation (*KH-interpolation* [6]), which also functions in sparse rule bases. The Mamdani-type algorithms are thus replaced by

$$B_{\alpha C}^* = \frac{\sum_{i=1}^{r} \frac{1}{d(A_{\alpha C}^*, A_{i\alpha C})} B_{i\alpha C}}{\sum_{i=1}^{r} \frac{1}{d(A_{\alpha C}^*, A_{i\alpha C})}},$$

where the normalized degree of similarity for fixed α and C is the reciprocal distance of the observation from the corresponding antecedent, divided by the sum of all these distances. It must be noted, however, that B^* reconstructed directly from the above α-cuts does not always exist, as various abnormalities in the shape of the conclusion might necessitate some transformations, which eventually result in obtaining sub-normal conclusions (cf. [6]). It should be noted that in practical applications, it is enough to do calculations for $\alpha \in \mathrm{B}$, the breakpoint set of the membership functions, which is four points altogether [7] in the case of the most widely applied trapezoidal functions $\{0,1; L, U\}$.

In all these approaches, the approximated mapping is "fuzzy set to fuzzy set":

$$\tilde{\Phi} : \tilde{P}(X) \to \tilde{P}(Y),$$

where $\tilde{P}(Z)$ denotes the fuzzy power set of *Z*, i.e., all fuzzy sets of the universe of discourse *Z,* so that a fuzzy observation is always mapped into a fuzzy conclusion by the respective rule base and inference engine: $\tilde{\Phi}(A^*) = B^*$. In the special case where the observation is a crisp singleton, we have $\tilde{\Phi}(x^*) = B^*$. In control applications the conclusion set should always be defuzzified, so that $y^* = defuzz(B^*)$. With both crisp singleton inputs and outputs, we get $y^* = defuzz(\widetilde{\Phi}(x^*)) = \Phi(x^*)$. (Some of the most important Φ-s are explicitly given in [2].) In this sense, fuzzy rule based controllers can be considered as multivariate real function approximators.

The respective interpolative extension for the equation for the Takagi-Sugeno-algorithm is

$$y^* = \frac{\sum_{i=1}^{r} s(A_i, A^*)(b_{i1}x^* + b_{i0})}{\sum_{i=1}^{r} s(A_i, A^*)},$$

where $\forall i : b_{i1} = 0$ in the case of the Sugeno-controller. Here, the similarity degree is an overall one that can be derived from the reciprocal of an overall distance of the two fuzzy sets in question, e.g. by defining it as

$$s(A_i, A^*) = \frac{1}{\int\limits_{\alpha_L \in (0,1] \wedge \alpha_U \in (0,1]} d(A^*_{\alpha C}, A_{i\alpha C}) d\alpha}.$$

In the case of trapezoidal membership functions this can be simplified to

$$s(A_i, A^*) = \frac{4}{d(A^*_{0L}, A_{i0L}) + d(A^*_{1L}, A_{i1L}) + d(A^*_{1U}, A_{i1U}) + d(A^*_{0U}, A_{i0U})},$$

because both the left and right pairs of flanks of the antecedents define trapezoidal areas.

This latter interpolation has the advantage of obtaining directly defuzzified consequents, which fact eliminates the problem of the abnormal conclusion shape. This interpolation method will be investigated in more detail in the future.

The investigation of the KH-interpolation has shown that using a more radical similarity degree improves the convergence properties tremendously. If s is defined as

$$s(A_{i\alpha C}, A^*_{\alpha C}) = \frac{\frac{1}{d^n(A^*_{\alpha C}, A_{i\alpha C})}}{\sum_{i=1}^{r} \frac{1}{d^n(A^*_{\alpha C}, A_{i\alpha C})}},$$

where n is the dimension of the input sets, it will be guaranteed that when the consequent membership functions are known with ε accuracy, the conclusion function will be also calculable with $\delta(\varepsilon)$ accuracy [9]. Due to this result, the input-output function of the general KH-interpolation obtained by means of the radical similarity defined above,

$$B^*_{\alpha C} = \frac{\sum_{i=1}^{r} \frac{1}{d^n(A^*_{\alpha C}, A_{i\alpha C})} B_{i\alpha C}}{\sum_{i=1}^{r} \frac{1}{d^n(A^*_{\alpha C}, A_{i\alpha C})}},$$

possesses the universal approximation property for every fixed α, i.e., every continuous function defined on a compact set can be approximated arbitrarily accurately with respect to the $L_p, (p \in \{[1,\infty)\} \cup \infty)$ norm with functions of this form. By this, the mathematical stability of the method can be formulated simply as follows: If the antecedents or the observation of a rule base change only slightly, it does not cause a significant change in the conclusion.

General KH-interpolation has strong connections with the so-called Balázs-Shepard interpolator operator well known in approximation theory. This operator is also capable of approximating real valued continuous functions arbitrarily well, where the convergence of the approximation depends on the modulus of continuity (a sort of smoothness factor) of the approximated real function. It has been shown that for every fixed value of α, the function generated by the general KH interpolation is a Balázs-Shepard operator. On the other hand, for practically important piecewise linear membership functions it can be proved that it is sufficient to calculate the conclusion for the elements of the breakpoint set of the membership functions, and in the regions between the breakpoint levels the conclusion can be approximated well by linear functions. Using these results, general KH-interpolation can be considered as a generalized function approximator being able to approximate fuzzy-to-fuzzy mappings on the Cartesian product of compact universes of discourse.

There are further methods for rule interpolation that we do not discuss here in detail (cf. [10], [11], [12]).

4. Radial Basis Functions in neural networks

An alternative approach where sub-symbolic information is added is the artificial neural network model. Three-layer feedforward neural networks using *radial basis functions* (RBF) are similarly function approximators as are rule based fuzzy systems. Park and Sandberg proved that any continuous function f could be arbitrarily well approximated on a compact subset $X \subset \Re^k$ in L_∞ norm [13]:

$$f(x) = \sum_{i=1}^{r} \lambda_i \varphi(\|x - \xi_i\|),$$

where $\varphi : \Re^+ \to \Re$, ξ_i are called the centres of the functions φ_i and λ_i are real numbers. If there are r known points of the function to be approximated, $\{y_i = F(x_i), i = 1...r\}$, by substituting them into the above, a set of linear equations for λ_i is obtained [14]:

$$\begin{pmatrix} y_1 \\ \cdots \\ y_r \end{pmatrix} = \begin{pmatrix} \varphi_{11} & \cdots & {}_{1r} \\ \cdots & & \cdots \\ \varphi_{r1} & \cdots & \varphi_{rr} \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \cdots \\ \lambda_r \end{pmatrix},$$

where

$$\varphi_{ij} = \varphi(\|x_i - \xi_j\|).$$

By denoting $\Phi = \begin{pmatrix} \varphi_{11} \cdots \varphi_{1r} \\ \cdots \quad \cdots \\ \varphi_{r1} \cdots \varphi_{rr} \end{pmatrix}$, $\lambda = \begin{pmatrix} \lambda_1 \\ \cdots \\ \lambda_r \end{pmatrix}$, $y = \begin{pmatrix} y_1 \\ \cdots \\ y_r \end{pmatrix}$, this is simply written as $y = \Phi\lambda$.

From here, the solutions for λ_i can be obtained by

$$\lambda = \Phi^{-1} y$$

if Φ is regular. For a rather large set of various functions it has been proved that Φ is always regular (see e.g. [15]), if $\xi_i \, (i = 1,\dots,r)$ are distinct vectors. For example, $x, x^2 \log x, x^3$, $(x^2 + c)^{\alpha}$ $(c > 0, 0 < \alpha < 1)$ and e^{-cx^2} $(c > 0)$. These functions belong to two different groups: *localized* functions which satisfy $\varphi(\rho) \to 0$, when $\rho \to 0$, and so Φ is positive definite, while *non-local* functions $\varphi(\rho)$ become unbounded, when $\rho \to 0$, and so Φ is not positive definite, having *r*–1 negative eigenvalues and only one positive one [16].

The RBF approach can be extended to multiple output mappings, and more general approximation, where the number of known points rather larger than the number of functions φ_i, using the Moore-Penrose pseudo inverse (cf. [14], [17]):

$$\Phi^{+} = (\Phi^{T}\Phi)^{-1}\Phi^{T},$$

where $\Phi^{+}\Phi = E_r$, the × identity matrix.

Three-layer feedforward neural networks using radial basis functions and with fully interconnected layers are similarly suitable function approximators as the fuzzy systems discussed in section 2. The number of units in the input layer should be *k,* the number of input variables. Each connection between the input layer and the hidden layer should be assigned a value ξ_{ij}, where $i \in \{1 \dots r\}$ indicates the subscript of the corresponding unit centre, and $j \in \{1 \dots k\}$ refers to the respective component. (In the basic case *r*=*k*.) The *i*th hidden unit computes the norm $\|x^* - \xi_i\|$, and its mapping by φ. The output layer consists of a single unit in the one output variable case, and this unit is connected to the *i*th hidden unit by a weight λ_i, and is linear, so it generates the value $f(x^*) = \sum_{i=1}^{r} \lambda_i \varphi(\|x^* - \xi_i\|)$ (See Figure 4.). Multiple unit output layers can similarly implement multiple output variable cases.

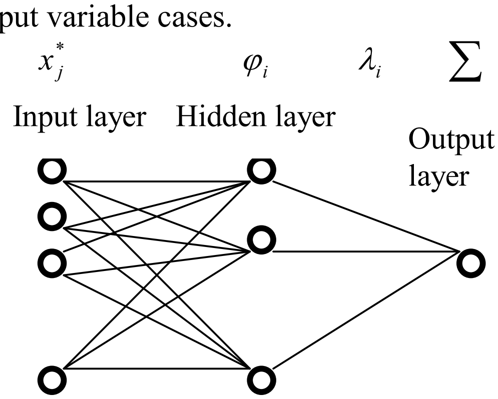

Figure 4. Three-layer feedforward neural network with RBF activation functions in the hidden layer

5. The connection of fuzzy algorithms and RBF

Comparing the formulae of various fuzzy interpolation techniques and RBF based feedforward networks, some similarities can be detected. Fuzzy reasoning and control systems are approximators on two different levels. They approximate primarily fuzzy-to-fuzzy mappings in the following sense:

$$S:\widetilde{P}(X)\rightarrow\widetilde{P}(Y),$$

S is the generalization of the notion of a set-to-set function. As every fuzzy set is described by a membership function, for every given $\alpha\in(0,1]$ and fixed C ("lower or upper end"), S realizes a function

$$S_{\alpha C}:X\rightarrow Y,$$

and fuzzy interpolation means the solution of an ordinary interpolation problem for every $S_{\alpha L/U}$. The r rules define r vectors $(A_{i\alpha C},B_{i\alpha C})\in X\times Y$, so that the interpolation problem is equivalent to fitting a function to these r control points. Fuzzy interpolation means finding a family of functions $y=F(x;\alpha,C),\alpha\in(0,1],C\in\{L,U\}$ interpolating these families of points. For a fixed α-cut and C, the problem to solve is identical with that of ordinary interpolation.

On the other hand, if a fuzzy system is considered with crisp observations and defuzzified outputs, the whole system is equivalent to a single approximator. Because of the different defuzzification techniques are rather complicated to be expressed in terms of the approximation error involved, in this context we refer to the explicit formulae of the most frequently used fuzzy controllers [2], but will not discuss this aspect here in detail.

Getting back to the first type of problem, it can be recognized that Mamdani-control has a similarity to RBF approximation, but is less specific concerning the type of functions used.

Obviously, the original Mamdani-method is rather different from the RBF approach, as in $B^*=\max_i\{\min\{height(A^*(x),A_i(x)),B_i(y)\}\}$ not addition but a max operator combines the functions $height(A^*(x),A_i(x))$, and also the "weighting" is done by min rather than multiplication. So, the formula can be written like this:

$$F(x;\alpha,C)=\max_{i=1}^{r}\{\min\{\lambda_i(\alpha,C),A_i(x)\}\},$$

where $\lambda_i(\alpha,C)=B_{i\alpha C}(y)$ if B^* is calculated point by point. There is a striking analogy with the formulae of the RBF-s, especially as in the theory of possibility measures, the max operator replaces the addition and min might play a similar role as multiplication in probability theory. So, the above equation can be interpreted as a *"possibilistic linear combination"* of the consequent function points. An additional essential difference is however the lack of radial behavior.

According to many results, Mamdani-type approximators are universal [18], [19], [20], at least in the sense that their full versions including defuzzification can arbitrarily well approximate an arbitrary continuous function over a compact domain. It is well worth investigating in the future whether this statement remains true for arbitrary continuous fuzzy-to-fuzzy mappings as well.

If Larsen-interpolation is applied with centre of gravity defuzzification, the equivalent conclusion can be written point by point as

$$F(x;\alpha,C)=\sum_{i=1}^{r}\lambda_i(\alpha,C)A_i(x),$$

where $\lambda_i(\alpha,C)$ is the same as in the Mamdani-formula. Here, the only difference to the RBF-case is the non-radial nature of the general antecedent function.

Statement 1.

If in the Larsen-algorithm all antecedent functions are symmetric around the centre of the kernel: $A_i(\xi_i-\delta)=A_i(\xi_i+\delta)$, and further, if $A_i(x)$ differ from each other only in a constant factor, this special fuzzy controller implements an RBF-approximator for every fixed α.

It has been shown in [2], that the transfer function of symmetric Sugeno-controllers is identical with the transfer function of Larsen-algorithm using the centre of gravity method. Exploiting this functional equivalency, we can have the following

Corollary 1.

Sugeno-controllers with symmetric shaped antecedent sets around the centre of the kernel: $A_i(\xi_i-\delta)=A_i(\xi_i+\delta)$, further, if $A_i(x)$ differ from each other only in a constant factor, this special fuzzy controller implements an RBF-approximator for every fixed α, as well.

Considering the results of [20], it would be interesting to investigate whether in general, any convex and normal fuzzy (CNF) membership function could generate a suitable RBF with positive definite matrix Φ, where the only necessary condition for the radial basis function is that $\varphi(0)=1;\varphi(\rho)\to 0,$ when $\rho\to\infty$ and $\varphi(\rho)$ is monotone decreasing.

Considering now the formulae of the linear KH-interpolation,

$$F(x;\alpha,C)=\sum_{i=1}^{2}\lambda_i(\alpha,C)\varphi(x),$$

where $\varphi(x)=\dfrac{1}{d(A^*_{\alpha C}(x),A_{i\alpha C}(x))}$ and $\lambda_i(\alpha,C)=\dfrac{B_{i\alpha C}}{\sum_{i=1}\dfrac{1}{d(A^*_{\alpha C}(x),A_{i\alpha C}(x))}}$, is obviously an RBF-approach for every fixed level and end point.

Similar constructions apply for the general KH-interpolation, where

$$F(x;\alpha,C)=\sum_{i=1}^{r}\lambda_i(\alpha,C)\varphi(x),$$

where $\varphi(x)=\dfrac{1}{d(A^*_{\alpha C}(x),A_{i\alpha C}(x))}$ and $\lambda_i(\alpha,C)=\dfrac{B_{i\alpha L/U}}{\sum_{i=1}^{r}\dfrac{1}{d(A^*_{\alpha C}(x),A_{i\alpha C}(x))}}$, for every fixed level and end point. Analogously, for the stabilized KH-interpolator also similar constructions apply, where we take the *s*th power of the distance function in the previous expressions, where $s \geq n$ and n stands for the number of variables.

Hence we have the following result:

Statement 2.

The general KH-interpolation implements an RBF-approximator for every fixed α. Here, due to the construction of $\varphi(x)$ and $\lambda_i(\alpha, C)$,.the vectors $\xi_i (i = 1,...,r)$ need not to be distinct to assure the regularity of the coefficient matrix Φ.

A very similar statement can be said about the stabilized version of the KH-interpolation, where however the distance function is raised to the power s ($s \geq n$ and n is the number of variables). Consequently we have:

Statement 3.

The stabilized general KH-interpolation also implements an RBF-approximator for every fixed α. Here, due to the construction of $\varphi(x)$ and $\lambda_i(\alpha, C)$,.the vectors $\xi_i (i = 1,...,r)$ need not to be distinct to assure the regularity of the coefficient matrix Φ.

6. Conclusion

In this paper new examples have been given for the approximate functional equivalence of fuzzy and neural models. We pointed out that under certain conditions Larsen type controllers and Sugeno type controllers implement RBF approximation scheme.

We have also shown that the (stabilized) general fuzzy KH interpolation and the RBF approximation scheme (usually implemented by RBF networks); both being considered as universal approximators, can be connected straightforwardly. The general fuzzy KH interpolators can be considered as a special RBF approximation scheme for fixed α-cuts $\alpha \in (0,1]$, and $C \in \{L, U\}$.

Acknowledgement

The authors want to thank to János Levendovszky for his valuable suggestions on the connection of KH-interpolation and RBF functions.

References

[1] L. A. Zadeh: Fuzzy sets, *Information and Control* **8** (3) (1965), 338-353.

[2] L. T. Kóczy and M. Sugeno: Explicit functions of fuzzy control systems, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* **4** (1996), 515-535.

[3] E. H. Mamdani and S. Assilian: An experiment in linguistic synthesis with a fuzzy logic controller, *.Int. J. of Man Machine Studies* **7** (1975), 1-13.
[4] L. A. Zadeh: Outline of a new approach to the analysis of complex systems and decision processes, *IEEE Trans. on SMC* **3** (1973), 28-44.
[5] M. Sugeno and T. Takagi: Fuzzy identification of systems and its applications to modeling and control, *IEEE Trans. on SMC* **15** (1985), 116-132.
[6] L. T. Kóczy and K. Hirota: Approximate reasoning by linear rule interpolation and general approximation, *Internat. J. Approx. Reason.* **9** (1993), 197-225.
[7] Y. Shi and M. Mizumoto: Some considerations on Kóczy's interpolative reasoning method, *Proc. of the FUZZ-IEEE/IFES '95*, Yokohama (1995), 2117-2122.
[8] L. T. Kóczy: Algorithmic aspects of fuzzy control, *International J. of Approximate Reasoning* **12** (1995), 159-217.
[9] D. Tikk, I. Joó, L. T. Kóczy, P. Várlaki, B. Moser and T. D. Gedeon: Stability of interpolative fuzzy KH-controllers. To appear in *Fuzzy Sets and Systems*.
[10] T. D. Gedeon and L. T. Kóczy: Conservation of fuzziness in rule interpolation, *Proc. of the Symp. on New Trends in Control of Large Scale Systems,*. Herľany (1996), vol. 1, 13-19.
[11] P. Baranyi and T.D. Gedeon: Rule interpolation by spatial geometric representation. *Proc. of the IPMU'96*, Granada (1996), 483-488.
[12] D. Tikk and P. Baranyi: Comprehensive analysis of a new fuzzy rule interpolation method. *IEEE Trans. on Fuzzy Systems* **8** (3) (2000), 281-296.
[13] J. Park and I. W. Sandberg: Universal approximation using radial-basis-function networks, *Neural Computation* **3** (1991), 246-257.
[14] K. Hlavacková and R. Neruda: Radial basis function networks, Inst. of Information and Computer Science, Czech Academy of Sciences, Prague, report, w.y.
[15] A. C. Faul and M. J. D. Powell: Krylov subspace methods for radial basis function interpolation, *DAMPT* 1999/NA11, University of Cambridge, 1999, 25p.
[16] C. A. Micchelli: Interpolation of scattered data: Distance matrices and conditionally positive definite functions, *Constructive Approximation* **2** (1986), 11-22.
[17] D. S. Broomhead and D. Lowe: Multivariable functional interpolation and adaptive networks, *Complex Systems* **2** (1988), 321-355.
[18] L. X. Wang: Fuzzy systems are universal approximators, *Proc. of the IEEE Int. Conf. on Fuzzy Systems*, San Diego (1992), 1163-1169.
[19] B. Kosko: Fuzzy systems as universal approximators, *Proc. of the IEEE Int. Conf. on Fuzzy Systems*, San Diego (1992), 1153-1162.
[20] J. Castro: Fuzzy logic controllers are universal approximators, *IEEE Trans. on SMC* **25** (1995), 629-635.