Architectural Computer-Aided Design Systems: An Example

Ivan Petrovic &Igor Svetel[1]

Introduction

Aim of the Paper

In 1987, a project was initiated at The IMS Design Research Workshop with the aim to study the effects of the introduction of CAAD methodology on the possible qualitative improvement of architectural design in the Institute's proprietary Building System "GIMS". The system consists of a precast prestressed concrete skeleton structure that incorporates various sub-systems, and has been applied extensively in housing and public building in Yugoslavia and many other countries [Banic 86].

Generally speaking, all well-defined and well-structured aspects of architectural design, mainly dealing with the technical aspects and/or graphical presentations, have been successfully modeled and merged with the computer application and applied particularly in the detailed design phases. This is not so with the ill-structured problems and fluid situations that dominate the conceptual phase. Many decisions here depend on the subjective judgements of the designer. Knowledge-based systems have been applied in the selected domains to aid decisions based on experience and difficult to model by algorithmic methods. However, how to increase quality of design process is an open-ended question. The semantic aspects of design have not been treated to a great extent in CAAD research projects so far.

The paper describes some of the project results - the CAAD methods and tools to be used as aids in conceptual design of the IMS family houses. The tools have been developed to a prototype level, with limited, but adequate testing of their performance. The present versions are applicable on the IBM personal computers.

"Object - Oriented Design Methodology"

Design Objects

Let us define design process as a "modeling", of design problems, ideas, proposals, physical objects, contexts. In this paper, we shall use the term "object" to refer to various "models" of the real architectural object that is going to be built, a GIMS house. Model type depends on its role in the design process. For example, an abstract model can be comprised of the expected design requirements of a house, while a geometric model shall contain the description of a physical structure with the required characteristics. Use of object models may differ from designer to designer, from one to another user of design information, but can be "standardized" and used by many users in similar situations [Tolman 90], [Penttila 90].

"Object model problems". Definition of an object model depends on what the model is going to be used for, where (e.g. in what phase of design process), how (e.g. what tools shall be applied), by whom, etc. If the designer is not sure what model to use, there exists an "object model problem".

[1]The IMS Institute, POB 834 11000 Belgrade, Yugoslavia

Design Process

We shall now claim that design is "modeling of the desired change of a design object". This definition is based on the writings by J. C. Jones [Jones 70], and G. H. von Wright's "Logic of action" [von Wright 67]. Description of change includes the descriptions of: a) the initial state of the object (when the change was initiated), b) the goal state of the object (when the change was completed), and c) the interims states leading from a) to b). Design process functions in such a way that the design methods are applied to perform various tasks, e.g. a) to define what is required (in the initial state), b) to propose and represent an idea of a kind of physical object that corresponds to such requirements (in the end state), and c) to evaluate object's attributes in order to assure the designer that the design tasks are achieved (in the interim states).

"Design-method problems". If the designer is not sure what method to use, or there exist more methods leading to the similar results, there exists a "design method problem".

Design Methodology

Design methodology is a body of knowledge that can help our understanding why certain design methods are used for achievement of certain tasks in various design situations, or, explains approaches to solving "object model problems" and "design methods problems". Design methodology in the new, systematic sense [Jones 62] is contrasted to the "traditional" one, claiming that the design process can be externalized, and the rational design decision-making applied throughout. The dilemma of the general applicability of such approach is not quite settled yet. Examples of methodological problems are the treatment of mixed objective and subjective assessments of both requirements and achievements, and absence of the "dependable" methods for transformation of the requirements expressed in words into the description of "novel" physical objects.

How designers work. A rational methodologist may follow the described linear design sequence of definition of change, and apply various algorithmic procedures for "composition" (and decomposition) of objects using rational decision procedures. A traditional designer may start in the middle of the process, go from the end to the beginning of the process and back, changing ideas and design methods as he/she sees fit, and even ending with a proposal that radically differs from the given requirements but supported by the "subjective judgements". The favorite design method is the "trial-and-error" technique which depends on designer's intuition and experience. While the rational design techniques are apt for computer modelling, the traditional design process is not. "Knowledge-based design systems" contain the designer's "heuristic" experience, and can, to some extent, simulate designer's behavior in certain situations [Lansdown 82], [Gero 85], [Coyne 88].

Proposed Approaches

Approach to methods-problems. If a CAADesign program is to be of use to a demanding type of an expert designer, it should contain a "requisite variety" of methods which shall at least tally or be greater of the variety of problems [Ashby 61]. This requirement comes natural to a traditional designer. Naturally, the methods should be able to cope with problems arising during the design process.

Approach to object-model problem. As design methods are applied to objects, there must also exist a requisite variety of object model representations adequate to the design problem at hand.

Fusion of object and method. Each object model (data structure) is implicitly associated with a set of design methods, techniques and tools that have participated in its formation. As a proper match between the object and the method is desirable, the latter can change the former to a different state. It is a common-sense design economy to use the "fused methods and objects" in design process. A similar idea appears in "object-oriented programming" [Goldberg 83]. More objects may be compatible with the same methods, but the latter may produce different objects. We are facing now the problem of choice of design method. The requisite variety of fused objects and methods must be provided by the program.

We leave the selection of the methods to an expert designer, to apply them in a way appropriate to his/her objectives, intuition, and the applied design logic defining the nature of the task to be performed. In our experience, many a seasoned expert applies only a few of the favourite methods at the "critical" points in design process on the conceptual level, being able to judge the appropriateness of each applied evaluation criterion and devise a plan for further object transformation upon such partial results. This leads to the eventual creation of an "expert's expert design system".

A cyclic structure of design process. Instead of the following the "linear" design procedures following the description of change, we propose instead the cyclic, "whirling" design process structure, where one can start at any point (beginning, end or at an interim state), and go between the states in accordance to the designer's plan [Hickling 80]. This means that the designer may propose the final solution first, than go back to the beginning to see whether the solution is acceptable, and than go through the interim stages back and forth between the initial and end stage until an acceptable/satisficing/optimized solution is produced. Many variations on this theme are possible. However this approach may seem hectic, in the authors experience, it may work quite efficiently.

The described concept enables:

a) designer's implicit selection of the method that suits the problem at hand and not vice-versa,
b) application of a method or their sequential mix as many times as necessary at any phase of the process,
c) object's transformation or replacement providing the new one responds to the available methods, and
d) inclusion of new methods and objects.

At present we stay with the interactive application of the program and shall continue seeking the justifiability of its automation in future.

How to deal with object semantics? We place this problem to the "subjective" area of design, and would like to enable designer's/user's participation during the process in assessing what kind of semantic meaning an object ought to have, or what kind of meaning is achieved. To this purpose we propose the use of a neural-net based tool, with a 3-d modeller and a "semantic differential" diagram as an input/output device [Osgud 57]. Both the 3-d model and the diagram are easily comprehended, the diagram enables easy description of approximate meanings, but, the working of such tool escapes the theoretical formalization. We see no reason to exclude the investigation of usability of such non-formal methods in systematically organized design contexts [Petrovic 1991a, 1991b].

Methods in Use: An Example

Examples of GIMS Methods and tools

Plans synthesizers: ARCH. The prototype program is based on the exhaustive search using depth-first branch-and-bound algorithm for layout synthesis. The problem and the solution are represented by the same data-structure (state-space) which contains the description of all elements. The properties can be defined by the user or by the program defaults. The second data structure contains the binary relations between pairs of elements, and is implemented as an associative memory matrix. The third data structure is the knowledge-base, still to be implemented. In each step of the search, the program adds one element to the layout, and evaluate its links with other elements, both located or unlocated. If the relationship of the element with its context is proven to be satisfactory, the next element is proceeded, else, the next position of element in the layout is tested. When all properties are (nearly) defined, the arrangement is taken to represent the solution [Svetel 90]. Room shapes are rectangular without any special dimensional constraints.

Plans synthesizers: GIMS-EXPERT. This is a "consultant" to a designer of individual family houses to be built in the "GIMS building system, in this instance featuring the modular structural skeleton grid of 4,20 x 4,20 m. Following the user's preference of the shape of the house, type and number of functional spaces and the location data, the program generates a set of feasible design alternatives using the elements from the GIMS Catalogues and rules on their combinations, both contained in the Data/Knowledge Base. The catalogues include the "functional elements" such as living rooms, entrance-stairs-WC-or-bathroom units, bedrooms with corridor etc., all designed to fit the structural module. The ground-floor and first- floor plans are generated separately, and then fit together. The sub-program Roof Planner, a small production system by itself, produces all possible roof alternatives for all selected design alternatives.

It should be pointed out that although the functional uses are indicated in plans, the internal partitions and elaboration of facades are missing, as their elaboration is treated in a separate project in progress. However, the information obtained on this (conceptual) level is in fact sufficient for further project programming phases: builder can start planning the production and infra-structure as the bill of quantities of the load-bearing structure is known, together with the main inputs and outlets for energy, water and waste. The finalization of the projects of individual houses can be completed with user participation.

The sketches are subsequently evaluated and sorted out according to the heuristics rules on the relation between indoor spaces and outdoor specifics: micro-climate conditions, vistas etc. Each alternative is supported by the explanation of the expert's (changeable) evaluation method of design decisions taken. For each alternative, ground-floor and first-floor plans, isometric or perspective surface-model views of the house with all possible roof solutions are presented on the screen and offered to be drawn on the plotter [Petrovic 87].

Evaluators: OYSTER. A prototype rule-production shell, allowing forward and backward chaining, and inclusion of new information in form of "What if" command, allowing a change of suppositions in course of the process and creation of alternatives. Tested on an number of "small" expert system applications, such as a simple "Aseismic design consultant for architects". At present, it is used for the additional assessment of attributes of GIMS objects, and for providing proofs of the object's attributes confirmation to the building regulations [Miric 90].

Graphic tools: Little 3-D Modeller. Program for interactive surface modelling of 3-d objects. Version 1.0 allows the basic operations with the primitives (translation, rotation, reflection). Although the program allows the general application, at present it is used as a graphic tool associated with GIMS objects [Svetel Ibid]

Hybrid tools. These unorthodox design tools based on the parallel distributed processing paradigm for knowledge processing known as a "neural net", [Rumelhart 86a] have been suggested as " tools for exploring associative reasoning in design" [Coyne 89]. They are significantly different from the AI methods, such as rules and knowledge-bases. Instead of having a separate knowledge-base which is interpreted by some control program, a connectionist system consists of a framework of large number of simple processing elements (nodes), and is both, the container of learned knowledge, and the processor of that knowledge.

These systems:

a) do not require the formal theory of design method what makes them applicable for non-algorithmic design procedures,
b) can "learn" from examples, thus making compensation for absence of theory, and
c) can generalize the learned knowledge on the basis of learned examples and draw "conclusions" about the new instances, providing they belong to the same class of objects.

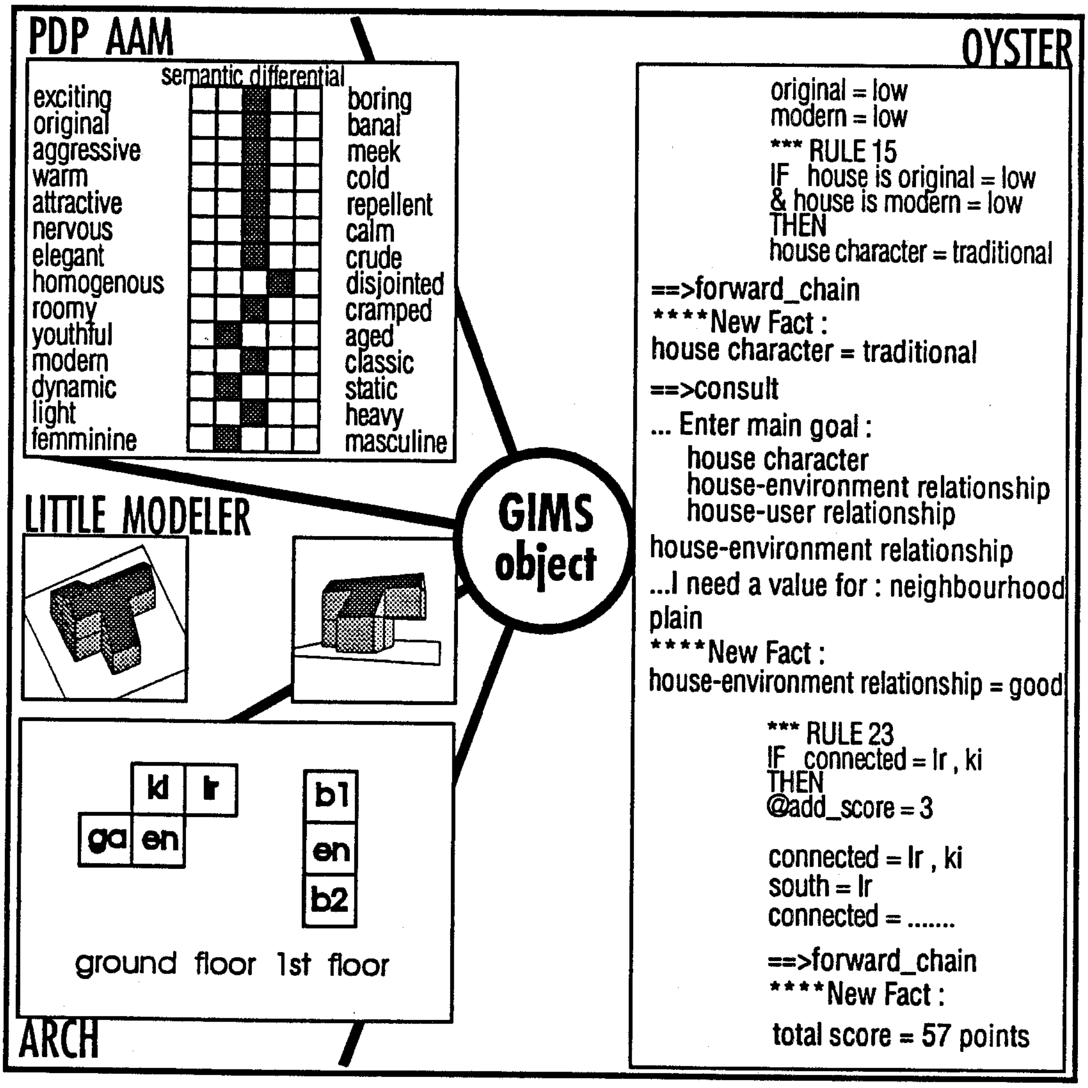
PDP-AAM: Parallel Distributed Processing - Analogical Architectural Modeller. The program uses the back-propagation algorithm [Rumelhart 86b], [Knight 89]. The "Little 3-d Modeller" and a semantic differential diagram are used as input/output devices. The tool enables the subjective assessment of object's form in two main ways: a) learning from examples and producing a semantic differential for an unknown GIMS building form, and b) learning from examples and producing a GIMS building form for the given semantic differential. The typical queries are: "What GIMS building is such that is associated with the given semantic differential?" and "What is the semantic differential of this GIMS building?" in view of the user's subjective preferences learned by the program. Despite the fact that the buildings or semantic differentials may be new to the program, the neural net shall produce the answers. The 3-d Modeller attached to the neural net enables the instant visualization of design proposals [Svetel 91].

Methods at Work

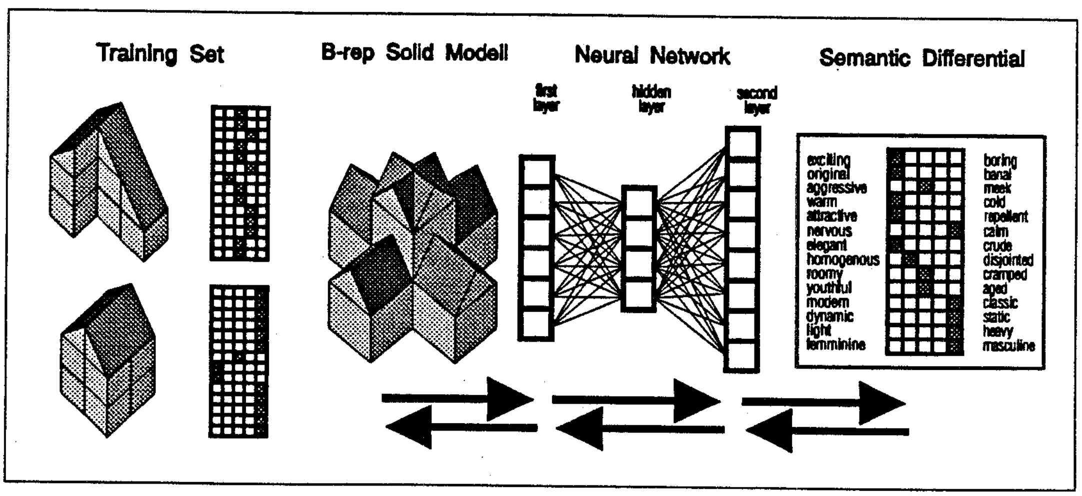
Designing GIMS houses. GIMS-EXPERT, as an automatic design system produces an answer that is one of the possible states of a two-storey object that belongs to the field constrained in 3 x 3 structural module of 4.2 m. The output is treated by the Little Modeller producing 3-d representation of the chosen object.

ARCH also produces GIMS houses, but in various modular structural spans. The rest of the produced design information is identical with that of GIMS-EXPERT.

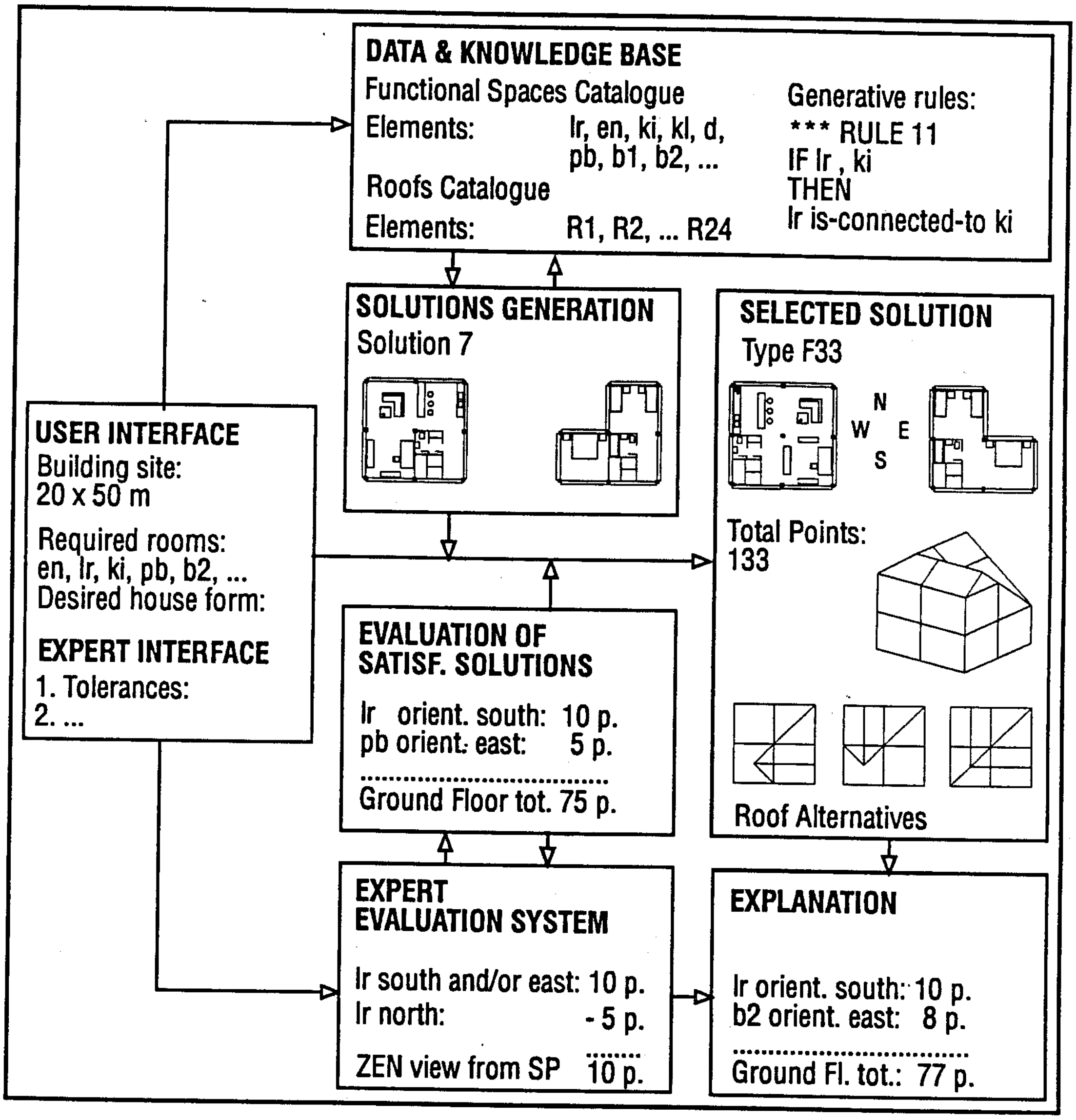
Semantic evaluation. The semantic evaluation of GIMS houses is made with the help of PDP-AAM. The obtained house is "evaluated" by the program containing someone's preferences; be it the designer himself, the investor, the user, or anyone else. The interpretation of the semantic differential is made by OYSTER. For example: an output from the semantic differential, "house is original = low" is taken as a fact. Rule no. 15 states that IF house is original = low, THEN, house character = traditional. In forward chain, starting from the facts, we would conclude that the house is traditional. In backward



Application of ARCH, OYSTER and PDP=AAM Programs



Functioning of PDP-AAM Program



GIMS-EXPERT Architecture

chain, we can find the values of the goals not obtained in forward chaining because the corresponding facts were missing. Starting with a goal statement: "house environment relationship", if the program finds the fact missing, it will ask for the value. If the value is as required (traditional), the program shall conclude that "house relationship" is good.

The application of the methods can also be in the opposite direction. We can start with the required house character, which shall define the semantic differential, and further propose the 3-d volume and the plans. We can start with any existing method, apply it individually, or combine it with any other method in any direction.

All described methods act on the object, instantiating the object data structure and using the knowledge base. It is the intention to keep both data and knowledge within the object model. An interface is used to connect the object and the method, and in fact provides the "fusion" of both.

Conclusions

The results of the first phase of the project produced a discrete design process, based on a set of independently applicable design tools related to the critical design decisions. As for now, the idea "works" but further elaboration is needed to enrich the possibilities, eliminate the deficiencies, and find more possible uses. The present applicability of the project results is primarily in the field of experimental research and education. The possibility of the subjective assessment of designed objects and easy introduction of new information indicates some new paths for further research in CAADesign methodology.

Acknowledgements

The project is financed by the research grants from the Scientific Council of the Republic of Serbia and the IMS Institute. The project director: Ivan Petrovic.

References

[Ashby 61] Ashby, W.R., *An Introduction to Cybernetics*, London, Chapman & Hall Ltd, 1961

[Banic 86] Banic, M., Grujic, B., Dimitrijevic, R., Petrovic, I. and Z. Lazovic , *IMS SYSTEM: A Prefabricated Skeleton Building System*, The IMS Institute, Belgrade, 1986

[Coyne 88] Coyne, R.D., Rosenman, M.A., Radford, A.D., Balachandran, M and J.S. Gero, *KNOWLEDGE-BASED DESIGN SYSTEMS*, Addison-Wesley Publishing Company, 1988

[Coyne 90] Coyne, R.R , "Tools for Exploring Associative Reasoning in Design" in McCulloch, M, Mitchell, W.J., & P. Purcell (Eds.): *THE ELECTRONIC DESIGN STUDIO*, The MIT Press, 1990

[Evans 82] Evans, B., Powell, J. & R. Talbot (Eds), *CHANGING DESIGN*, John Wiley & Sons , 1982

[Gero 85] Gero, J.S., "An overview of knowledge engineering and its relevance to CAAD", Proceedings, *CAAD FUTURES CONFERENCE*, Delft, 1985

[Goldberg 83] Goldberg, A., and P. Robson, *Smalltalk-80: The Language and its Implementation*, Addison-Wesley., 1983

[Hickling 82] Hickling, A., "Beyond a linear iterative structure" in Evans, B., Powell, J. & R. Talbot (Eds) *CHANGING DESIGN*, John Wiley & Sons pp 275-294, 1982

[Jones 63] Jones, J.C. and D.G. Thornley (Eds.) *CONFERENCE ON DESIGN METHODS*, Pergamon Press , 1963

[Jones 70] Jones, J.C., *DESIGN METHODS - SEEDS OF HUMAN FUTURE*, Wiley-Interscience, COID, London, 1970

[Knight 89] Knight, K., *A Gentle Introduction to Subsymbolic Computation: Connectionism for an AI Researcher*, Report CMU-CS-89-150, Carnegie-Mellon University , 1989

[Miric 90] Miric, A., Svetel, I. and I. Petrovic "OYSTER: A Shell", *THE IMS BULLETIN*, Vol 4, No 1, The IMS Institute, Belgrade, 1990

[Osgud 57] Osgud, C., Suci, G.J., and P.H. Tannenbaum, *THE MEASUREMENT OF MEANING*, Urbana: The University of Illinois Press, 1957

[Pentilla 90] Pentilla , H., Talonpoika, R., Hult, S., Tiainen, P. & Bjork, B.C., "RATAS Building Product Model: Experiences with Prototype" in Terai, T. (Ed), *Computer Integrated Construction*, CIB Proceedings Publication 138, CIB W78 & W74 Seminar, Tokyo, Japan, pp. 25 - 32, Sept. 1990

[Petrovic 87] Petrovic, I., Novkovic, M, Cubric, M., Svetel, I. and Z. Minjevic, " GIMS-EXPERT: Consultant for Design of Family Houses in GIMS Building System", PROCEEDINGS, III *INTERNATIONAL CONFERENCE ON RESEARCH AND DEVELOPMENT OF BUILDING TECHNOLOGIES IMS'87*, Belgrade, 227-245, 1987

[Petrovic 91a] Petrovic, I., "Subjective Object Modelling in CAAD: Problems and Approaches", Second International Workshop on *COMPUTER BUILDING REPRESENTATION FOR INTEGRATION*, Aix-les-Bains, France, 1991

[Petrovic 91b] Petrovic, I. , "Integrative CAAD Systems", CIB W78 Seminar *COMPUTER INTEGRATED FUTURE*, Eindhoven. Holland

[Rumelhart 86] Rumelhart D.E. and J.L. McClelland (Eds), *Parallel Distributed Processing*, The MIT Press, Cambridge, MA, 1986

[Svetel 90] Svetel, I., "ARCH: A Program for Synthesizing of Architectural Plans", *THE IMS BULLETIN*, Vol 4, No 1, The IMS Institute, Belgrade, 1990

[Svetel 91] Svetel, I., *PDP-AAM Manual*, The IMS Institute Report, 1991

[Tolman 90 Tolman, F.P., Kuiper, P., & Luiten, G.T., "Product Modeling at Work",in Terai, T. (Ed), *Computer Integrated Construction*, CIB Proceedings Publication 138, CIB W78 & W74 Seminar, Tokyo, Japan, pp. 25 - 32, Sept. 1990

[Wright 67] Wright, G.H. von , "The Logic of Action - A Sketch": in Resher, N. (Ed): *THE LOGIC OF DECISION AND ACTION*, Univ. of Pittsburgh Press, pp 121-136, 1967