IEEE Congress on Evolutionary Computation – CEC 2009
Trondheim, Norway, 18-21st May 2009

A cognitive system based on fuzzy information processing and multi-objective evolutionary algorithm

Michael S. Bittermann, Özer Ciftcioglu, *Senior M*ember*, IEEE* and I. Sevil Sariyildiz

***Abstract*— A cognitive system is presented, which is based on coupling a multi-objective evolutionary algorithm with a fuzzy information processing system. The aim of the system is to identify optimal solutions for multiple criteria that involve linguistic concepts, and to systematically identify a most suitable solution among the alternatives. The cognitive features are formed by the integration of fuzzy information processing for knowledge representation and evolutionary multi-objective optimization resulting in a decision-making outcome among several equally valid options. Cognition is defined as final decision-making based not exclusively on optimization outcomes but also some higher-order aspects, which do not play role in the pure optimization process. By doing so, the decisions are not merely subject to rationales of the computations but they are the resolutions with the presence of environmental considerations integrated into the computations. The work describes a novel fuzzy system structure serving for this purpose and a novel evolutionary multi-objective optimization strategy for effective Pareto-front formation serving for the goal. The machine cognition is exemplified by means of a design example, where a number of objects are optimally placed according to a number of architectural criteria.**

Keywords: soft computing, multi-objective optimization, Pareto front, fuzzy neural tree, cognitive design

I. INTRODUCTION

The complexity of the modern technological real world problems makes it problematic for a decision maker to specify the relative importance among criteria a-priori. The problem is that it is difficult to foresee the implications of such commitment prior to investigating available solutions. Therefore multi-objective optimization algorithms are needed and being developed. With the advent of evolutionary algorithms in the last decades, multi-objective evolutionary algorithms (MOEAs) are extensively being investigated for solving multi-objective optimization problems [1-5]. Evolutionary algorithms are particularly suitable for this, since they evolve simultaneously a population of potential solutions. These solutions are investigated in non-dominated solution space, so that the optimized solutions in a multi-objective functions space form a front which is known as Pareto surface or front. It is emphasized that the multi-objective approach is based on postponing the commitment on the relative importance among the objectives until the Pareto front is established, and let the decision maker select one among the Pareto solutions with great awareness. In this respect the multi-objective optimization is a *cognitive* approach. Namely cognition is understood as the process of bringing second-order preferences into play based on awareness of the options available in an environment [6-8]. This means cognition goes beyond a mere optimization process, but it involves higher-order considerations. In multi-objective optimization these considerations yield the selections among the Pareto optimal solutions, which entail specification of the relative importance among the objectives. It is to be noted that in the existing multi-objective optimization approaches the cognitive component is due to the considerations by the human decision maker.

In the present work a novel multi-objective optimization system is presented, where the machine identifies suitable second-order preferences selecting a most desirable solution on the Pareto front. This means machine cognition is exercised. This is accomplished by coupling a MOEA with a fuzzy information processing system. The fuzzy system is used to evaluate the fitness of the solutions with respect to the objectives. Due to the special type of fuzzy information processing system employed, different objectives are treated on a common ground. That is, the objectives are considered as complex linguistic concepts, and their fulfilment is measured as a membership degree between zero and one. This way the machine is able to distinguish the suitability among the solutions on the Pareto front, although they are equivalently valid in Pareto sense, and determine the particular solution having maximal overall suitability for the final purpose. It is noted that the diversity of solutions on the front is important for effective execution of the machine cognition. The strict search of non-dominated regions in the multi-objective solution space prematurely excludes some of the potential solutions. This is due to very low selection pressure towards the Pareto front in Pareto dominance-based evolutionary multi-objective (EMO) algorithms [9]. This results in aggregated solutions in objective space. In the present work a novel method for diversity preservation is employed termed as *relaxed dominance*. It refers to a degree of dominance in the terminology of MOEAs. It is noted that the present work is an extension of a design system presented earlier [10] with cognitive features.

The cognitive system is exercised in an application concerning an architectural design task, which involves a number of conflicting, linguistic criteria.

The organization of the work is as follows. Section II describes the fuzzy information processing system. Section III deals with solution diversity in evolutionary multi-objective optimization. Section IV describes a design experiment. This is followed by conclusions.

Manuscript received February 27, 2009. The authors are Delft University of Technology, 2628CR, Delft, The Netherlands.
(phone: +31-39250915 +31-39250901 ; e-mails: m.s.bittermann@tudelft.nl, o.ciftcioglu@tudelft.nl , i.s.sariyildiz@tudelft.nl)

II. FUZZY-NEURAL TREE MODELING DOMAIN KNOWLEDGE

For human-like information processing the methods of soft computing are presumably the most convenient. The salient soft computing methods are in the paradigms of neural nets and fuzzy logic [11]. In this work a neural tree is considered to assess the suitability of a solution in a human-like manner. 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 attribute introducing a non-linearity simulating a neuronal activity. In the present case, this attribute is established by means of a Gaussian function which has several desirable features for the intended goals; namely, it is a radial basis function ensuring a solution and the smoothness. At the same time it plays the role of a fuzzy 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 figure 1. Detailed structures of a neural tree are shown in figure 2. Figure 2a shows a terminal node connected to an inner node, and figure 2b and 2c show the connections among inner nodes.

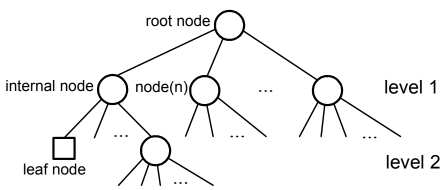


Fig. 1. The structure of a neural tree

Each terminal node, also called *leaf*, is labelled with an element from the terminal set $T=\{x_1, x_2, \ldots, x_n\}$, where x_i is the i-th component of the external input vector $\boldsymbol{x}$. Each link (i,j) represents a directed connection from node i to node j. A value w_{ij} is associated with each link as seen from figure 2. In a neural tree, the root node is an output unit and the leaf nodes, or 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 node is denoted x_i and it is introduced to another node j. A non-terminal node consists of a Gaussian radial basis function.

$$f(X) = w\phi(\| X - c \|^2) \tag{1}$$

where $\phi(.)$ is the Gaussian basis function, $\boldsymbol{c}$ is the center 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 σ_j at node j is used to measure the uncertainty associated with the inputs to this node, designated as external input X_j. X_j is related to the output of node i denoted as μ_i by relation

$$X_j = \mu_i w_{ij} \tag{2}$$

where w_{ij} is the weight connecting node i to node j. The centers of the basis functions are the same as the input weights of that node.

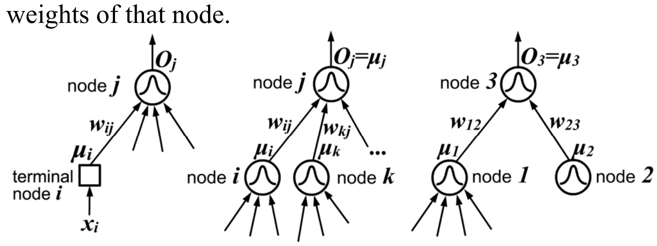


Fig. 2. Detailed structures of a neural tree with respect to different type of node connections

The output of node j is given by

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

which reduces to

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

We can express (4) in the following form

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

This implies that the width of the Gaussian is scaled by the input weight w_{ij}. In other words, as to the width, the shape of Gaussian fuzzy membership function is dependent on the input weights w_{ij} determined by the domain knowledge. It should be noted that this is a novel type of computation at each node which is quite different than conventional radial basis function (RBF) type computation, where the centers are determined by other means, clustering for instance. However, a non-terminal node itself can be seen as an RBF having different width for each dimension. For such a node, there should be at least two inputs with appropriate connection weights. The connection weights of a node should be normalized, so that the sum of the weights becomes equal to 1.

An example of a neural tree is shown in figure 3. It is to be noted that the tree has a hierarchical structure, where the root node describes the ultimate goal subject to maximization, and the tree branches form the objectives constituting this goal. It is noted that in the multi-objective optimization case the weights w_{1620}, w_{1720}, w_{1820}, *and* w_{1920} in figure 3 are not specified a-priori, but they are subject to identification after the optimization process is accomplished.

III. MULTI-OBJECTIVE EVOLUTIONARY OPTIMIZATION

A. System overview

In this work a cognitive system for design is developed using fuzzy information processing and evolutionary algorithm. The task of the cognitive system is to generate solutions matching the design criteria, which involve soft requirements. The design task is an architectural design,

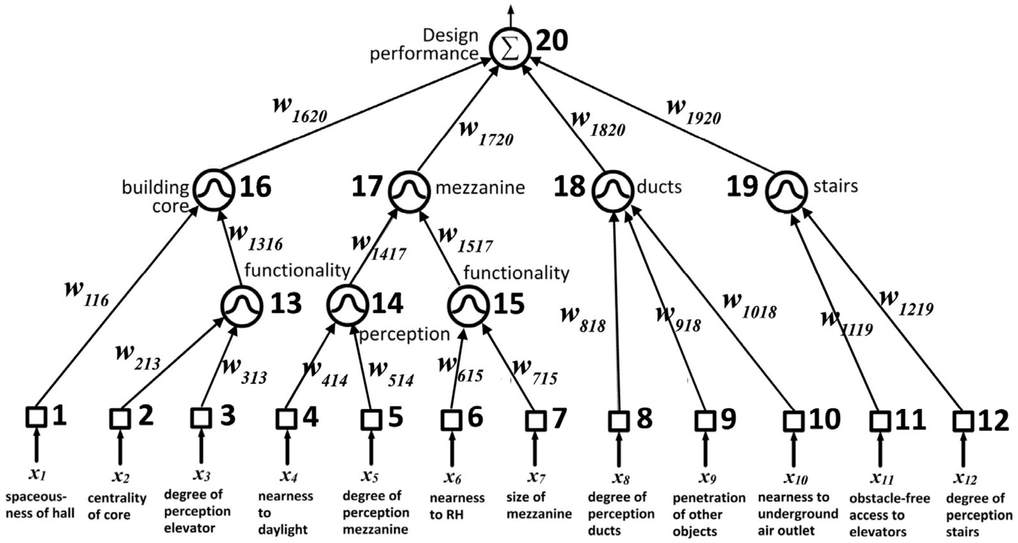


Fig. 3. The fuzzy neural tree used in the application example; the weights are subscribed according to the convention w_{ij}

where design variables are mainly soft in nature, i.e. they are given via associations to linguistic concepts. The softness is treated by means of fuzzy information processing. This is accomplished by means of fuzzy neural tree mentioned in the preceding section.

The neural tree plays the role of a fuzzy information processing system, providing feedback to the evolutionary algorithm about the effectiveness of the multi-objective optimization (MO).

Figure 4 shows the cognitive system. It is noted that the design process is cognitive, involving abundant visual information subject to processing. Therefore the investigations are carried out in virtual reality. The design task is accomplished in the following way. The genetic algorithm creates a population of random solutions, which are instantiated as scenes. The solutions are then evaluated by a virtual observer providing the system with virtual measurements of certain design features. The outcomes from the measurements are then fed into the neural tree model and processed to determine the degree of satisfaction of several criteria. It is noted that this information is obtained at the outputs of the nodes on the penultimate level of the tree. This provides the feedback information to the system. This information is used in a novel way in this work to rank the solutions in a Pareto sense. The novelty will be explained in the next section. Based on the ranking the genetic algorithm performs the genetic operations on the solutions as this is well known, generating new solutions and the process is iterated until a Pareto front is established. It is emphasized that the entire operation aims to maximize the outputs at the penultimate level of the neural tree in Pareto sense. Due to the fuzzy logic concept involved the maximal values at the outputs are unity.

B. *Neural tree as fuzzy information processor*

In figure 3 the neural tree nodes play the role of information processors that perform fuzzy AND operations. It is noteworthy to point-out that the input information to a node first is fuzzified by means of a Gaussian membership functions, thereafter AND operation is performed, to this information. The fuzzification is accomplished being directly related to the associated input weight. The weights are domain knowledge and they sum up to unity. This means the knowledge provided to the neural tree is directly used together with the input information, so that the commensurate outputs corresponding to all inputs are obtained with fuzzy AND operation. This is a novel way of performing AND operation in the sense that fuzzy numbers

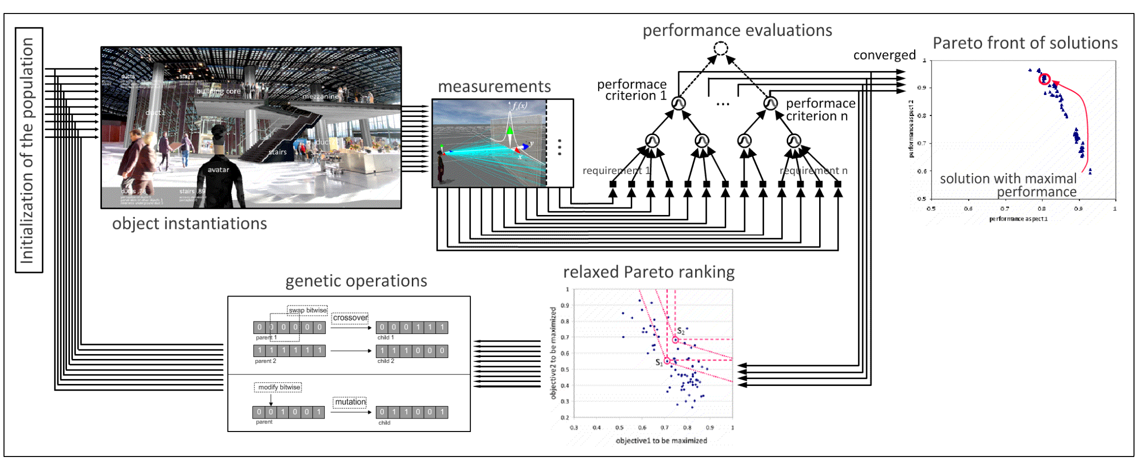


Fig. 4. The cognitive system

are directly obtained from the fuzzy membership functions and they are directly multiplied as an arithmetic operation, where directly means the whole process takes place inside the node, without explicit fuzzy set operations. This is explained schematically in figure 5, referring to figure 2c.

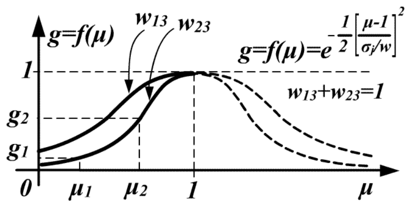
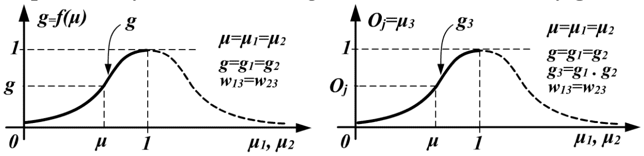


Fig. 5. Input fuzzification

In this figure only two inputs are considered without loss of generality. The variables w_{13} and w_{23} are input weights determining the width of the Gaussian membership functions. It is noted that in the figure $w_{13}<w_{23}$. This is clear from (5).

For two inputs, two distinct standard deviations are defined. In particular, the inputs can be equal, i.e., $\mu_1 = \mu_2$. This particular case occurs when the outputs of the two nodes delivering the inputs to the node we are considering are equal. This case is illustrated in figure 6a. In figure 6b it is clear that, if w_1 and w_2 are equal then the AND operation is expressed by means of a single Gaussian denoted by g.



(a) (b)

Fig. 6. Gaussian member as to the respective connection weights, where w=w_{13} = w_{23}

Since σ_j is a free parameter, by giving an appropriate width via (5), the result of the AND operation is given by

$$\begin{aligned} &\mu = \mu_1 = \mu_2 \\ &f(\mu) = f(\mu_1) = f(\mu_2) = g_1 = g_2 \\ &O_j = g_1 \cdot g_2 \end{aligned} \qquad (6)$$

This is illustrated in figure 7 where the left part of the Gaussian is approximated by a straight line.

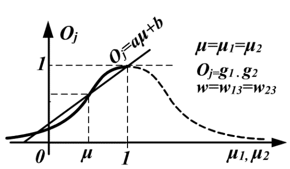


Fig. 7. Linear approximation to Gaussian function

In figure 7, optimizing the σ_j parameter, we obtain

$$O_j \cong \mu \qquad (7)$$

for the values μ and O_j can take between zero and one. In any case, for a node in the neural tree, (7) is satisfied for $\mu=O_j=0$ (approximately) and for $\mu=O_j=1$ (exact) inherently, while g_1 and g_2 are increasing function of μ_1 and μ_2. Therefore a linear relationship between O_j and μ in the range between 0 and 1 is a first choice from the fuzzy logic viewpoint; namely, as to the AND operation at the respective node, if inputs are equal, that is $\mu=\mu_1=\mu_2$ then the output of the node of μ_1 AND μ_2 is determined by the respective *triangular membership functions* in the antecedent space. Triangular fuzzy membership functions are the most prominent type of membership functions in fuzzy logic applications. For five inputs to a neural tree node, these membership functions are represented by the data sets given by Table I and Table II.

TABLE I
DATASET AT A NEURAL TREE NODE INPUT

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

TABLE II
DATASET AT A NEURAL TREE NODE OUTPUT

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

In general, the data sets given in Table I and Table II are named in this work as '*consistency conditions*'. They are used to calibrate the membership function parameter σ. This is accomplished by optimization.

At this point a few observations are due, as follows. If a weight w_{ij} is zero, this means the significance of the input is zero, consequently the associated input has no effect on the node output and thus also the system output. Conversely, if a w_{ij} is close to unity, this means the significance of the input is highest among the competitive weights directed to the same node. This means the value of the associated input is extremely important and a small change about this value has big impact on the node output O_j. If a weight w_{ij} is somewhere between zero and one, then the associated input value has some possible effect on the node output determined by the respective AND operation via (5). In this way, the domain knowledge is integrated into the logic operations.

The general properties of the present neural tree structure re as follows.

- If an input of a node is small (i.e., close to zero) and the weight w_{ij} is high, then, the output of the node is also small complying with the AND operation
- If a weight w_{ij} is low the associated input cannot have significant effect on the node output. This means, quite naturally, such inputs can be ignored.
- If all input values coming to a node are high (i.e., close to unity), the output of the node is also high complying with the AND operation
- If a weight w_{ij} is high the associated input x_i can have significant effect on the node output.

It might be of value to point out that, the AND operation in a neural-tree node is executed in fuzzy logic terms and the associated connection weights play an important role on the effectiveness of this operation.

The fuzzy logic interpretation of this process is illustrated in figure 8, referring to figure 2c. From the figure it is seen that the input x_1 and x_2 yield membership degrees at several membership functions in the antecedent space. After multiplication of these membership degrees, the membership

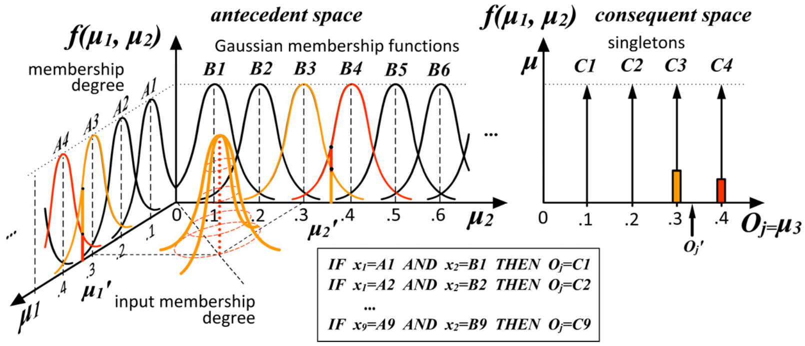


Fig. 8 Illustration of fuzzy logic information processing at an individual neural tree node with two inputs

degrees in the consequent space are determined at singleton membership functions, where every singleton belongs to the rule involving the corresponding Gaussian membership functions. The fuzzy rules are written in the figure as well.

C. Multi-objective optimization with a relaxed dominance concept

To deal with multi-objectivity, evolutionary algorithms with genetic operators are effective in defining the search direction for rapid and effective convergence. Basically, in a multi-objective case the search direction is not one but may be many, so that during the search a single preferred direction cannot be identified and even this is not desirable. In the evolutionary computation case a population of candidate solutions can easily hint about the desired directions of the search and let the candidate solutions during the search process be more probable for the ultimate goal. Next to the principles of genetic algorithm-directed optimization, in multi-objective (MO) algorithms, in many cases the use of Pareto ranking is a fundamental selection method. Its affectivity is clearly demonstrated for a moderate number of objectives, which are subject to optimization simultaneously [12]. Pareto ranking refers to a solution surface in a multidimensional solution space formed by multiple criteria representing the objectives. On this surface, the solutions are termed Pareto solutions. They are diverse but they are assumed to be equivalently valid as there are no other solutions which might surpass the Pareto solutions. Selection of one of the solutions among those many is based on some higher-order preferences, which require more insight into the problem at hand. This is necessary in order to make more refined decisions before selecting any solution represented along the Pareto surface. From the cognitive viewpoint, this means among the solutions available for the task, one is selected with conscience. The above construction is crucial for a cognitive system design. Namely, the problem formulation is not purely optimization-based but the final outcome is dependent on the availability and the nature of availability of the solutions. Even solutions may be sub-optimal as a trade-off for diversity, when cognition plays important role in decision-making.

The formation of the Pareto front is based on objective functions of the weighted *N* objectives which are of the form

$$F_i(\boldsymbol{x}) = f_i(\boldsymbol{x}) + \sum_{j=1, j\neq i} a_{ji} f_j(\boldsymbol{x}), i = 1,2,\ldots,N\ ) \quad (8)$$

where $F_i(x)$ is the new objective function; a_{ij} is the designated amount of gain in the *j*-th objective function for a loss of one unit in the *i*-th objective function. Therefore the sign of a_{ij} is always negative. The above set of equations require fixing the matrix *a*, which has all ones as diagonal elements. For the Pareto front we assume that, a solution parameter vector $\boldsymbol{x}_1$ dominates another solution $\boldsymbol{x}_2$ if $F(\boldsymbol{x}_1) \geq F(\boldsymbol{x}_2)$ for all objectives, and a contingent equality is not valid for at least one objective. $F_i(\boldsymbol{x})$ functions define the contour lines which form a convex hull.

For the *greedy* application of the MO algorithm, one uses the orthogonal contour lines at the point ***P*** as shown in figure 9.

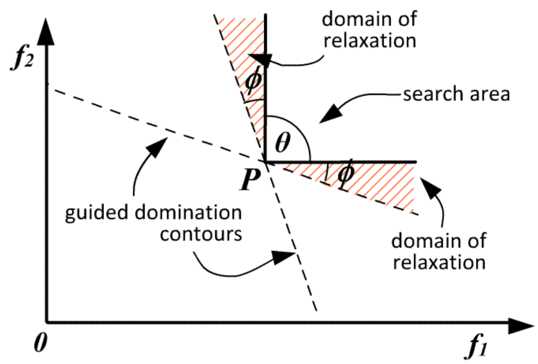


Fig. 9 Contour lines defining the search areas

In this figure the point ***P*** denotes one of the individuals among the population in the context of genetic algorithm (GA) based evolutionary search. In the greedy search many potential favourable solutions are prematurely excluded from the search process. This is because each solution in the population is represented by the point ***P*** and the dominance is measured in relation to the number of solutions falling into the *search domain* within the angle $\theta=\pi/2$. To avoid the premature elimination of the potential solutions, a relaxed dominance concept is implemented where the angle θ can be considered as the *angle for tolerance* provided $\theta>\pi/2$. The resulting Pareto front corresponds to a non-orthogonal *search domain* as shown in figure 9. The wider the angle beyond $\pi/2$ the more tolerant the search process and vice versa. For $\theta<\pi/2$, θ becomes the *angle for greediness*.

Domains of relaxations are also indicated in figure 9. In the greedy case the solutions are expected to be more effective but aggregated. In the latter case, the solutions are expected to be more diversified but less effective. In both cases, the fitness of the solutions can be ranked by the fitness function

$$\boldsymbol{R}_{fit} = \frac{1}{N(\theta) + n} \tag{9}$$

where n is the number of potential solutions falling into the *search domain*. Although $N(\theta)$ can be on-line modified during the search, it is expectedly constant once θ is determined. However, without the analysis of the functionality of $N(\theta)$ it is difficult to establish such a function by experiments.

Above considerations and the ad hoc formulation given by (8) can be put into more precise mathematical terms based on the formulation given by (10), as follows. Let (8) be expressed by

$$\begin{aligned} F_1 &= f_1 + a_{21} f_2 + ... + a_{n1} f_n \\ F_2 &= a_{12} f_1 + f_2 + ... + a_{n2} f_n \\ &.......................... \\ F_n &= a_{1n} f_1 + a_{2n} f_2 + ... + a_{nn} f_n \end{aligned} \tag{10}$$

In matrix equation form, (10) becomes

$$F = \begin{bmatrix} F_1 \\ F_2 \\ ... \\ F_n \end{bmatrix} = \begin{bmatrix} a_{11} \; a_{21} \; ..... \; a_{n1} \\ a_{12} \; a_{22} \; ..... \; a_{1n} \\ .... \\ a_{1n} \; a_{2n} \; ..... \; a_{nn} \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ ..... \\ f_n \end{bmatrix} \tag{11}$$

For the sake of simplicity in the description below only two objectives are considered while the results are valid for any dimension. The objective functions for this case are given by

$$\begin{aligned} F_1 &= f_1 + a_{21} f_2 \\ F_2 &= a_{12} f_1 + f_2 \end{aligned} \tag{12}$$

In a two-dimensional coordinate system, the contour lines in figure 9 are orthogonal and non-orthogonal respectively. The search area in the latter case includes also the domains of relaxations, which are added to the search area of the orthogonal system as seen in figure 9. In the non-orthogonal system, the area of selected solutions is relatively wider while some of the solutions are not dominating the solution at the point ***P***. However, as a trade-off it provides more diversity at the final Pareto front, while the front is not totally non-dominated. The solutions at the front are more probably non-dominated in the middle part of the front where f_1 and f_2 are close to each other. Conversely, the solutions may be more dominated at the regions close to edges of the front [13]. This situation occurs since the greedy algorithm is applied with respect to the non-orthogonal system taking the point ***P*** as origin. By doing so, the search algorithm *remains the same*, but it uses the coordinates of the new non-orthogonal system [14]. However, this approach does not address the problem of aggregation especially in the higher multi-dimensional optimization. This means, the Pareto front is potentially wider without resolving the aggregation phenomenon. That is. the potentially wider Pareto front is left ineffective.

As a novel approach, in this work, during the genetic search, each member of the population is considered to be represented by the point ***P*** seen in figure 9, and the solutions falling into the relaxation domains are included to the non-dominated solutions. In other words, some dominated solutions are accrued to the non-dominated ones to form the next-generation solutions. This means, the orthogonal system is *not replaced* by the non-orthogonal system but the greedy non-dominated orthogonal search space is relaxed. The relaxed domains simply contribute to the greedy search domain with some additional, potentially lucrative solutions. Interestingly, this situation is similar to the classical gradient-based optimization method, where each iteration the step length towards the global maxima or minima determined by the gradient. The step length should be small enough to ensure the stability of the convergence [15, 16]. If the step length is zero, approach to minima or maxima does not occur. If it is too big, convergence does not occur. For similar reasons, in the evolutionary computation the angle ϕ defining the relaxation domain should be kept small. In this way the stability of the algorithm is maintained and the effectiveness is enhanced. It should be noted that, although the angle ϕ is small it plays role for each population member at each generation making the net effect highly significant. The role of angle ϕ is comparable to that of step length of gradient-based optimization. If ϕ is zero, greedy search in orthogonal system occurs. Then the final result in the extreme case aggregates to one solution. If it is too big, convergence does not occur.

In (12) the small-enough designation of the parameters a_{ij} is crucial for the performance of the evolutionary computation. It is characterized by the cosines of the angle between respective coordinate axes, i.e., ϕ and it is expected to be equal to 10^o or less, although it is also application dependent. The normalization of these cosines yields the directive cosines of the coordinate systems. The coordinate transformation between orthogonal and non-orthogonal systems is determined by these direction cosines. If we denote the direction cosines as q_{ij}, the transformation matrix becomes

$$Q = \begin{bmatrix} q_{11} \; q_{12} \; ..... \; q_{1n} \\ q_{21} \; q_{22} \; ..... \; q_{2n} \\ .... \\ q_{n1} \; q_{n2} \; ..... \; q_{nn} \end{bmatrix} \tag{13}$$

which transforms the non-orthogonal system to the orthogonal system and vice versa via

$$x = Q\,x' \tag{14}$$

$$x' = Q^{-1}\,x \tag{15}$$

where x' denotes the non-orthogonal system $\boldsymbol{x'}=[x_1',x_2', ... ,x_n']^T$. The directive cosine row vectors of (13) are given by

$$d_i = \left[d_{1i}\; d_{2i} \,..d_{ni}\right] = \frac{1}{\sqrt{\sum_{j=1}^{n} a_{ji}^2}} [a_{1i} \;\; a_{2i} \;\; .... \;\; a_{ni}] \tag{16}$$

Direction cosine row vector corresponds to column vectors in (13) so that for each column in (13)

$$\sqrt{\sum_{i=1}^{n} q_{ji}^2} = 1 \qquad j = 1,2,....,n \tag{17}$$

The direction cosines matrix formed by the respective direction cosine row vectors is related to Q in (14) by

$$D = \begin{bmatrix} [d_{11}\ d_{21}\ \ .....\ d_{n1}\ ] \\ [d_{12}\ d_{22}\ \ .....\ d_{n2}] \\ .... \\ [d_{1n}\ d_{2n}\ \ .....\ d_{nn}] \end{bmatrix} = Q^T \tag{18}$$

In two-dimensional case the directive cosine row vectors with respect to (18) are given by

$$\begin{aligned} d_1 &= [d_{11}\ d_{21}] = \left[a_{11}/\sqrt{1+a_{21}^2} \qquad a_{21}/\sqrt{1+a_{21}^2}\right] \\ d_2 &= [d_{12}\ \ d_{22}] = \left[a_{12}/\sqrt{1+a_{12}^2} \qquad 1/\sqrt{1\ +a_{12}^2}\right] \end{aligned} \tag{19}$$

The coordinate transformation of points falling into the relaxed search domains as seen in figure 9 is given by

$$\begin{aligned} \begin{bmatrix} f_1^P \\ f_2^P \end{bmatrix} &= \begin{bmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{bmatrix} \begin{bmatrix} F_1^P \\ F_2^P \end{bmatrix} \\ &= \begin{bmatrix} d_{11} & d_{21} \\ d_{12} & d_{22} \end{bmatrix} \begin{bmatrix} F_1^P \\ F_2^P \end{bmatrix} \end{aligned} \tag{20}$$

and

$$\begin{aligned} \begin{bmatrix} F_1^P \\ F_2^P \end{bmatrix} &= \begin{bmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{bmatrix}^{-1} \begin{bmatrix} f_1^P \\ f_2^P \end{bmatrix} \\ &= \begin{bmatrix} d_{11} & d_{21} \\ d_{12} & d_{22} \end{bmatrix}^{-1} \begin{bmatrix} f_1^P \\ f_2^P \end{bmatrix} \end{aligned} \tag{21}$$

The importance of coordinate transformation becomes dramatic especially in higher dimensions. In such cases the spatial distribution of domains of relaxation becomes complex and thereby difficult to implement. Namely, in multidimensional space the volume of a relaxation domain is difficult to imagine, and more importantly it is difficult to identify the population in such domains. Therefore one needs a systematic approach for identification by computation and not by inspection or by something else. This systematic approach is the coordinate transformation as follows. Basically for each solution point, say ***P*** in figure 11, the point is temporarily considered to be a reference point as origin, and all the other solution points in the orthogonal coordinate system are converted to the non-orthogonal system coordinate by (15). For instance for three objectives, we write

$$F = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix} = \begin{bmatrix} 1\ a_{21} & a_{31} \\ a_{12}\ \ 1 & a_{1n} \\ a_{13}\ \ a_{23} & 1 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix}, \tag{22}$$

where three parameters should be designated, namely a_{12}, a_{13}, and a_{23} in advance if the coefficient matrix is taken to be symmetrical. They are expectedly around the range 0.1 and the coefficients can be modified during the search, if necessary. After conversion, all points which have positive coordinates in the non-orthogonal system correspond to potential solutions contributing to the next generation in the evolutionary computation. If any point possesses a negative component in the new coordinate system, the respective solution is discarded meaning eliminated from the offspring possibility. This fundamental process employing (15) is easy to implement even in any higher dimensional space. However, in higher dimensions the situation is extremely complex and relaxation domains are difficult to be taken into account by other means. This is one of the essential motivations to explore many different methods for effective Pareto front formation in literature [17, 18]. The presence of the relaxation guarantees the prevention of aggregation as the relaxation is meant for. This is exemplified in the following section of experimentation, where the demonstration example is taken as an architectural design.

IV. Application

A. Fuzzy neural tree as intelligent system for cognition

In the following implementation the aim is to compare the effectiveness of the greedy vs. a relaxed Pareto ranking with respect to Pareto optimal front formation in a four-dimensional objective space. It is emphasized that during the search the fuzzy information processing described above is used to compute the extent a solution fulfils the objectives. The objectives in the present application are modeled by the neural tree branches below the nodes 16-19 in figure 3, which are the penultimate nodes in the neural tree. During evaluation of a design alternative the tree is provided with inputs at its leaf nodes and the fuzzification processes are carried out. The fuzzification yields the satisfaction of an elemental requirement at the terminal nodes of the neural tree. These requirements are some desirable features expressed by means of fuzzy membership functions at the terminal nodes of the tree. An example of a design requirement is shown in figure 10. It is a requirement on the perception of an object in the design, which is the stairs. The requirement is fully satisfied for a degree of perception about 0.02, and it diminishes otherwise as seen from figure 10. The degree of perception of the object is computed using a probabilistic perception theory [19]. Using the membership function in figure 10 the perception degree is converted into a degree of satisfaction for the requirement. In the same way other quantities in the design are measured and converted into satisfactions using specific membership functions at the terminals. The fuzzified information is then processed by the inner nodes of the tree. These nodes perform the AND operations using Gaussian membership functions as described above.

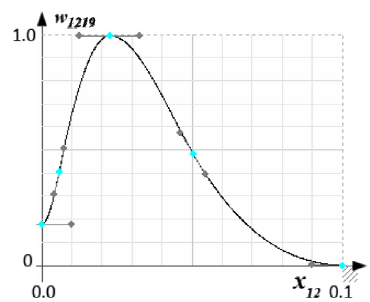


Fig. 10 Membership function at node 12 in figure 3

Finally the sequence of logic operations starting from the model input yield the performance at the penultimate node outputs of the model. This means the more satisfied the elemental requirements at the terminal level are, the higher the outputs will be at the nodes above, finally increasing the design performance at the root node of the tree. It is to be noted that the design performance in this example depends on the performance of the design objects in the scene, which are subject to optimal positioning. Clearly, the better every design object is, the better the whole scene is. Next to the evaluation of the design performance score, due to the fuzzy logic operations at the inner nodes of the tree, the performance of any sub-aspect is obtained as well. This is a desirable feature in design, which is referred to as *transparency*.

Having established the performance evaluation model, it is used for the evolutionary search process aiming to identify designs with high design performance. In the present case we are interested in a variety of alternative solutions that are equivalent in Pareto sense. The design is therefore treated as a multi-objective optimization as opposed to a single-objective optimization. In single-objective case exclusively the design performance, i.e. the output at the root node of the neural tree, would be subject to maximization. In the latter case, the solution would be the outcome of a mere convergence and any cognition aspect would not be exercised. In the multi-objective implementation the outputs of the nodes 16-19, which are the penultimate nodes, are subject to maximization. Their values are used in the fitness determination procedure of the genetic algorithm [20, 21]. Employing the fuzzy neural tree in this way the genetic search is equipped with some human-like reasoning capabilities during the search. The part of the tree beyond the penultimate nodes is for the defuzzification process, which models cognition, so that ultimately the design performance is obtained at the root node.

B. *Analysis of the Pareto front*

The results from the design with multi-objective optimization are presented in figures 11 and 12.

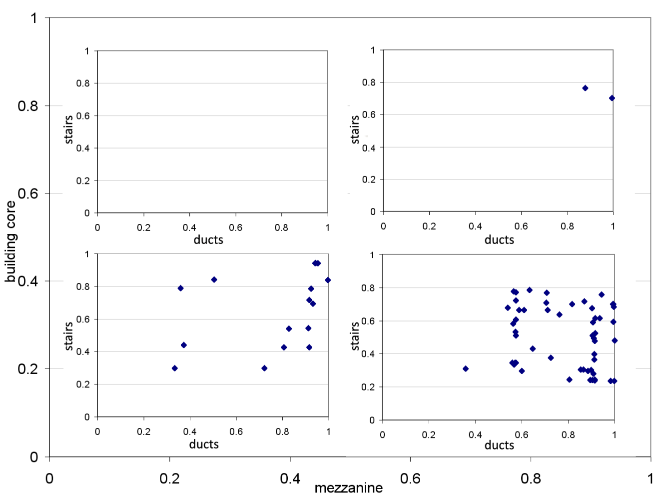


Fig. 11 Pareto optimal designs with respect to the four objective dimensions using *greedy* Pareto ranking

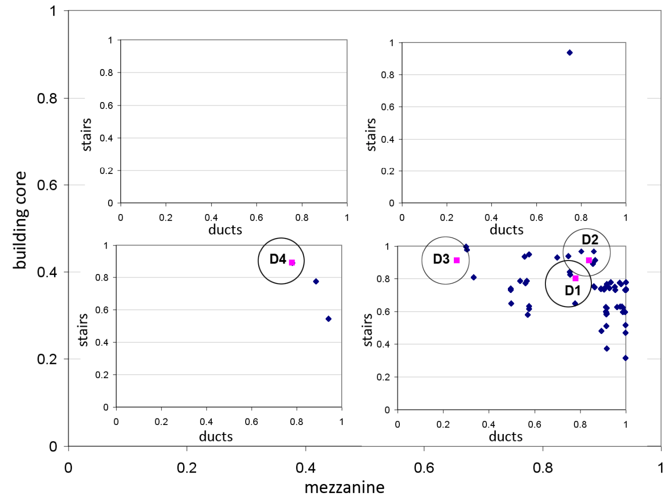


Fig. 12 Pareto optimal designs with respect to the four objective dimensions using *relaxed* Pareto ranking

The figures respectively show the result with and without relaxation algorithm. Figure 11 shows the front when the greedy Pareto dominance concept is applied. Figure 12 shows the front after the same amount of generations applying the relaxed dominance concept. The *angle of relaxation* is taken as $\phi = 10^o$. The positive effect of relaxation is clearly seen comparing the fronts. In the greedy case the front did not establish very distinctly. This is seen from figure 11, where some solutions that are inferior in the ducts/stairs part of objective space remain in the population, while these are not present in the relaxed case. From figure 11 it is also noted that some solutions exist with a low score in the mezzanine/building core part of objective space, while in the relaxed case the solutions have a higher score in this respect. We can say that the solutions in the relaxed case are more 'motivated' to come to the front compared to the greedy case. This is explained considering that in the four dimensional objective space the size of the space is large, so that merely few solutions are dominated in a strict, i.e. greedy, sense. This results in low pressure towards the Pareto front. In the relaxed case some inferior solutions are counted as being dominant, due to the expanded *angle for tolerance*, i.e. $\theta = 110^o$. This way the selection pressure is finely adjusted, since the information on a greater portion of the population is contributing to the front formation.

Two resulting Pareto-optimal designs are shown in figures 13 and 14. The designs belong to different regions on the Pareto front as shown by circles in figure 12. The design *D1* has a better evaluated performance of the mezzanine compared to *D2*. As a trade-of *D1* has a slightly inferior performance of the stairs. This is due to the conflicting nature of the requirements for these objects. The relaxed dominance approach employs 10^o between the respective orthogonal and non-orthogonal system axes. Therefore the basic computations for this yields the F matrix in (10) as

$$\begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} = \begin{bmatrix} 1 & -.176 & -.176 & -.176 \\ -.176 & 1 & -.176 & -.176 \\ -.176 & -.176 & 1 & -.176 \\ -.176 & -.176 & -.176 & 1 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \quad (23)$$

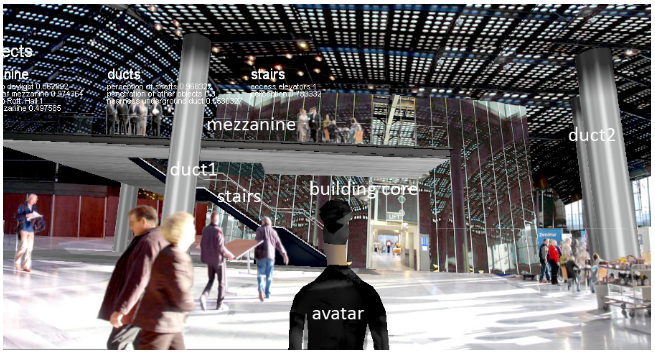


Fig. 13 Pareto optimal design *D1*

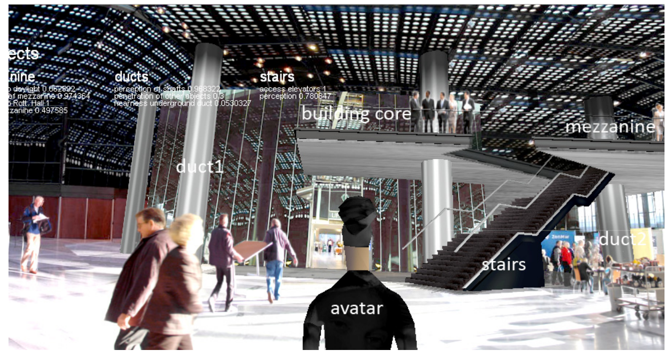


Fig. 14 Pareto optimal design *D4*

so that, the cosine direction matrix given by (18) becomes

$$D=\begin{bmatrix} .956 & -.168 & -.168 & -.168 \\ -.168 & .956 & -.168 & -.168 \\ -.168 & -.168 & .956 & -.168 \\ -.168 & -.168 & -.168 & .956 \end{bmatrix}=Q^T \quad (24)$$

The corresponding weighted objectives F_1 and F_2 are given by (21) as

$$\begin{bmatrix} f_1^P \\ f_2^P \\ f_3^P \\ f_4^P \end{bmatrix}=\begin{bmatrix} .956 & -.168 & -.168 & -.168 \\ -.168 & .956 & -.168 & -.168 \\ -.168 & -.168 & .956 & -.168 \\ -.168 & -.168 & -.168 & .956 \end{bmatrix}\begin{bmatrix} F_1^P \\ F_2^P \\ F_3^P \\ F_4^P \end{bmatrix} \quad (25)$$

and the inverse of (25) becomes

$$\begin{bmatrix} F_1^P \\ F_2^P \\ F_3^P \\ F_4^P \end{bmatrix}=\begin{bmatrix} 1.220 & .331 & .331 & .331 \\ .331 & 1.220 & .331 & .331 \\ .331 & .331 & 1.220 & .331 \\ .331 & .331 & .331 & 1.220 \end{bmatrix}\begin{bmatrix} f_1^P \\ f_2^P \\ f_3^P \\ f_4^P \end{bmatrix} \quad (26)$$

The results given by (25) and (26) are interesting because the Pareto front determination by evolutionary multi-objective optimization is considerably simplified. The selection of a_{ij} coefficients is a marginal issue and it remains marginal as they are mainly application dependent. However, it is clear that the offline a_{ij} coefficients in (26) should be small compared to unity. For an increasing amount of dimensions, they should be diminished commensurately for a stable convergence.

From figure 3, at the root node, the performance score is computed by the defuzzification process given by

$$w_1 f_1 + w_2 f_2 + w_3 f_3 + w_4 f_4 = p \quad (27)$$

where f_1 is the output of the node $O_2(1)$; f_2 of node $O_2(2)$; f_3 of node $O_2(3)$; and f_4 of node $O_2(4)$. That is, they denote the performance values of the design objects forming the scene, which are subject to maximization. The variable p denotes the design performance which is also requested to be maximized. In (27) w_1;…; w_4 denote the connection weights $w_2(1)$;…; $w_2(4)$ shown in figure 3 respectively. It is noted that $w_1+w_2+w_3+w_4=1$.

In this design exercise, the cognitive design viewpoint plays important role. This means it is initially uncertain what values $w_1,\dots w_4$ should have. Namely, the node outputs $f_1, \dots, f_4$ can be considered as the *design feature vector,* and the reflection of these features can be best performed if the weights w_1 ; …; w_2 define the same direction as that of the feature vector. Hence the components of the unit vector along the feature vector are computed as

$$u_1=\frac{f_1}{\sqrt{f_1^2+f_2^2+\dots+f_n^2}}\,; \quad u_2=\frac{f_2}{\sqrt{f_1^2+f_2^2+\dots+f_n^2}}\,; \quad \dots \quad u_n=\frac{f_n}{\sqrt{f_1^2+f_2^2+\dots+f_n^2}} \quad (28)$$

Normalising the components and equating them to the weights yields

$$w_1=\frac{f_1}{f_1+f_2+f_3+f_4}\,; \quad w_2=\frac{f_2}{f_1+f_2+f_3+f_4}$$
$$w_3=\frac{f_3}{f_1+f_2+f_3+f_4}\,; \quad w_4=\frac{f_4}{f_1+f_2+f_3+f_4} \quad (29)$$

In general, if there are n objectives at the penultimate layer of the neural tree, we can write that

$$u_1=\frac{f_1}{\sqrt{f_1^2+f_2^2+\dots+f_n^2}}\,;\dots;\; u_n=\frac{f_n}{\sqrt{f_1^2+f_2^2+\dots+f_n^2}} \quad (30)$$

Above computation implies that, the performance p for each genetic solution is given by

$$p=\frac{f_1^2+f_2^2+\dots+f_n^2}{f_1+f_2+\dots+f_n} \quad (31)$$

Therefore, (31) is computed for all the design solutions on the Pareto front. Then the *solution with maximal performance* is selected among the Pareto solutions. This way the particular design is identified as a solution candidate with the corresponding $w_1, w_2, \dots, w_n$ weights. These weights form a priority vector $\boldsymbol{w^*}$. In the present application (31) becomes

$$p=\frac{O_2(1)^2+O_2(2)^2+O_2(3)^2+O_2(4)^2}{O_2(1)+O_2(2)+O_2(3)+O_2(4)} \quad (32)$$

Where $O_2(n)$ is the n-th output on the penultimate level of the neural tree. If for any reason this candidate solution is not appealing, the next candidate is searched among the available design solutions with a desired design feature vector and the relational attributes, i.e., $w_1, w_2, \dots, w_n$. One should note that, although performance does not play role in the genetic optimization, Pareto front offers a number of design options with fair performance leaving the final choice dependent on other environmental preferences. Using (32) second-order preferences are identified that are most promising for the task at hand, where ultimately maximal design performance is pursued. This is the essential cognitive component in this particular task being exercised.

It is emphasized that the entire operation is a form of machine cognition, where there is a systematic distinction made among solutions that are equally valid in Pareto sense, implying that a suitable second-order criterion is identified by computation.

V. Conclusions

Integration of cognitive aspects into an evolutionary system for design is described. Based on an evolutionary design system published earlier [10] the extension of that work with some cognitive features is investigated. The investigation is accomplished in a virtual reality environment by multi-objective-optimization-based positioning of objects. The novelty of the research is the relaxation of the MO search and the integration of fuzzy information processing into the search process, where the multi-objective functions are four nodal outputs of the fuzzy system, so that machine cognition is enabled. That is, due to the special fuzzy logic implementation the system is able to identify the most suitable vector specifying the relative importance among the objectives. That is, a second-order aspect is systematically introduced, based on the contingent availability of solutions. This makes the system a cognitive system beyond being merely an optimization process with fuzzy information processing. The search method used to identify a desired solution is evolutionary computation with the Pareto front based on a novel formulation for fitness gradation, where non-dominated search domains are relaxed with some dominated search domains. As result of this method the analysis revealed that the results from strictly non-dominated search are inferior to the relaxed counterpart. This means the relaxation of the strict non-dominated domain search favors the potential solutions, so that they are not prematurely excluded in the search process. In other words, long term benefits are favored against the short term gains with regard to the solutions in Pareto sense. The Pareto front expands into complex domains in the multidimensional-space, i.e. more diversified positions towards to non-dominated regions which are feasible regions for solutions. The complexity is dealt with the coordinate transformation from orthogonal search space to non-orthogonal search space. The fitness function in the genetic search makes use of fuzzy information processing in the form of a neural tree for soft computation. This way the soft information at the tree inputs, which are the design requirements, is dealt with. The computations are based on fuzzy logical AND operations by means of Gaussians at the neural tree nodes ultimately providing rationales for the design as well as improving it. The fuzzy neural system is able to handle the complexity of criteria as well as their fuzzy nature at the same time, equipping the evolutionary search with human-like reasoning capabilities during the fitness evaluation.

Combining fuzzy information processing with multi-objective optimization applying a relaxed dominance concept makes the research a unique cognitive system implementation in the context of cognitive design.

References

[1] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. Grunert, "Performance assessment of multiobjective optimizers:An analysis and review," *IEEE Trans. on Evolutionary Computation,* vol. 7, pp. 117-132, 203.

[2] C. A. C. Coello, D. A. Veldhuizen, and G. B. Lamont, *Evolutionary Algorithms for Solving Multiobjective Problems*. Boston: Kluwer Academic Publishers, 2003.

[3] K. Deb, *Multiobjective Optimization using Evolutionary Algorithms*: John Wiley & Sons, 2001.

[4] H. Ishibuchi and T. Murata, "Multi-objective genetic local search algorithm," in *1996 IEEE International Conference on Evolutionary Computation*, Nagoya, Japan, 1996, pp. 119-124.

[5] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, "Combining convergence and diversity in evolutionary multi-objective optimization," *Evolutionary Computation,* vol. 10, pp. 263-282, 2002.

[6] G. Strube, C. Habel, L. Konieczny, and B. Hemforth, "Kognition," in *Handbuch der Kunstlichen Intelligenz*, G. Görz, C.-R. Rollinger, and J. Schneeberger, Eds. Munchen: Oldenbourg Wissenschaftsverlag, 2003, pp. 19-72.

[7] E. Ahle and D. Söffker, "A concept for a cognitive-oriented approach to build autonomous systems," in *2005 IEEE Int. Conf. on Systems, Man and Cybernetics*, Big Island, Hawaii, 2005, pp. 2929-2935.

[8] E. Ahle and D. Söffker, "A cognitive-oriented architecture to realize autonomous behaviour – part I: Theoretical background," in *2006 IEEE Conf. on Systems, Man, and Cybernetics*, Taipei, Taiwan, 2006, pp. 2215-2220.

[9] H. Sato, H. E. Aguirre, and K. Tanaka, *Controlling dominance area of solutions and its impact on the performance of MOEAs*. Heidelberg: Springer, 2007.

[10] Ö. Ciftcioglu and M. S. Bittermann, "Solution diversity in multi-objective optimization: A study in virtual reality," in *Wold Congress Comp. Intelligence*, Hong Kong, 2008.

[11] S. Mitra, S. K. Pal, and P. Mitra, "Data mining in soft computing framework: A survey," *IEEE Trans. on Neural Networks,* vol. 13, pp. 3-14, 2002.

[12] K. C. Tan and J.-X. Xu, "(Eds.) Proc. 2007 IEEE Congress on Evolutionary Computation," in *2007 IEEE Cong*, Singapore, 2007.

[13] K. Deb, P. Zope, and A. Jain, "Distributed computing of pareto-optimal solutions with evolutionary algorithms," in *9th annual conference on Genetic and evolutionary computation* London, England, 2003, pp. 532-549.

[14] J. Branke, T. Kaussler, and H. Schmeck, "Guiding multi-objective evolutionary algorithms towards interesting regions," AIFB University of Karlsruhe, Germany 2000.

[15] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear Programming*. New York: John Wiley & Sons, Inc, 1993.

[16] J. Kuester, *Optimization Techniques with FORTRAN*: McGraw-Hill College, 1973.

[17] A. Jaszkiewicz, "On the computational efficiency of multiple objective metaheuristics: The knapsack problem case study," *European Journal of Operational Research,* vol. 158, pp. 418-433, 2004.

[18] E. J. Hughes, "Evolutionary many-objective optimisation:many once or one many?," in *IEEE Congress on Evolutionary Computation CEC'2005*, Edinburgh, Scotland, 2005, pp. 222-227.

[19] M. S. Bittermann, I. S. Sariyildiz, and Ö. Ciftcioglu, "Visual perception in design and robotics," *Integrated Computer-Aided Engineering,* vol. 14, pp. 73-91, 2007.

[20] D. E. Goldberg, *Genetic Algorithms*. Reading, Massachusetts: Addison Wesley, 1989.

[21] Z. Michalewicz, *Genetic Algorithms and Data Structures=Evolution Programs*. Berlin: Springer, 1996.