A Survey of Hierarchical Fuzzy Systems (Invited Paper)

Di Wang, Xiao-Jun Zeng and John A. Keane

***Abstract*—It is widely recognized that standard fuzzy systems suffer the "curse of dimensionality" and that such has become the bottleneck when applying fuzzy systems to solve complicated and high dimensional application problems. In recent years, hierarchical fuzzy systems have emerged as an effective alternative to overcome the curse of dimensionality of fuzzy systems and have attracted considerable attention. This paper presents a survey of hierarchical fuzzy systems with the focus in understanding what has been achieved and identifying the remaining open problems and further research directions.** *Copyright © 2006 Yang's Scientific Research Institute, LLC. All rights reserved.*

***Index Terms*—Fuzzy Logic, Hierarchical Fuzzy Systems, Curse of Dimensionality, Fuzzy Systems.**

I. Introduction

Standard fuzzy systems have been successfully used in most applications, such as approximation [1]-[4], system control [5]-[11], fuzzy classification [12]-[14] and fuzzy clustering [15]-[16]. These successful applications have shown the flexibility of fuzzy systems, the expressive ease of fuzzy rules, and have obtained related theoretical results [1], [2], [5], [17] for fuzzy logic. Fuzzy systems are not black-box models, and their associated transparency enables both interpretation and analysis.

Standard fuzzy systems are general applicable, in the sense that they are universal approximators which can approximate arbitrary continuous functions to any accuracy [1]-[5], [17]. Because of the transparency and their general applicability, substantial progress both in the theory and application of fuzzy systems has been achieved during the last four decades. However, as fuzzy systems have been applied to more complicated and high dimensional systems, the "curse of dimensionality" has become increasingly apparent as the bottleneck to wider application. The curse of dimensionality of fuzzy systems can be summarized as follows (a more detailed discussion in given in Section II):

--*Rule dimensionality*: The total number of rules in the fuzzy rule base increases exponentially with the number of input variables;

--*Parameter dimensionality*: The total number of parameters in the mathematical formulas of fuzzy systems increases exponentially with the number of input variables.

--Data or information dimensionality: The number of data or knowledge set required to identify fuzzy systems increases exponentially with the number of input variables

As a consequence of the curse of dimensionality, transparency and interpretation (important advantages of fuzzy systems) is damaged as humans are incapable of understanding and justifying hundreds or thousands of fuzzy rules and parameters. Further, as there are often only a limited data available in application, a large number of rules and parameters result in over-fitting, which destroys the generalizability of fuzzy systems.

To overcome the curse of dimensionality of fuzzy systems, hierarchical fuzzy systems were proposed in the early 90 by Raju, Zhou and Kisner [7] and have attracted considerable attentions in recent years. In hierarchical fuzzy systems, instead of using a flat or standard high-dimensional fuzzy system, a number of lower-dimensional Sub-Fuzzy Systems (SFS) are linked in a hierarchical manner. A sub-fuzzy system is also termed a module or a Fuzzy Logic Unit (FLU). In addition to using hierarchical fuzzy systems to overcome the curse of dimensionality, some researchers have focused their research on hierarchical fuzzy systems but to refine the final solution [8], [18]. The exceptions of the lower leveled sub-fuzzy systems will be revised by the upper leveled sub-fuzzy systems. These refined algorithms of hierarchical fuzzy systems are used as fuzzy classification [12]-[14], clustering [15]-[16], trajectory planning and tracking system [11].

In response to this growing interest and to summarize recent progress, this paper presents a survey of hierarchical fuzzy systems with the focus on understanding what has been achieved and what needs further research and investigation.

In this paper, we only consider the case with Multiple Inputs and Single Output (MISO). This is without loss of generality as systems with Multiple Inputs and Multiple Outputs (MIMO) can be represented as several MISO systems and solved by using the methods and results based on MISO systems.

This paper is organized as follows: Section II discusses the "curse of dimensionality" problem in standard fuzzy systems;

Manuscript received February 8, 2006. This work is supported by U.K. EPSRC under Grant EP/C513355/1.

All the authors are with the School of Informatics, the University of Manchester, Manchester, M60 1QD, U.K. (phone: 0161 306 3362; fax: 0161 306 3346; e-mail: di.wang@manchester.ac.uk, x.zeng@manchester.ac.uk, johnkeane@manchester.ac.uk).

Publisher Item Identifier S 1542-5908(06)10102-5/$20.00

Section III introduces different types of hierarchical fuzzy system structures; Section IV discusses the universal approximation property of hierarchical fuzzy systems; Section V investigates problems in hierarchical fuzzy systems that are introduced by intermediate variables; Section VI shows that hierarchical fuzzy systems have better generalization capability than standard fuzzy systems; learning algorithms for hierarchical fuzzy systems are discussed in Section VII; and finally we give a brief summary in Section VIII.

II. Exponential growth problems in standard fuzzy systems

In a standard fuzzy system, a grid-based definition of fuzzy terms and their membership functions is usually used to partition the input space and all possible combinations of these fuzzy terms and their membership functions form the fuzzy rule

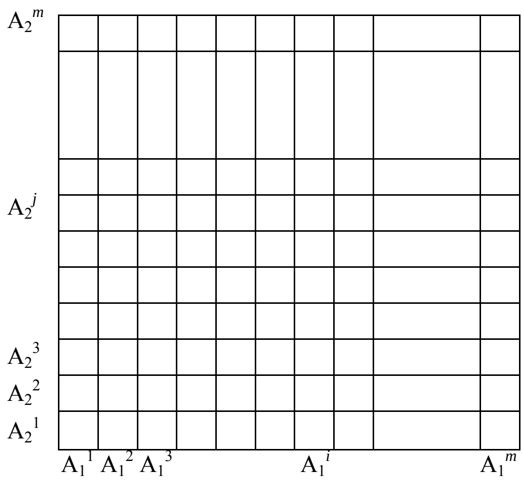

Fig. 2. An example of grid definition for rules and membership functions with two input variables.

base. As a consequence, there is an exponential growth in the number of rules as the number of input variables increases. Fig.1. shows a grid-based definition of the membership functions with two input variables, X_1 and X_2, each having m fuzzy values (A_1^1, A_1^2, …, A_1^m and A_2^1, A_2^2, …, A_2^m). A fuzzy rule can be defined as:

IF X_1 is $A_1^i$and X_2 is A_2^j, THEN Y is B^{ij}, where i=1…m and j=1…m.

Then the total number of fuzzy rules is m^2. This set of fuzzy rules cover the entire definition domain and is termed a complete rule set [19]. Assume that there are n variables, and each variable has m membership functions (fuzzy values), a complete set of rules should have m^n different fuzzy rules. Considering a problem with 10 input variables each of which has 10 membership functions, there will be 10^{10} fuzzy rules altogether. It is infeasible to design a standard fuzzy system for such a complex application with this number of input variables.

The number of parameters also increases exponentially corresponding to the number of input variables. Assume p parameters are needed for each fuzzy rule, a problem with 10 input variables and 10 membership functions for each input variable needs $p \times 10^{10}$ parameters.

The data needed to train standard fuzzy systems also increases exponentially corresponding to the number of input variables [20]-[22]. The number of training samples in the training dataset should at least be the same as the number of parameters. It has been discussed that the number of parameters increases exponentially corresponding to the number of input variables. Hence the data needed increases exponentially corresponding to the number of input variables as well.

This exponential growth of fuzzy rules, parameters and data is termed the "curse of dimensionality" problem [7], [16]. This curse of dimensionality results in several negative consequences as follows:

--Transparency and interpretation (important advantages when fuzzy systems are applied) is damaged or lost as a human is incapable of understanding or justifying hundreds or thousands of fuzzy rules and parameters.

--Over-fitting learning often occurs. As there is often only limited data available to an application, a large number of rules and parameters result in over-fitting, which destroys the generality of fuzzy systems. In other words, learning from data becomes difficult or impossible.

--The requirements for more computation and more memory become excessive [23];

In brief, the curse of dimensionality makes it very difficult or impossible to apply fuzzy systems to solve complicated and high dimensional application problems. This has become the bottleneck to extend fuzzy systems methods from successfully solving simple application problems to a powerful intelligent system tool to solve more and more complicated and high dimensional application problems.

III. Types of hierarchical structure V.S. number of fuzzy rules

To overcome the curse of dimensionality of fuzzy systems, hierarchical fuzzy systems were proposed in the early 90s by Raju, Zhou and Kisner [7]. In hierarchical fuzzy systems, instead of using a flat high-dimensional fuzzy system, a number of lower-dimensional sub-fuzzy systems (defined in terms of standard fuzzy systems) are linked in a hierarchical manner. Hierarchical fuzzy systems are characterized by having several sub-fuzzy systems contributing to the computation of the final solution. The lowest leveled sub-fuzzy systems receive the original input variables, and their outputs are used as the inputs to higher leveled sub-fuzzy systems. In general, hierarchical fuzzy systems are characterized by the number of hierarchical levels, the number of sub-fuzzy systems in each level, and whether the original inputs are just used at the lowest level or not.

A general structure for hierarchical fuzzy systems is shown in Fig. 2. There might be multiple levels, and multiple sub-fuzzy systems in each level. The outputs of certain sub-fuzzy systems (lower leveled sub-fuzzy systems) are used as the inputs of other sub-fuzzy systems (neighboring upper leveled sub-fuzzy systems). The inputs to the lowest level are all original input variables. The inputs to the lth (l>1) level can be the outputs from its lower level, or the combination of its lower leveled outputs and original input variables.

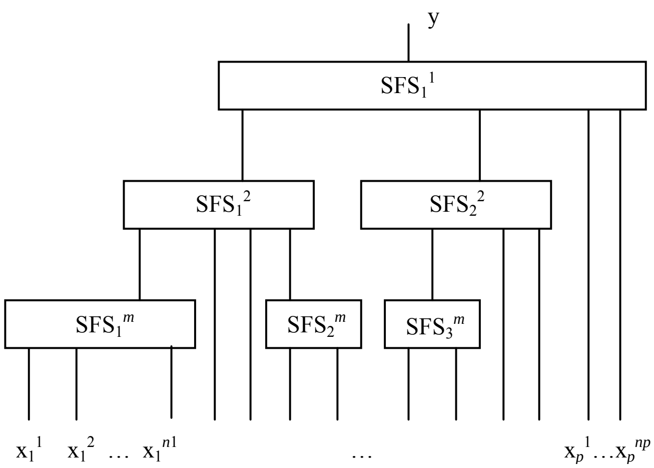

Fig. 2. A general structure for hierarchical fuzzy systems.

Chung and Duan's [24] classified hierarchical fuzzy systems as incremental structures (as shown in Fig. 3), aggregated structures (as shown in Fig. 4) and cascaded structures. Cascaded structures are the combination of incremental structures and aggregated structures (as shown in Fig. 2).

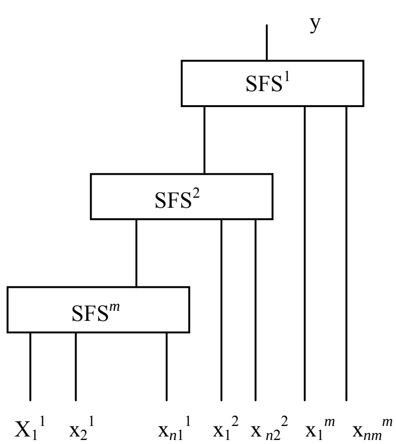

Fig. 3. Incremental hierarchical structures.

Incremental structures are also termed multiple-stage reasoning. One stage is represented by one level in a hierarchical fuzzy system. In the incremental structures (as shown in Fig. 3), there is one sub-fuzzy system in each level.

The lth level uses the combination of the output from the (l-1)th level: y_{l-1}, and original input variables ($x_l^1, \ldots, x_l^{nl}$) as its inputs. Chung and Duan [24] then concluded that the most important input variable should be assigned to the first level, and that less important input variables should be assigned to the next level, and so on. Chung and Duan also designed a method to rank the input variables by their importance to the final solution; similar work was also done by Wang and colleagues [17], [25]-[28]. This approach can also be explained as follows: sub-fuzzy systems with more specific information are applied first, and sub-fuzzy systems with less specific information are applied later. The output of the lower level is propagated to the upper level without being affected by subsequent levels.

For aggregated structures (as shown in Fig. 4), there are multiple sub-fuzzy systems in each level. All the original input variables are input to the lowest level, and none of the original input variables are input to the lth (l>1) level.

One key problem for aggregated structure designation is that

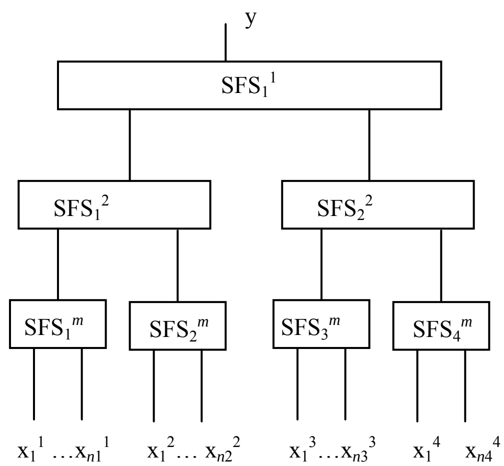

Fig. 4. Aggregated hierarchical structures.

correlated or coupled input variables should be assigned to the same sub-fuzzy system (in the lowest level). Chung and Duan [24] also designed a method to determine the correlated or coupled input variables by introducing a correlation matrix.

A special case of incremental structures is the chaining rule proposed by Domingo and Sierra [29] (as shown in Fig. 5). There is one sub-fuzzy system in each level. All the original input variables are input to the first level, and none of the original input variables are input to the lth (l>1) level. The output of a sub-fuzzy system is used as an input variable to its higher leveled sub-fuzzy system (module). However, this structure is designed uniquely for multi-stage reasoning, and does not contribute to addressing the "curse of dimensionality" problem.

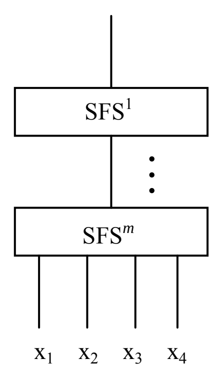

Fig. 5. Hierarchical structure for chaining rule.

Another special case of incremental structures was proposed by Wang and colleagues [17], [25]-[28]. Their proposed scheme with triangle membership functions was proven to be a universal approximator, which can approximate arbitrary nonlinear function to any accuracy. Suppose n be the number of original input variables, m be the number of membership functions for each input variable, and L be the number of levels, ni be the number of inputs in the ith level, which includes the output from the (i-1)th level and the original input variables (to the ith level), then the total number of fuzzy rules is $\sum_{i=1}^{L} m^{n_i}$. The minimal number of fuzzy rules is

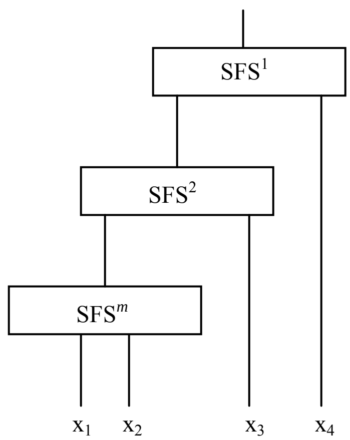

Fig. 6. Hierarchical structure by Wang.

$\sum_{i=1}^{n-1} m^2 = (n-1)\times m^2$ when ni=2 (as shown in Fig.6), while the maximal number of fuzzy rules is m^n when ni=L, which is the same as a standard fuzzy system.

For an example: n=m=4, L=3, n1=n2=n3=2, then the number of rules is $3\times 4^2 = 48$; if we use one level (L=1), then the number of rules is $4^4 = 256$. However, even though Wang and colleagues managed to decrease the exponential growth of fuzzy rules, the exponential growth problem of parameters is still inherently there. There must be sufficient number of free parameters in order for their hierarchical fuzzy systems to become universal applroximators. To gain enough free parameters, Wang and colleagues used the extended T-S fuzzy systems where the polynomials are chosen as the conclusion parts in fuzzy rules.

Later, Wang and colleagues' scheme was improved by Campello and Amarel [30] by using Gaussian functions as membership functions and also gave satisfactory results.

In Joo and Lee's work [31]-[33], they proposed a general hierarchical fuzzy structure (as shown in Fig. 2). The difference is that, the outputs of the lower levels are not used as the inputs (the IF part) of the upper levels. Instead the outputs of the lower levels are only used as the THEN part of the following levels. This is an elegant method to deal with the intermediate variables introduced by hierarchical fuzzy systems, and is discussed in more detail in Section V. However, their scheme involved more parameters to compute the THEN part for each sub-fuzzy system. The THEN part of one fuzzy rule for the dth sub-fuzzy system of the kth level is defined as:

$$y_{kd} = \sum_{j=1}^{d_{k-1}} \beta_{kd}(j) y_{(k-1)j} + \sum_{(u,v)\in_{d_{k-1}} C_2} \alpha_{kd}(u,v) y_{(k-1)u} y_{(k-1)v} + \gamma_{kd}$$

where

$$a_{kd}(u,v) = \frac{\sum_{i=1}^{I_{kd}} u^i(\boldsymbol{x}_{kd}) w_{kd}^i(u,v)}{\sum_{i=1}^{I_{kd}} u^i(\boldsymbol{x}_{kd})},$$

$$\beta_{kd}(j) = \frac{\sum_{i=1}^{I_{kd}} u^i(\boldsymbol{x}_{kd}) p_{kd}^i(j)}{\sum_{i=1}^{I_{kd}} u^i(\boldsymbol{x}_{kd})},$$

$$\gamma_{kd} = \frac{\sum_{i=1}^{I_{kd}} u^i(\boldsymbol{x}_{kd}) \gamma_{kd}^i}{\sum_{i=1}^{I_{kd}} u^i(\boldsymbol{x}_{kd})},$$

where $I_{kd} = \coprod_{j=1}^{n_{kd}} m_{kd}(j)$, $m_{kd}(j)$ is the number of membership functions defined for $x_{kd}(j)$, $\mu(\boldsymbol{x}_{kd})$ is the fitness value of the corresponding fuzzy rule. Then the number of parameters is $\frac{d_{k-1}\times(d_{k-1}+3)}{2}\times\coprod_{j=1}^{n_{kd}} m_{kd}(j)$ for this single rule for the THEN part (not including the parameters in its IF part), where is much more than those used in other schemes discussed in this survey.

Zeng and Keane [34] focused on the general hierarchical structure for fuzzy systems (shown in Fig. 1). This has analyzed the representation capability of general hierarchical fuzzy systems and proven that general hierarchical fuzzy systems are universal approximators. The number of fuzzy rules in general hierarchical fuzzy systems increases polynomially with the growth of input variables. For an example, for a general hierarchical fuzzy system with two levels: the upper level and the lower level (as shown in Fig. 7), the number of rules is $\sum_{j=1}^{m}\left(\prod_{k=1}^{n_{2j}} N_{k^{(2j)}}\right) + \prod_{j=1}^{m} N_{2j} \times \prod_{k=1}^{n_1} N_{k^1}$, where m is the number of sub-fuzzy systems in the lower level, n2j is the number of the original input variables to the jth sub-fuzzy system in the lower level, n1 is the number of original input variables to the upper level, $N_{k^{(2j)}}$ is the number of membership functions of the kth input variable to the jth sub-fuzzy systems in the lower level, N2j is the number of membership functions of the output from the jth sub-fuzzy system in the lower level, N_{k^1} is the number of membership functions of the kth original input variable to the upper level. The super bound of total number of rules and parameters for a hierarchical fuzzy system is $c\left(\sum_{j=1}^{m} M^{n_{2j}} + M^{n_1+m}\right)$, while the super bound of total number of rules and parameters for a standard fuzzy system is M^n, where $M = \max\left\{ \max_{k=1}^{n_1}(N_{k_1}), \max_{j=1}^{m}(N_{2j}), \max_{k=1}^{n_{2j}}(N_{k^{(2j)}}) \right\}$ and $n = \sum_{j=1}^{m} n_{2j} + n_1$. We notice that $\frac{\sum_{j=1}^{m} M^{n_{2j}} + M^{n_1+m}}{M^n} \to 0$ as $M \to \infty$. So hierarchical fuzzy systems help to reduce the number of rules and parameters compared with standard fuzzy systems when M is large.

Other research on hierarchical fuzzy systems has been focussed on the refinement of the final solution [8], [18]. To do this, all (or part in some cases) of the input variables are input to all levels. The sub-fuzzy system in the ith level gives an approximate solution to the problem that is then updated by the sub-fuzzy system in the (i+1)-level to compute a refined solution. This hierarchical structure can be shown in Fig. 8.

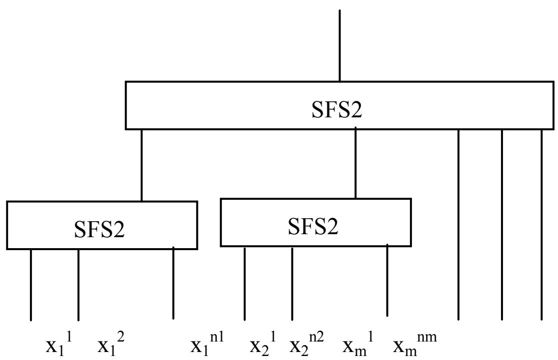

Fig. 7. A two-layered hierarchical structure [34].

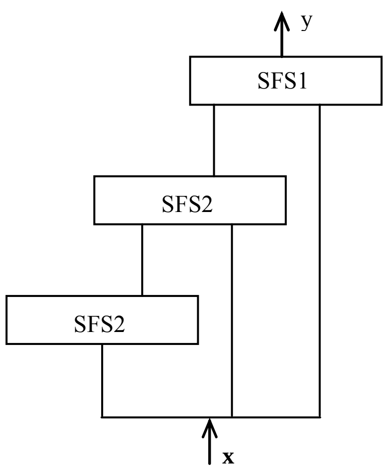

Fig. 8. Hierarchical structures for refined learning.

Unfortunately, it is not easy to develop a proper architecture of a hierarchical fuzzy system. It is not easy to determine the number of levels, the number of sub-fuzzy systems in each level and the proper input variables to each sub-fuzzy system. The outputs of the (i-1)th level, along with the original input variables to the ith level, have to be grouped in order to be input to the upper level, the ith level. It is also not easy to define the rules for the intermediate inputs (such as their domains and the partitions) to the upper levelled sub-fuzzy systems.

IV. Approximation of Hierarchical Fuzzy Systems

To justify the general applicability of hierarchical fuzzy systems, many researchers have analyzed the representation capability and proven the approximation capability of their proposed schemes to construct hierarchical fuzzy systems.

Wang and colleagues [25]-[27] have proven that hierarchical systems based on extended T-S fuzzy systems are universal approximators for continuously differentiable functions with three input variables. For a continuously differentiable problem $g(x_1, x_2, x_3)$ with three input variables on $U = [\alpha_1, \beta_1] \times [\alpha_2, \beta_2] \times [\alpha_3, \beta_3]$, there is a hierarchical fuzzy system $f(x_1, x_2, x_3)$ which can achieve the approximation accuracy $\|g(x_1,x_2,x_3) - f(x_1,x_2,x_3)\| \le \sum_{i=1}^{3}\left(\left\|\frac{\partial g}{\partial x_i}\right\| + \left\|\frac{\partial f}{\partial x_i}\right\| \right) b_i$ where b_i is the width of the partition on $[\alpha_i, \beta_i]\,(i = 1,2,3)$; Based on this approximation error bound and by choosing b_i to be small enough, it was implied that, for any continuous function $g(x_1, x_2, x_3)$ on the compact set $U = [\alpha_1, \beta_1] \times [\alpha_2, \beta_2] \times [\alpha_3, \beta_3]$ and an arbitrary error $\varepsilon > 0$, there is a fuzzy system $f(x_1, x_2, x_3)$ in the form of f_1 and f_2 as follows: $f(\mathbf{x}) = f(x_1, x_2, x_3) = f_2(f_1(x_1, x_2), x_3)$ such that: $\|g(x_1,x_2,x_3) - f(x_1,x_2,x_3)\| \le \varepsilon$.

Generalizing their work, Wang has proven in [27] that for a continuously differentiable problem with n input variables $g(x_1, x_2, ..., x_n)$ on $U = [\alpha_1, \beta_1] \times [\alpha_2, \beta_2] \times ... \times [\alpha_n, \beta_n]$, based on extended T-S fuzzy systems, there is a hierarchical fuzzy system $f(x_1, x_2, ..., x_n)$ such that: $\|g - f\| \le \sum_{i=1}^{n}\left(\left\|\frac{\partial g}{\partial x_i}\right\| + \left\|\frac{\partial f}{\partial x_i}\right\| \right) b_i$; similarly, for any continuous problem $g(x_1, x_2, ..., x_n)$ with n input variables on $U = [\alpha_1, \beta_1] \times [\alpha_2, \beta_2] \times ... \times [\alpha_n, \beta_n]$, given an arbitrary error $\varepsilon > 0$, there is a hierarchical fuzzy system $f(x_1, x_2, ..., x_n)$ such that: To investigate the approximation capability of general hierarchical fuzzy systems, Zeng and Keane [34] have introduced the concept of a natural hierarchical structure when a function can be decomposed as a composition of several lower dimensional functions or a system consists of several lower dimensional components. This has shown that any continuous functions with natural hierarchical structure can be naturally and effectively approximated by hierarchical fuzzy systems, in the sense that it is proved theoretically the hierarchical fuzzy systems need fewer rules and parameters than the standard fuzzy systems to achieve the same accuracy. For a function without natural hierarchical structure or unknown hierarchical structure, Zeng and Keane [34] analyzed the approximation capability of general hierarchical fuzzy systems for an arbitrary continuous function based on Kolmogorov's theorem [35], [36]: for any given continuous function $G(x_1,...,x_n)$ on $U=\prod_{i=1}^{n}[\alpha_i,\beta_i]$, there exists one continuous functions $g(x_1,...,x_n)$ and $\psi_q(q=0,1,...,2n)$, such that $G(x_1,...,x_n)=\sum_{q=0}^{2n} g\left[\sum_{i=1}^{n}\lambda_i\psi_q(x_i)\right]$, where $\lambda_i(i=1,...,n)$ are real constants. From Kolmogorv's theorem [35], [36], any continuous function $G(x_1,...,x_n)$ on $U=\prod_{i=1}^{n}[\alpha_i,\beta_i]$ can be represented by a sum of (2*n*+1) continuous function with separable hierarchical structure (with *n* components) [37]-[39]. Hence for any given $\varepsilon>0$, there is a fuzzy system $F(x_1,...x_n)$ which has the hierarchical structure with (2*n*+1) sub fuzzy systems: $F(\boldsymbol{X})=F_0(F_1(\boldsymbol{X}),...,F_{2n+1}(\boldsymbol{X}))$ such that: $\|G-F\|_\infty<\varepsilon$ and $\|G_i-F_i\|_\infty<\varepsilon$.

Huwendiek and Brockmann [40]-[42] proposed networks of adaptive nodes (NetFAN), a special type of hierarchical fuzzy systems, and a universal approximator, which increased the number of hierarchical levels during training.

Similarly, Joo and Lee's work [31]-[33] proved that hierarchical fuzzy systems with the hierarchical structure given in Fig. 2 are universal approximators with enough hierarchical levels being used.

In spite of the above results relating to the universal approximation property of hierarchical fuzzy systems, there is not a complete answer to the following questions:

1) For any continuous function, whether there is always a hierarchical fuzzy system which achieves the same accuracy with both fewer rules and parameters than standard fuzzy systems? In other words, whether hierarchical fuzzy systems are always better than standard fuzzy systems in overcoming the curse of dimensionality? It is likely that the answer to this question is NO and this leads to the second question;

2) Under what conditions can hierarchical fuzzy systems achieve the same approximation accuracy with both fewer rules and parameters than standard fuzzy systems?

So far, the second question has been partially answered by Zeng and Keane in [34] which showed that, for functions or systems with the natural hierarchical structure, hierarchical fuzzy systems are capable to achieve the same accuracy with both fewer rules and parameters. Therefore, further research is needed in order to have a better and more complete answer to the above questions.

Remark 1.When the input and output spaces are limited to a special domain such as the Boolean spaces, a problem with the general applicability of hierarchical fuzzy systems has been identified by Kikuchi and colleagues [43], [44]: i.e. not all functions (for example, some Boolean functions) can be decomposed into a series of hierarchical structured functions. Based on this and the result in [34], question 1 above can be asked in a different way: whether hierarchical fuzzy systems are better approximators than standard fuzzy systems if functions or systems to be approximated are without natural hierarchical structure in their definition domain?

V. INTERMEDIATE VARIABLES V.S. TRANSPARENCY

Hierarchical structures will introduce intermediate variables, which are the outputs of the lower leveled sub-fuzzy systems, and are propagated to the upper leveled sub-fuzzy systems. If a task can be decomposed into sub-tasks, the application itself has a natural hierarchical structure. The resultant hierarchical fuzzy system should have the same hierarchical structure as the problem prototype. In this case, the intermediate variables possess some physical meaning and it is easy to design such a hierarchical fuzzy system. Unfortunately, in many applications, it is difficult to decompose a main task into sub-tasks. In these applications, the intermediate variables usually do not possess any physical meaning and consequently make hierarchical fuzzy systems hard to design.

In addition, these intermediate variables without 'actual meaning' also causes difficulties with the transparency of fuzzy logic. Hierarchical fuzzy systems give a set of multi-stage rules:

IF X1 is A1 and …and Xn is An, THEN Y1 is B1 and …and Ym is Bm

IF Y1 is B1 and …and Ym is Bm, THEN Z is C

If the intermediate variables: Y1 … Ym have no actual meaning, the multi-stage reasoning becomes less understandable. Furthermore, fewer partitions of the intermediate variables will cause information loss.

To overcome this problem, Lee and Chung [45] developed hierarchical fuzzy systems involving intermediate fuzzy rules in the middle layers of those hierarchical structures. They proposed a new kind of mapping rule-based schemes where the outputs of the previous levels and the input of the next levels are defined as intermediate mapping variables. Using these intermediate mapping variables, one can easily design the involved fuzzy rules in the middle layers of the hierarchical structures. The intermediate mapping variables then can then be given a physical meaning. Their scheme is realized by fixing other input variables in a sub-fuzzy system and finding all the possible outputs for that sub-fuzzy system by changing one variable. These possible outputs are defined as the intermediate mapping variable. This search requires an exponential computation corresponding to the number of input variables. It should be noted that, Lee and Chung's method can only be used to discrete problems, not for continuous problems.

In Joo and Lee's work [31]-[33], they proposed a learning scheme for hierarchical fuzzy structures, in which, the outputs of the lower levels are not used as the inputs of the upper levels. Instead of using the outputs from the lower levels as the IF part of it following levels, they only used these outputs as the THEN part of the following levels. All of the IF parts are derived from the original input variables. The output of the *i*th level y_i is never used as the IF part of the (*i*+1)th level, but the THEN part of the (i+1)th level. Thus the resulting fuzzy rules come with clear physical meaning and become easy to design. However, their scheme needs too many parameters. $\frac{d_{k-1} \times (d_{k-1} + 3)}{2} \times \prod_{j=1}^{n_{kd}} m_{kd}(j)$ parameters are needed for the THEN part of a single rule (discussed in detail in Section III); too many parameters make the fuzzy system less understandable.

For a standard fuzzy system, the defuzzification process transforms the fuzzy conclusion into a crisp (usually numerical) output value. Some information is lost during this dufuzzification process [46]; in the design of a hierarchical fuzzy system, defuzzification of lower sub-fuzzy systems cause some information loss. This information loss will be propagated to the upper sub-fuzzy systems. This information loss introduced by the intermediate variables [46] is termed the spread of fuzziness or fuzzy explosion. The spread of fuzziness in multi-stage (hierarchical) fuzzy systems has been extensively studied by Maeda and colleagues [46], [47] and by Babuska [48]. This propagation of fuzzy explosion in hierarchical fuzzy systems will make the final solution unknown. The more modules that are considered, the more uncertainty is likely.

In addition, when the number of fired rules increases, in turn the uncertainty of the output is increased. So the designation of membership functions and the conclusion (THEN) part is also affected by the spread of fuzziness. We want fewer rules to fire at a time to reduce the spread of fuzziness and to also try to keep the fuzzy rules transparent. Reducing the overlaps of different membership functions is an effective method. Gaussian functions and triangle functions are two popular membership functions used in fuzzy systems. By using Gaussian membership functions more rules fire for the overlap of different membership functions. Triangular membership functions have comparatively better performance to overcome these overlaps if carefully designed. In the schemes presented in [25], [26] and [34], triangular membership functions are applied. Specially, if the edge of one triangular membership function does not get across to the middle point of its neighbouring triangular membership functions, then only 2^n rules fire each time, where n is the number of input variables. Anther factor to consider is the sensitivity of membership functions and the conclusion (THEN) part of a fuzzy rule. Sharper membership functions result in greater sensitivity and smoother membership functions result in lower sensitivity. Small values of the conclusion (the THEN part) of sub-fuzzy systems result in lower sensitivity of the output of these sub-systems to the upper levels, and vice versa. Rattasiri and Halgamuge [23] proposed a type of hierarchical fuzzy systems termed Hierarchical Classifying-Type Fuzzy Systems (HCTFS). This scheme does not repeat the defuzzification process between sub-fuzzy systems to avoid losing information.

To date little work has been done investigating the intermediate variables in hierarchical fuzzy structures. Such developments should enable hierarchical fuzzy systems to be used in wider areas.

VI. GENERALIZATION CAPABILITY

Hierarchical fuzzy systems have better generalization performance than standard fuzzy systems. As was discussed in Section II, the data needed increases exponentially corresponding to the number of input variables. The reason is that the number of parameters increases exponentially corresponding to the number of input variables and meanwhile the data needed is proportional to number of parameters.

In real world applications, it is hard to obtain a sufficient data set with many input variables that cover the whole input space. In most cases there usually are limited samples to train the (hierarchical or standard) fuzzy systems. Given a fixed number of training samples and a fixed number of membership functions for each input variable, the number of parameters needed by a hierarchical fuzzy system is much smaller than for a standard fuzzy system (exponential growth corresponding to the number of input variables). A standard fuzzy system (supported by more parameters) is more liable to over-fit than a hierarchical fuzzy system (supported by fewer parameters).

The better generalization capability of hierarchical fuzzy systems can be explained in another way. The accuracy of the approximation for the testing data depends on the density of the observed data samples in the input space. Suppose there are n input variables, each of which has m membership functions. Then the whole input space will be partitioned into $(m\text{-}1)^n$ sub-spaces for a complete fuzzy set. There should be at least one input sample (or more to gain better performance) located in each sub-space. That is the number of training samples should be proportional to m^n. Otherwise, if the number of training samples is less than m^n, there will be some sub-input spaces with no samples to train them. An unexpected result will be obtained if the testing sample is located in these sub-input spaces with no training data. On the contrary, in hierarchical fuzzy systems, suppose each intermediate variable also has m membership functions, the whole input space will be partitioned into $(m\text{-}1)^{n1}$ sub-spaces for a complete fuzzy set in the first (lower) level, and will be partitioned into $(m\text{-}1)^{n2}$ sub-spaces for a complete fuzzy set in the second level, and so on, where $n1$ and $n2$ are the number of inputs to the first and second level. The whole input space will be partitioned into $(m\text{-}1)^{ni}$ sub-spaces for a complete fuzzy set in the ith level. Altogether, for a hierarchical fuzzy system with L level, the whole input space is partitioned into $\max_{i=1}^{L}(m-1)^{ni}$ sub-spaces for a complete fuzzy for all levels, which is much smaller than $(m\text{-}1)^n$. The probability of a sub-space having no training data in a hierarchical fuzzy system is much smaller than the associated probability in a standard fuzzy system. Therefore, when using hierarchical fuzzy systems, less data is needed for description of the input-output relationships in the subspace.

Hierarchical fuzzy systems reduce the number of fuzzy rules and parameters used in standard fuzzy systems and this reduction prevents the whole model from over-fitting. The obtained fuzzy systems therefore have better generalization capability.

VII. Learning algorithms

For a standard fuzzy system the Least Square Method (LSM) is usually used to gain an optimal result. If the expected intermediate variables (outputs of the lower level and inputs to the upper level) are supposed to be known, LSM might be applied for training a hierarchical fuzzy system (as is done in a standard fuzzy system). However, it is not easy to apply LSM when developing a hierarchical fuzzy system because in many cases the intermediate variables have no physical meaning. A possible solution is for an expert to supply the required hierarchical structure knowledge needed to develop a system. Alternatively, machine learning or optimization techniques can be used to try to identify such systems. It should be noted that the structure of a hierarchical system should be carefully considered before optimizing the parameters.

A. Gradient-descent method

Gradient-descent techniques, such as the error backpropagation algorithm, are a popular method to optimize the parameters in hierarchical fuzzy systems. A hierarchical fuzzy system can be based on Mamdani or T-S fuzzy models. And two types of membership functions are usually applied for both models. They are triangular membership functions [27] and Gaussian membership functions [49]. Compared with the gradient-descent technique used in a standard (flat) fuzzy system, the error of the final output is propagated back from the upper fuzzy level to the lower fuzzy level. The parameter updating of the lower levels is based on the errors propagated back from the upper fuzzy level.

For an example, based on the Mamdani fuzzy model (by using either triangular membership functions or Gaussian membership functions), the formulas to train hierarchical fuzzy systems are shown as follows.

a) For Gaussian membership functions [49]:

$$\mu^i_{p,j}(k) = \exp\left[-\left(\frac{x_{p,j}(k)-c^i_{p,j}(k)}{\sigma^i_{p,j}(k)}\right)^2\right]$$

$$z^i_p(k) = \prod_{j=1}^{n_p} \exp\left[-\left(\frac{x_{p,j}(k)-c^i_{p,j}(k)}{\sigma^i_{p,j}(k)}\right)^2\right] = \prod_{j=1}^{n_p}\mu^i_{p,j}(k)$$

$$a_p(k) = \sum_{i=1}^{l_p} w^i_p(k) z^i_p(k)$$

$$b_p(k) = \sum_{i=1}^{l_p} z^i_p(k)$$

$$\hat{y}_p(k) = \frac{a_p(k)}{b_p(k)}$$

where $\mu^i_{p,j}(k)$ is the Gaussian membership function for the jth input of the ith fuzzy rule to the pth sub-fuzzy system, $x_{p,j}(k)$ is the jth input the pth sub-fuzzy system, $c^i_{p,j}(k)$ is the centre of the jth input of the ith fuzzy rule to the pth sub-fuzzy system, and $\sigma^i_{p,j}(k)$ is the centre of the jth input of the ith fuzzy rule to the pth sub-fuzzy system at time k. $w^i_p(k)$ is the THEN part of the ith fuzzy rule to the pth sub-fuzzy system. $\hat{y}_p(k)$ is the model output of the pth sub-fuzzy system and $y(k)$ is the actual output. $z^i_p(k)$, $a_p(k)$ and $b_p(k)$ are some intermediate parameters in computing $\hat{y}_p(k)$.

The errors propagated back are:

$$e_0(k) = \hat{y}_0(k) - y(k)$$

$$e_p(k) = 2\times\frac{\hat{y}_q(k)-w^i_q(k)}{b_q(k)}\times z^j_q(k)\times\frac{\hat{y}_q(k)-c^i_{q,p}(k)}{(\sigma^i_{q,p}(k))^2}\times e_q(k)$$

where $e_0(k)$ is the error between the actual output $y(k)$ and the model output $\hat{y}_0(k)$. $e_p(k)$ is the propagated error of sub-fuzzy system p, which is propagated from its neighboring upper leveled sub-fuzzy system q. $e_q(k)$ is the propagated error of sub-fuzzy system q.

Parameter updating formulas are:

$$w^i_p(k+1) = w^i_p(k) - \eta\times\frac{z^i_p}{b_p}\times e_p$$

$$c^i_{p,j}(k+1) = c^i_{p,j}(k) - 2\times\eta\times\frac{z^i_p(k)\times(w^i_p(k)-\hat{y}_p(k))\times(x^i_{p,j}(k)-c^i_{p,j}(k))}{b_p(k)\times(\sigma^i_{p,j}(k))^2}\times e_p(k)$$

$$\sigma^i_{p,j}(k+1) = \sigma^i_{p,j}(k) - 2\times\eta\times\frac{z^i_p(k)\times(w^i_p(k)-\hat{y}_p(k))\times(x^i_{p,j}(k)-c^i_{p,j}(k))^2}{b_p(k)\times(\sigma^i_{p,j}(k))^3}\times e_p(k)$$

where η is the learning rate.

(b) For triangular membership functions:

$$\mu_{p,j}^i(k) = \begin{cases} (x_{p,j}(k) - a_{p,j}^i(k)) \big/ (b_{p,j}^i(k) - a_{p,j}^i(k)), & a_{p,j}^i(k) \le x_{p,j}(k) < b_{p,j}^i(k) \\ (c_{p,j}^i(k) - x_{p,j}(k)) \big/ (c_{p,j}^i(k) - b_{p,j}^i(k)), & b_{p,j}^i(k) \le x_{p,j}(k) < c_{p,j}^i(k) \\ 0, & x_{p,j}(k) < a_{p,j}^i(k) \quad or \quad x_{p,j}(k) > c_{p,j}^i(k) \end{cases}$$

$$Z_p^i(k) = \prod_{j=1}^{n_p} \mu_{p,j}^i(k)$$

$$A_p(k) = \sum_{i=1}^{l_p} w_p^i(k) Z_p^i(k)$$

$$B_p(k) = \sum_{i=1}^{l_p} Z_p^i(k)$$

$$\hat{y}_p(k) = \frac{A_p(k)}{B_p(k)}$$

where $\mu_{p,j}^i(k)$ is the triangular membership function for the jth input of the ith fuzzy rule to the pth sub-fuzzy system, $x_{p,j}(k)$ is the jth input the pth sub-fuzzy system, $a_{p,j}^i(k)$, $b_{p,j}^i(k)$ and $c_{p,j}^i(k)$ are parameters for a triangular membership function: $\Delta(x_{p,j}, a_{p,j}^i, b_{p,j}^i, c_{p,j}^i)$ at time k. $w_p^i(k)$ is the THEN part of the ith fuzzy rule to the pth sub-fuzzy system. $\hat{y}_p(k)$ is the model output of the pth sub-fuzzy system and $y(k)$ is the actual output. $Z_p^i(k)$, $A_p(k)$ and $B_p(k)$ are some intermediate parameters in computing $\hat{y}_p(k)$.

The errors propagated back are:

$$e_0(k) = \hat{y}_0(k) - y(k)$$

$$e_p(k) = \begin{cases} \dfrac{\hat{y}_q(k) - w_q^i(k)}{B_q(k)} \times Z_q^j(k) \times \dfrac{1}{(x_{p,j}(k) - a_{p,j}^i(k))} \times e_q(k), & a_{p,j}^i(k) \le x_{p,j}(k) < b_{p,j}^i(k) \\ \dfrac{\hat{y}_q(k) - w_q^i(k)}{B_q(k)} \times Z_q^j(k) \times \dfrac{1}{(x_{p,j}(k) - c_{p,j}^i(k))} \times e_q(k), & b_{p,j}^i(k) \le x_{p,j}(k) < c_{p,j}^i(k) \\ 0, & x_{p,j}(k) < a_{p,j}^i(k) \quad or \quad x_{p,j}(k) > c_{p,j}^i(k) \end{cases}$$

where $e_0(k)$ is the error between the actual output $y(k)$ and the model output $\hat{y}_0(k)$. $e_p(k)$ is the propagated error of sub-fuzzy system p, which is propagated from its neighboring upper leveled sub-fuzzy system q. $e_q(k)$ is the propagated error of sub-fuzzy system q.

$$w_p^i(k+1) = w_p^i(k) - \eta \times \frac{z_p^i}{b_p} \times e_p$$

$$b_{p,j}^i(k+1) = \begin{cases} a_{p,j}^i(k) - \eta \times \dfrac{\hat{y}_q(k) - w_q^i(k)}{B_q(k)} \times Z_q^j(k) \times \dfrac{1}{a_{p,j}^i(k) - b_{p,j}^i(k)} \times e_q(k), & a_{p,j}^i(k) \le x_{p,j}(k) < b_{p,j}^i(k) \\ a_{p,j}^i(k) - \eta \times \dfrac{\hat{y}_q(k) - w_q^i(k)}{B_q(k)} \times Z_q^j(k) \times \dfrac{1}{c_{p,j}^i(k) - b_{p,j}^i(k)} \times e_q(k), & b_{p,j}^i(k) \le x_{p,j}(k) < c_{p,j}^i(k) \\ 0, & otherwise \end{cases}$$

$$a_{p,j}^i(k+1) = \begin{cases} a_{p,j}^i(k) - \eta \times \dfrac{\hat{y}_q(k) - w_q^i(k)}{B_q(k)} \times Z_q^j(k) \times \dfrac{(x_{p,j}(k) - b_{p,j}^i(k))}{(x_{p,j}(k) - a_{p,j}^i(k)) \times (b_{p,j}^i(k) - a_{p,j}^i(k))} \times e_q(k), & a_{p,j}^i(k) \le x_{p,j}(k) < b_{p,j}^i(k) \\ 0, & otherwise \end{cases}$$

$$c_{p,j}^i(k+1)=\begin{cases} c_{p,j}^i(k)-\eta\times\dfrac{\hat{y}_q(k)-w_q^i(k)}{B_q(k)}\times Z_q^j(k)\times\dfrac{(x_{p,j}(k)-b_{p,j}^i(k))}{(x_{p,j}(k)-c_{p,j}^i(k))\times(b_{p,j}^i(k)-c_{p,j}^i(k))}\times e_q(k), & b_{p,j}^i(k)\le x_{p,j}(k)<c_{p,j}^i(k) \\ 0, & \text{otherwise} \end{cases}$$

where η is the learning rate.

For hierarchical fuzzy systems with the T-S model, the THEN part of a rule is a function of its inputs. There are different schemes to construct these functions and there are various parameter optimization formulas for these functions.

Based on his previous theoretical work for hierarchical structure with T-S fuzzy systems [26], [28], Wang [27] derived a gradient descent algorithm for tuning the parameters of the hierarchical fuzzy systems to match the input-output pairs. The resulting hierarchical structure gives good approximation accuracy. However, although the exponential growth of fuzzy rules is avoided, the parameters to be tuned still increase exponentially with the number of the input variables. Joo and Lee [31] also used a gradient descent algorithm to optimize the parameters for their scheme with constraint fuzzy rules. Campello and Amaral [30] optimize their models using a conjugate gradient descent algorithm by Fletcher and Reeves. Chung and Duan [24] also used a gradient descent algorithm to optimize the parameters for their scheme to construct hierarchical fuzzy systems.

It should be noted that the structure of a hierarchical system should be carefully considered before optimizing the parameters. Gradient-descent techniques have been widely used to optimize the parameters in hierarchical fuzzy systems. However, it is not applicable to optimize model structures of hierarchical fuzzy systems by using gradient-descent techniques. Instead of gradient-descent techniques, other researchers prefer to use genetic and evolutionary algorithms for optimization (for both structures and parameters).

B. Genetic and evolutionary algorithms

Genetic and evolutionary algorithms can be used to optimize both the structures and the parameters for hierarchical fuzzy systems. For optimizing hierarchical fuzzy systems using a genetic algorithm, a coding scheme for the rule bases must be designed properly (usually in the form of binary strings). The input variables are assigned to sub-groups; then the sub-conclusions are referenced from these sub-fuzzy rules and coded as sub-binary strings to the upper level. The final solution is referenced from these sub-solutions based on these optimized sub-binary strings. The rule set is initialized randomly; then genetic operations such as selection, crossover and mutation are applied to optimize the rule set. Hoffmann and Pfister [50], [51] proposed an automatically designed model for hierarchical fuzzy controllers by using genetic algorithms. They used genetic algorithms to select the fuzzy terms for the hierarchical fuzzy controllers and their method demonstrated good performance. Other work by Tachibana and Furuhashi [20] used a Multiple Objective Genetic Algorithm (MOGA) to optimize both the corresponding variables to each sub-fuzzy system (sub-model) and to optimize the number of fuzzy rules to combine the input variable for each sub-fuzzy system (sub-model).

Unfortunately, for optimizing hierarchical fuzzy systems using genetic algorithms, a coding scheme for the rule bases must be designed properly. Furthermore, genetic algorithms and evolutionary algorithms are notoriously time consuming.

C. Hybrid approaches

Researchers have also investigated hybrid approaches to training hierarchical fuzzy systems, for example, combining a genetic algorithm with the backpropagation algorithm. Shimojima [52] used the genetic algorithm to optimize the structure and backpropagation (gradient-descent) algorithm to fit the parameters. Cheong and Lai [53] used evolutionary algorithms to learn the membership function and the conclusion (the THEN part) of the fuzzy rules. Mohamadian and Stonier [10] also used evolutionary algorithms to learn the structures of hierarchical models. Furuhashi and colleagues [21], [22] proposed a method with an uneven division of the input space of each sub-fuzzy system. The new division of the input space of each sub-fuzzy system is based on the model error. They used a multiple objective genetic algorithm (MOGA) [20] to select the input variables for each sub-fuzzy system.

The hybrid approach of different machine learning or optimization techniques is a promising method for hierarchical fuzzy system learning, however, little work has been done in this areas.

D. Limitations

Although a number of learning algorithms have been developed, they remain largely ineffective due to the difficulty caused by intermediate variables. Furthermore, little research has been done in developing the learning algorithms to identify the hierarchical structure of a function or system. Therefore, substantial effort and research in learning algorithms are needed in order to realize the benefits and potentials of hierarchical fuzzy systems in applications.

VIII. SUMMARY

This survey has reviewed the motivation, current state and future challenges of hierarchical fuzzy systems. The "curse of dimensionality" problem in standard fuzzy systems is well recognised as the motivation behind the development of hierarchical fuzzy systems. In complex applications with a large number of input variables, the number of rules, parameters and data need to increase exponentially according to the number of input variables; hence it is simply often infeasible to define a standard fuzzy system for such a complex problem. It has been validated both theoretically and practically that hierarchical fuzzy systems are a very effective alternative in overcoming the "curse of dimensionality" in many areas.

This paper specifically reviews most of the latest results in constructing hierarchical fuzzy systems, their universal approximation capability, and associated learning algorithms. It is clear that, despite significant progresses being achieved, there are many open problems which need further investigation, such as determining when hierarchical fuzzy systems are better than standard fuzzy systems; how to handle intermediate variables; and how to train hierarchical fuzzy systems efficiently. Further research and developments in these areas should result in wider applications of hierarchical fuzzy systems and, in turn, help to extend fuzzy systems to address more complex and high dimensional problems.

REFERENCES

[1] H. Ying, "Sufficient conditions on general fuzzy systems as function approximators", in Automatica, Vol.30, pp.521-525, 1994.

[2] X. J. Zeng and M. G. Singh, "Approximation theory of fuzzy system — MIMO case", in IEEE Transactions on Fuzzy Systems, Vol.3, No.2, pp.219-235, 1995.

[3] X. J. Zeng and M. G. Singh, "Approximation theory of fuzzy systems — SISO case", in IEEE Transaction on Fuzzy Systems, Vol.2, No.2, pp.162-176, 1994.

[4] X. J. Zeng and M. G. Singh, "Approximation accuracy analysis of fuzzy system as function approximators", in IEEE Transactions on Fuzzy Systems, Vol.4, No.1, pp.44-63, 1996.

[5] J. J. Buckley, "Universal fuzzy controller", in Automatica, Vol.28, pp.1245-1248, 1992.

[6] R. R. Yager, "On of hierarchical structure for fuzzy modelling and control", in IEEE Transactions on Systems, Man, and Cybernetics, Vol.23, pp.1189-1197, 1993.

[7] G. V. S. Raju and J. Zhou, "Adaptive hierarchical fuzzy controller", in IEEE Transactions on System Man and Cybernetics,Vol.23, No.4, pp. 973-980, 1993.

[8] C. L. Karr, "A synergistic architecture for adaptive, intelligent control system development", in Proceedings of IEEE International Conference Industrial Electronics, Control Instrumentation, pp.2986-2991, 2000.

[9] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modelling and control", in IEEE Transactions on Systems Man and Cybernetics, Vol.15, pp.116-132, 1985.

[10] M. Mohamadian and R. J. Stonier, "Hierarchical fuzzy logic control", in Proceedings of IPMU Conference, pp. 622-629, Madrid, Spain, 2000.

[11] X. Z Fan, N. Y Zhang and S. J. Teng, "Trajectory planning and tracking of ball and plate system using hierarchical fuzzy control scheme", in Fuzzy Sets and Systems, Vol.144, pp.297-312, 2003.

[12] U. Priber and W. Kretzschmar, "Inspection and supervision by means of hierarchical fuzzy classifiers", in Fuzzy Sets and Systems, Vol.85, pp.263-274, 1997.

[13] G. Tsekourasa, H. Sarimveisb and E. K. George, " Ahierarchical fuzzy-clustering approach to fuzzy modelling", in Fuzzy Sets and Systems Vol.150, pp.245–266, 2005.

[14] P. Billaudelb and G. V. Lecolierc "A fuzzy hybrid hierarchical clustering method with a new criterion able to and the optimal partition" in Fuzzy Sets and Systems, Vol.128, No.3, pp.323 – 338, 2002.

[15] Y. EI-Sonbaty and M. A.,Ismail, "Fuzzy clustering for symbolic data", in IEEE Transactions on Fuzzy Systems, Vol.6, No.2, pp.195-204, 1998.

[16] G. V. S. Raju and R. A. Kisner, "Hierarchical fuzzy systems," in International Journal of Control, Vol.54, No.12(5), pp.1201-1216,1991.

[17] L. X. Wang, "Fuzzy systems are universal approximptors", in Proceedings of Conference on Fuzzy Systems, pp.1163-1170, San Diego,1992.

[18] L. Magdalena, "Hierarchical fuzzy control of a complex system using meta-knowledge", in Proceedings of IPMU conference, pp.622-629, Madrid, Spain, 2000.

[19] X. J. Zeng and M. G. Singh, "Decomposition property of fuzzy systems and its applications", in IEEE Transactions on Fuzzy Systems, Vol.4, No.2, pp.149-165, 1996.

[20] K. Tachibana and T. Furuhashi, "A structure identification method of submodels for hierarchical fuzzy modelling using the multiple objective genetic algorithm", in International Journal of Intelligent Systems, Vol.17, pp.495-513, 2002.

[21] S. Nakayama, T. Furuhashi and Y. Uchikawa, "A proposal of hierarchical midelling", in Journal of Japan Society Fuzzy Theory Systems, Vol.1, No.5, pp.1155-1168, 1993.

[22] S. Matsushita, A. Kuromiya and M. Yamaoka, T. Furuhashi, and Y. Uchikawa, "Determination of antecedent structure of fuzzy modelling using genetic algorithm", in Proceedings of IEEE International Conference on Evolutionary Computation (ICEC96), pp. 235-238, 1996.

[23] W. Rattasiri and S. K. Halgamuge, "Computationally advantageous and stable hierarchical fuzzy systems for active suspension", in IEEE Transactions on Industrial Electronics, Vol.50, No.1, pp. 48-61, 2003.

[24] F. L Chung and J. C. Duan, "On multistage fuzzy neural network modeling", in IEEE Transactions on fuzzy systems, Vol.8, No.2, pp.125-142, 2000.

[25] C. Wei and L. X. Wang, "A note on universal approximation by hierarchical fuzzy systems", in Information Science, Vol.123, pp.241-248, 2000.

[26] L. X. Wang, "Universal approximation by hierarchical fuzzy systems", in Fuzzy Sets and Systems, Vol.93, pp.223-230, 1998.

[27] L. X. Wang, "Analysis and design of hierarchical fuzzy systems", in IEEE Transactions on Fuzzy systems, Vol.7, No.5, pp.617-624, 1999.

[28] L. X. Wang, "A course in fuzzy systems and control", Prentice-Hall, Englewood Cliffs, NJ,1997.

[29] M. Domingo and C Sierra, "A knowledge level analysis of taxonomic domains" in International Journal of Intelligent Systems, Vol.12 No.2, pp.105-135, 1997.

[30] R. J. G. B. Campello and W. C. Amaral, "Optimization of hierarchical neural fuzzy models", in Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks, Vol.5, pp.24-27, July 2000 also in pp.8 – 13, vol.5, 2000. IJCNN 2000.

[31] M. G. Joo and J. S. Lee, "Universal approximation by hierarchical fuzzy system with constraints on the fuzzy rule", in Fuzzy Sets and Systems, Vol.130, pp.175-188, 2002.

[32] M. G. Joo and J. S. Lee, "A class of hierarchical fuzzy systems with constraints on fuzzy rules", in IEEE Transactions on Fuzzy Systems, Vol.13, No. 2, pp.194- 203, 2005.

[33] M. G. Joo and J. S. Lee, "Hierarchical fuzzy control scheme using structured Takagi-Sugeno type fuzzy inference", in Proceeding of IEEE transaction on Fuzzy Systems CHECK THIS, pp.178-183, Seoul, Korea, Aug. 1999.

[34] X. J. Zeng and J. A. Keane, "Approximation capabilities of hierarchical fuzzy systems", in IEEE Transactions on Fuzzy Systems, Vol.13, No.5, pp. 659-672, 2005.

[35] A. N. Kolmogorov, "On the representation of continuous functions of many variables by superposition of continuous functions of one variables and addition", in Transactions of American Math Society, Vol.2, No.28, pp.55-59, 1963.

[36] A. N. Kolmogorov, "on the representation of continuous functions of many variables by superposition of continuous functions of one variables and addition", in Dokl. Akad. Nauk. BSSR Vol.144, pp.679-681, 1957.

[37] G. G. Lorentz, "The 13th problem of Hilbert", in F. Browder Ed. Mathematical Developments arising from Hilbert problems, AMS, Providence, RI, 1976.

[38] G. G. Lorentz, "Metric entropy, widths and superposition of functions", in American Math. Monthly, Vol.69, pp.469-485, 1962.

[39] —, "On the structure of continuous functions of several variables", in Transactions of American Math Society, Vol.115, pp.340-355, 1965.

[40] O. Huwendiek and W. Brockmann, "NetFAN — a structured adaptive fuzzy approach", in Proceedings of IEEE International Conference on Neural Networks, pp.1079-1084, Pascakaway, 1996.

[41] O. Huwendiek and W. Brockmann, "On the applicability of the NetFAN-approach to function approximation", in Proceedings of IEEE International Conference on Fuzzy Systems, pp.477-482. Piscataway,1997.

[42] O. Huwendiek and W. Brockmann, "Funciton approximation with deposed fuzzy systems", in Fuzzy Sets and Systems, Vol.101, pp.273-296, 1999.

[43] H. Kikuchi, A. Otake and S. Nakanishi, "Function completeness of hierarchical fuzzy modelling", in Information Science, Vol.110, pp.51-60, 1998.

[44] H. Kikuchi and N. Takagi, "Hierarchical fuzzy modelling and jointly expandable functions", in International Journal of Intelligent Systems, Vol.17, pp.515-529, 2002.

[45] M. L. Lee, H. Y. Chung and F. M. Yu, "Modelling of hierarchical fuzzy systems", in Fuzzy Sets and Systems, Vol.138, pp.343-361, 2003.

[46] H. Maeda, "An investigation on the spread of fuzziness in multi-fold multi-stage approximation reasoning by pictorial representation ─ under sup-min composition and triangular type membership function", in Fuzzy Sets and Systems, Vol.80, pp.133-148, 1996.

[47] H. Maeda, H. Yonelura, Y. Nobusada and S. Murakami, "A study on the spread of fuzziness in multi-fold multi-stage ─ effect of two-dimensional multi-fold fuzzy membership function forms", in Proceedings of Iizuka conference, pp.355-356, Iizuka, Japan, 1994.

[48] R. Babuska, "Construction of fuzzy system interplay between precision and transparency", in ESIT 2000, 12-15 Sep. Aachen, Genmany.

[49] A. Marco, A. Moreno and Y. Wen, "Novel learning algorithms for hierarchical fuzzy neural networks", Proceedings of ISCI, pp.167-175, Mexico, 2005

[50] F. Hoffmann and G. Pfister, "Automatic Design of Hierarchical Fuzzy Controllers Using Genetic Algorithms", The Second European Congress on Intelligent Techniques and Soft Computing (EUFIT94), Aachen, September 1994.

[51] F. Hoffmann and G. Pfister, "A New Learning Method for the Design of Hierarchical Fuzzy Controllers Using Messy Genetic Algorithms", PTOC, IFSA'95 (Sao Paulo), Vol. l, pp.249-25, 1995.

[52] K. Shimojima, T. Fukuda and Y. Hasegawa, "Self-tuning fuzzy modelling with adaptive membership function, rules, and hierarchical structure based on genetic algorithms" in Fuzzy Sets and Systems, Vol.71, No.3, pp.295-309, 1995.

[53] F. Cheong and R. Lai, "Designing a hierarchical fuzzy logic controller using differential evolution", in Proceedings of IEEE International Conference on Fuzzy Systems, pp. I-277-282, Seoul, Korea, 1999.