Artificial Design and Planning Support: Interactive Plan Generation and Coordination in Distributed Decision-Making

K. Alexiou and T. Zamenopoulos
University College London
Bartlett School of Graduate Studies
Centre for Advanced Spatial Analysis
London
UK

ABSTRACT

In this paper we discuss some basic issues pertaining to artificial plan design as a paradigm for architectural design and urban planning support. We present a model for artificial design generation based on learning control methodologies. Plan design is seen as a search for "coordinated" solutions (changes) that satisfy distributed domain requirements and views expressed by human or artificial agents. Learning control is used as a method to search for solutions that direct partial descriptions produced by agents, to follow their dynamically defined targets -despite conflicting requirements. The model is simulated for land use and layout plan design, involving decisions for the location and physical configuration of a hypothetical housing and retail development.

1 INTRODUCTION

The use of computational models in the context of decision support systems has been a subject of investigation in architectural design and urban planning for some 40 years now. In this history of computer-aided design and planning, very many different approaches to support based on formal methodologies have been developed, that turn any attempt for classification into a tedious and subjective task (Timmermans 1994). The diversity of approaches toward computational intelligence in the context of decision support systems has been augmented after the mid 70s when the lure of computational models as operational tools fades out (Lee 1973), and formal models are used loosely as analogies (Batty 1984) and as tools to "think with".

A typical paradigm in architectural and urban planning support is the artificial generation of plans, which is sometimes referred to as plan design (Schlager 1972), automatic generation of designs (Steadman 1970, Cross 1977) or generation of alternatives (Brill et al 1990). The realisation that design and planning are cyclic processes that involve a continuous interaction among the different design tasks has established the importance of this paradigm in design and decision support. Artificial plan design has facilitated the generation of proposals that can be evaluated from the early phases of design through to the final stages of solution formation.

In urban planning a typical application addresses the problem of land-use plan design (e.g. Feng and Lin 1999, Aoki and Muraoka 1997, Anderssen and Ive 1992), while in architecture the dominant example is in building layout design (e.g. Chakrabarty 1990, Steadman 1970, Jo and Gero 1998, Mitchell et al 1976). In this paper the simultaneous generation of land-use and layout plan design is elaborated.

The problem is defined as a search for locations and physical layout proposals that satisfy distributed and time-variant requirements or targets. Expert knowledge for this search is not explicitly incorporated in the model but a Neural Network (NN) architecture is used instead to discover and represent knowledge captured as interdependencies among decision variables expressed by distributed sources (decision makers or their domain models). We present a model-tool that learns from user interaction and then uses this knowledge to search and generate design proposals. For the simulation of this model we take a hypothetical urban development assignment that aims to the development of a housing and retail unit. The attractive point in this framework is that we have to consider a simultaneous and constant generation of alternative plans, both in the architectural and the urban scale, from the preliminary stages of the plan design. Additionally, requirements and targets are typically distributed among different teams and vary in time according to the emergence of new conditions (Cadman and Topping 1985).

2 ARTIFICIAL PLAN DESIGNING

Before we proceed with the presentation of the model it would be useful to see the broader picture in artificial designing and discuss some theoretical and methodological issues.

Optimisation has been the predominant approach to automated plan design, in urban planning as well as in architectural and engineering design (Gero 1985, Harris and Batty 1993). The design problem is translated into a search for design(s) that represent optimum solutions. Thus appropriate methodologies need to be devised to generate and choose solutions that optimise some utility or cost function under a number of constraints. There are different formulations that fit to this paradigm which employ techniques ranging from linear programming (Anderssen and Ive 1982) to multi-objective (Balling et al 1999) and genetic programming (Aoki and Muraoka 1997, Chakrabarty 1990). Another approach to automatic plan generation includes the development of algorithms for the exhaustive investigation of all possible (or feasible) solutions pertained to a design problem (Haubrich and Sanders 2000, Alexander 1962, Steadman 1970). A third paradigm considers plan design as a search for creative solutions. Evolutionary algorithms have been used for creative design in architecture and engineering (Bentley 1999, Frazer 1995). Shape generation based on algebras or grammars is another potential plan generation process based on selection, creative exploration and emergence (Stiny 1994). Arguably, creativity and innovation are important issues in plan designing which usually relate to a task of employing known solutions to a new context (Gero 2000). Case Based Reasoning (CBR) deals with such issues of creativity. CBR as has been used in design automation, starts from the recognition that knowledge is distributed to design cases which can be adapted and reused in similar contexts to support creative reasoning (Maher and Pu 1997, Yeh and Shi 1999).

This plethora of methodologies discloses a plethora of ways to understand designing and planning. In this paper we see plan design as a search for "coordinated" solutions (changes) that satisfy distributed domain requirements and views. Learning control is a method used to search for solutions that direct partial descriptions to follow their (dynamic) targets despite conflicting requirements. There are three hypotheses behind this view: the first is that decision making is distributed among multiple agents, bearers of different types of knowledge and individual needs; the second is that some kind of coordination needs to be reached among these diverse requirements and purposes; and the third is that knowledge cannot be defined a-priori in this context, but some learning mechanism needs to be devised to capture distributed knowledge and effectively use it to generate plan designs.

The first hypothesis is related to methodological issues. The shift from designing plans based on individual action to designing plans based on collective-distributed action has pointed towards a reconciliation of normative and positive approaches (Batty 1984). The idea behind this shift is that changes occur not because of a centrally controlled action but rather they emerge from a distributed decision-making process. In other words the normative activity of change is under the weight of a collective dynamic. In the same view knowledge is also distributed, not only because plans are collectively formed by communities (or multidisciplinary groups), but also because even expert reasoning is fragmented to diverse goals, criteria and evaluations.

Naturally, in the context of distributed decision making, plan design involves searching for configurations that reduce or resolve conflict among distributed goals. Different computational paradigms have been used to formalise the conflict resolution problem. Broadly speaking we can distinguish three approaches. The first appoints a collective function that needs to be optimised for the sake of a "social welfare", the second leaves the dynamic among the involved parts to determine the distribution of welfare, and the third directs the distribution of welfare equally among the involved parts. Bargaining is one way to formalise the design problem, others including negotiation, conflict resolution, social choice, consensus or cooperation (Kleindorfer et al 1993). In this research, plan design -in the light of distributed decision making and conflict resolution- is considered as a coordination problem. Coordination is extensively discussed in the context of organisational decision support systems (Grandori 2001, Malone and Crowston 1990) and is a recurring issue in the literature about distributed artificial intelligence and multi-agent systems (Ossowski 1999, Jennings 1996). Whether talking about actors or agents, human or artificial, coordination is what makes them act as a distributed system and reach solutions on the basis of managing interdependencies among individual requirements. In the following we will introduce the idea of coordination as a process of capturing interdependencies among individual knowledge sources, and reach equilibrium, despite disturbances or conflicting requirements expressed by distributed agents. In this context innovation and creativity lies on the possibility of unforeseen solutions to emerge through agent interaction.

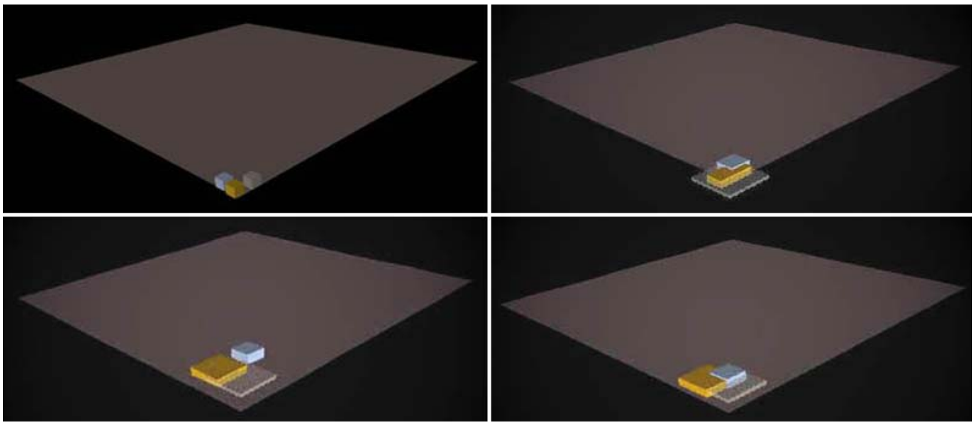
Finally, the third hypothesis relates to the question of how domain knowledge about the system to be designed is incorporated within the model (e.g. land use interaction matrices, or adjacency criteria for room layout problems). This is a field of research on its own as it is connected with problems of knowledge representation and acquisition, so we won't thoroughly discuss it. But there are some methodological issues concerning human-model interaction that are associated with the way domain knowledge is incorporated to the model. Very often, domain knowledge is seamless with the proposed model. For instance, facility planning has been extensively addressed with respect to studies on user behaviour, thus building models (e.g. gravity models) that represent this behaviour. Human-model interaction is defined mainly by choosing and introducing to the system alternative scenarios for specific problems (e.g. emergency facilities location) without however affecting the reasoning base of the system (which for instance, will always be based on min-max relationships). Other methodologies avoid these strict dependencies using for instance Genetic Algorithms methodology (Aoki and Muraoka 1997) but there still seems to be a need to have an explicit (formal) description of the domain behaviour incorporated within the model. A different paradigm -where domain knowledge is updated dynamically by user interaction- is again Case Based Reasoning. This research has been developed on the idea that if we can devise a tool that learns patterns of reasoning from the distributed resources, we can then use this knowledge to produce coordinated solutions based on partial evidences. This task can be supported and improved if learning is introduced in the computational model. We use Neurocontrol as a paradigm for the artificial generation of coordinated plan designs. In the following we discuss a formal model of coordination using the paradigm of learning control.

3 PLAN DESCRIPTION

We consider that plan descriptions are built on distributed domain problems and/or partial proposals developed and controlled by agents (human or artificial). For instance, a trivial location and space layout problem may involve various groups of agents: one that defines the appropriate location, another that designs a suitable distribution of volumes, a third that designs a potential spatial distribution of rooms and a last one that is involved in the structural engineering of the building. Each agent within a group is self-interested and represents a partial component of the overall description.

In the context of this paper, plan descriptions are generated within a virtual reality (VR) world and are composed by aggregated objects introduced by users. Typically, objects are justified on the basis of a "purpose" and represent a domain or a partial description of the plan. The specifications of these objects are dynamically identified and modified by human actors directly or via the use of formal models. As an example, for the simulation described in this paper we used three objects (initially in the form of three cubes) located in a hypothetical virtual city, which represent the preliminary development goals for a housing unit, a retail facility and an open space (figure 1). We should note that the way objects are defined is critical for the model because it determines the way control is distributed. The definition of objects determines the subject of control wielded by human operators or their models.

Figure 1: **Representation of agents as objects within a VR environment**



The objects within the virtual environment are built on three classes of information: Structural, Behavioural and Functional (SBF). The meaning of the SBF framework for the plan design has been extensively discussed in literature and in a variety of different contexts (e.g. Gero 2000, Gorti et al 1998, Szykman et al 2000, Narasimhan et al 1997). In this paper we will only discuss briefly how this framework is adopted in the context of urban planning.

More formally, each object is specified as a row matrix: $A_i=[S_{ai}, B_{ai,} F_{ai}]$. The overall plan description will be the column matrix P = [Ai] of all these objects. Structural information depicts the physical components of the plan. In the simple example we present, structural information defines the natural state of each object-cube, that is location [x y] and volume dimensions $[z_x\ z_y\ z_z]$. Behavioural information describes the way each object reacts to changes of its state and its environment. In our example we use behavioural functions of "motion" and "cost". The former represents the tendency of moving land uses close to (or far from) other facilities and is expressed by variables of distance d=[dij] and its second derivatives (i is the introduced facility and j the target land uses). The latter represents an estimation of the development cost c for the current component, based on land value and floor area ratio. Finally, we consider that functional information represents the ontology of the proposed object expressed as land use –in our case housing, retail, and open space.

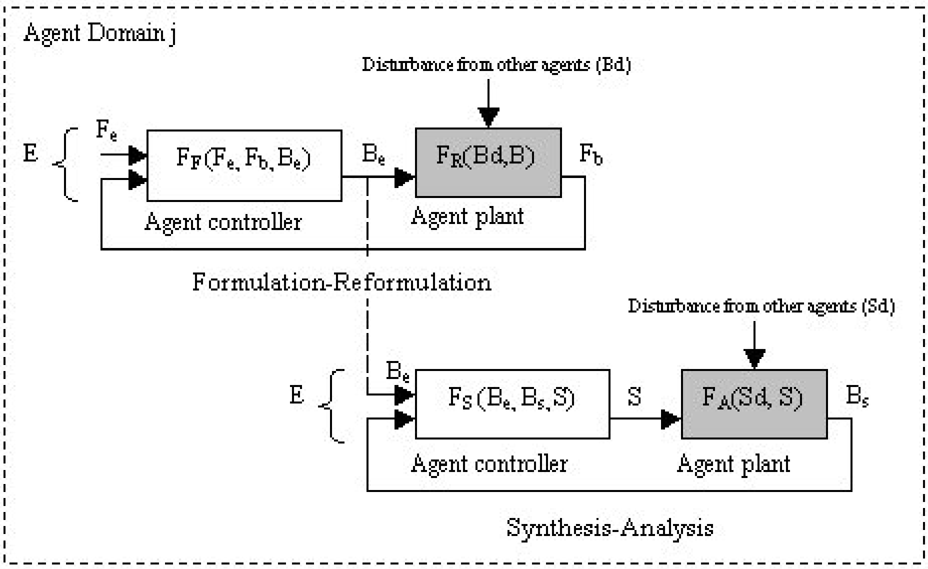
4 ARTIFICIAL PLAN DESIGNING AS CONTROL BASED COORDINATION

The design problem is formulated as a coordination problem among self-interest agents (which are represented as cubes in the VR world) and is addressed via a distributed learning control methodology. In general the idea can be summarised as follows: a learning algorithm is used to train a neural network to discover associations among Structural, Behavioural and Functional attributes (in this paper we use off-line training). This knowledge is then used to generate plan descriptions, based on partial information presented to the NN, which will satisfy a temporal (preliminary) reference target for the SBF attributes. For each agent we assign a control architecture which seeks to stabilise SBF interdependencies while facing internal variations and external disturbances presented by the other agents. Even though there are a lot of different control-based formulations that might be reasonable for coordination problems (for a different formulation refer to Alexiou and Zamenopoulos, 2001), we will present here one where we address coordination and conflict as a search for equilibrium despite inherent uncertainties and despite exogenous disturbances.

More analytically, each self-interested agent carries out two combined control-based activities: the first alludes to a synthesis-analysis-evaluation route expressed as a function among Structural Decisions S, Expected Behaviour B_e and Actual Behaviour B_s. The second activity alludes to an evaluation-formulation-reformulation route expressed as a function of Actual Behaviour B_s, Expected Function F_e and Actual Function F_b.

The objective of each agent is to find a suitable path of structures S that lead the behaviours B_s, to follow a reference (expected) behaviour B_e, despite uncertainties and despite exogenous disturbances Sd produced by other agents' decisions. The expected behaviour B_e is defined by a reference model, which is developed following a similar control process. The objective in that case is to find the appropriate behaviours B_e that lead the function F_b, to follow a reference (expected) function F_e, despite uncertainties and despite exogenous disturbances Bd (figure 2). 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 $\{F_e, B_e\}$. The control system attempts to make the plant model follow the reference output B_e asymptotically:
$\lim_{t\to\inf} | B_e - B_s | < \varepsilon$, where ε is a positive integer.

Figure 2: **Plan generation as a control process**



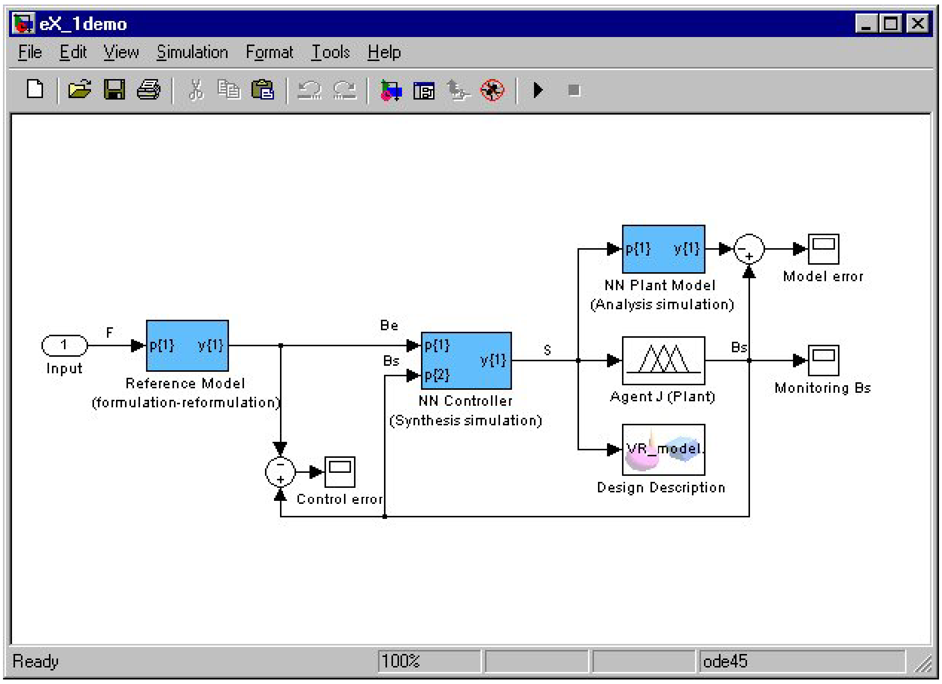
To sum up, what we call *synthesis* is the control process that aims to stabilise the state space (behaviour) of an agent according to a reference value for the behaviour B_e; and *formulation* is the control process that aims to stabilise the state space (function) of an agent according to a reference value $F_{e.}$ *Evaluation* is the process of measuring the degree of "matching" between the two control systems. The control signals St,…, St+n produced by this combined control process consist a set of evolving plans (proposals) for the design and planning problem in hand. The process of artificial generation of plans based on learning control is a process of self-adaptation of agents that leads to coordination of their distributed descriptions.

5 SIMULATION

The above model is developed and simulated in a MATLAB-SIMULINK (Mathworks, Inc) environment. We are experimenting with Adaptive Backthrough Control architectures. These structures typically use two neural networks: the Controller (the system that controls) and the Plant Model (a model of the system to be controlled) (figure 3). First, the plant model is trained to approximate the plant by learning on line or off-line input-output patterns of the agent behaviour. Then, these patterns are used "backwards" as a guideline for the controller (Kecman, 2001).

In our case the plant has been implemented as a compact block of three agents that represent the design and planning reasoning of the three objects that stand for the different development goals. For the purposes of this simulation we do not introduce human operators but we rest on formal descriptions to represent them.

Figure 3: **The control model**



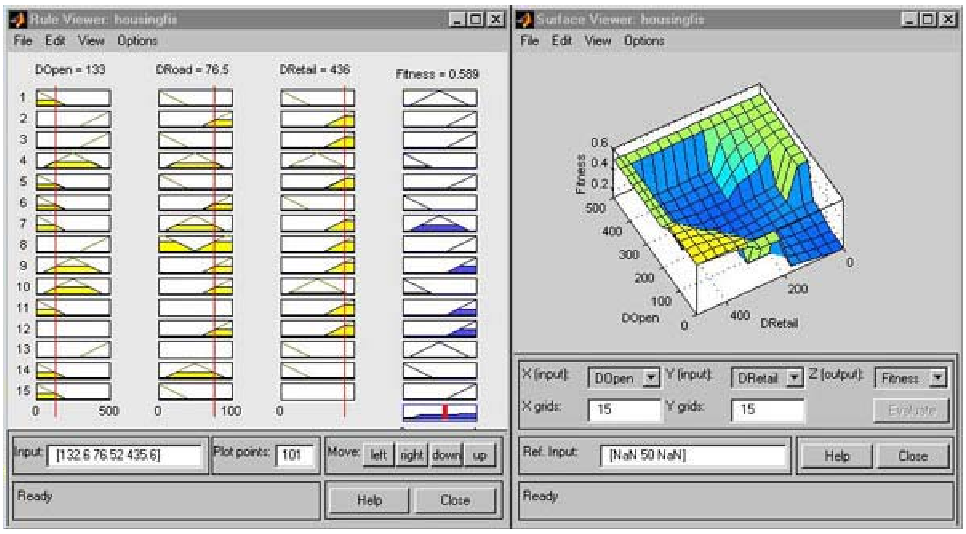
We have experimented with mathematical formulations that model agent behaviour (like motion, shape transformation and costs) based on state space methodology, as well as with fuzzy systems. As an example, the "moving behaviour" of the land use j is described by n equations (for n land uses) as follows:

$$m_j x_j'' = \sum_{i=1}^{n} k_{ij}(x_i - x_j),$$

where m_j is the floor area of the land use j, x_j is position, k_{ij} is the interaction matrix between land use j and i, and x_j'' is the second derivative of the distance. Fuzzy systems are built on the basis of fuzzy IF-THEN rules, which for example may represent qualitative evaluations about the fitness of a specific location based to criteria of proximity with neighbouring facilities (figure 4).
An important component of the Reference Model Controller we are presenting here is the reference model. It is essentially a prototype model of the system producing the target behaviours that train the controller towards a desirable state, and corresponds in our case to the formulation-reformulation phase of the design description generation. The structure of the reference model as described previously, is a control architecture similar to the one focused on the synthesis-analysis process.

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 evolves in the three dimensional space. Something we are currently working on is the connection of the VR world with a spatial database, so that we can retrieve information from the environment that is crucial for the agent reasoning, and is not incorporated in either of the agents. So far we have focused on the interaction among the three objects within a neutral space, but this may give the possibility to expand control beyond the limits of the three objects alone.

Figure 4: **Agent reasoning as a fuzzy system**



6 REFERENCES

Alexander, C. and M. L. Manheim (1962) The use of diagrams in highway route location: an experiment, RR-R62-3, Department of Civil Engineering, MIT, Cambridge, MA.

Alexiou, K. and T. Zamenopoulos (2001) A connectionist paradigm in the coordination and control of multiple self-interested agents, *Proceedings of the 2nd National Conference Input 2001: Information Technology and Spatial Planning, Democracy and Technologies*, Tremiti Islands, June 27-29, 2001, pp. 25.

Anderssen, R. S. and J. R. Ive (1982) Exploiting structure in linear-programming formulations for land-use planning, *Environment and Planning B: Planning and Design,* 9 (3), pp. 331-339.

Aoki, Y. and N. Muraoka (1997) An optimisation method for facility location using a genetic algorithm, in Timmermans, H. (ed.), *Decision Support Systems in Urban Planning*. E & F N Spoon, London, pp. 175-191.

Balling, R. J., Taber, J. T., Brown, M. R. et al (1999) Multiobjective urban planning using genetic algorithm, *Journal of Urban Planning and Development* 125 (2), pp. 86-99.

Batty, M. (1984) Plan design as committee decisionmaking, *Environment and Planning B: Planning and Design* 11 (3), pp. 279-295.

Bentley, P. J. (1999) *Evolutionary Design by Computers*. Morgan Kaufmann, San Francisco, California.

Brill, E. D., Flach, J. M., Hopkins, L. D. et al (1990) MGA: a decision support system for complex, incompletely defined problems, *IEEE Transactions on Systems, Man, and Cybernetics* 20 (4), pp. 745-757.

Cadman, D. and R. Topping (1995) *Property Development*. E & F N Spoon, London.

Chakrabarty, B. K. (1990) Models for the optimal design of housing development systems, *Environment and Planning B: Planning and Design* 17 (3), pp. 331-340.

Cross, N. (1977) *The Automated Architect*. Pion Limited, London.

Feng, C-M., and J- J. Lin (1999) Using a genetic algorithm to generate alternative sketch maps for urban planning, *Computers, Environment and Urban Systems* 23 (2), pp. 91-108.

Frazer, J. (1995) *An Evolutionary Architecture*. Architectural Association, London.

Gero, J. S. (1985) *Design Optimisation*. Academic Press, London.

Gero, J. S. (2000) Computational models of innovative and creative design processes, *Technological Forecasting and Social Change* 64, pp. 183-196.

Grandori, A. (2001) *Organisation and Economic Behavior*. Routledge, Taylor & Francis Group, London and New York.

Gorti, S. R, Gupta, A., Kim, et al (1998) An object-oriented representation for product design processes, *Computer-Aided Design* 30 (7), pp. 489-501.

Harris, B. and M. Batty, (1993) Locational Models, Geographic Information and Planning Support Systems, *Journal of Planning Education and Research* 12, pp. 184-198.

Haubrich, J. J. A. and F. M. Sanders (2000) Potential surface analysis: a planning and design support system, *Proceedings of the 5th International Conference in Design and Decision Support Systems in Urban Planning*, Nijkerk, The Netherlands, August 22-25, pp. 128-143.

Jennings, N. R. (1996) Coordination techniques for distributed artificial intelligence, in O'Hare, G. M. P. and N. R. Jennings (eds.) *Foundations of Distributed Artificial Intelligence*, John Wiley & Sons, New York, pp. 187-210.

Jo, J. H. and J. S. Gero (1998) Space layout planning using an evolutionary approach, *Artificial Intelligence in Engineering* 12 (3), pp. 149-162.

Kecman, V. (2001) *Learning and Soft Computing. Support Vector Machines, Neural Networks and Fuzzy Logic Models*. The MIT Press, London.

Kleindorfer, P. R., Kunreuther, H. C. and P. H. J. Schoemaker (1993) *Decision Sciences: an integrative perspective*. Cambridge University Press, Cambridge.

Lee, D. B. (1973) Requiem for large-scale models, *Journal of the American Institute of Planners* 39, pp. 163-178.

Maher, M. L and P. Pu (eds.) (1997) *Issues and Applications of Case-Based Reasoning in Design*. Lawrence Erlbaum Associates, Mahwah, NJ.

Malone, T.W. and K. Crowston (1990) What is coordination theory and how can it help design cooperative work systems?, *Proceedings of the Conference on Computer-Supported Cooperative Work*, Los Angeles, CA USA, October, 1990, pp. 357-370.

Mitchell, W. J., Steadman, J. P. and R. S. Liggett (1976) Synthesis and optimisation of small rectangular floor plans, *Environment and Planning B* 3 (1), pp. 1-141.

Narasimhan, S., Sykara, K. P. and D. Navin-Chandra (1997) Representation and synthesis of non-monotonic mechanical devices, in Maher, M. L. and P. Pu (eds.), *Issues and Applications of Case-Based Reasoning in Design*. Lawrence Erlbaum Associates, New Jersey, pp. 187-222.

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

Schlager, K. J. (1972) A land use plan design model, in Robinson, I. M. (ed.), *Decision-Making in Urban Planning*. Sage Publications, Beverly Hills and London, pp. 139-150.

Steadman, P. (1970) The automatic generation of minimum-standard house plans, Land Use and Built Form Studies, Working paper 23, Department of Architecture, University of Cambridge, Cambridge.

Stiny, G. (1994) Shape rules: closure, continuity and emergence, *Environment and Planning B: Planning and Design* 21, pp. s49-s78.

Szykman, S., Racz, J., Bochenek, C. et al (2000) A web-based system for design artifact modeling, *Design Studies* 21 (2), pp. 145-165.

Timmermans, H. (1994) Decision support systems in urban planning and urban design, *Environment and Planning B: Planning and Design* 21 (1), pp. 1-4.

Yeh, A.G.O. and X. Shi, (1999) Applying case-based reasoning to urban planning: a new planning-support system tool, *Environment and Planning B: Planning and Design* 26 (1), pp. 101-115.