

# Multiobjective Optimization for Cognitive Design

Ö. Ciftcioglu

Department of Building Technology  
Delft University of Technology  
Delft, 2628 CR, The Netherlands  
o.ciftcioglu@tudelft.nl

M. S. Bittermann

Department of Building Technology  
Delft University of Technology  
Delft, 2628 CR, The Netherlands  
m.s.bittermann@tudelft.nl

**Abstract** - An innovative neural fuzzy system is considered for cognitive design using a neural tree structure with nodes of neuronal type, where Gaussian function plays the role of membership function. The total tree structure effectively works as a fuzzy logic system. The structure of the tree is determined by domain knowledge and each node represents an entity of the domain of concern. The states of these entities are dependent on the stimuli at the input and the relationships between the stimuli and the states form the core for cognitive design. Namely, for each stimulus the status of the system is known and interpreted not only at the output but also at the granulated level concerning the system sub-domains. The research is described in detail and demonstrative applications are reported.

## I. INTRODUCTION

By the introduction of fuzzy-neural concept in 90s, soft computing gained an important dimension in the sense that, the black-box feature of neural networks has turned to gray with the transparency of the logic structure of fuzzy systems. Although this exciting combination has found many soft computing applications and established its own technology, one should note that this is at the cost of a trade-off between transparency vs complexity or alternatively fuzziness vs exactness. Namely, the transparency of fuzzy system is limited by the complexity due to what is known as *curse of dimensionality* and in contrast with this, in neural systems, the complexity can be high due to expensive training algorithms. In contrast with transparency, fuzziness in fuzzy systems is an important issue in engineering systems especially if the application is critical. In such situations however, with neural networks certain performance can be guaranteed although the transparency cannot be fully provided. This situation can be explained in other terms, referring to parametric and non-parametric models. Fuzzy models are parametric models in contrast with the neural network models which are non-parametric. Fuzzy-neural systems are hybrid systems where non-parametric neural systems are partially parameterized by the fuzzy systems. Perhaps another trade-off worth to mention is the pragmatic approach by fuzzy systems vs elaborated approach by neural systems. Favourably, with the neuro-fuzzy systems some feasible solutions are obtained which are not obtainable otherwise. This work endeavours to make such a novel step towards to this direction. A special structure developed for this aim is given in detail and the cognition ability of the system is exemplified with application. In parallel with the developments of computer technology, brain research has gained a marked momentum towards to cognitive systems as a natural trend stemming from intelligent systems.

The work is motivated by architectural design with the support of machine cognition. The work is organized as follows. Section II describes the neural-tree-fuzzy system developed in this research as a base for the cognitive system being developed. Section III describes the fuzzy neural tree in the cognition framework. Section IV gives an application example which is followed by conclusions.

## II. NEURAL TREE FUZZY MODEL

Neural tree networks are in the paradigm of neural networks with marked similarities in their structures[1-4]. A neural tree is composed of terminal nodes, non-terminal nodes, and weights of connection links between two nodes. The non-terminal nodes represent neural units and the neuron type is an element introducing a non-linearity simulating a neuronal activity. In the present case, this element is a Gaussian function which has several desirable features for the goals of the present study; namely, it is a radial basis function ensuring a solution and the smoothness. At the same time it plays the role of membership function in the tree structure which is considered to be a fuzzy logic system as its outcome is based on fuzzy logic operations and thereby associated reasoning. An instance of a neural tree is shown in Fig. 1.

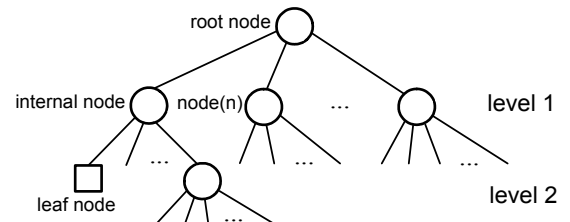


Fig. 1. The structure of a neural tree

Each terminal node, also called *leaf*, is labelled with an element from the terminal set  $T=\{x_1, x_2, \dots, x_n\}$ , where  $x_i$  is the  $i$ -th component of the external input  $x$  which is a vector. Each link  $(j,i)$  represents a directed connection from node  $j$  to node  $i$ . A value  $w_{ij}$  is associated with each link. In a neural tree, the root node is an output unit and the terminal nodes are input units. The node outputs are computed in the same way as computed in a feed-forward neural network. In this way, neural trees can represent a broad class of feed-forward networks that have irregular connectivity and non-strictly layered structures. In particular, in the present work the nodes are similar to those used in a radial basis functions network with the Gaussian basis functions.

In the neural tree considered in this work the output of  $i$ -

th terminal node is denoted  $w_i$  and it is introduced to a non-terminal node. A non-terminal node consists of a Gaussian radial basis function.

$$f(X) = w\phi(\|X - c\|^2) \quad (1)$$

where  $\phi(\cdot)$  is the Gaussian basis function,  $c$  is the centre of the basis function. The Gaussian is of particular interest and used in this research due to its relevance to fuzzy-logic. The width of the basis function  $\sigma$  is used to measure the uncertainty associated with the node inputs designated as external input  $X$ . The output of  $i$ -th terminal node  $w_i$  is related to  $X$  by the relation

$$X_i = w_i w_{ij} \quad (2)$$

where  $w_{ij}$  is the weight connecting terminal node  $i$  to terminal node  $j$ . It connects the output of a basis function to a node in the form of an external input. This is shown in Fig. 2.

The centres of the basis functions are the same as the input weights of that node. Therefore, for a *terminal node connected to a non-terminal node*, we can express the non-terminal node output denoted by  $O_j$ , as

$$O_j = \exp\left(-\frac{1}{2} \sum_i^n \left[\frac{X - w_{ij}}{\sigma_j}\right]^2\right) \quad (3)$$

which becomes due to (2)

$$O_j = \exp\left(-\frac{1}{2} \sum_i^n \left[\frac{w_{ij}(w_i - 1)}{\sigma_j}\right]^2\right) \quad (4)$$

where  $j$  is the layer number;  $i$  denotes the  $i$ -th input to the node;  $w_i$  is the degree of membership at the output of the terminal node;  $w_{ij}$  is the weight associated with the  $i$ -th terminal node and the non-terminal node  $j$ .

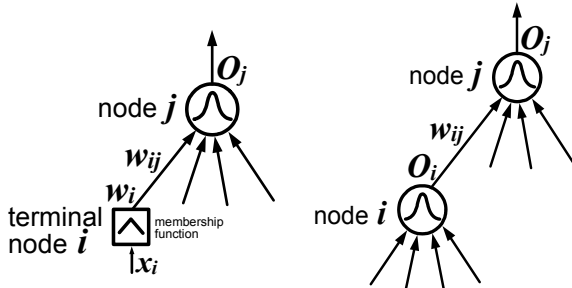


Fig. 2. The detailed structure of a neural tree with respect to different type of node connections

For a *non-terminal node connected to a non-terminal node*, (3) becomes

$$O_j = \exp\left(-\frac{1}{2} \sum_i^n \left[\frac{w_{ij} O_i - w_{ij}}{\sigma_j}\right]^2\right) \quad (5)$$

and further

$$O_j = \exp\left(-\frac{1}{2} \sum_i^n \left[\frac{w_{ij}(O_i - 1)}{\sigma_j}\right]^2\right) \quad (6)$$

We can express (4) and (6) in the following form

$$O_j = \exp\left(-\frac{1}{2} \sum_i^n \left[\frac{(w_i - 1)}{\sigma_{w_{ij}}}\right]^2\right) \quad (7)$$

$$O_j = \exp\left(-\frac{1}{2} \sum_i^n \left[\frac{(O_i - 1)}{\sigma_{w_{ij}}}\right]^2\right) \quad (8)$$

where

$$\sigma_{w_{ij}} = \frac{\sigma_j}{w_{ij}} \quad (9)$$

Referring to (7)-(9), the following observations are essential.

- for each node, parsimoniously, a common width is defined in place of several ones each for a term in the summation.
- the center of a Gaussian is always equal to unity.
- the width of a Gaussian is scaled by the input weight  $w_{ij}$ . In other words, as to width, the shape of Gaussian fuzzy membership function is dependent on the input weights  $w_{ij}$  of the respective Gaussian.
- the derivative w.r.t.  $w_i$  for the terminal nodes and w.r.t.  $O_i$  for non-terminal nodes is always positive. Namely, for a terminal node, from (7)

$$\frac{\partial O_j}{\partial w_i} = -\frac{w_{ij}^2}{\sigma_j^2} (w_i - 1) O_j \quad (10)$$

Since  $0 < w_i \leq 1$  and  $O_j$  is positive, it follows that

$$\frac{\partial O_j}{\partial w_i} \geq 0 \quad (11)$$

In the same way, for a non-terminal node, from (8)

$$\frac{\partial O_j}{\partial O_i} = -\frac{w_{ij}^2}{\sigma_j^2} (O_i - 1) O_j \quad (12)$$

Since  $0 \leq O_i \leq 1$  and  $O_j$  is positive, it follows that

$$\frac{\partial O_j}{\partial O_i} \geq 0 \quad (13)$$

Above, (11) and (13) are important conclusions; namely, the neural tree output is an increasing function of the inputs. This means with the increasing values of the input, the tree output is always increasing and vice versa. This implies that, in this structure, only the left side of the Gaussian membership functions of the non-terminal nodes are used. This is shown in figure 3. Therefore in the figure the right side of the Gaussian is shown with a broken line.

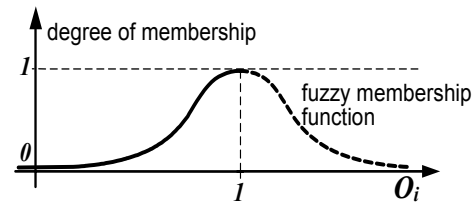


Fig. 3. Fuzzy membership function at non-terminal node.

All the input weights of the nodes are dependent on the neural tree structure and determined by the domain knowledge. At the terminal nodes membership functions are not necessarily Gaussian; they can be triangular, among many other types depending on the application. Note that degree of

membership is denoted by  $w_i$  for this case. For the input  $w_1=1, w_2=1, \dots, w_n=1$ , if the terminal node membership function is a Gaussian, then the output at the non-terminal node is also 1; namely, in (7), the centres of the basis functions are given by a vector  $c = \{1, 1, 1, \dots, 1\}$ , that is  $c_i=1$ . This implies that the Gaussian fuzzy membership functions have their maximum value at the point where all inputs are unity as seen in (7) or (8). In this neural tree structure, only the root node performs a simple weighted summation of the inputs coming from the immediate layer below. Terminologically, this is the defuzzification process for the final outcome. Therefore the weight connected to the root node sums up to one. By means of the above described approach, the locations of the Gaussian membership functions at the non-terminal nodes are well-defined.

In the above discussion the shape of the fuzzy membership functions at the non-terminal nodes are Gaussians due to logic operations. Namely, each input to a node has contribution to the output of that node based on logic AND operation. The centre location of the  $i$ -th Gaussian membership function is selected as  $w_{ij}$  so that certain features described above are maintained. In contrast to the case in neural networks, here the weights are determined by domain knowledge whereas in neural networks they are determined by training. This is accomplished as follows. If there are several inputs weights which belongs to a certain non-terminal node, these weights are determined with respect to each other. They can all be pair-wise compared and normalized afterwards for convenience. That is after normalization if some weights are not relatively significant, they can be discarded. However, normalization is not necessary action as this is explained below. The respective Gaussian width is scaled by the input weight for each term in the summation in (7) and (8). After scaling we have a scaled Gaussian width  $\sigma_j/w_{ij}$  as seen in (9). The determination of the Gaussian width is performed by the imposition of what is called in this framework as “consistency conditions” which serve as boundary condition for the neural tree model. This is explained below.

The neural tree output follows the trend of input  $w_i$  representing the degree of membership associated. Considering this property, the *consistency* refers to the fact that, in the knowledge domain if all the inputs  $w_i$  are unity, all system determinants have the value where the associated fuzzy membership functions at the terminal node take the value of 1; as result of this, all the non-terminal node outputs are accordingly 1 and therefore system output at the root node is also 1. This condition is inherently satisfied in the present neural tree structure and this is easily seen by (7) and (8); namely if all  $w_i$  are 1, then all non-terminal node outputs  $O_i$  are 1 and then the neural tree output is 1. This is a marked difference between a neural network and a neural tree in this work. Namely, in neural networks the weights are determined by learning and the network is a “universal approximator”. In the neural tree presented, weights are determined by domain knowledge and it is not a “universal approximator”. It is an soft computing estimator estimating the reflection of the input

composition to the output with the presence of given structural relationships. This is more explicitly explained by the following example. Consider that neural tree represents the safety of a vehicle. If all the safety determinants of the vehicle at the input are high, then accordingly the safety of the vehicle is high at the output and vice versa. In engineering terms, as a metaphor, the safety determinants may be the reliability of each component. For any safety scores at the input we obtain corresponding safety score at the output where the safety scores may be somehow connected to some safety measures based on the reliabilities. At any point in the tree structure, if the reliability measure goes high, then the reliability score at the output bound to be high too. Summarizing, because of the increasing function property of the tree, as explained earlier, any improvement of any input score will be reflected as an improvement at the output. However, the degree of improvement is still dependent on the tree structure representing the domain knowledge established. This consistency condition is imposed on the tree structure by means of two consistency data sets and the Gaussian widths complying with these conditions are determined by training. The consistency data sets are shown in Table II and Table III as input and output applied to the tree for a supervised training, in the terminology of neural networks. It is worth to note that if all the inputs are unity in (7) and (8), then output is also unity and therefore this input-output pair is not included in to training set. It should be pointed out that, the data sets in Tables II and III are application specific and valid in most applications in the framework of soft computing. However, for special applications the validity of the given consistency data sets should be reviewed and altered, if necessary. For instance, above in the metaphoric reliability example, data sets in Tables II and III should be altered according to the reliability scheme being considered. However, one should consider that such a case is far from being a soft computing exercise and therefore the example is qualified as “metaphoric” although the validity of the fuzzy neural tree concept may still prevail [5].

TABLE II  
DATASET AT THE NEURAL TREE INPUT FOR CONSISTENCY CONDITION

.1	.1	.1	.1	.1	.1	.1	.1	.1	.1	.1	.1	.1	.1	.1
.2	.2	.2	.2	.2	.2	.2	.2	.2	.2	.2	.2	.2	.2	.2
.3	.3	.3	.3	.3	.3	.3	.3	.3	.3	.3	.3	.3	.3	.3
.4	.4	.4	.4	.4	.4	.4	.4	.4	.4	.4	.4	.4	.4	.4
.5	.5	.5	.5	.5	.5	.5	.5	.5	.5	.5	.5	.5	.5	.5
.6	.6	.6	.6	.6	.6	.6	.6	.6	.6	.6	.6	.6	.6	.6
.7	.7	.7	.7	.7	.7	.7	.7	.7	.7	.7	.7	.7	.7	.7
.8	.8	.8	.8	.8	.8	.8	.8	.8	.8	.8	.8	.8	.8	.8
.9	.9	.9	.9	.9	.9	.9	.9	.9	.9	.9	.9	.9	.9	.9

TABLE III  
DATASET AT THE NEURAL TREE OUTPUT FOR CONSISTENCY CONDITION

.1	.2	.3	.4	.5	.6	.7	.8	.9
----	----	----	----	----	----	----	----	----

As result of the training, the consistency conditions of the tree are established and all the Gaussian widths are determined. An example neural tree structure is shown in figure 5 where the output is designated as performance of a design. The detailed

description of this tree model is given elsewhere [6]. Basically, it deals with an architectural design task where positioning of three high-rise buildings complying with some perceptual conditions is aimed. Accordingly, the inputs are some distances and some perceptions measured.

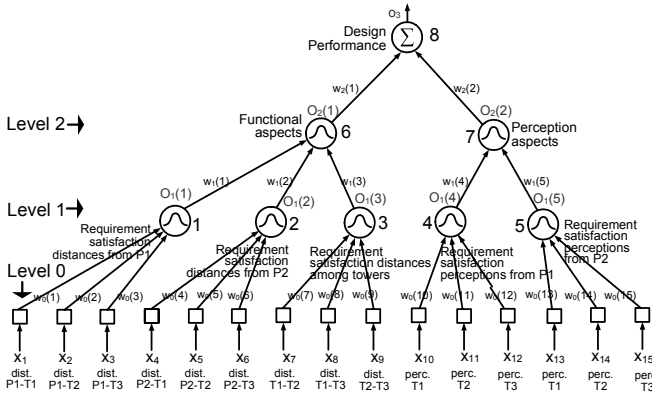


Fig. 4. An example of a neural tree employed for design

There are two options for consistency training. In one option, if all the inputs to the tree are uncorrelated and the tree is in the form of several branches, then the training is performed for each branch. A branch is defined as independent set of nodes connected to the root node. The output of a branch is taken as the node one below the root node. For example in figure 4, there are two branches. The branch at the left side has 9 inputs and the output node is designated as node number 6. The other branch has 6 inputs and the output node is designated as node number 7. When the consistency conditions are established for all branches, it includes inherently the root node also. In another option, if the inputs to the tree are correlated, then to allow for this correlation, the training for the consistency condition includes also the root node meaning that for the whole tree structure one common training is performed. Referring to (9), after consistency training only the relative values of the weights coming to a node are matter indicating the effectiveness of each weight on the node relative to each other. In essence, such relativity is passed on the Gaussian widths, eventually.

The consistency conditions having been established by training, for the inputs at the terminal nodes, appropriate fuzzy membership functions are used. These are based on domain knowledge. The employment of these membership functions is due to two essential reasons as follows. The neural tree accepts input between zero and one as a dimensionless quantity so that during the computations, we are not concerned with units and all the node outputs are also between zero and one providing easiness for the interpretation of the results not only at the root node but also throughout the tree structure.

Another reason worth to mention is the circumvention of standardization of the inputs. Namely, since the inputs to the tree are some physical quantities, in general they are all measured with their respective units and there can be a big difference among these units. This is termed as “stiffness” in science. Since larger values at the input produce larger outputs, it would not be fair to apply directly the measurement

values with their respective units. By means of the membership functions all inputs are “standardized” in some way similar to the standardization of statistical data according to a Gaussian distribution, in the statistical analysis. By doing so, the “stiffness” is circumvented.

So far a novel neural tree is described in detail for a system modelling based on ontological knowledge integration into the system. This is a knowledge driven modelling where fuzzy modelling concept plays important role. One can consider that, in neural tree the information flow from input to output is according to fuzzy logic AND operations in the form of *rule chaining*, that is, a *consequent* or a rule output is an input for another rule in the form of a chain. Gaussian radial basis functions play the role of membership functions. In the tree network, there is no cognitive component yet, since everything is based on the external representation of the environment; namely, based on the knowledge provided in advance. In other words, referring to figure 5, the tree has the property that high level inputs provide high performance while it does not provide information about which input composition is preferred for the same design performance or for the same amount of design performance increase. In general we can assume that there can be several input composition for the same performance or performance increase. In a cognitive system, one anticipates that the system in some sense “comprehend” the environment and acts in a way or offer a solution in a way that fits the goal in a most suitable way conforming to the limitations of the environment. This means, a choice among the same performance or performance increase is made by distinguishing among them. In particular, some higher-order demands are dealt with afterwards i.e., after the basic (first-order) demands are fulfilled and commensurate solutions established. This cognition concept is dealt with in the following section.

### III. FUZZY NEURAL TREE IN A COGNITIVE NETWORK FRAMEWORK

As a cognitive system, fuzzy neural tree is anticipated so to speak “comprehend” the environment in the sense that among several equivalent outputs it provides the most appropriate information or offer as a solution fitting to the goal. In this definition the final outcome is mainly due to some higher level preferences given to the system in advance. These higher level preferences do not play role during the formation of the environment generated by the tree itself; however, they are in effect after tree is queried. As an example, in figure 5, the tree queried for a higher performance and several input compositions are obtained for this performance. Here we have a selection problem. Therefore we are still not ready until tree suggests more detailed solution providing more insight into the response initiated by the query. The more insight is based on a higher-order demand which is among a number of such similar demands and these demands are associated with the valid solutions at the time point of query is issued and not before. Valid solution is defined as the solution meeting at

least one higher-order demand in a respective demand repertoire.

In this research, above given considerations are realized by means of formation of Pareto front of solutions. Such solutions are subject matter of multi-objective optimization (MO) methods in the evolutionary algorithms framework and these methods are extensively discussed by Deb[7]. Referring to figure 5, the Pareto surface is in two dimensions formed by the *functional aspects* and *perception aspects*. Strictly speaking, we deal with a Pareto curve in this example. These two aspects are supposedly competitive and any of them is high at the cost of the other. We know that, higher inputs applied to the tree provide higher scores at the nodes. However, because of this competitive behaviour functional and perception aspects go to saturation at some arbitrary level and arbitrary composition if a blind increase of inputs is exercised. Therefore a search must be done for appropriate input compositions for diverse solutions which provide the demanded performance. The diversity of solutions on the Pareto surface is essential for cognition. The establishment of the Pareto surface can be obtained by means of search algorithms among which evolutionary algorithms are favourable. An overview about multi-objective evolutionary algorithms (MOEA) algorithms and the multi-objective optimization issues is given by Fonseca [8]. Concerning the diversity of solutions along the Pareto front, the issue is a hot topic in science and important advancements are made in the last decade [9, 10]. Concerning the tree model in figure 4, let us denote the demanded score of functional aspects by  $f_1$  and demanded score of perception aspects as  $f_2$ . The associated Pareto surface is schematically shown in figure 5 where the Pareto surface is shown as a bold convex curve. The scattered distinct solutions on this curve are the valid solutions which are all different but equivalent. In the figure the area delimited by the Pareto front for the suboptimal dominated solutions. The convex hull with the angle  $\theta = \pi/2$  represent the domain of non-dominated solutions during the search process. However, for the sake of diversity of solutions, this condition is relaxed allowing  $\theta \geq \pi/2$  so that more solutions are considered to be a candidate solution and transferred to the next generation at each generation of the evolutionary search. This will be explained in detail below.

Once a Pareto front is established, it contains all non-dominated solutions that are searched for. In order to elaborate on this let us consider a higher-order demand as

$$w_2(1)f_1 + w_2(2)f_2 = c \quad (14)$$

where  $w_1 + w_2 = 1$ ;  $c$  is a constant representing a prescribed performance score. Let this score be selected as  $c = 0.83$  so that we write

$$w_1(1)f_1 + w_2(2)f_2 = 0.83 \quad (15)$$

which represents a line with respect to  $f_1$  and  $f_2$  and this is shown in figure 13. The intersections of the Pareto surface and the line given by (14) are denoted by  $S_1$  and  $S_2$  which are the solutions to be selected on the Pareto front with respect to

higher order preferences. One should note that, in this research during the Pareto front formation the performance score at the root node does not play any role during the genetic search. Therefore it is counted to be in the higher order demand category

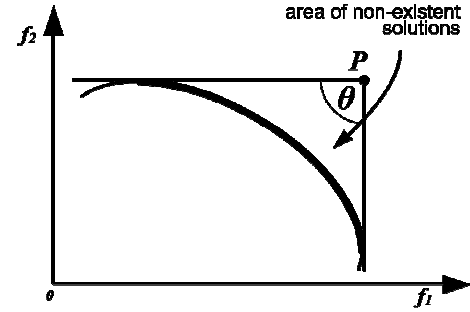


Fig. 5. Contour lines defining the dominated region with respect to the point  $P$ : greedy dominance (a); relaxed dominance (b).

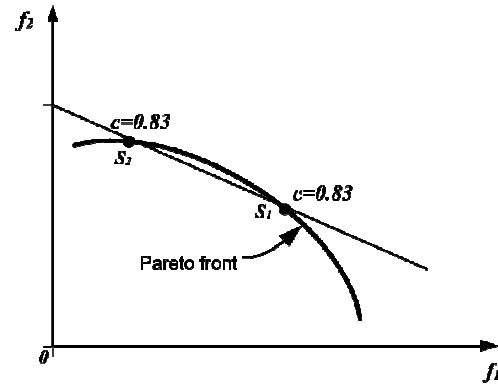


Fig. 6. Particular solutions on the Pareto surface corresponding to a fixed Performance score which is 0.83, in this case.

The corresponding weighted objectives  $F_1$  and  $F_2$  are given by

$$\begin{aligned} F_1 &= f_1 + a_{12} f_2 \\ F_2 &= a_{21} f_1 + f_2 \end{aligned} \quad (16)$$

Taking  $w_1 = 0.6$  and  $w_2 = 0.4$  for the visual privacy and  $a_{21} = a_{12}$ , we obtain

$$\begin{bmatrix} F_1 \\ F_2 \end{bmatrix} = \begin{bmatrix} 1 & -.66 \\ -.66 & 1 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \quad (17)$$

The corresponding direction cosines designated as  $d^{(1)}$  and  $d^{(2)}$  are given by

$$d^{(1)} = \begin{bmatrix} 1/\sqrt{1^2 + .66^2} & -.66/\sqrt{1^2 + .66^2} \end{bmatrix} \quad (18)$$

$$d^{(2)} = \begin{bmatrix} -.66/\sqrt{1^2 + .66^2} & 1/\sqrt{1^2 + .66^2} \end{bmatrix}$$

Transforming  $F$  coordinates to  $f$  coordinates and vice versa is accomplished by means of coordinate transformation:

$$\begin{bmatrix} f_1^{S_1} \\ f_2^{S_1} \end{bmatrix} = \begin{bmatrix} d_1^{(1)} & d_1^{(2)} \\ d_2^{(1)} & d_2^{(2)} \end{bmatrix} \begin{bmatrix} F_1^{S_1} \\ F_2^{S_1} \end{bmatrix} \quad (19)$$

The corresponding  $F$  coordinate system is schematically shown in figure 14 where the guided domination contours passing from the point  $S_I$  are also schematically illustrated with the broken lines. With the numerical values in (17), the angle  $\theta$  from the cosine directives corresponds to  $150^\circ$ . The present approach is not seeking for not only the dominating solutions but also those dominated but still close to dominating solutions and ranking the solutions.

Apparently for two objectives the MO optimization is relatively easy considering the angle  $\theta$ . However for higher number of conflicting objectives selection of the non-dominated solutions becomes problematic. This is because in the multidimensional space it is a problem to establish the domain of non-dominated solutions. However, the same approach, namely (18) can easily be extended to higher dimensionality.

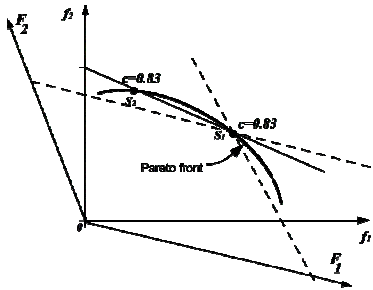


Fig. 7. Particular solutions on the Pareto surface corresponding to a fixed Performance score which is 0.83, in this case.

For instance for four objectives, we write

$$\begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} = \begin{bmatrix} 1 & a_{12} & a_{13} & a_{14} \\ a_{21} & 1 & a_{23} & a_{24} \\ a_{31} & a_{32} & 1 & a_{34} \\ a_{41} & a_{42} & a_{43} & 1 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \quad (20)$$

where five parameters should be assessed, namely  $a_{12}$ ,  $a_{13}$ ,  $a_{14}$ ,  $a_{23}$  and  $a_{24}$  in advance since the coefficient matrix is taken to be symmetrical. The parameters are assessed easily from the cosine directives considering the fuzzy-neural tree being considered for such a case as exemplified above; namely, figure 4 contains 2 objective- nodes below the root node and the direction cosines are given by (18). Identifying the effective parents for generating favorable offspring in consecutive generations as described is essential as such it underlies the relaxed dominance approach applied in this research. The genetic search is carried out in  $F$ -space and the final results are transformed to  $f$ -space as given by (19) for a two-dimensional case.

#### IV. IMPLEMENTATION

Both works on neural tree and multi-objective Pareto optimization have received ample attention in the literature. However, fuzzy neural tree in a cognitive network framework

with the integration multi-objective optimization is a novel concept being revealed as a subject-matter of this presentation. Implementations related to this work in a neural tree and multi-objective optimization framework is described in detail earlier [11] higher dimensional aspects being excluded. Referring to figure 7, a Pareto front obtained from the search is shown in figure 8 where the multi-objective optimization is for a design process of a dwellings location[12].

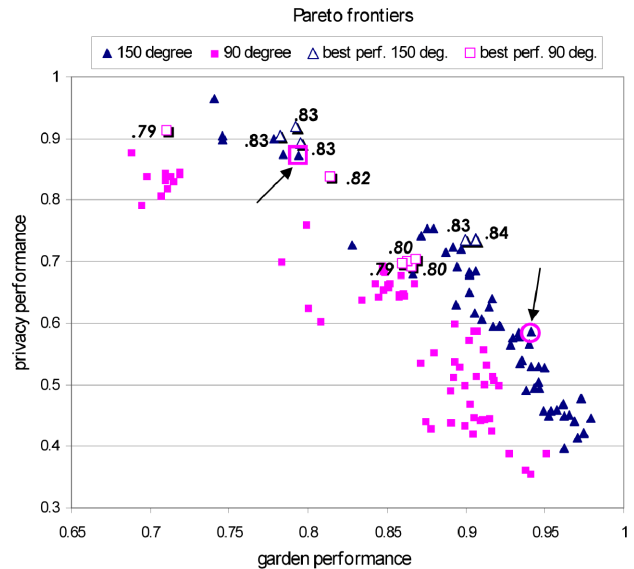


Fig. 8. Particular solutions on the Pareto surface corresponding to a Comparison of the Pareto optimal frontiers for the angle of tolerance  $\theta=90^\circ$  vs.  $\theta=150^\circ$  after 15 generations; the design performance values of the best solutions of each population are indicated

In this figure there are two different outcomes. Namely, one is for  $\theta=90^\circ$  and the other for  $\theta=150^\circ$ . The latter implies that at the genetic selection process some limited number of dominated solutions is also considered for the next generation. In each case, the best design solutions in Pareto sense are indicated by arrows in the figure. These final outcomes of the design are shown in figure 9a and 9b respectively. Here the design determinants are visual perceptions and they are measured from vantage points at the location of the inhabitants. Here, by means of cognition “machine perceived” environment is generated by the system so that for any query the fuzzy neural tree system responds in most suitable way based on this very concept of environment. It is to emphasize that, the system decides to propose a solution which is not prescribed in advance like for instance by means of a supervised training or anything else one might think of. The final decision is made afterwards and it is based on the higher order demands like privacy considerations. This is especially due to multi-objectives where each objective is satisfactorily addressed and manifested by the Pareto front. The cognitive design results are due from the assessment of the design performance at the root node with the selection of a design based on rationales among several competitive design choices.

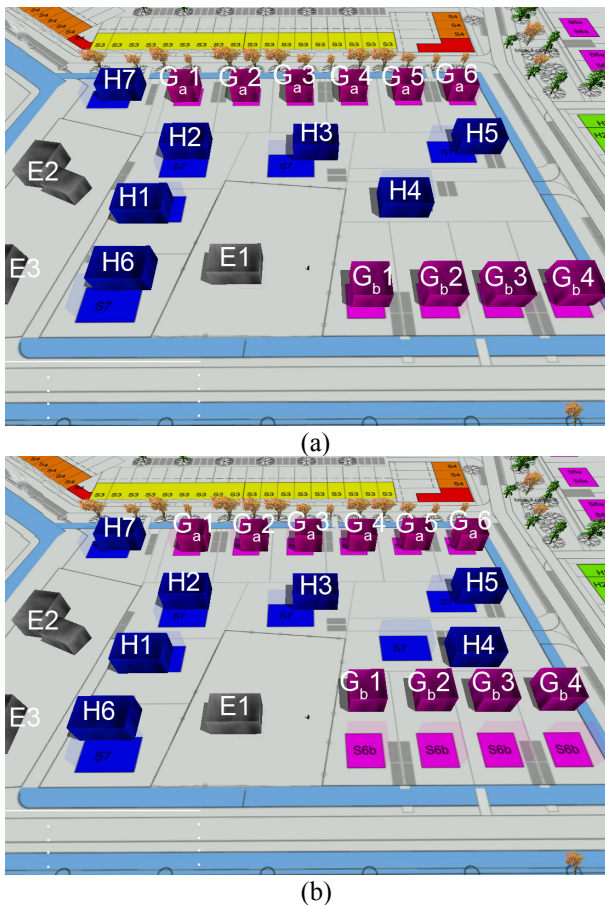


Fig. 9. Two Pareto-optimal designs found by the genetic algorithm

Therefore in each generation and for each scene of potential solution, the rendering of that very scene is necessary for the ongoing visual perception computations. Therefore the MOEA used is computationally intensive and the entire computation for an outcome takes about some 5 minute involving 20 iterations. For further iterations the fluctuations about the Pareto front become less yielding a smoother Pareto front.

## V. CONCLUSIONS

A novel fuzzy neural network as a cognitive design is described where the definition of cognition in this context is clearly defined. The novelty stems from the particular neural tree structure bearing fuzzy logic concepts and the associated details which make altogether the machine cognition works. This is demonstrated by means of an example while making use of some earlier researches of the same authors on multi-objective optimization with evolutionary algorithms and integrating the novel “machine cognition” concept into these works. Here the term “cognition” is justified thanks to both multi-objectivity and the neural tree structure by means of which a human-like information processing is performed with the underlying accumulated domain knowledge. The whole structure can be seen as a cognitive map where accumulated knowledge is connected with new information, namely the stimulus at the input. This connection made is a learning process and this is made meaningful by means of “learning via

consistency” in this work. Meaningful learning is most likely occur when information is presented in a potentially meaningful way and the learner is encouraged to anchor new ideas with the establishment of links between old and new material [13].

The motivation of the research presented in this work is especially for architectural design assessment viewpoint where the complexity in such designs prevail making objective assessments easier that are difficult otherwise. Machine cognition arrives at decisions which are based on the ample knowledge about the design given in advance to the fuzzy neural tree system and the decision is deemed to be objective within a capacity proportional to the provided knowledge. Apart from the motivation of the research, in general, research on cognitive systems is an important activity in science, especially in the last decade. Machine cognition can make important impact especially in automation for effective production in industry and other services. The present work is a novel significant step along this direction.

## VI. REFERENCES

- [1] G. L. Foresti and C. Micheloni, "Generalized neural trees for pattern classification," *Neural Networks, IEEE Transactions on*, vol. 13, pp. 1540-1547, 2002.
- [2] A. Sankar and R. J. Mammone, "Neural tree networks," in *Neural Networks: Theory and Applications*, R. J. Mammone and Y. Zeevi, Eds. New York: Academic, 1991, pp. 281 -302.
- [3] J. A. Sirat and J. P. Nadal, "Neural tree networks," *Network*, vol. 1, pp. 423-438, 1990.
- [4] F. d'Alché-Buc, D. Zwierski, and J. P. Nadal, "Trio learning: a new strategy for building hybrid neural trees," *Int. J. Syst.*, vol. 5, pp. 259-274, 1994.
- [5] O. Ciftcioglu, "A fuzzy-neural tree for possibilistic reliability," presented at Joint 4th Int. Conf. on Soft Computing and Intelligent Systems (SCIS & ISIS), September 17-21, Nagoya, Japan, 2008.
- [6] Ö. Ciftcioglu, M. S. Bittermann, and I. S. Sariyildiz, "A neural fuzzy system for soft computing," presented at NAFIPS 2007, San Diego, USA, 2007.
- [7] K. Deb, *Multiobjective Optimization using Evolutionary Algorithms*: John Wiley & Sons, 2001.
- [8] C. M. Fonseca, "An overview of evolutionary algorithms in multiobjective optimization," *Evolutionary Computation*, vol. 3, pp. 1-16, 1995.
- [9] H. Sato, H. E. Aguirre, and K. Tanaka, "Controlling dominance area of solutions and its impact on the performance of MOEAs," in *Lecture Notes on Computer Science, Evolutionary Multi-Criterion Optimization*. Heidelberg: Springer, 2007.
- [10] Z. P. a. J. A. Deb K., "Distributed computing of Pareto-optimal solutions using multi-objective evolutionary algorithms," in *Proc. EMO 2003 pp. 534-449*, C. Fonseca, P.J. Fleming, E. Zitzler, K. Deb, L. Thiele (Eds.), 2003.
- [11] I. S. Sariyildiz, B. M. S., and Ö. Ciftcioglu, "Performance-based Pareto optimal design," *Proc. TMCE 2008, April 21-25, Izmir, Turkey*, I. Horvath and Z. Rusak (Eds.), 2008.
- [12] O. Ciftcioglu and M. S. Bittermann, "Solution Diversity in Multi-Objective Optimization: A study in Virtual Reality," presented at IEEE World Congress on Computational Intelligence, Hong Kong, 2008.
- [13] L. Irvine, "Can concept mapping be used to promote meaningful learning in nursing education?," *Journal of Advanced Nursing, Blackwell Science Ltd.*, vol. 25, pp. 1210-1219, 1997.