Computer-Aided Creativity and Learning in Distributed Cooperative Human-Machine Networks

Theodore ZAMENOPOULOS and Katerina ALEXIOU
Centre for Advanced Spatial Analysis, Bartlett School of Graduate Studies, UCL, UK

Keywords: collaboration, creativity, e-learning, human-machine networks

Abstract: In this paper we discuss designing abilities, such as creativity and learning, as abilities that emerge though interaction in cooperative human-machine networks. We concentrate in a design system that can exhibit and support creative behaviour using knowledge learnt through distributed human-machine interaction. In this context, conflict resolution and coordination is a main issue, as well as a main indicator for the creative and adaptive ability of the design system. More specifically, we are going to present a model of coordination developed using learning control and multi-agent systems methodologies and techniques. A prototype system is tested in a virtual collaborative design assignment for simple location and three-dimensional configuration problems.

1 INTRODUCTION

The paradox in developing CAAD systems for creativity support is that any attempt to support creative design will lead (implicitly or explicitly) to a new definition and understanding of creative design. In effect, how can we define the meaning of computer-aided creative design? In order to briefly outline the different meanings and paradigms of creativity support we discuss CAAD systems according to the question of 'where creativity is' as opposed to 'what is creativity' for these systems (Csikszentmihalyi 1988). Most CAAD systems developed are based on the assumption that creativity is a purely individual cognitive ability of human designers. It follows, that creativity might be supported or enhanced if computer systems facilitate design tasks such as combinatorial or exploratory tasks (Segers et al. 2001); enable the externalisation and visual representation of ideas (Candy 1997); or stimulate cognitive aspects of creativity such as emergence (Reffat and Gero 2000). In some cases computer systems are developed so as to exhibit creative behaviour themselves (rather than to support it) with or without interaction with human designers (Rosenman and Gero 1999). In this context, the motivation is to improve the creative abilities of computer systems and thereafter challenge the creative ability of designers.

However, what if the development of CAAD systems put the emphasis on creativity built on the interaction between human designers and artificial tools, as opposed to creativity built inside the human or artificial mind? In the following, we embrace the view that creativity may arise not only because designers can be creative, neither because computers can support it effectively (being creative themselves or not) -but because humans and computational constructs interacting with each other form a complex whole that demonstrates creative behaviour as such. A similar view is discussed in Fischer (1999) and it is also considered relevant to research on situated cognition and constructive memory (Gero 1999). The motivation behind this approach is the development of more open, evolutionary and scalable systems.

The objective of this research is to investigate efficient conditions that will enable generative and creative behaviour to emerge through human-computer network interaction. We argue that in order to enable such abilities to emerge we need to devise design systems that are able to interact with humans, learn, and adapt their behaviour according to this interaction, so as to support the constructive generation of coordinated solutions. To this end, we present a model of coordination developed using learning control and multi-agent systems methodologies and techniques. A prototype system is tested for a virtual collaborative design assignment considering simple location and three-dimensional configuration problems.

2 ASSUMPTIONS AND HYPOTHESES FOR A CAAD SYSTEM

CAAD research is typically based on three classes of studies: the study of the design artefact and its representation, the study of the design processes and methods, and the study of the design knowledge and its structure. In order to delineate the core hypotheses for the development of the proposed prototype system, we need to have an adequate perception about the meaning and scope of designs generated in human-computer design networks, but we also need to outline some of our critical assumptions regarding design knowledge and processes.

2.1 Designs as Plans

Before anything else, CAAD systems deal with issues related with the formation, retrieval and representation of information regarding decisions to be implemented (solution space), performance attributes and expectations (performance space), requirements and constrains (problem space), as well as interdependencies between these spaces, usually in a time-dependant manner. In a typical architectural design problem, and if we restrict ourselves to the preliminary design stages, *designs* are specifications of a spatial configuration such as room layout and volume composition. Designing usually starts from an ‘idea’ regarding the expected building, and with the aim to satisfy conflicting requirements and goals associated with this idea. In this sense, designs might be seen as constructs that address a current design situation based on expectations and ideas developed from previous knowledge, but activated and constructed according to the current perception of the past. Based on this assumption, designing is seen as a situated act based on constructive memory (Gero 1999), and designs might be seen as memories (e.g. solutions) coming from previous experiences 'reinterpreted' and reconstructed in/for the current design context.

We wish to discuss another important dimension here, what we could call the 'memory of the future', and which relates to a view of architectural designing as a purposeful constructive process. When an architect comes up with a first abstract idea, let's say of a shopping mall cut in two by a semi-public axis, this is not only an act that 'reinterprets' the past (and in parallel rebuilds the memory of the system), but it is also an action that implicitly or explicitly constructs future design opportunities. It is an action that suggests a future linear organisation of stores in connection with the proposed axis. This decision guides the search processes in the next step as in recalling in a sense, future configurations. Hence, constructive memory is not only a 'reinterpretation of the past through the lens of the present' (Gero 1999) but also an interpretation of the future for that matter. In this sense we will consider designs as plans, and hence designing as an action of forming plans.

2.2 Designing as a Distributed Process

The development of the proposed system starts with the assumption that *design knowledge and processes are distributed* in human-computer networks. This statement implies that the generation of an idea, concept, design action, or design entity is an emergent property of the interaction that evolves between human and computer systems. In one sense this assumption is inspired by (human or artificial) connectionist systems where the emergence of a concept, action or entity is the result of the interaction of (non-symbolic) units. Early research on connectionist systems in design applications indicates that these systems appear to be able to synthesize and innovate (Coyne 1990). However, this can be extended to see creative designing within the context of complex socio-technical systems. As Fischer puts it (Fischer 1999), 'Distributed cognition emphasizes that that the heart of intelligent human performance is not the individual human mind but groups of minds in interaction with each other and minds in interactions with tools and artefacts'.

A wide range of computer applications for creative design is based on the hypothesis that creativity might be supported or enhanced in the form of information that stimulates the generation of new associations; as 'architects can find ideas anywhere' (Segers et al. 2001). These statements recognize the fact that in building CAAD systems we have to cope with (or exploit) the fact that knowledge is distributed- not only because designs (or plans) are formed collectively by different experts or interdisciplinary groups, but also because even expert reasoning involves employing diverse roles and models of action (see also Goldschmidt 1995).

The implications of the above assumption are the following: First, if design knowledge and processes are distributed, then designing must involve some kind of knowledge construction. In this sense, learning becomes a desirable function that represents the ability of agents, human or artificial, to construct knowledge about their environment, and reflect on their own actions and actions of other agents. This also implies that there is no clear division of labour between humans and machines, as the final artefact (plan) is the product of a process of interaction, adaptation and mutual learning. Second, if design knowledge and processes are distributed, then the targets of design (problem space) cannot be defined a-priori but need to be reconstructed as new solutions are produced by the interaction of distributed agents. This implies that problem and solution spaces need to co-evolve in time (Maher 2000). Third, if design knowledge and processes are distributed, then what makes the overall system -for an external observer- to work as a group of interacting components that jointly achieve an added value, 'is a processes of permanent mutual adaptation' (Ossowski 1999). This reflects the ability of the distributed system to *coordinate* its activities. Designing becomes a reflective process of learning and teaching, adaptation and guidance, so that individual and global goals may be satisfied. The action of distributed agents is built not according to a central control mechanism but through partial/distributed control over the overall plan.

2.3 Designing as Distributed Learning Control

Following our two main assumptions that designs represent a constructive process, and that design knowledge and processes are distributed, we are led here to the proposition that (plan) designing can be seen as a coordination problem between distributed agents and addressed via *distributed learning control*. According to this view, agents act as 'controllers' that aim to generate and reconstruct problem and solution formulations so as to satisfy time-variant targets, performance constraints and expectations - despite endogenous uncertainty and exogenous disturbances produced by other agents. In this framework, learning has a dual function and meaning: first, to capture and restructure interdependencies between the problem, solution and performance spaces so as to improve agent understanding about the domain problem and reduce uncertainty and second, to improve the generative ability of each agent towards solutions that fulfil time-variant expectations and performance constraints and reduce conflict that may arise from the interaction with other agents. The generative ability of the overall system is evaluated through its ability to produce coordinated solutions (plans). Through the view of decentralised control, coordination becomes an emergent property of the overall system, resulting from a continuous process of learning and self-adaptation. In the following we will show how this is formalised by adopting and adapting the Function-Behaviour-Structure (FBS) proposed by Gero (2000).

3 THE PRINCIPLES OF THE PROPOSED SYSTEM

3.1 The Construction of Plans

We consider that plans are composed in a 'collective space' by human and artificial agents that control parts of the overall description (Figure 1). The artificial design agents are introduced by users on the basis of a purpose or domain problem, but in principle, artificial agents should be able to control their own definition and population. The interface between human and artificial agents is built by objects (which are collections of variables) embedded in a Virtual Reality (VR) world. These objects are dynamically identified and modified by agents. We should note that human actors (and their computational constructs), and the way they manipulate objects, form the 'knowledge' or 'reasoning sources' for the artificial agents. The artificial design agents learn through the interactions that take place in the collective (VR) space and adapt their behaviour according to this knowledge. Their function is to steer the plan designing process towards (scalable) decision and problem formulations that may lead to the coordination of their distributed requirements. Artificial agents do not have global knowledge about the overall system and there is no central coordination mechanism. In effect, coordination is an emergent property of the system, triggered by the process of learning and self-adaptation of agents in the local level.

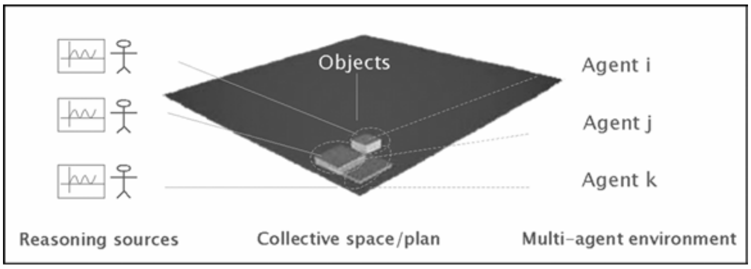

Figure 1 How Plans are Built

3.2 The FBS Framework

The meaning of the FBS framework has been extensively discussed in the design literature and in a variety of different contexts. FBS is a formalism that uses associations between functional, behavioural and structural information to formalise design as an activity that involves the processes of synthesis, analysis, formulation, re-formulation and evaluation. We will try to show here how this framework is adapted to support the formation of plans as a distributed process.

Structural information specifies the components of the proposed plan, their attributes and their relations. For the simulations presented later in this paper structural information depicts the physical components of objects and their topological relations. So, for instance, for an object (house), structural information includes location *[x, y]*, volume dimensions $[z_X, z_Y, z_Z]$ and relations with other objects such as: distance to other facilities (like retail and open space) and adjacency. Behavioural information specifies the way each object reacts to changes of its state and its environment. Behaviour is a description of structural change of the design objects in order to reach their intended functions. For instance, behaviours describe land use attractiveness (tendency of land uses to be attracted to -or repelled by- other facilities) or cost (according to land value and floor area ratio). We should distinguish between structural behaviour, which is directly derivable from structure (for example the cost of a building is related to its total floor area) and expected behaviour, which is derived from the function (for example there is a minimum floor area expected for a housing unit).Finally, we consider that functional information represents the teleology or purpose for the proposed objects. In one sense, function is equivalent to the memory of the future discussed previously. However, in the simulation we present, we adopt a more traditional view of function as activity contained in objects (e.g. land use). These three classes of variables are linked together by processes which transform one another; namely the processes of synthesis, analysis, evaluation, formulation and reformulation (Gero 2000). We will discuss these processes by adopting a control-based approach in the following section.

The important point in this framework is that it allows us to represent information about the problem or requirement space (Functional information), the solution space (Structural information) and the performance space (Behavioural information). Moreover, in our framework these three classes of information may be used to represent different components of a plan in different scales: for example function may represent functions within a building (living, dining, working) as well as land uses (housing, retail, open space). We should also note that behavioural information in specific is particularly critical: it represents high-level knowledge and plays an important role in linking function and structure and providing an evaluation mechanism.

3.3 Coordination as Distributed Learning Control

Each agent carries out two combined control-based activities: the first alludes to a synthesis-analysis-evaluation route and the second alludes to an evaluation-formulation-reformulation route (Figure 2). The objective of each agent is to find a suitable path of structures *S* that lead the behaviours *Bs* to follow a reference (expected) behaviour *Be* despite uncertainties and despite exogenous disturbances *Sd* produced by other agents' decisions. The expected behaviour *Be* is defined by a reference model, which is developed following a similar control process. The objective in that case is to find the appropriate functions *F* that lead the expected behaviour *Be*, to follow a reference structural behaviour *Bs* despite uncertainties and despite exogenous disturbances *Fd*. Hence, the desired performance of the synthesis-analysis system is evaluated (denoted by *E* in the figure) through the reference model (formulation-reformulation) which is defined by its input-output pair *{F, Be}*.

The control system attempts to guide the plant model to follow the reference output *Be* asymptotically: as seen in (1), where ε is a positive integer.

$$\lim_{t\to inf} |Be-Bs| < \varepsilon \quad (1)$$

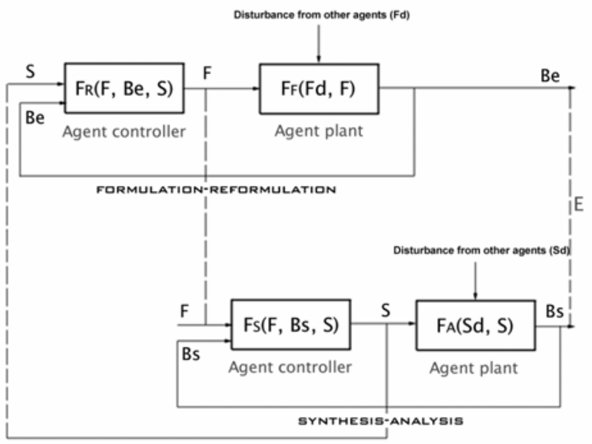

Figure 2 The Structure of an Agent

To sum up, what we call synthesis is a control process that generates solutions (structures) so that time-variant expectations can be satisfied. Reformulation is the control process that aims to redefine the problem formulation (function) so that new expectations developed may respect the constraints originated from the domain knowledge. The two processes of problem reformulation and solution construction co-evolve in time. Evaluation is the process of measuring the degree of 'matching' between the two control systems. The control signal *St,…,St+n* produced by this combined control process might be interpreted in three ways: as a set of partial actions (solutions) that build a global solution in time; as a course of complete actions (solutions) that the agent has to follow in order to reach its targets; or as a set of actions that regulate a given set of variables in order to satisfy time-variant targets. The latter option was implemented for the model presented here.

4 SIMULATION OF AN EXPERIMENT

The proposed plan design system is a prototype developed in a MATLAB-SIMULINK (Mathworks, Inc) environment. The typical architecture of an artificial agent is shown in Figure 3. Each box in the figure represents a subsystem of the overall system built using Adaptive Backthrough Control architectures. These architectures typically use two neural networks: the Controller (the system that controls) and the Plant Model (a model of the system to be controlled). First, the plant model is trained to approximate the reasoning sources (human operators or their computational models), by learning, on-line or off-line, input-output patterns of FBS attributes. Then, these patterns are used ‘backwards’ as a guideline for the controller.

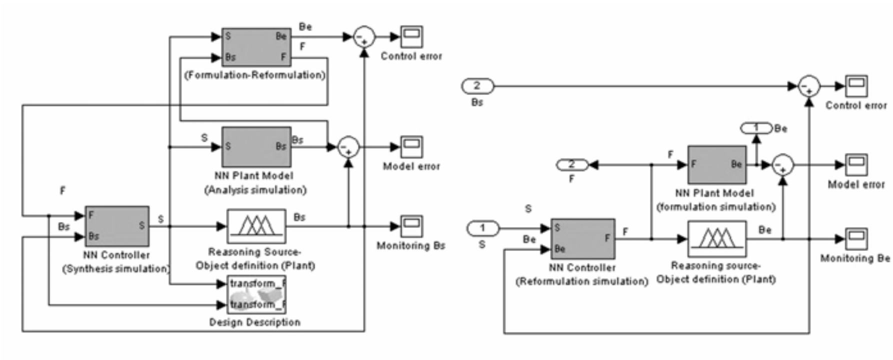

Figure 3 The Model of an Artificial Agent in Matlab-Simulink

For the simulation, we have experimented with mathematical formulations to simulate the reasoning sources (i.e. model the behaviour of objects, such as motion, shape transformation and costs) based on state-space methodology, as well as with fuzzy systems. As an example, Figure 4 shows a fuzzy system built on fuzzy IF-THEN rules used to represent expected proximity between land uses according to their volume dimensions.

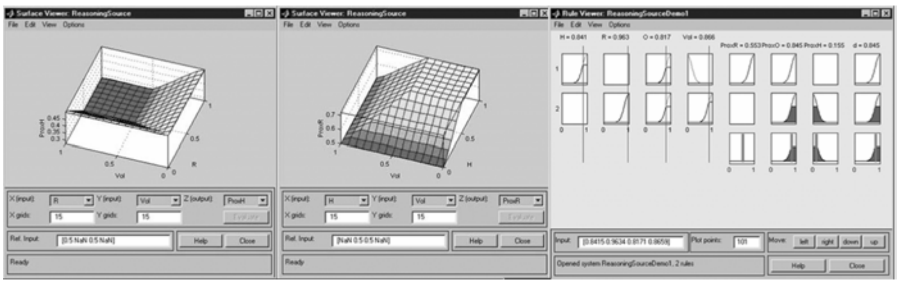

Figure 4 Simulating the Reasoning Sources Using Fuzzy Systems

The Virtual Reality toolbox offered the possibility to visualise the evolution of the design-decision space. We can directly retrieve and manipulate the location and shape variables of the three objects and view the conflict as it evolves in the three dimensional space. Conflict in the current version of the model is introduced as ‘disturbance’ in control terminology. Conflict may arise due to the fact that agents control parts of the overall description, so there might be variables that affect the performance of the agents and cannot be anticipated. Agents need to find the appropriate solutions (functions or structures) despite this disturbance. To illustrate this in more detail we present below a conflict situation.

In Figure 5 the agent that controls the housing unit *H*, has developed an expectation to move close to the retail facility *R*. In contrast, the agent that controls the retail facility has developed an expectation to move far from the housing unit (the third cuboid denotes open space *O* and remains unchanged). This conflict situation can be resolved in two ways. Given that the retail cuboid remains unchanged, the agent that controls the housing unit might: 1) Change function and hence expectations for the plan design process (through reformulation) and adopt a mixed land use *H'*, in order to follow the structural behaviour and keep far from the existing retail unit (Figure 5, bottom left), or 2) Change radically its structure to *H1H2* so that it can potentially attract retail, and therefore satisfy its original expectation (in Figure 5, bottom centre, the retail unit can potentially move close to the extended housing), or 3) Generate another retail agent/cuboid *R2* whose expected behaviour is to move close to housing (figure 5, bottom right).

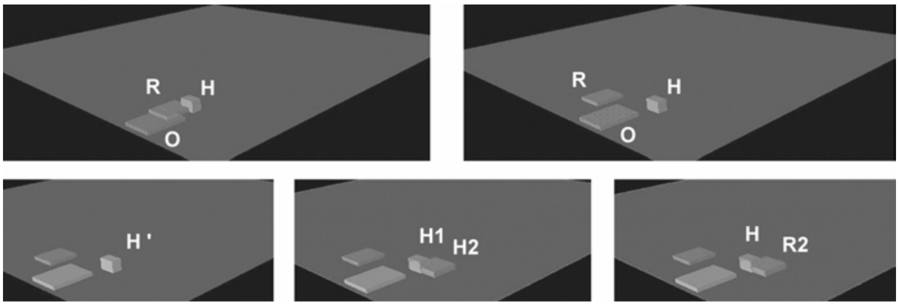

Figure 5 A Conflict Situation (top) and its Resolution (bottom)

5 DISCUSSION

The simulations presented here are still under development so, little can be claimed regarding the overall performance of the system and hence the validity of our hypotheses. Validating such -generative in nature- systems is however an important question. One approach is to have the resulting plan descriptions evaluated by domain experts. The other approach, adopted so far, is to stage different conflict scenarios as the one described, and review the rationality of the results and 'degree' of creativity (or novelty) in each case, and in respect to the expectations of individual agents. As we have mentioned, coordination (or conflict resolution) is the main indicator of the creative ability of the system as a whole. This ability is supported by the ability of agents to restructure problem and solution formulations in a timely manner. On the other hand, the ability of the system to reformulate problem definitions and/or relax certain constraints, might be problematic if agents do not reflect realistically the human preferences. Weak learning performance from the part of the artificial agents could lead to a ceaseless problem reformulation loop.

Similarly, generalisations produced in the learning process may be innefficient in coping with conflicting or alternative preferences expressed by individual humans. In any case, the systems' efficiency is primarily related to its learning ability and much attention should be paid in devising and using the appropriate learning algorithms.

REFERENCES

Candy, L. 1997. Computers and Creativity Support: Knowledge, Visualisation and Collaboration. *Knowledge-Based Systems,* 10: 3-13. Elsevier Science.

Coyne, R.D. 1990. Design Reasoning without Explanations. *AI Magazine,* 11(4): 72-80. AAAI.

Csikszentmihalyi, M. 1988. Society, Culture, and Person: a Systems View of Creativity, *The Nature of Creativity,* eds. R.J Sternberg, 325-249. Cambridge University Press.

Fischer, G. 1999. Symmetry of Ignorance, Social Creativity, and Meta-design. *Creativity & Cognition 1999*, 116-123. ACM.

Gero, J.S. 1999. Constructive Memory in Design Thinking. *Design Thinking Research Symposium: Design Representation*, eds. G. Goldschmidt and W. Porter, 29-35. Cambridge: MIT.

Gero, J.S. 2000. Computational Models of Innovative and Creative Design Processes. *Technological Forecasting and Social Change,* 64: 183-96. Elsevier Science.

Goldschmidt, G. 1995. The Designer as a Team of One. *Design Studies,* 16: 189-209. Great Britain: Elsevier Science.

Maher, M.L. 2000. A Model of Co-evolutionary Design. *Engineering with Computers,* 16: 195-208. Springer.

Ossowski, S. 1999. *Co-ordination in Artificial Agent Societies: Social Structure and its Implications for Autonomous Problem-Solving Agents*. Berlin Heidelberg: Springer- Verlag.

Reffat, R.M. and J.S. Gero. 2000. Towards Active Support Systems for Architectural Designing. *Proceedings of eCAADe 2000*, ed. D. Donath, 143-147. Weimar: Bauhaus Universitat Weimar.

Rosenman, M. and J. Gero. 1999. Evolving Designs by Generating Useful Complex Gene Structures. *Evolutionary design by computers*, ed. P. Bentley, 345-364. San Francisco, CA: Morgan Kauffmann.

Saunders, R. and J.S. Gero. 2001. Artificial Creativity: A Synthetic Approach to the Study of Creative Behaviour. *Computational and Cognitive Models of*

Creative Design V, eds. J.S. Gero and M.L. Maher, 113-140. Sydney: University of Sydney, Key Centre of Design Computing and Cognition.

Segers, N.M., Achten, H.H., Timmermans, H.J.P. and B. de Vries. 2000. A Comparison of Computer-aided Tools for Architectural Design. *Proceedings of the 5th International Conference on Design and Decision Support Systems in Architecture*, ed. H.J.P. Timmermans, 325-339. Eindhoven: Eindhoven University.