A Computational Design System with Cognitive Features Based on Multi-objective Evolutionary Search with Fuzzy Information Processing

Michael S. Bittermann
Delft University of Technology, The Netherlands

A system for architectural design is presented, which is based on combining 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 system possesses cognitive features, where cognition is defined as final decision-making based not exclusively on optimization outcomes, but also on some higher-order aspects, which do not play role in the pure optimization process. That is, the machine is able to distinguish among the equivalently valid solution alternatives it generated, where the distinction is based on second order preferences that were not pin-pointed by the designer prior to the computational design process. This is accomplished through integrating fuzzy information processing into the multi-objective evolutionary search, so that second-order information can be inductively obtained from the search process. 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.

Introduction

Design is complex. This is because it involves conflicting goals that are often vague. For example a design is demanded to be functional, look appealing and have moderate costs. The vagueness of these objectives makes it problematic to precisely compare alternative solutions during design, and the conflicting nature of the objectives make it problematic to

J.S. Gero (ed.): *Design Computing and Cognition'10*, pp. 505–524.

reach optimality. Another source of complexity is that the amount of possible solutions is excessively large in general. This is due to the combinatorial explosion in the parameter domain, making it difficult to ensure a designer does not miss a superior solution during the design process. A third source of complexity is that prior to the design it is generally not clear how important goals are relative to each other. For example it is difficult to specify exactly how important the functionality of a design should be taken compared to cost aspects prior to knowing what the implication of such a commitment is. That is, before finding most suitable solutions for the goals, and thereby becoming aware of the nature of the inevitable trade-offs occurring in the task at hand, it is premature to commit to a relative importance among high-level goals.

The complexity makes it difficult to accomplish scientific means for design enhancement, i.e. providing designers with means that ensure to some extend the suitability of their designs for an intended purpose. There have been a number of works addressing this issue. The first ones were using methods of classical artificial intelligence [1-3]. These methods are based a rigid inference mechanism. Namely, the identification of a deficiency in a design has a set of actions associated for improving the deficiency that is defined a-priori. For example in [3], when it is detected that a room has insufficient openness, the computations move or remove elements blocking the view. This predefined association of goal and action is clearly a problematic approach when *openness* for instance is not the only objective at hand, but other factors play a role as well, such as costs, functionality etc. Also the approach is challenged when the action to take involves a number of parameters to decide upon in combination, e.g. determining the most suitable location to move the object blocking the view to.

A promising approach in this respect is the emerging information processing paradigm known as computational intelligence (CI), also referred to as soft computing [4]. CI methodologies are superior over the classical AI methodologies in particular with respect to dealing with vagueness of objectives and combinatorial explosion in the parameter domain. Among the CI methodologies, in particular evolutionary algorithms became popular in the last decade for identifying optimal solutions for different aspects in design problems [5-9]. A difference of evolutionary approaches compared to the classical AI approach lies in the fact that the former does not involve a-priori defined solutions for preconceived conditions. Instead the solutions are the result of a stochastic process involving an evaluation-generation loop, where the evaluation and the generation are *separate* sub-processes, i.e. they are not associated rigidly and a-priori as it is the case in classic AI approaches. Among the evolutionary algorithms the class known as multi-objective evolutionary algorithms (MOEA) is particularly promising for design, as it is able to deal with the multiplicity of conflicting criteria that is common in design. However, it is important to note that the ability of MOEAs to handle multiple objective dimensions is generally limited to about four or five [10]. This issue will be further explained in section three. Design usually involves much more than five requirements, while some of the requirements may be vague in nature. Therefore it is clear that another approach to represent the objectives is needed compared to conventional applications of MOEAs.

This paper presents a novel system for performance-based computational design that addresses the complexity issues mentioned above using a combination of two CI methodologies. One bottleneck addressed concerns MOEAs ability to deal with many objectives at the same time. The strategy used in the present work is to reduce the amount of objective dimensions. This is achieved by establishing a neuro-fuzzy model in which a number of elemental requirements are aggregated forming fewer, more complex and more abstract concepts. In this way the model mimics the human-like comprehension of the design objectives. It is noted that aspects of the approach have been presented in an earlier paper [11], where the focus was on enhancing the effectiveness of multi-objective evolutionary algorithm. This paper elucidates the role of the cognitive approach in design applications. In particular attention is paid to the role of fuzzy information processing in a computational design process, and to the implications of the machine cognition concept on alleviating the decision making.

The paper is structured as follows. In section two fuzzy modeling for design evaluation is described. In section three multi-objective evolutionary search is described. In section four the two components are combined yielding an intelligent system for performance-based design exhibiting cognitive features. This is followed by conclusions.

Evaluating Design Performance

During design we need to estimate the suitability of a solution for our intended purpose. This means beyond observing the direct physical features of a solution, they need to be interpreted with respect to the goals pursued. For example, designing a space it may be desirable that the space is *large* or it is *nearby* another space. Clearly these requirements have to do with the size of the space, and the distance among spaces respectively, which are physical properties of the design. However, it is clearly noted that largeness is a concept, i.e. it does not correspond immediately to a physical measurement, but it is an abstract feature of an object. It is also noted that there is generally no sharp boundary from on which one may attribute such a linguistic feature to an object. For instance there is generally no specific size of a room from on which it is to be considered large, and below which it is not large. Many design requirements have this character, i.e. they do not pin-point a single acceptable parameter value for a solution, but a range of values that are more or less satisfactory. This is essentially because design involves conflicting requirements, such as spaciousness versus low cost. Therefore many requirements are bound to be merely partially fulfilled.

Such requirements characterized as *soft*, and they can be modelled using fuzzy sets and fuzzy logic from the soft computing paradigm [12]. A fuzzy set is characterized via a function *termed fuzzy membership function*, which is an expression of some domain knowledge. Through a fuzzy set an object is associated to the set by means of a membership degree μ. Two examples of fuzzy sets are shown in Figure 1.

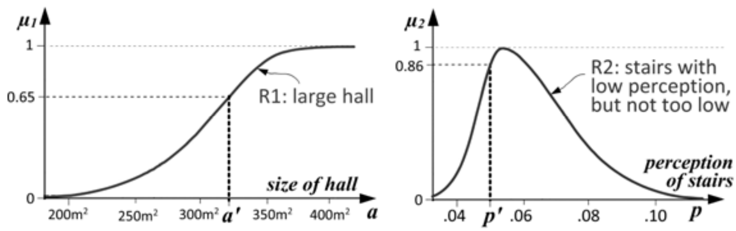


Fig. 1. Two fuzzy sets expressing two elemental design requirements

By means of fuzzy membership functions a physical property of a design, such as size, can be interpreted as a degree of satisfaction of an elemental requirement. The degree of satisfaction is represented by the membership degree.

The requirements considered here are relatively simple, whereas the ultimate requirement for a design - namely a high design performance - is complex and abstract. Namely the latter one is determined by the simultaneous satisfaction of a number of elemental requirements.

In this work the performance is computed using a fuzzy neural tree [13]. It is particularly suitable to deal with the complex linguistic concepts like design performance. A neural tree is composed of one or several model output units, referred to as *root nodes* that are connected to input units termed *terminal nodes,* and the connections are via logic processors termed *internal nodes.* An example of a fuzzy neural tree is shown in Figure 2.

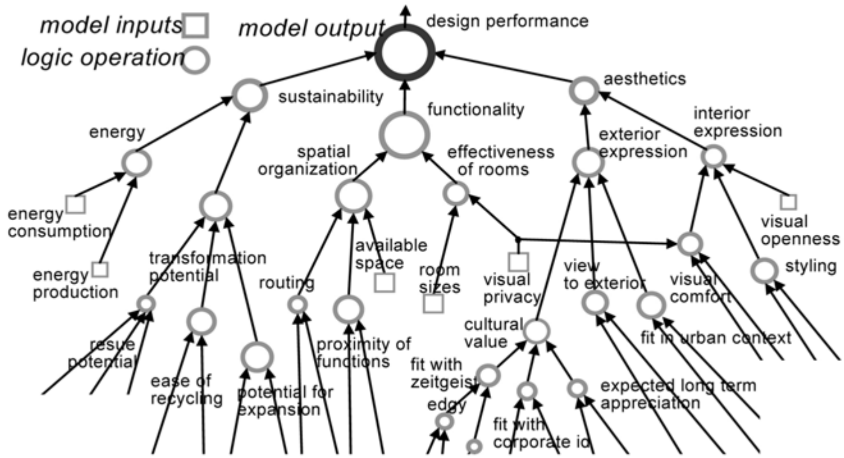
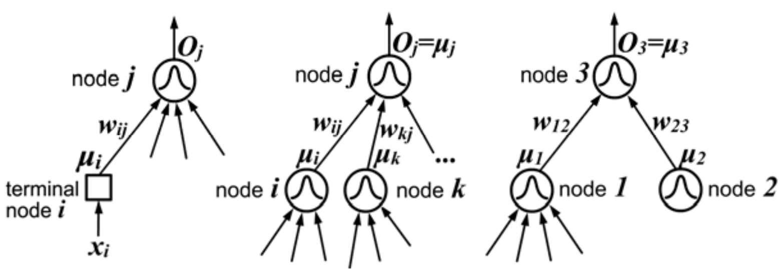


Fig. 2. The structure of a fuzzy neural tree model for performance evaluation

The neural tree is used for performance evaluation by structuring the relations among the aspects of performance. The root node takes the meaning of *high design performance* and the inner nodes one level below are the aspects of the performance. The meaning of each of these aspects may vary from design project to project and it is determined by experts. The model inputs are shown by means of squares in Figures 2 and 3, and they are fuzzy sets, such as those given in Figure 1.



(a) (b) (c)

Fig. 3. Different type of node connections in the neuro-fuzzy model in Figure 2

The detailed structure of the nodal connections with respect to the different connection types is shown in Figure 3, where the output of i-th node is denoted μ_i and it is introduced to another node j. The weights w_{ij} are given by domain experts, expressing the relative significance of the node i as a component of node j.

The centres of the basis functions are set to be the same as the weights of the connections arriving at that node. Therefore, for a *terminal node* connected to an *inner node*, the inner node output denoted by O_j, is obtained by [13].

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

where j is the number of the node; i denotes consecutive numbers associated to each input of the inner node; n denotes the highest number of the inputs arriving at node j; w_i denotes the degree of membership being the output of the *i-th* terminal node; w_{ij} is the weight associated with the connection between the i-th terminal node and the inner node j; and σ_j denotes the width of the Gaussian of node j.

It is noted that the inputs to an inner node are *fuzzified* before the AND operation takes place [11]. This is shown in Figure 4a. It is also noted that the model requires establishing the width parameter σ_j at every node. This is accomplished by means of imposing a consistency condition on the model [13].

This condition is to ensure that when all inputs take a certain value, then the model output yields this very same value, i.e. $\mu_1=\mu_2\approx O_j$ This is illustrated in Figure 4b by means of linear approximation to the Gaussian. The consistency is ensured by means of gradient adaptive optimization, identifying optimal σ_j values for each node.

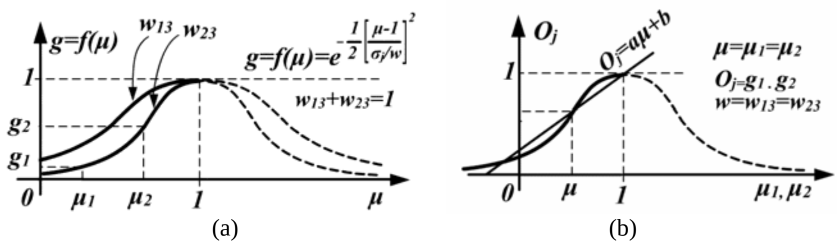


Fig. 4. Fuzzification of an input at an inner node (a); linear approximation to Gaussian function at AND operation (b)

It is emphasized that the fuzzy logic operation performed at each node is an AND operation among the input components μ_i coming to the node. This entails for instance that in case all elemental requirements are highly fulfilled, then the design performance is high as well. In the same way, for any other pattern of satisfaction on the elemental level, the performance is computed and obtained at the root node output. The fuzzy neural tree can be seen as a means to aggregate elemental requirements yielding fewer requirement items at higher levels of generalization compared to the lower level requirements. This is seen from Figure 5.

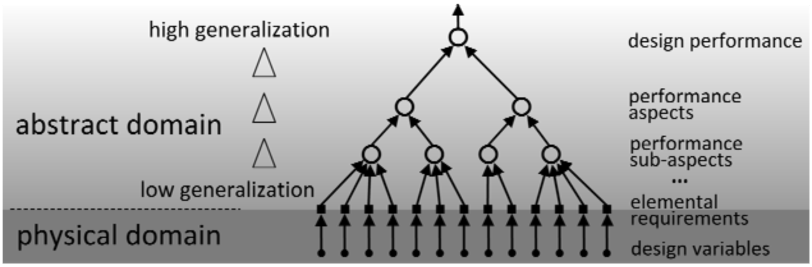


Fig. 5. Degrees of generalization in the neuro-fuzzy performance evaluation

Multi-objective Evolutionary Search with a Relaxed Dominance Concept

In design generally multiple objectives are subject to simultaneous satisfaction. Such objectives are for example high functionality and low cost. To deal with multi-objectivity, evolutionary algorithms with genetic operators are effective in defining the search direction for rapid and effective convergence [14]. 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 provoke the emergence of candidate solutions during the search process that are more suitable 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 effectiveness is clearly demonstrated for a moderate number of objectives, which are subject to optimization simultaneously [15]. Pareto ranking refers to comparing solutions of a population regarding their degree of being non-dominated by other solutions in the population. The evolutionary search using Pareto ranking converges to a set of solutions that lie on a surface in the multidimensional objective space. On this surface, the solutions are termed Pareto optimal solutions. They are different, both in terms of the solution parameters and their associated features. But they are assumed to be equivalently valid as there are no other solutions outperforming them for every objective dimension at the same time, i.e. the solutions on the Pareto surface are all *non-dominated*. Selection of one of the solutions among these is based on some higher-order preferences, which require further 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 consciously. The above are 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 an 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

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

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.

Applying the Pareto concept in its conventional strict form has drawbacks when the problem involves more than four or five objectives. The cause of this issue is the *greediness* of the conventional Pareto ranking. Namely with many objectives most solutions of the population will be considered non-dominated, although the search process is still at a premature stage. This means the search has little information to distinguish among solutions, so that the selection pressure pushing the population into the desirable region is too low. This means the algorithm prematurely eliminates potential solutions from the population, exhausting the 'creative' potential inherent to the population. As a result the search arrives at an inferior Pareto front, and with aggregation of solutions along this front. This is a well known topical issue in the area of evolutionary multi-objective optimization [10, 16, 17]. For the *greedy* application of the MO algorithm, one uses the orthogonal contour lines at the point ***P*** as shown in Figure 6. 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 6.

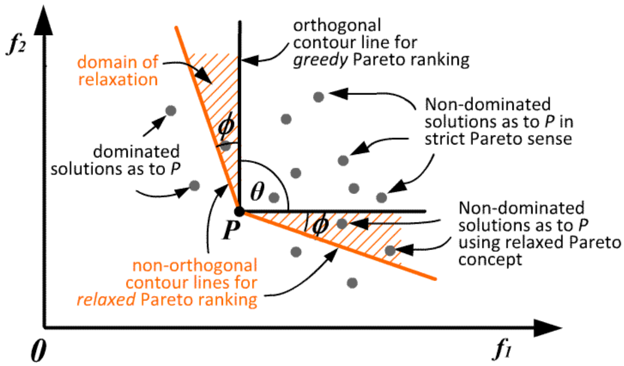


Fig. 6. Contour lines defining the search areas

The wider the angle beyond π/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 6. In the greedy case the solutions are expected to be more effective but to be 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

$$\boldsymbol{R}_{fit} = \frac{1}{N(\theta)+n} \qquad (3)$$

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 through experiments.

To obtain n in Eq. (3), for each solution point - say $\boldsymbol{P}$ in Figure 6 - 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 coordinates. This is accomplished through a matrix operation [11]. The importance of this 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 the approach through the coordinate transformation is a systematic and elegant approach. In this way the bottleneck of conventional Pareto ranking dealing with many objectives is alleviated to some extend, so that the evolutionary paradigm becomes more apt for applications in architectural design usually containing a great many requirements. It is noted that the relaxed Pareto approach requires significantly less computations to rank the population compared to the existing strategies aiming to avoid aggregation on the Pareto front, such as *niched* Pareto ranking [17].

Multi-objective Evolutionary Algorithm & Fuzzy Neural Tree = A Computational Design System with Cognitive Features

Next to relaxing the greediness of multi-objective evolutionary algorithm (MOEA), a second measure that alleviates the limitations of evolutionary search for design applications is to couple the algorithm with the fuzzy model in section 2. This yields the system shown in Figure 7. In the system the fuzzy model is the playing the role of fitness function in the MOEA. When we first consider a design containing many elemental requirements, on can consider that within the fuzzy model the amount of objectives is reduced by aggregating some requirements forming fewer, more complex concepts, which become the objectives subject to simultaneous satisfaction. Examples of such complex objectives are *functionality* and *sustainability*. Using the fuzzy information processing in this way entails that the search process makes use of human-like reasoning during its strive for optimality.

From Figure 7 we note that the computational design system starts its processing by generating a population of random solutions within the boundaries put forward by the designer in advance. Then several properties of these solutions are measured, such as sizes, distances, and perceptual properties. These are interpreted with respect to the elemental design requirements at the input layer of a fuzzy neural tree. This information is propagated through the tree yielding the degree of satisfaction of the solution at the penultimate level right below the root node. That is the evaluation using the neural tree is able to express the features of a solution in a few abstract, linguistic terms. For example it provides the performance regarding functionality, perception and cost effectiveness. These outputs are then used to compare the randomly generated solutions regarding their respective non-dominance using the relaxed Pareto concept.

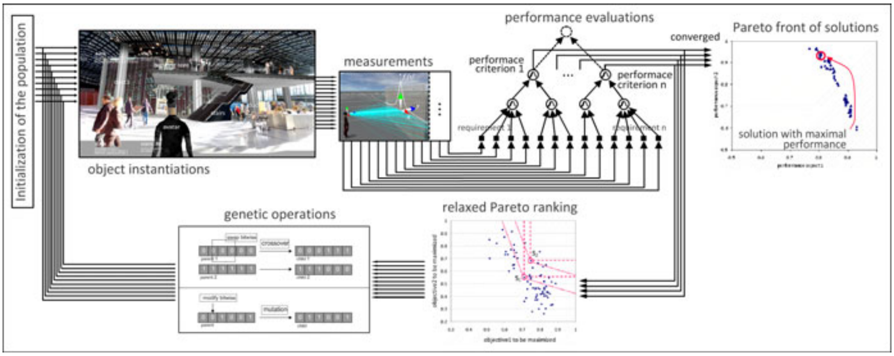


Fig. 7. Cognitive system based on MOEA and fuzzy neural tree

Relatively non-dominated solutions are then favoured for reproduction and the genetic operations, so that the next generation is more likely to contain non-dominated solutions. This generation-instantiation-evaluation-loop is executed for a number of generations, finally resulting in a set of Pareto optimal solutions. A designer or decision-maker is then able to compare these solutions in order to select his favourite design among the apparently equally valid solutions. In case the favourite solution is completely satisfying the designer's preferences, the design solution is found. Otherwise the designer may change the criteria of the computational design process and re-run the algorithm. This process iterates as shown in Figure 8, where the box containing the term *IDO (Intelligent Design Objects)* represents the system shown in Figure 7.

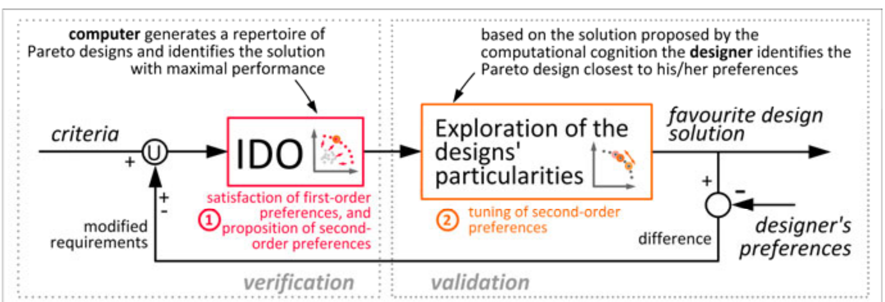


Fig. 8. Cognitive design approach

The Cognitive Features of the System

In contrast to conventional multi-objective optimization, due to the special fitness evaluation in this work involving a fuzzy model, the solutions on the Pareto front are not completely equivalent despite being all non-dominated. They may be distinguished as follows.

At the root node of the neural tree, the performance score is computed by the de-fuzzification process given by

$$w_1(1)\,f_1 + w_2(1)\,f_2 + ... + w_n(1)\,f_n = p\,, \tag{4}$$

where $w_1+w_2+\ldots+w_n=1$; f_1-f_n are the outputs at the penultimate nodes; p is the design performance, which is requested to be maximum. The vector $\boldsymbol{w}$ containing the weights on the penultimate level is termed priority vector. The node outputs f_1 - f_n can be considered as the *design feature vector* $\boldsymbol{f}$. It is important to note that a certain feature vector $\boldsymbol{f}$ will yield the greatest performance p at the model output if the weights w_1-w_n define the same direction as that of the feature vector. In order to consistently compare among different solutions' feature vectors with their different magnitudes, it is necessary to obtain the unit vector $\boldsymbol{u}$ along every feature vector. This unit vector is given by

$$u_1 = \frac{f_1}{\sqrt{f_1^2 + f_2^2 + ... + f_n^2}}\;;\quad u_2 = \frac{f_2}{\sqrt{f_1^2 + f_2^2 + ... + f_n^2}}\;;\;\; ...\;;\;\; u_n = \frac{f_n}{\sqrt{f_1^2 + f_2^2 + ... + f_n^2}} \tag{5}$$

In order to meet the condition imposed by the de-fuzzification process in Eq. (4), namely that the weight components sum up to unity, it is necessary to normalize the components u_1, …., u_n of the unit vector. Explicitly this is given by

$$\begin{gathered} u'_1 = \frac{u_1}{u_1 + u_2 ... + u_n} = \frac{\dfrac{f_1}{\sqrt{f_1^2 + f_2^2 + ... + f_n^2}}}{\dfrac{f_1 + f_2 + ... f_n}{\sqrt{f_1^2 + f_2^2 + ... + f_n^2}}} = \frac{f_1}{f_1 + f_2 + ... + f_n}; \\ ... \\ u'_n = \frac{u_n}{u_1 + u_2 ... + u_n} = \frac{\dfrac{f_n}{\sqrt{f_1^2 + f_2^2 + ... + f_n^2}}}{\dfrac{f_1 + f_2 + ... f_n}{\sqrt{f_1^2 + f_2^2 + ... + f_n^2}}} = \frac{f_n}{f_1 + f_2 + ... + f_n}\,. \end{gathered} \tag{6}$$

Equating the normalized components to the components of a priority vector $\boldsymbol{w}_{max}$ yields

$$w_{1,\max} = \frac{f_1}{f_1 + f_2 + ... + f_n}\;;\quad w_{2,\max} = \frac{f_2}{f_1 + f_2 + ... + f_n}\;;\quad ...\;; w_{3,\max} = \frac{f_n}{f_1 + f_2 + ... + f_n} \tag{7}$$

The steps in Eq. 5-7 are illustrated in Figure 9 for a case with two objectives. Due to Eq. 7 the performance given by Eq. 4 becomes the maximal performance p_{max}

$$p_{\max} = \frac{f_1^2 + f_2^2 + ... + f_n^2}{f_1 + f_2 + ... + f_n} \tag{8}$$

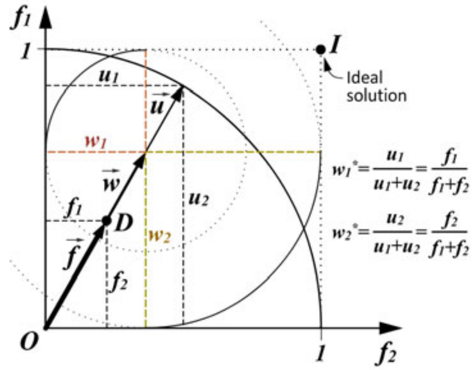


Fig. 9. Feature vector ***f***, unit vector ***u***, and priority vector ***w*** in objective space

Every solution on the Pareto front has an associated p_{max} value that characterizes it. This value gives the maximum design performance the solution attains, when there is no a-priori preference regarding the objectives. The solutions can be compared regarding their p_{max} value, and the solution with the highest value is a preferable choice among the Pareto solutions. This solution has a characteristic priority vector $\boldsymbol{w}^*$. It is noted that the computation used to identify p_{max} is elegant in the sense that it does not require, for instance, solving a linear programming problem to maximize p for a given feature vector $\boldsymbol{f}$.

This vector $\boldsymbol{w}^*$ implies that the computer advises the decision maker which goal he should take as more or less important in the present task, while this information was not known prior to the search process. This means the machine performs an act beyond mere optimization through intelligent information processing. Namely it is an act of cognition, yielding information about second-order aspects that were not included in the criteria given by the human decision maker. The artificial cognition alleviates decision-making in the sense that the designer need not explore the entire Pareto front, but has information on proficient areas along the front.

Let us consider the possibility that we obtain solutions having identical p_{max} values while their components are different. For example a solution with $\boldsymbol{f}$=(0.9, 0.1, 0.1) yields the same p_{max} value as a second solution (0.1, 0.1, 0.9). For clarifying this issue it is noted that Eq. (8) can be transformed in such a way that it describes a spherical surface in the multidimensional objective space, on the surface of which p_{max} is constant. In a two objective case the sphere becomes a circle given by [18]

$$f_1^2 + f_2^2 - pf_1 + pf_2 \equiv (x - x_1)^2 + (y - y_1)^2 - R^2 \text{, where} \quad (9)$$

$$
\begin{aligned}
x_1 &= p/2 \\
y_1 &= p/2 \\
R &= p/\sqrt{2}
\end{aligned}
\tag{10}
$$

The circle of constant maximal performance is shown in Figure 10. It is noted that due to the Pareto dominance criterion a Pareto front has a flatter curvature compared to the circle of constant maximal performance. Therefore the greatest p_{max} values are generally to be found at the extremities of the front, i.e. close to the axes. From Figure 10b is noted that for Pareto fronts that are asymmetrically shaped w.r.t. the line passing through the origin and (x_1,y_1) *where x1=y1* merely a single maximal p_{max} value may exist in the population. In the latter case provision of this solution clearly alleviates the decision making. From figure 10a we note that, tn case of symmetry of the front providing the p_{max} value to a decision maker alleviates the decision making as the two solutions will be very different regarding their respective features, so that the decision may be easier to make compared to the case p_{max} were unknown.

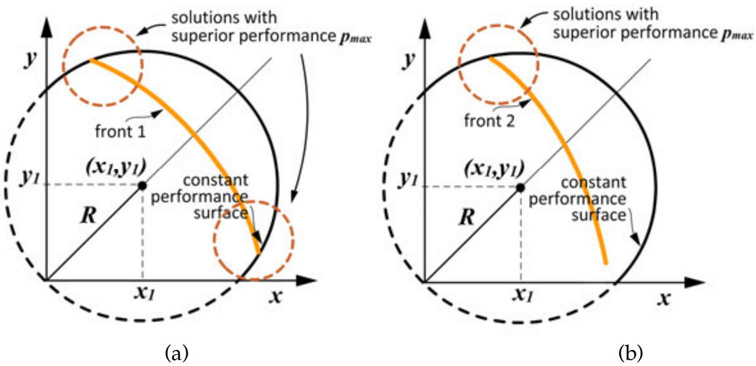
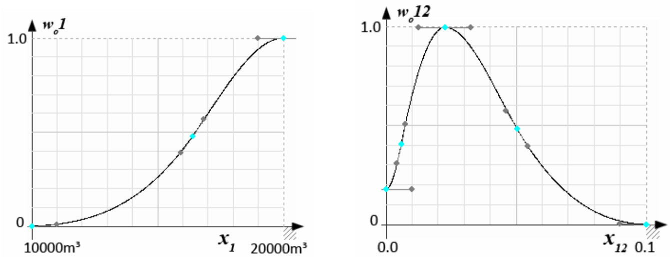


Fig. 10. Constant performance surface and Pareto front

It is also noted that a decision maker's preferences are also important with respect to determining to what extend an extremity is to be exercised.

Application

The design task concerns the design of an interior space. The space is based on the main hall of the World Trade Centre in Rotterdam in the Netherlands. The perception of a virtual observer plays a role in the design, as the task involves a number of perception-based requirements. An example is that the stairs should not be very noticeable from the entrance of the space, as seen from Figure 11(b). The perception computation yielding x_{12} in Figure 11(b) is accomplished using a probabilistic perception model [19]. Another example is that the building core should be positioned in such a way that the entrance hall is spacious, while the elevators should be easily perceived at the same time. This is seen from figure 11a.



(a) (b)

Fig. 11. Two requirements subject to satisfaction, concerning (a) spaciousness of the entrance hall; (b) perception of the stairs

The task is to optimally place the design objects satisfying a number of perception and functionality requirements. The objects are a vertical building core hosting the elevators, a mezzanine, stairs, and two vertical ducts. The goals are maximizing the performance of every design object forming the scene, as seen from the fuzzy neural tree structure in Figure 12. From the structure of the model it is seen that the amount of objectives to be maximized is four, namely the outputs of nodes 4-7, whereas the elemental requirements total an amount of 12.

The resulting Pareto optimal solutions are shown in Figures 13, and 14. Figure 13 shows the results using the greedy Pareto ranking approach, whereas Figure 14 shows the relaxed Pareto ranking approach. It is noted that the objective space has four dimensions, one for the performance of every design object. The representation is obtained by first categorizing the solutions as to which of the four quadrants in the two-dimensional objective space formed by the building core and mezzanine performance they belong, and then representing in each quadrant a coordinate system showing the stairs and ducts performance in this very quadrant. This way four dimensions are represented on the two-dimensional page. Comparing Figure 13 with Figure 14 we note that the relaxed approach is superior compared to the greedy case in Figure 13, as it yields a Pareto front having solutions with a higher performance p_{max}.

Two Pareto optimal designs are shown in Figure 15 and 16 for comparison. The maximal performance score as well as the performance feature vector for these solutions is shown in Table 1. From the table it is seen that design D2 outperforms design D4 with respect to the maximal performance p_{max} obtained using Eq. 8. It is also noted that the performance of D4 as to its features varies less compared to D2. The fact that D2 has a greater p_{max} confirms the theoretical expectation illustrated by Figure 10 that solutions with more extreme features generally have a greater maximal performance compared to solutions with little extremity.

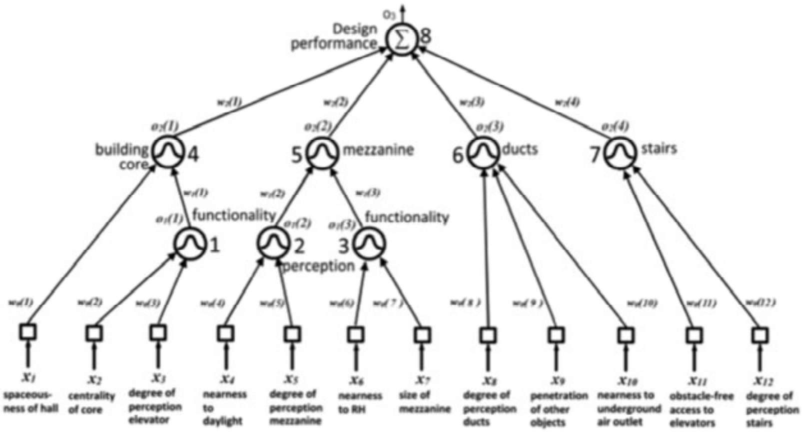


Fig. 12. Neural tree structure for the performance evaluation

The greatest absolute difference among D2 and D4 is the performance of the mezzanine. In D2 the mezzanine is located closer to associated functions, and this turns out to be more important compared to the fact that D4 yields more daylight on the mezzanine. Therefore D2 scores higher that D4 regarding the mezzanine. Additionally D2 slightly outperforms D4 regarding the performance of the ducts. This is because the ducts do not penetrate the mezzanine in D2, whereas in D4 they do. The latter is undesirable, as given by the requirements in D4. Regarding the building core D2 is inferior to D4, which is because the spaciousness in D4 is greater and also the elevators are located more centrally. Regarding the stairs' performance, the difference among D2 and D4 is negligible.

The latter exemplifies the fact that an objective may be reached in different ways, i.e. solutions that are quite different regarding their physical parameters may yield similar scores as to a certain goal. In the present case the greater distance to the stairs in D2 compared to D4 is compensated by the fact that the stairs is oriented sideways in D2, so that the final perception degree is almost the same. It is noted that D2 is the solution with the greatest maximal performance p_{max}, so that from an unbiased viewpoint it is the most suitable solution among the Pareto optimal ones. This solution is most appealing to be selected for construction.

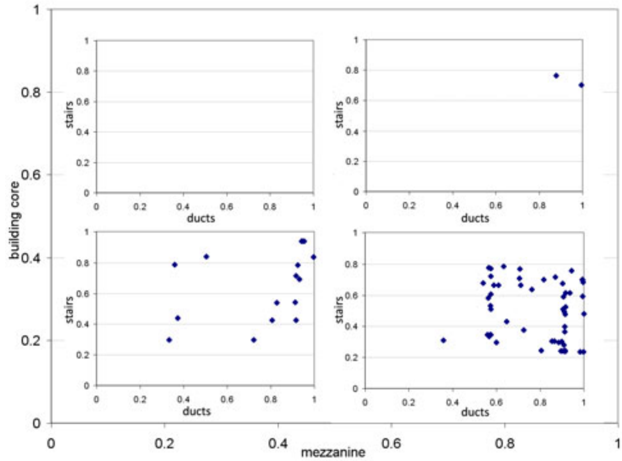


Fig. 13. Pareto optimal designs with respect to the four objective dimensions using greedy Pareto ranking

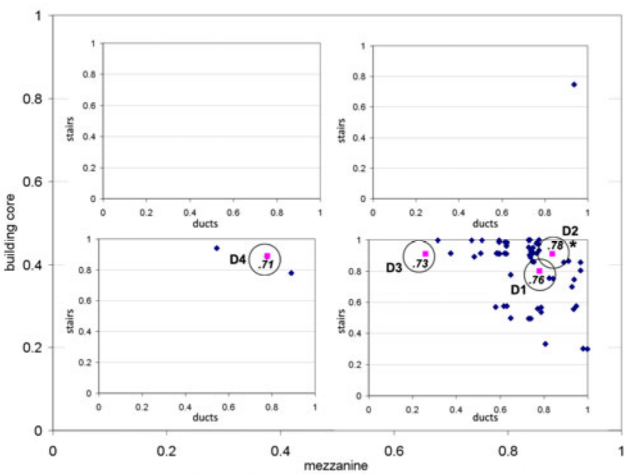


Fig. 14. Pareto optimal designs with respect to the four objective dimensions using relaxed Pareto ranking

This result is an act of machine cognition, as it reveals that pursuing maximal performance in the present task the stairs and ducts are more important compared to the building core from an unbiased viewpoint. This information was not known prior to the execution of the computational design process. It is interesting to note that the solution that was chosen by a human architect in a conventional design process without computational support was also similar to solution 2.

Table 1 Performance of design D2 versus D4

	core	**mezzanine**	**ducts**	**stairs**	p_{max}
D2	0.27	0.73	0.83	0.93	0.78
D4	0.48	0.49	0.78	0.89	0.71

This indicates that the architect must have had similar requirements in mind, and followed a similar reasoning as involved in the computational process. The benefit of the computational approach is that it ensures identification of most suitable solutions, their unbiased comparison, and precise information on their respective trade-off as to the abstract objectives. This is difficult to obtain using conventional means.

Fig. 15. Pareto-optimal design *D2* in Figure 14

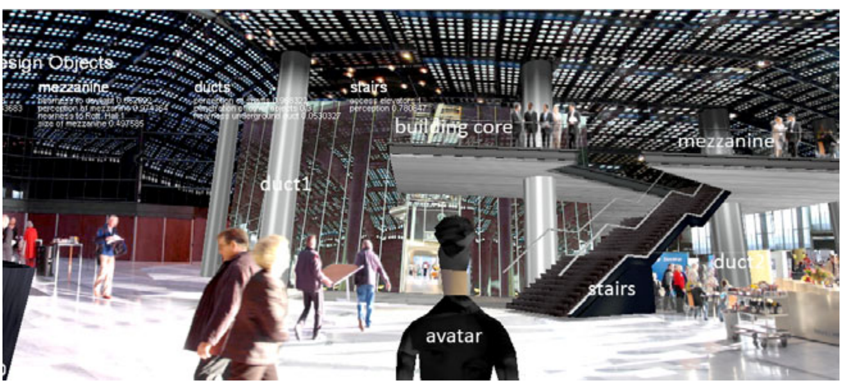


Fig. 16. Pareto-optimal design *D4* in Figure 14

Conclusion

A novel computational system for architectural design is presented. It generates designs that satisfy multiple criteria put forward by decision makers, such as architects.

The contribution of the present work is twofold. First, the use of the fuzzy information processing enables multi-objective evolutionary algorithm (MOEA) to deal with the many, soft requirements characterizing design tasks. Secondly it enables the machine cognition. Namely the fuzzy model entails that the objective space takes on a particular shape, being a unit hypercube. This way, different properties of a design are subject to consideration on a common ground, which allows distinction among the Pareto solutions regarding their unbiased maximal performance being an act of machine cognition. It is noted that cognition is understood in this work as the faculty to take a decision based on an awareness of, in some sense, equivalently valid solution alternatives fulfilling first-order objectives. The cognitive act is selection among the alternatives based on second-order preferences that were not included in the process yielding these solutions. Such second-order preferences are due to the particularities among the alternative solutions, which cannot be foreseen prior to designing.

Machine cognition is a feature of scientific and practical interest. From the scientific viewpoint the introduction of human-like reasoning into an evolutionary search process increases the applicability of evolutionary algorithms in particular for design problems. Thereby it may contribute to the growing use of MOEAs in this domain. From the practical viewpoint it is an innovative direction for alleviating decision making tasks. A designer gets to know - from an unbiased viewpoint - which trade-offs inherent to the design task are worthwhile to accept compared to others. The attribute *unbiased* refers that no objective is favored over another one a-priori. This way the decision maker has an indication which regions on the Pareto front are outstanding, and to what extend this is the case. This information is challenging to obtain when conventional means are used.

References

1. Eastman, C.M.: Automated space planning. Artificial Intelligence 4, 41–64 (1973)
2. Flemming, U., Woodbury, R.: Software environment to support early phases in building design (SEED): Overview. J. of Architectural Engineering 1, 147–152 (1995)

3. Koile, K.: An intelligent assistant for conceptual design. In: Gero, J.S. (ed.) Design Computing and Cognition 2004, pp. 3–22. Kluwer Academic Publishers, Massachusetts Institute of Technology, Boston, USA (2004)
4. Jang, J.S.R., Sun, C.T., Mizutani, E.: Neuro-Fuzzy and Soft Computing. Prentice Hall, Englewood Cliffs (1997)
5. Gero, J., Kazakov, V.: Evolving design genes in space layout problems. Artificial Intelligence in Engineering 12, 163–176 (1998)
6. Damsky, J., Gero, J.: An evolutionary approach to generating constraint-based space layout topologies. In: Junge, R. (ed.) CAAD Futures 1997, pp. 855–874. Kluwer Academic Publishing, Dordrecht (1997)
7. Jo, J., Gero, J.: Space layout planning using an evolutionary approach. Artificial Intelligence in Engineering 12, 163–176 (1998)
8. Mawdesley, M.J., Al-jibouri, S.H., Yang, H.: Genetic Algorithms for Construction Site Layout in Project Planning. J. Constr. Engrg. and Mgmt. 128, 418–426 (2002)
9. Caldas, L.: GENE_ARCH: An evolution-based generative design system for sustainable architecture. In: Smith, I.F.C. (ed.) EG-ICE 2006. LNCS (LNAI), vol. 4200, pp. 109–118. Springer, Heidelberg (2006)
10. Coello, C.A.C., Veldhuizen, D.A., Lamont, G.B.: Evolutionary Algorithms for Solving Multiobjective Problems. Kluwer Academic Publishers, Boston (2003)
11. Bittermann, M.S., Ciftcioglu, O.: A cognitive system based on fuzzy information processing and multi-objective evolutionary algorithm. In: IEEE Conference on Evolutionary Computation. IEEE, Trondheim (2009)
12. Zadeh, L.A.: Fuzzy logic, neural networks and soft computing. Communications of the ACM 37, 77–84 (1994)
13. Ciftcioglu, O., Bittermann, M.S., Sariyildiz, I.S.: Building performance analysis supported by GA. In: Tan, K.C., Xu, J.X. (eds.) 2007 IEEE Congress on Evolutionary Computation, pp. 489–495. IEEE, Singapore (2007)
14. Deb, K.: Multiobjective Optimization using Evolutionary Algorithms. John Wiley & Sons, Chichester (2001)
15. Tan, K.C., Xu, J.-X. (eds.): Proc. 2007 IEEE Congress on Evolutionary Computation. IEEE Cong., Singapore (2007)
16. Hughes, E.J.: Evolutionary many-objective optimisation: many once or one many? In: IEEE Congress on Evolutionary Computation CEC 2005, pp. 222–227. IEEE Service Center, Edinburgh (2005)
17. Horn, J., Nafploitis, N., Goldberg, D.E.: A niched Pareto genetic algorithm for multiobjective optimization. In: Michaelwicz, Z. (ed.) First IEEE Conf. on Evolutionary Computation, pp. 82–87. IEEE Press, Los Alamitos (1994)
18. Ciftcioglu, Ö., Bittermann, M.S.: Adaptive formation of Pareto front in evolutionary multi-objective optimization. In: Santos, W.P.d. (ed.), Evolutionary Computation. In-Tech, Vienna, pp. 417–444 (2009)
19. Bittermann, M.S., Sariyildiz, I.S., Ciftcioglu, Ö.: Visual perception in design and robotics. Integrated Computer-Aided Engineering 14, 73–91 (2007)