Artificial Intelligence (AI) versus Computational Intelligence (CI) for treatment of complexity in design

Michael S. Bittermann
Delft University of Technology, The Netherlands

The complexity of design tasks has a number of aspects. Three of them are the vagueness of objectives, conflicting nature of objectives, as well as the large amount of possible solutions. This paper considers two major approaches addressing treatment of these complexity aspects, namely approaches based on methods from the domain of classical artificial intelligence (AI) and approaches using methods from the emerging paradigm of computational intelligence (CI). Challenges of the methodologies in treating complexity issues are pinpointed, and alleviation is exemplified by means of an approach that is based on two CI methodologies addressing an architectural design problem.

Complexity of design

There are a number of issues characterizing the complexity of a design task. One issue is that generally the amount of possible solutions is excessive. This occurs due to a phenomenon known as combinatorial explosion. Design solutions can be regarded as particular combinations of values for the parameters of a solution. In architectural design e.g., the parameters may concern the sizes and proportions of a building, its rooms, the dimensions and kind of material of the building elements etc. Considering that for every parameter many values are feasible, it is clear that the amount of feasible solutions is enormous. Therefore a challenge for computational methods addressing design problems lies in identifying suitable solutions among the feasible ones. A second issue is the vagueness of objectives in design, which may be also referred to as their *soft* nature. For example a design is demanded to be functional, look appealing, or have moderate costs. This property makes it problematic to precisely compare alternative solutions during the design process. The vagueness also entails that the sat-

Design Computing and Cognition DCC'10. J.S Gero (ed), pp. xx-yy.

isfaction of a design goal is generally a matter of degree and not either total or not at all. It is noted that cause for this generally partial satisfaction of requirements is that design goals are in conflict with each other to some extend. For example, on one hand a design is demanded to be functional, on the other hand it is demanded to yield few cost. This is a third issue contributing to complexity, as conflicting objectives make it difficult to identify a unique solution direction.

To address these complexity issues several computational methodologies have been developed. The methodologies can be categorized with respect to the information processing paradigm they belong to, namely either the domain of classical artificial intelligence (AI) or the domain of computational intelligence (CI).

It is noted that approaches that are considered as *multi-agent systems* are to be classified into the category of either CI or classical AI depending on the essential computational machinery determining the behavior of the agents. For example *particle swarms* used in the particle swarm optimization (PSO) technique can be considered as a multi-agent system. They belong into the CI paradigm, as this will be explained below.

Approaches based on methods from classical artificial intelligence (AI)

A number of computational design approaches use methods of classical artificial intelligence, e.g. [1-3]. The methods are based a rigid inference mechanism. Namely, a deficiency in a design has a set of actions associated for improving the deficiency and this set is defined a-priori. Such systems are referred to as expert systems, as they generally contain expert knowledge in the form of rules to be executed [4]. 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. That is, the multi-objective nature of design problems severely challenges a purely rule based approach. Also AI based approaches are challenged when an action to take involves a number of parameters to decide upon in combination, e.g. determining the most suitable location to move an object blocking the view to. And finally the issue concerning the vagueness of requirements is not addressed in approaches based on classical AI. For example the detection of *low openness* is based on a sharp boundary value for openness, above which a space is considered open enough, and below which it is not. That is, the premise part of a rule is either fulfilled or not fulfilled, having only two possible degrees of truth associated with it, true or false. Such sharp boundaries imply an abrupt change in satisfaction with the gradual modification of a physical parameter. Although this might occasionally happen, it is rather unnatural in general. For example as openness increases, satisfaction of a demand for *high openness* should be considered to increase gradually as well, i.e. satisfaction should not suddenly appear or disappear. The simplification, to use sharp boundaries in the inference mechanism, implies that generally a significant amount of information is ignored and not available for use during a search for most suitable solutions, which hampers the effectiveness and robustness of the approach.

Approaches based on methods from computational intelligence (CI)

A recent information processing paradigm designed to handle complexity occurring in diverse application areas is computational intelligence (CI) [5]. CI methodologies are evolutionary computation (EC), fuzzy logic (FL), and artificial neural networks (ANN). Among the CI methodologies, in particular evolutionary computation became popular in the last decade for identifying optimal solutions for different aspects in design problems [6-10]. It is noted that EC is a bio-inspired search and optimization methodology based on multiple agents with stochastic behavior. Next to the well-known method of genetic algorithm, EC also includes methods like genetic programming, evolutionary programming, and cultural algorithms, as well as artificial immune systems. Also recent methods like particle swarm optimization (PSO) are considered as part of the EC paradigm, although sometimes they are referred to as belonging to a novel CI methodology termed as swarm intelligence (SI) [11]. A difference of evolutionary approaches compared to the classical AI approach lies in the fact that the former do not involve a-priori defined solution actions 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 the *rules* involved in the classic AI approach. Therefore EC is superior over the classical AI methodologies in particular addressing two of the complexity issues mentioned above, namely the conflicting nature of objectives, when multi-objective evolutionary algorithms are used, and the combinatorial explosion in the parameter domain. However, EC conventionally does not deal with requirements that are vague in nature. Therefore EC should be combined with other methodologies to deal with design objectives. Most prominent methodologies for this purpose are ANNs [12] and fuzzy logic systems [13]. ANNs are suitable to model the relation between object features as model input, and abstract design qualities as model output, based on available data sets containing both, input and output patterns, while the data may contain vagueness, imprecision and uncertainty. Establishing this relation in this way is referred to as supervised learning. Fuzzy systems, which are equivalent with ANNs under certain conditions [14], are also suitable for modeling this relation, with the difference that fuzzy systems generally retain transparency in the model and ANNs do not. This is because the base from which the fuzzy model is built is not data, but expert knowledge that already contains a semantic structure. That is, the meaning of the parameters constituting the model is known in case of FL, whereas ANNs generally are black-box type models. Transparency is a desirable feature, because it allows tractability of the reasoning, and it also permits to make use of expert knowledge that may be more readily available compared to large amount of suitable performance data. It is noted that with complexity the amount of training data samples increases drastically, and may become excessive in design problems with many parameters.

With this understanding it is emphasized that in order to let computation address design effectively, synergistic combinations of CI methods should be used. This is because they allow combining the strength of several CI methods, where each method takes care of a certain issue constituting the complexity of the problem. The CI methods mentioned above as well as the classical AI approach are compared in table 1 regarding their suitability to treat different complexity issues in design tasks.

Table 1 General suitability of computational methodologies for addressing complexity issues in design.

Complexity issue	**FL**	**ANN**	**EC**	**classical AI**
Vagueness of objectives	+	+	-	-
Transparency	+	-	-	+
Multi-objectivity	-	-	+	-
Combinatorial explosion in parameter domain	-	-	+	-

A synergistic system addressing the complexity issues

A computational system for design that is based on two CI methods is shown in figure 1 [15]. The methods are evolutionary computation, specifically multi-objective evolutionary algorithm [16], and a hybrid method combining features of ANNs and FL known as fuzzy neural tree [17]. This way the system addresses the whole range of complexity issues mentioned above. The 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 energy performance related properties, such as façade orientation and composition etc. These are interpreted with respect to abstract design objectives, such as functionality and energy performance, using the neuro-fuzzy model. This way the degree of satisfaction on the design objectives is obtained. These outputs are then used to compare the solutions regarding their suitability in a multi-objective sense. Relatively suitable solutions are then favoured for reproduction during 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. The result is a set of solutions, where each solution satisfies the design objectives equivalently in a multi-objective sense. This permits a decision maker to compare the solutions with respect to his second-order preferences, so that he takes his decision with great awareness of its implications.

As an example, two optimal designs that are obtained during a single execution of the system are shown in figure 2 [18]. It is noted that the designs

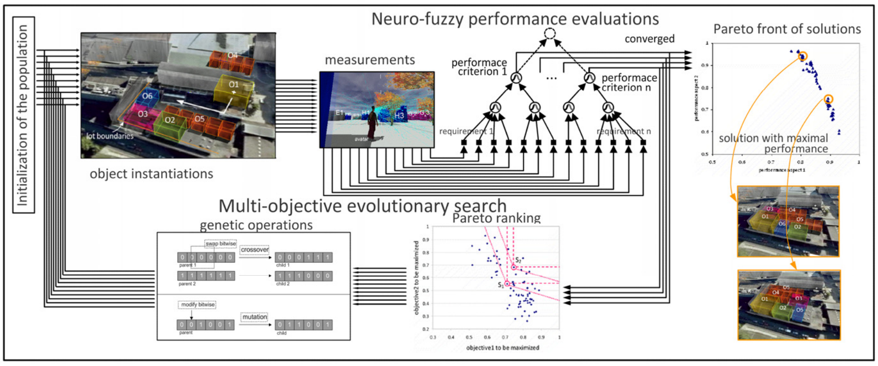

Fig. 1. A computational design system based on evolutionary computation and neuro-fuzzy modelling

are equivalent in multi-objective sense with different trade-off as to the goals. The goals are maximizing functionality, form and energy performance, and the task is layout of a building complex. From the example it is noted that addressing this task using classical AI methods this would be problematic. Namely it is formidably difficult to pinpoint a-priori unique rules, the execution of which would ensure that the design objectives will be reached. This is due to the involvement of multi-objectivity, vagueness of objectives and the large amount of possible solutions in the task.

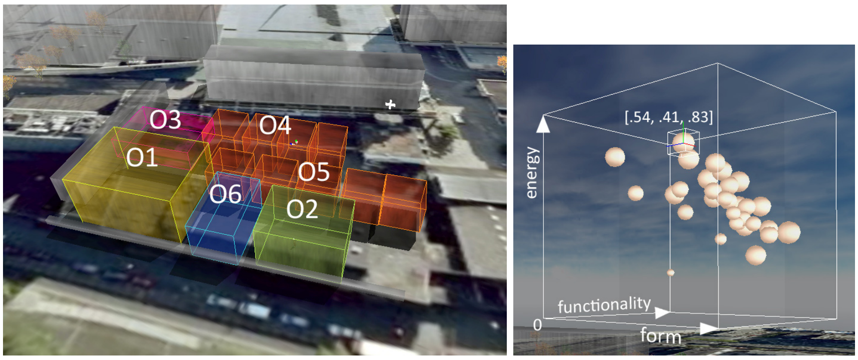

Fig. 2. Optimal design with a high energy performance (.83 of 1.0); in parameter space (a); the same solution in objective space (b) among the other optimal ones.

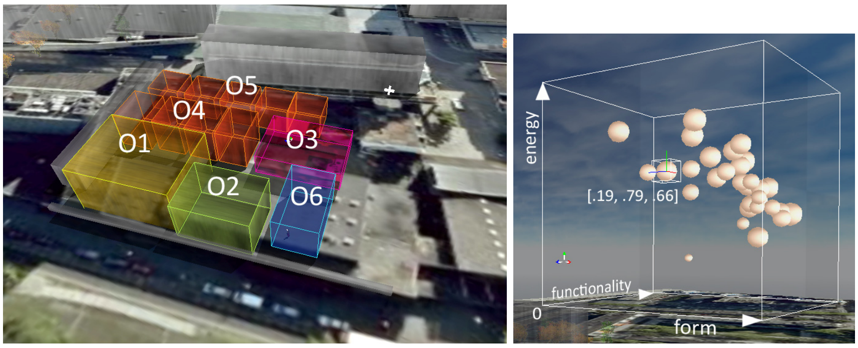

Fig. 3. Optimal design with a high functionality performance (.79 of 1.0); in parameter space (a); the same solution in objective space (b).

Conclusion

Approaches based on Classical AI are inferior compared to approaches based on CI regarding the treatment of most complexity issues in design. In particular this concerns dealing with vagueness, multi-objectivity and large amount of possible solutions. Therefore, application of classical AI is limited to problems that minimally involve these issues. As design tasks are generally characterized by these issues, application of the classical AI approach for such tasks is questionable in general.

References

1. Eastman, C.M.: Automated space planning. Artificial Intelligence **4** (1973) 41-64
2. Flemming, U., Woodbury, R.: Software environment to support early phases in building design (SEED): Overview. Journal of Architectural Engineering **1** (1995) 147-152
3. Koile, K.: An intelligent assistant for conceptual design. In: Gero, J.S. (ed.): Design Computing and Cognition'04. Kluwer Academic Publishers, Massachusetts Institute of Technology, Boston, USA (2004) 3-22
4. Kasabov, N.K.: Foundadtions of Neural Networks, Fuzzy Systems, and Knowledge Engineering. MIT Press, Cambridge (1996)
5. Jang, J.S.R., Sun, C.T., Mizutani, E.: Neuro-Fuzzy and Soft Computing. Prentice Hall (1997)
6. Gero, J., Kazakov, V.: Evoloving design genes in space layout problems. Artificial Intelligence in Engineering **12** (1998) 163-176
7. Damsky, J., Gero, J.: An evolutionary approach to generating constraint-based space layout topologies. In: Junge, R. (ed.): CAAD Futures 1997. Kluwer Academic Publishing (1997) 855-874
8. Jo, J., Gero, J.: Space layout planning using an evolutionary approach. Artificial Intelligence in Engineering **12** (1998) 163-176
9. Caldas, L.: GENE_ARCH: An evolution-based generative design system for sustainable architecture. In: Smith, I.F.C. (ed.): Lecture Notes in Artificial Intelligence Springer-Verlag Berlin Heidelberg (2006) 109-118
10. Bandaru, S., Deb, K.: Automated discovery of vital design knowledge from Pareto-optimal solutions: first results from engineering design. IEEE Congress on Evolutionary Computation - CEC2010, Barcelona, Spain (2010)
11. Engelbrecht, A.P.: Computational Swarm Intelligence. Wiley, Chichester, GB (2005)
12. Haykin, S.: Neural Networks: A Comprehensive Foundation. Prentice Hall (1998)
13. Zadeh, L.A.: Fuzzy logic, neural networks and soft computing. Communications of the ACM **37** (1994) 77-84

14. Hunt, K.J., Haas, R., Murray-Smith, R.: Extending the functional equivalence of radial basis function networks and fuzzy inference systems. IEEE Trans. Neural Networks **7** (1996)
15. Bittermann, M.S.: Intelligent Design Objects (IDO) - A cognitive approach for performance-based design. Department of Building Technology, Vol. PhD. Delft University of Technology, Delft, The Netherlands (2009) 235
16. Coello, C.A.C., Veldhuizen, D.A., Lamont, G.B.: Evolutionary Algorithms for Solving Multiobjective Problems. Kluwer Academic Publishers, Boston (2003)
17. Ciftcioglu, Ö., Bittermann, M.S., Sariyildiz, I.S.: A neural fuzzy system for soft computing. NAFIPS 2007. IEEE, San Diego, USA (2007) 489-495
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 (2009) 417-444