Expert Systems with Applications, Vol. 22, No. 2, 2002, pp. 169-178

EXPERT SYSTEM APPLACTION ON PRELIMINARY DESIGN OF WATER RETAINING STRUCTURES

K.W. Chau
Department of Civil & Structural Engineering
The Hong Kong Polytechnic University, Hong Kong

F. Albermani
Department of Civil Engineering
University of Queensland, Australia

ABSTRACT
Design of liquid retaining structures involves many decisions to be made by the designer based on rules of thumb, heuristics, judgment, code of practice and previous experience. Various design parameters to be chosen include configuration, material, loading, etc. A novice engineer may face many difficulties in the design process. Recent developments in artificial intelligence and emerging field of knowledge-based system have made widespread applications in different fields. However, no attempt has been made to apply this intelligent system to the design of liquid retaining structures. The objective of this study is, thus, to develop a knowledge-based system (KBS) that has the ability to assist engineers in the preliminary design of liquid retaining structures. Moreover, it can provide expert advice to the user in selection of design criteria, design parameters and optimum configuration based on minimum cost. The development of a prototype KBS for the design of liquid retaining structures (LIQUID), using blackboard architecture with hybrid knowledge representation techniques including production rule system and object-oriented approach, is presented in this paper. An expert system shell, Visual Rule Studio, is employed to facilitate the development of this prototype system.

INTRODUCTION

In the past decade, the potential of artificial intelligence (AI) techniques for providing assistance in the solution of engineering problems has been recognized. Knowledge-based systems (KBS) are considered suitable for solving problems that demand considerable expertise, judgment or rules of thumb. As a result of years of research in artificial intelligence, KBS have emerged as a most promising application covering a wide range of applications (Adeli & Hawkins 1991, Chau 1992, Chau & Chen 2001, Chau & Ng 1996, Chau & Yang 1992, Chau & Yang 1994, Chau & Zhang 1995, Lin & Albermani 2001, Tah & Carr 2001, Waheed & Adeli 2000). KBS have developed into practical problem solving tools that can reach a level of performance comparable to that of a human expert in some specific problem domains. All these applications can be broadly classified into the following categories: diagnosis; design; data interpretation; planning; and education. Areas of early applications of KBS technology include medical diagnosis, mineral exploration and chemical spectroscopy. Many researchers are developing programs that borrow AI concepts to automate common engineering analyses.

AI has made widespread applications in different fields. In structural engineering field, there is a need to develop programming environments that can incorporate engineering judgment along with algorithmic tools. (Kitzmiller & Kowalik 1987) However, no attempt has been made to apply this intelligent system to the design of liquid retaining structures. The objective of this work is, thus, to develop a KBS prototype that has the ability to assist engineers in the design of liquid retaining structures and to provide expert advice to the user in selection of design criteria, design parameters and optimum structural section based on minimum cost. The development of a KBS in design of liquid retaining structures (LIQUID), using blackboard architecture with hybrid knowledge representation techniques including production rule system and object-oriented approach, is presented. An expert system shell, Visual Rule Studio, is employed to facilitate the development of this prototype system. It is a coupled system in which AI-based symbolic processing is combined with the traditional numerical processing. The KBS developed is based on British Standards Code of Practice BS8007: 1987: Design of concrete structures for retaining aqueous liquids (British Standards Institution 1987).

KBS are interactive computer programs that mimic the decision making and reasoning processes of human experts in solving a specific complex problem, by providing expert advice, answering questions, and justifying their conclusions. Figure 1 shows the schematic view of a KBS. Three basic components are knowledge base, context and inference mechanism. The knowledge base is a collection of general facts, rules of thumb and causal models of the behavior specific to the problem domain. The context contains facts that reflect the current state of the problem, constructed dynamically by the inference mechanism from the information provided by the user and the knowledge base. The inference mechanism guides the decision making process by using the knowledge base to manipulate the context.

In addition, three other desirable components are a user interface, an explanation facility and a knowledge acquisition module. The interface is responsible for translating the interactive input as specified by the user to the form used by the KBS. The explanation module provides explanations of the inferences used by the KBS: why a certain fact is requested and how a conclusion was reached. The knowledge acquisition module serves as an interface between the experts and the KBS and provides a means for entering domain specific knowledge into the knowledge base.

For a conventional program, the order of execution is predetermined. Updates need considerable effort other than by the programmer. The programmer must ensure completeness and uniqueness of the solution. The user, perceiving the program as a blackbox, has no idea to why certain results have been produced. On the contrary, KBS eliminate the above impediments by partitioning between the knowledge base and the control strategy. This allows for incremental addition of knowledge without manipulation of the overall program structure and hence the programmer need not guarantee completeness. By ranking several alternatives with inexact inference methods, several solutions with different confidence factors can be provided for a particular input condition. The user can also question the results through the explanation module.

Several approaches for declarative representation of knowledge are available in the AI literature, i.e. rule-based production system, frames and object-oriented programming. A production system is a collection of rules and is believed to be good at describing heuristic knowledge. A frame system is suitable for a complex and rich representation of knowledge. Object-oriented programming concept is used, in which a computer program consists of a number of independent objects that process jobs by exchanging information they need via messages. It seems a good idea to utilize a hybrid approach combining two representations to solve structural design problems.

Several tools are available for developing a knowledge-based system, i.e. traditional programming languages, high-level tools or expert system shells. The programming language designed for AI (LISP or PROLOG) can be used, but the programming effort required will be tremendous. High-level tools can provide an integrated knowledge engineering environment combining features of the AI languages appropriately and efficiently. Typical examples of this type of approach are ART (Clayton 1985) and KEE (Intellicorp 1986). An expert system shell provides the skeleton of a knowledge-based system incorporating inference engine, user interface and knowledge storage medium. Once the domain knowledge is filled in, it can perform the desired function of a knowledge-based system with the specified inference strategy and representation method.

KNOWLEDGE ON DESIGN OF LIQUID RETAINING STRUCTURES

Different Configurations and Design Parameters

Liquid retaining structure is a structure which is designed and constructed to retain aqueous liquid. It is subject to lateral liquid pressure and earth pressure when it is located underground. In Hong Kong, most of these structures are constructed using reinforced concrete with design life of 50 years. Crack widths need to be checked to ensure impermeability of concrete and prevention from corrosion of reinforcement.

Normally, two kinds of classification are used regarding liquid retaining structures, i.e. according to the shape or the location. Based on the shape, it is classified as rectangular, circular or polygon. Based on its location, it is classified as underground or above ground.

Compared with circular tank structure with the same width, rectangular liquid retaining structure has larger volume. However, because of stress concentration at corners, rectangular structures will be more vulnerable to failure. Since a circular structure can be constructed monolithically without any construction joints, it has better strength quality. With precise structural analysis, circular structure has a better control in deflection, crack width, bending moment resistance, axial compression resistance, and shear resistance than rectangular structure. Polygon liquid retaining structure is usually used for aesthetic purposes, such as a fountain in a garden and the retaining height is usually not very high. Its major design consideration is on crack width to ensure its impermeability and is seldom used in industrial or domestic fields.

Underground liquid retaining structure usually has larger base area, which cannot easily be supported by structure above the ground. In some cases, the tank is connected to underground pipe network system, in order to reduce maintenance cost, it will be constructed underground to suit the invert level of the pipe network system. The underground structure is mainly subjected to lateral earth pressure and lateral water pressure, which is due to the underground water table. Besides checking structural failure mode, bearing capacity of soil and settlement of structure also need to be checked. If the soil bearing capacity does not satisfy the requirement, pile or raft foundation will be required. Liquid retaining structure above the ground is mainly subject to liquid pressure due to its own retaining liquid. The structure can either rest on ground concrete slab immediately or rest on supports. Earthquake loading is another concern especially for fire prevention tanks. However this type of loading will not be considered in this paper.

There are a great variety of factors affecting the decision in selecting design criteria and design method. These factors are: dimensions, location, ground condition, support condition, groundwater conditions, aesthetic properties, design life, exposure condition, usage, roofing, availability of construction materials. Liquid retaining structures, like any other engineering structures, should not fail to satisfy any of its performance criteria. According to the Code of Practice BS8007, the two main classes of limit state to be considered are serviceability limit state and ultimate limit state. Serviceability limit state is design against deflection and crack width.

Serviceability Limit State

Cracks in reinforced concrete structures cannot be avoided. Moreover, concrete expands and contracts with the increase in temperature during hydration of cement and the subsequent fall in temperature. The effects of thermal contraction and drying shrinkage may be controlled by the provision of reinforcements and movement joints. The design calculations of both the serviceability limit state of cracking due to thermal and moisture effects and flexural effects can be tedious and time consuming.

In the design of liquid-retaining structures it is essential to restrict the width of cracks in the structure for direct tension and flexure or restrained temperature and moisture effects. This assumes that, if cracks do not exceed maximum design surface crack widths, they can heal autogeneously. Normal crack width control is 0.2mm while, for severe cases, allowable crack width is 0.1mm.

Appendix A of BS 8007 provides calculations of minimum reinforcement, crack spacing and crack widths in relation to temperature and moisture effects. Two criteria should be met. First of all, minimum reinforcement required to control the early thermal and shrinkage cracking (within three days) is related to the ratio between the direct tensile strength of immature concrete and the characteristic strength of reinforcement. Moreover, it should also be greater than the reinforcement ratio ρ, based on area of surface zone, determined from the following equation:

$$\rho = (f_{ct}/f_b)(\phi/2 W_{max})(\alpha/2)(T_1+T_2) \qquad (1)$$

where f_{ct} = direct tensile strength of immature concrete
f_b = average bond strength between concrete and steel
ϕ = bar diameter
W_{max} = maximum design surface crack width
α = coefficient of thermal expansion of mature concrete
T_1 = temperature rise due to hydration of cement
T_2 = temperature fall due to seasonal variation

Appendix B of BS 8007 provides calculations of crack widths in mature concrete under structural loading. The service flexural capacity of different slab thickness and reinforcement arrangements under differing conditions of crack width limitation, concrete strength and cover differ. The design is based on the elastic theory for cracked section. Based on the equilibrium of forces, the neutral axis position is determined to be a function of the modular ratio and area of reinforcement of a given section and is independent of the applied moment. From BS 8007 Appendix B, the crack width W is determined from the following equation:

$$W = \frac{3a_{cr}\varepsilon_m}{1+2(\frac{a_{cr}-c}{h-x})} \qquad (2)$$

where ε_m is the average strain for calculation of crack width allowing for concrete stiffening effect
a_{cr} is the distance from the point considered to the surface of the nearest longitudinal bar
c is the minimum cover to the tension reinforcement
x is the neutral axis position
h is the slab thickness

In calculating crack width, BS 8007 gives allowance to the stiffening effect of concrete between cracks

$$\varepsilon_m = \varepsilon_1 - \varepsilon_2 \qquad (3)$$

where ε_1 is the average strain for calculation of crack width
ε_2 is the strain due to the stiffening effect of concrete between cracks

For crack width W of 0.2 mm

$$\varepsilon_2 = \frac{b(h-x)^2}{3E_s A_s (d-x)} \qquad (4)$$

For crack width W of 0.1 mm

$$\varepsilon_2 = \frac{1.5b(h-x)^2}{3E_s A_s (d-x)} \tag{5}$$

where d is the effective depth of section
b is the width of section
E_s is moduli of elasticity of reinforcement
A_s is area of reinforcement.

Ultimate Limit State

The ultimate flexural capacity of a range of slab thickness and reinforcement arrangements under differing conditions of service and concrete strength and cover is calculated and tabulated in the KBS based on British Standards Code of Practice BS8110: 1985: Structural use of concrete (British Standards Institution 1985). Concrete stress is limited to 0.45 of the characteristic strength whilst reinforcement stress is limited to 0.87 of the yield strength. Besides, the neutral axis position and the lever arm are limited to 0.5 and 0.95 of the effective depth respectively. Part of a sample table showing ultimate moment capacity and unit cost of section (for concrete slab thickness 200-225mm, grade 40/20 concrete, high yield reinforcement, concrete cover 40mm, unit cost of concrete \$500/m^3, unit cost of reinforcement \$3000/tonne) is shown in Table 1. Since all these design parameters can vary at different design situations and an enormous amount of data are involved, a Microsoft Access database file *Section* is used to store the table. Of course, in order to minimize the size of this database, some heuristics are used to limit the choice of some design parameters to only practical values. For examples: the concrete slab thickness is limited to 200, 225, 250, 275, 300, 350,400, 450, 500, 600, 700, 800, 900 and 1000 mm; allowable reinforcement sizes are 10, 12, 16, 20, 25, 32 and 40 mm; concrete cover is limited to 40, 50, 60 and 75 mm.

Structural Optimization

The unit cost per meter length of the concrete section (consisting of concrete area and reinforcement area) can be calculated and tabulated in the KBS. Each concrete section will have its service moment capacity, ultimate moment capacity, ultimate shear capacity, crack width limitation. Structural optimization can be effected so as to achieve the minimum unit cost per meter length of concrete section.

KBS FOR DESIGN OF LIQUID RETAINING STRUCTURES (LIQUID)

Blackboard Architecture

The blackboard architecture is intended to support development of systems in domains characterized by interaction between diverse sources of knowledge and hence provides a framework for integrating knowledge from several sources. The blackboard serves as a global data structure, which facilitates this interaction. A common analogy may be made to problem-solving in domains where a number of experts in different areas of specialties co-operate over the solution which any one of them could never achieve alone. In order to facilitate this process, they agree to use a blackboard to post (or write) any partial result they can contribute separately. Each expert takes turns to write on the blackboard and, in case more experts wish to write simultaneously, the conflict is resolved by some pre-defined strategy.

The blackboard architecture has been successfully used in solving a wide range of tasks, such as speech recognition, signal processing, and planning. (Engelmore & Morgan 1988) A blackboard system consists of a number of knowledge sources that communicate through a blackboard and are controlled by an inference mechanism. Figure 2 shows the architecture of the current blackboard system. The main components of a typical blackboard system are knowledge sources, entries, blackboard, and inference mechanism.

The knowledge base consists of a number of knowledge sources (KSs), which contain the knowledge. These knowledge sources are independent chunks of knowledge and do not directly communicate with each other. Instead, they participate in the problem solving process by creating entries in a global database – the blackboard. Knowledge modules look at the blackboard to see if suitable data is present to trigger their execution. If they are selected, execution results in new or altered data on the blackboard, which will then trigger other knowledge modules. Solving a problem using blackboard architecture is based on cooperation of the knowledge modules present. Each knowledge source consists of a condition-action pair. Actions are executed whenever the conditions are satisfied in the blackboard.

Entries are the immediate results produced by the system. In a typical system, each entry has a certainty factor as well as a specification. The blackboard or context consists of the information or entries generated by the knowledge sources during the problem solving process. It is organized into a number of levels each representing different aspects or stages of the solution process. Each level contains objects and attributes that are important to the representation of the problem. Normally, knowledge sources are specific to certain levels in the blackboard, i.e., the activation of a certain knowledge sources depends on the entries generated at certain levels in the blackboard, while the actions of the knowledge source modify entries at some other level. The main units in the blackboard are hypotheses, which at various levels are related through structural relationships.

Design context and the processes in the design, both represented as objects, are organized separately. Declarative knowledge, divided into two groups (Design Status and Design Concept), is stored in the blackboard. Design Status contains attributes that represent indicators used to keep information of the current stage of every design entities. The processes in the design are represented in the knowledge source level. Multi-formalism approach is employed consisting of objects, rules, procedural methods, extensive numerical algorithm and databases.

Usually, in a typical blackboard model, the inference mechanism consists of the agenda (or scheduler) and the monitor. The agenda keeps track of all the events in the blackboard and calculates the priority of execution for knowledge sources that were generated as a result of the activation of other knowledge sources. The monitor takes the element with the highest priority and executes it. However, there is no fixed agenda and monitor in the current blackboard model. Since a scheduler is not applied, control of design process is performed by Process Control Knowledge modules in knowledge sources. These modules act opportunistically upon being triggered by user or situation during the design process. Since design steps in this system are explicitly seen on the main screen display, the sequence of design processes are primarily selected by the user. However, validity of the sequencing is checked by Process Control Knowledge modules in the knowledge source. Without fixed agenda, the user is free to change input parameters and check intermediate results given by the system during the design session.

Because of its modularity, the blackboard architecture enables easy incremental development of a software system. Developers can integrate different methods of knowledge representation in a single system because of the modularity of knowledge sources.

Features of LIQUID

To facilitate development of the knowledge-based system, expert system shell containing specific representation methods and inference mechanisms is employed. The knowledge base and explanation facility of the system have been developed using a commercially available expert system shell called Visual Rule Studio which is a hybrid application development tool that integrates object-oriented techniques and expert system technology with traditional, procedural programming. (RuleMachine Corporation 1998) Visual Rule Studio installs as an integral part of Microsoft Visual Basic 5.0 as an ActiveX Designer. By isolating rules as component objects, separate from objects and application logic, the shell allows developers to leverage the proven productivity of today's component oriented development tools, such as Visual Basic. Rule development becomes a natural part of the component architecture development process. It produces objects that can interact with virtually any modern development product.

Objects are used to encapsulate knowledge structure, procedures, and values. An object's structure is defined by its class and attribute declarations within a RuleSet. Object behavior is tightly bound to attributes in the form of facets, methods, rules, and demons. Figure 3 shows the structure of Visual Rule Studio components. Each attribute of a class has a specific attribute type. The attribute types are compound, multicompound, instance reference, numeric, simple, string, interval, and time. Facets provide control over how the inference engines process and use attributes. Methods establish developer-defined procedures associated with each attribute. The set of backward-chaining rules that conclude the same attribute is called the attribute's rule group. The set of forward-chaining demons that reference the same attribute in their antecedents is called the attribute's demon group.

LIQUID combines expert systems technologies, object-oriented programming, relational database models and hypertext/graphics in Microsoft Windows environment. By defining various types of windows as different classes, such as Check Box, Option Button, List Box, Command Button, Text Box, etc., they can inherit common characteristics or/and possess their own special properties.

The knowledge used has been acquired mostly from written documents such as code of practice, textbooks and design manuals and complemented by experienced engineers involved with the design of liquid retaining structures. The domain knowledge is translated into procedures and methods using object-oriented representation. The system can be compiled and encrypted to create a run-only system. This run-only system can be installed on a microcomputer for office use. The user can always overrule any design options and recommendations provided by the system. In other words, it plays the role of a knowledgeable assistant only.

The input data provided by the user will be rejected if it is not within the range specified. It can explain its line of reasoning for obtaining an answer. It provides in multi-window graphics text display where graphic images are combined with valuable textual information. This kind of intelligent graphics is extremely valuable to structural designers because it enhances their confidence in the design provided by the knowledge-based system.

The system offers a friendly user interface. Mouse, keyboard, or both can be used to navigate the application. The use of a mouse or other pointing device makes the data entry a simple task even for novice computer users. As such, users simply point and click their way through the process to appreciate the dynamic behavior of the system. Input data entry is kept at minimum. Input data are provided by the user mostly through selection of appropriate values of parameters from the menus and providing answers to the queries made by the system.

The inference engines control the strategies that determine how, from where, and in what order a knowledge base draws its conclusions. These inference strategies model the reasoning processes an expert uses when solving a problem. The Process Control Knowledge module involves metalevel knowledge which establishes the problem solving strategy and controls the execution of the Design Knowledge modules. It evaluates the Design Status and decides what action should be performed mainly in data-driven forward chaining mechanism. The knowledge representations of the Design

Knowledge modules, however, need both forward and backward chaining inference mechanism to arrive at the solution.

EXAMPLE OF CONSULTATION SESSION

In order to demonstrate how LIQUID assist engineer in the preliminary design of liquid retaining structure, a sample run is demonstrated in this section. The purpose of the design is to find a feasible optimum structural section in term of minimum unit cost per meter length, based on the standard slab thickness and reinforcement size as well as bar spacing. Little explanation facility is required since each screen display was designed to be user-friendly to follow.

Upon execution of the package LIQUID, the main menu screen is displayed as shown in Figure 4. A number of command buttons representing different design functions, which are grouped into three groups: Preliminary Design, Detailed Design, and Design Summary, can be activated by a mouse. Alternatively, a pull-down menu can also be employed for the same purpose. Tool-tip-text, giving detailed description on the function of the command button, is also available when the mouse pointer is dragged and placed over the button. It should be noted that only a few buttons, including Structural specification Button, Help Button, Exit Button, Restart Button, are enabled and all the others requiring the completion of some pre-requisite processes are disabled.

To start the preliminary design of liquid retaining structure in this example, Structural specification in Preliminary design has to be selected. Structural specification display is then popped up as shown in Figure 5. The user is required to input the basic spatial requirements, which are volume, height, shape, location, exposure condition, density of liquid and, if the shape is rectangular, the width/breadth ratio. The validity of the data input will be checked and, if any data are not within normal limits, warning message will be displayed to guide the correct input. Simple structural analysis program will be evoked to determine the maximum service and ultimate moments, shear force, deflection, etc. By selecting Search for configuration Button, LIQUID will search for 15 feasible alternative sections from the database *Section* which will be shown in Figure 6.

Back to the main menu, the next process is to select Alternative Evaluation where the system will evaluate each of these 15 recorded feasible configuration by using engineering heuristic based on span/depth ratio, unit cost and stability. As shown in Figure 6, 15 best alternative sections will be tabled in order of minimum unit cost per meter length which can satisfy all the criteria including crack width and strength. The computation of unit cost per meter length includes only the material cost of concrete and reinforcement and excludes cost of formwork and excavation. Here, it must be noted that it is not necessary for the user to adopt the best alternative proposed by the system, which has the minimum unit cost per meter length. The user can override the system’s decision by clicking the User’s section Option Button and choose any among the 15 proposed alternatives by pointing it in the table. The user can also view the database of sectional properties as well as make changes to the default parameters by selecting Sectional property Button.

Screen showing details of sectional properties with default material properties, concrete cover and unit cost of materials is shown in Figure 7. The user can change any of these default parameters from a list of available choices. The database will then be updated, reflecting the corresponding changes, to give the revised service moment capacity, ultimate moment capacity, ultimate shear capacity and unit cost per meter length.

When the Crack width checking Button is selected from the main menu, details of the chosen parameters, the actual crack width and the statement on whether or not the crack width criterion is satisfied will be shown in Figure 8. In this example, the computed actual crack width under the current configuration is 0.17 mm which is less than the designed crack width of 0.2 mm and hence the crack width criterion is satisfied under the provision of BS 8007.

Further Development

In order to improve the accuracy of structural analysis results, a rigorous commercially available finite element analysis package can be evoked as an external program in the detailed design stage and integrated into knowledge-based system. A model generation program will be required to generate files containing information such as nodal coordinates, member connectivity, support condition and loading specification which are ready to be analyzed by the analysis program.

CONCLUSIONS

A great variety of factors affect decisions in selecting design criteria and design parameters. An integrated microcomputer knowledge-based system (LIQUID), which provides much information necessary to make decisions, was developed to combine expert knowledge with conventional algorithmic programming and relational databases. Advice of structural optimization can be effected in the preliminary design stage so as to achieve the minimum unit cost per meter length of concrete section. It has been demonstrated that the hybrid knowledge representation approach combining production rule system and object-oriented programming technique to the design of liquid retaining structures is possible with the implementation of blackboard system architecture. It is appropriate to integrate algorithmic and symbolic programming on structural design into a single computer-aided environment running under a Windows platform. The educational spin-off of knowledge-based systems in training novice engineers or in transferring knowledge cannot be overemphasized.

REFERENCES

Adeli, H. and Hawkins, D.W. (1991) A Hierarchical Expert System for Design of Floors in Highrise Buildings, *Computers & Structures*, 41(4): 773-778.

Engelmore, R. and Morgan, T. (1988) *Blackboard Systems*, Addison_Wesley, Wokingham.

British Standards Institution (1987) *BS 8007:Design of Concrete Structures for Retaining Aqueous Liquids*, British Standards Institution, London.

British Standards Institution (1985) *BS 8110:Structural Use of Concrete*, British Standards Institution, London.

Chau, K.W. (1992) An Expert System for the Design of Gravity-type Vertical Seawalls, *Engineering Applications of Artificial Intelligence*, 5(4): 363-367.

Chau, K.W. and Chen, W. (2001) An Example of Expert System on Numerical Modelling System in Coastal Processes, *Advances in Engineering Software*, 32(9): 695-703.

Chau, K.W. and Ng, Vitus (1996) A Knowledge-Based Expert System for Design of Thrust Blocks for Water Pipelines in Hong Kong, *Journal of Water Supply Research and Industry - Aqua*, 45(2): 96-99.

Chau, K.W. and Yang, Wen-wu (1992) A Knowledge-Based Expert System for Unsteady Open Channel Flow, *Engineering Applications of Artificial Intelligence*, 5(5): 425-430.

Chau, K.W. and Yang, Wen-wu (1994) Structuring and Evaluation of VP-Expert Based Knowledge Bases, *Engineering Applications of Artificial Intelligence*, 7(4): 447-454.

Chau, K.W. and Zhang, X.N. (1995) An Expert System for Flow Routing in a River Network, *Advances in Engineering Software*, 22(3): 139-146.

Clayton, B.D. (1985) "ART-Programming Tutorial, Vols 1-3", Inference Corporation, Los Angeles, California, U.S.A.

IntelliCorp (1986) "KEE Software Development User Manual", 3rd ed.

Kitzmiller, C.T. and Kowalik, J. S. (1987) Coupling Symbolic and Numeric Computing in Knowledge-Based Systems, *AI Magazine*, Summer: 5-90.

Lin, S. and Albermani, F. (2001) Lattice-Dome Design Using a Knowledge-based System Approach, *Computer-aided Civil and Infrastructure Engineering*, 16(4): 268-286.

RuleMachines Corporation (1998) Visual Rule Studio Developer's Guide, Indialantic.

Tah, J.H.M. and Carr, V. (2001) Knowledge-based Approach to Construction Project Risk Management, *Journal of Computing in Civil Engineering, ASCE*, 15(3): 170-177.

Waheed, A. and Adeli, H. (2000) A Knowledge-based System for Evaluation of Superload Permit Applications, *Expert Systems with Applications*, 18(1): 51-58.

Slab thickness (mm)	Bar size (mm)	Bar spacing (mm)	Concrete cover (mm)	Bar area percentage (%)	Ultimate moment (kNm/m)	Ultimate shear (kN/m)	Unit cost ($/m)
200	10	125	40	0.41	37	107	130
200	12	150	40	0.49	44	114	136
200	10	100	40	0.51	46	116	137
225	10	125	40	0.35	43	114	142
200	16	225	40	0.59	51	120	142
200	12	125	40	0.59	52	121	143
200	16	200	40	0.66	57	125	147
225	12	150	40	0.42	51	121	148
225	10	100	40	0.44	54	123	149
200	12	100	40	0.73	64	130	153
200	16	175	40	0.76	64	130	154

Table 1. Part of sample table showing ultimate moment capacity and unit cost of section (for concrete slab thickness 200-225mm, grade 40/20 concrete, high yield reinforcement, concrete cover 40mm, unit cost of concrete $\$50/m^3$, unit cost of reinforcement $3000/tonne)

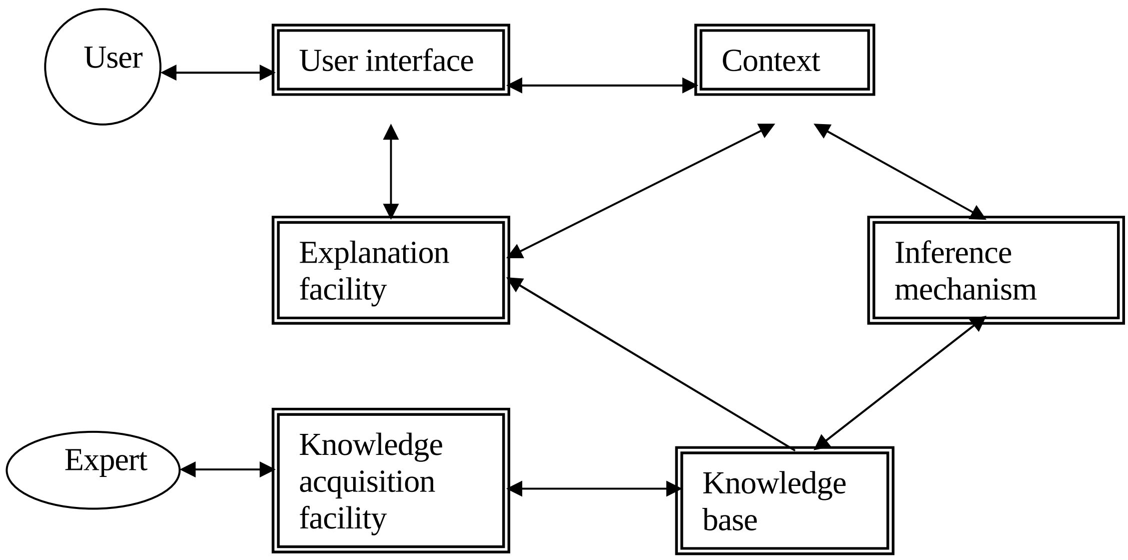


Figure 1 Schematic View of a Knowledge-based System

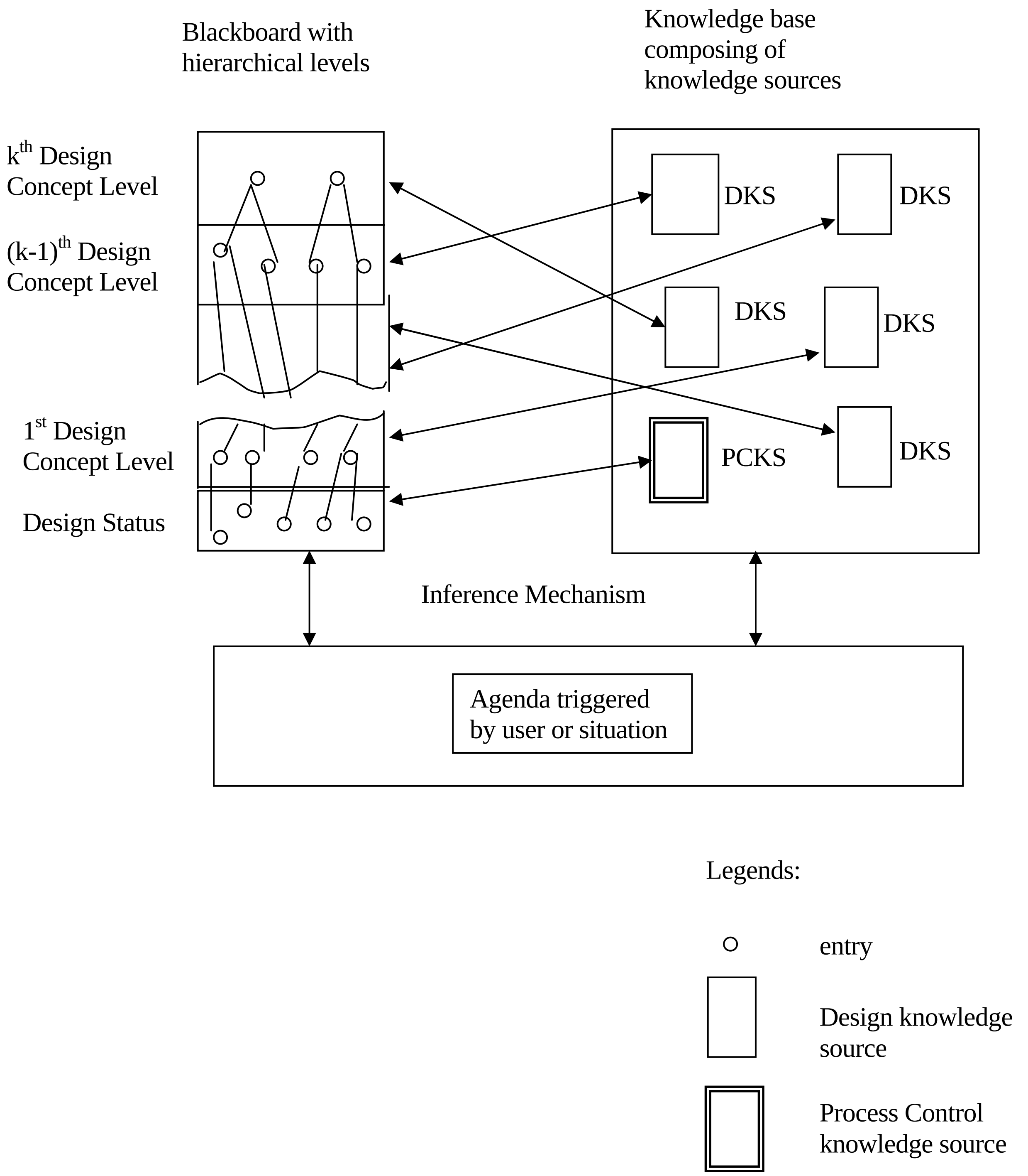


Figure 2 Blackboard Architecture of the Current KBS

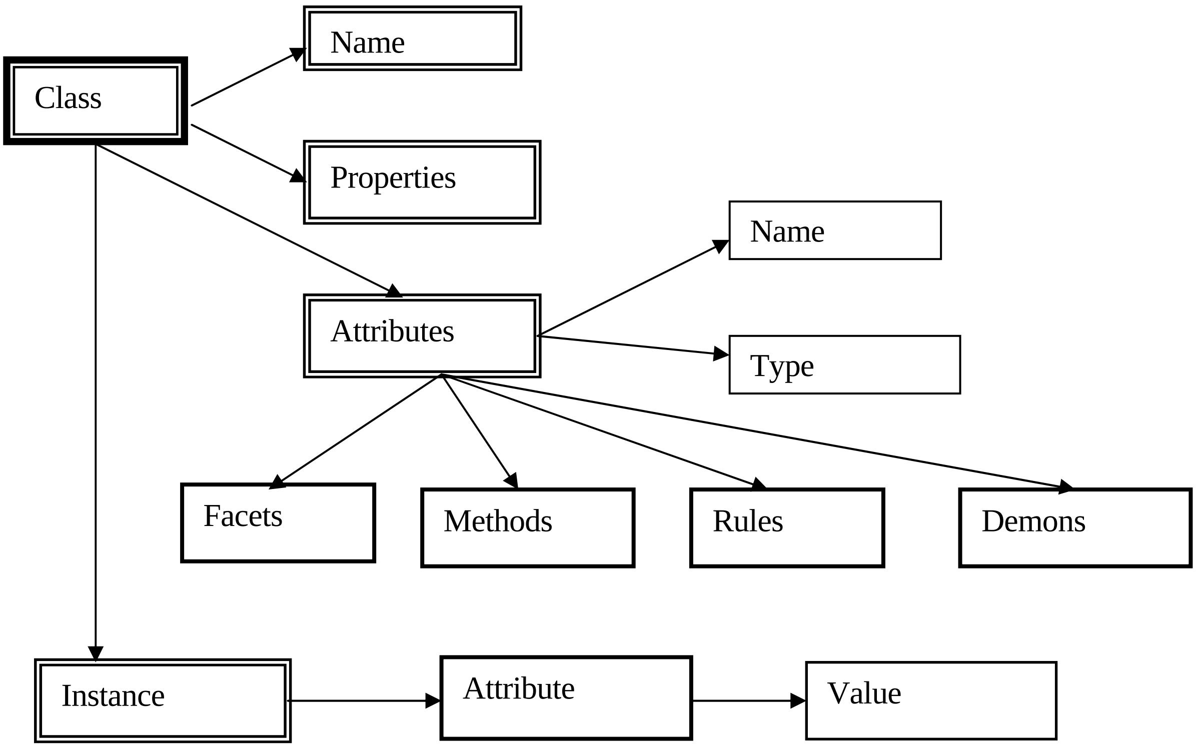


Figure 3 Structure of Visual Studio Components

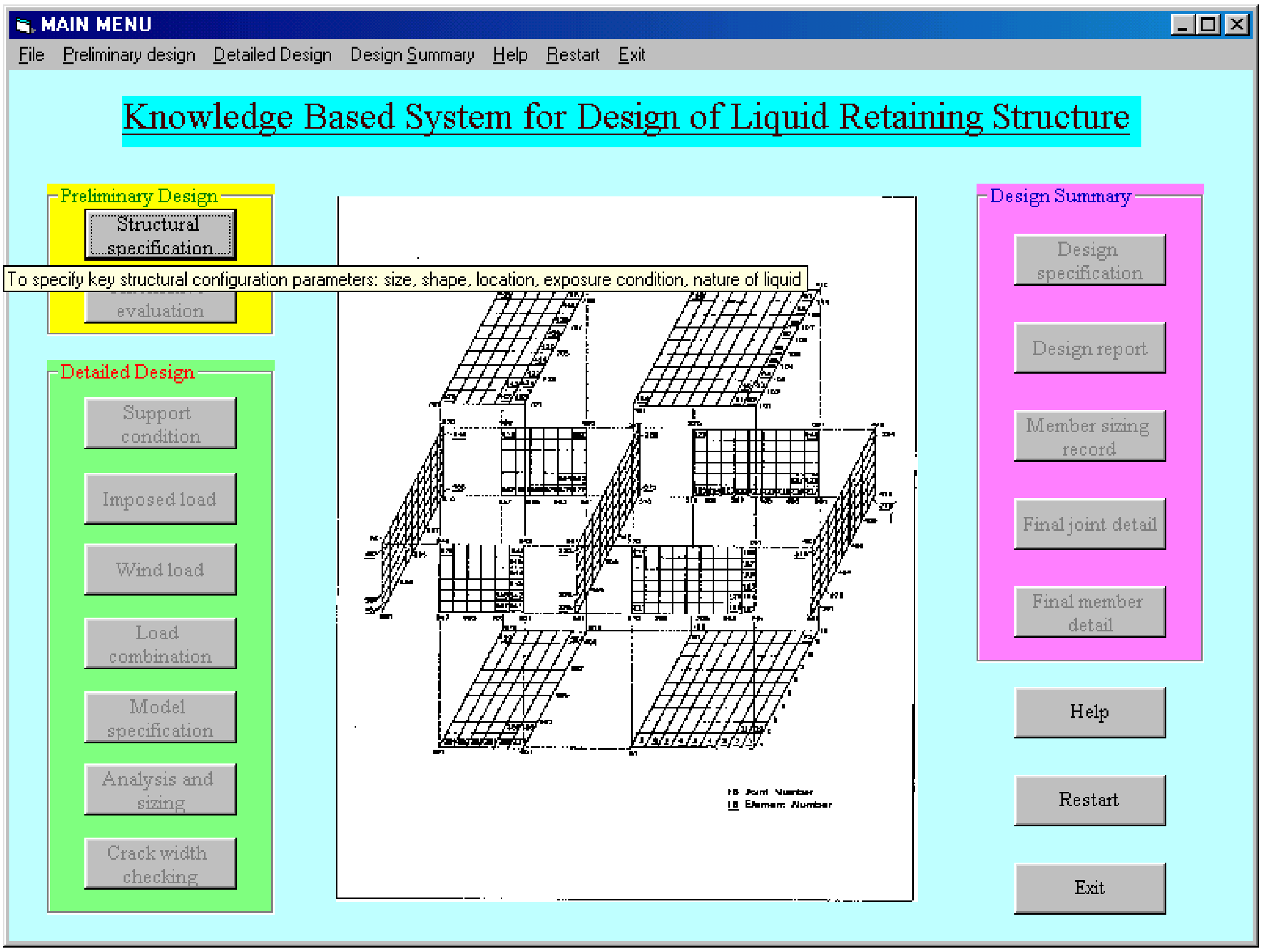


Figure 4 Main menu of the knowledge-based system

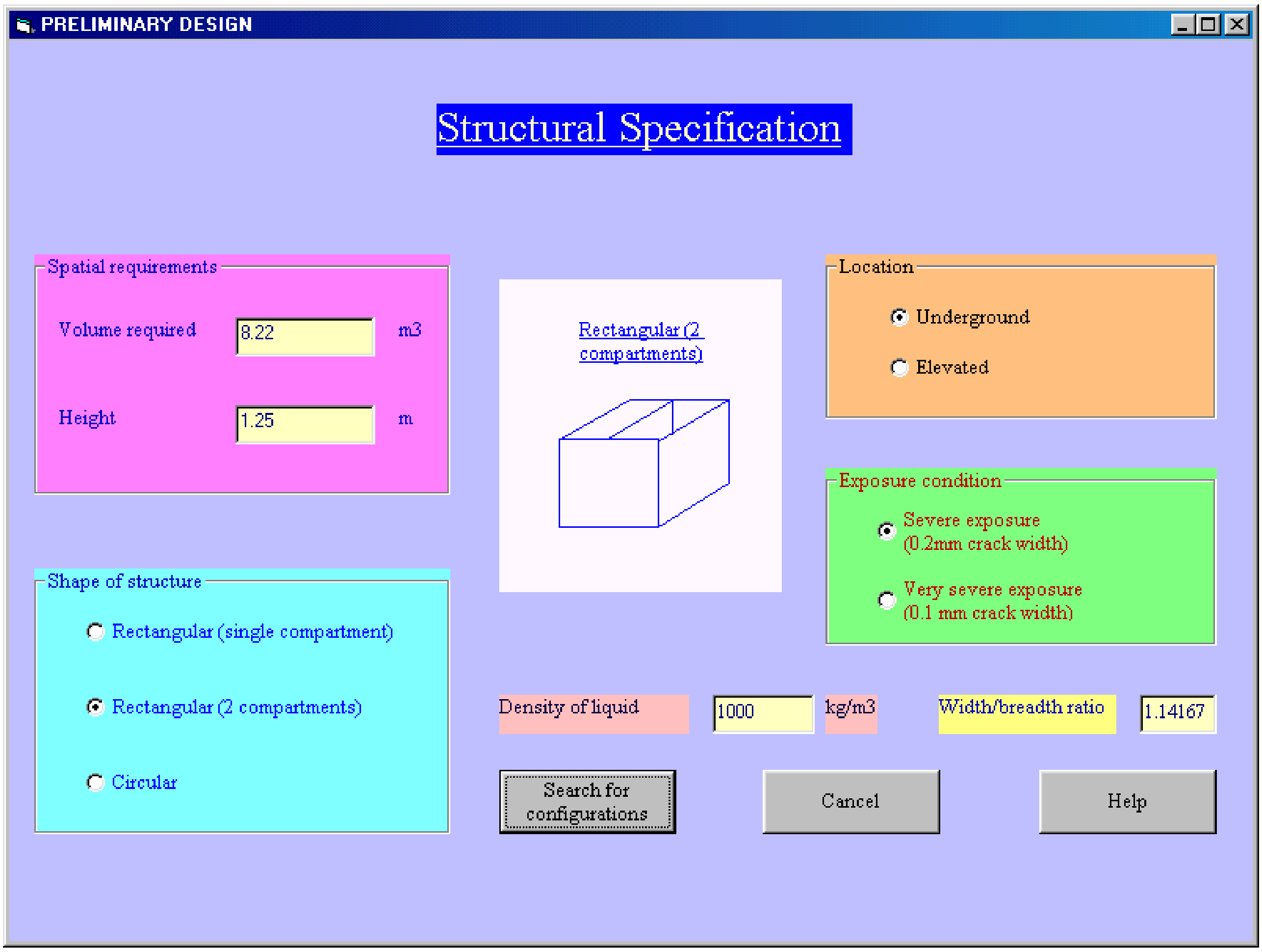
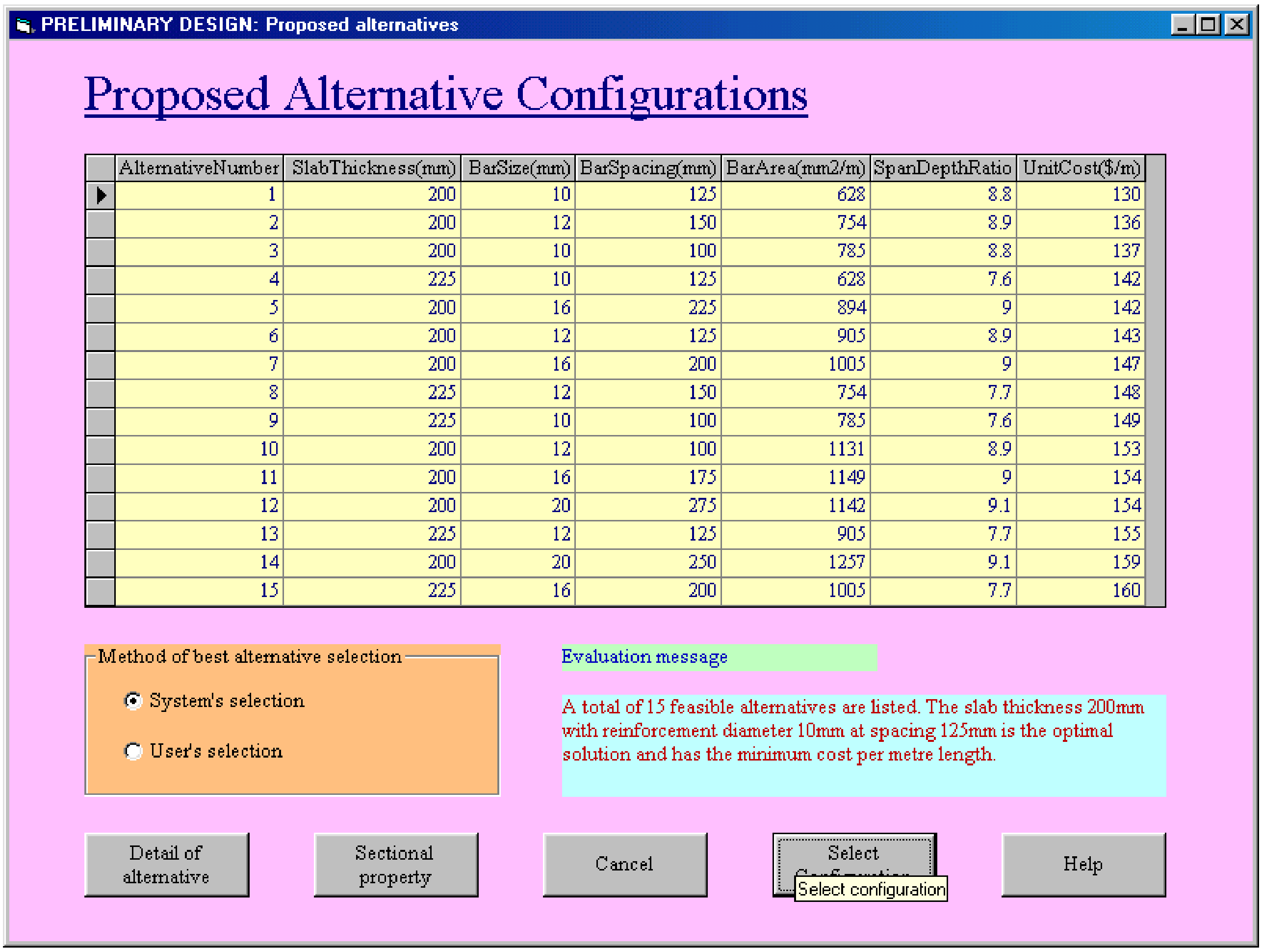


Figure 5 Screen showing structural specification in preliminary design



	AlternativeNumber	SlabThickness(mm)	BarSize(mm)	BarSpacing(mm)	BarArea(mm2/m)	SpanDepthRatio	UnitCost($/m)
▶	1	200	10	125	628	8.8	130
	2	200	12	150	754	8.9	136
	3	200	10	100	785	8.8	137
	4	225	10	125	628	7.6	142
	5	200	16	225	894	9	142
	6	200	12	125	905	8.9	143
	7	200	16	200	1005	9	147
	8	225	12	150	754	7.7	148
	9	225	10	100	785	7.6	149
	10	200	12	100	1131	8.9	153
	11	200	16	175	1149	9	154
	12	200	20	275	1142	9.1	154
	13	225	12	125	905	7.7	155
	14	200	20	250	1257	9.1	159
	15	225	16	200	1005	7.7	160

Figure 6 Screen showing 15 alternative sections with unit cost

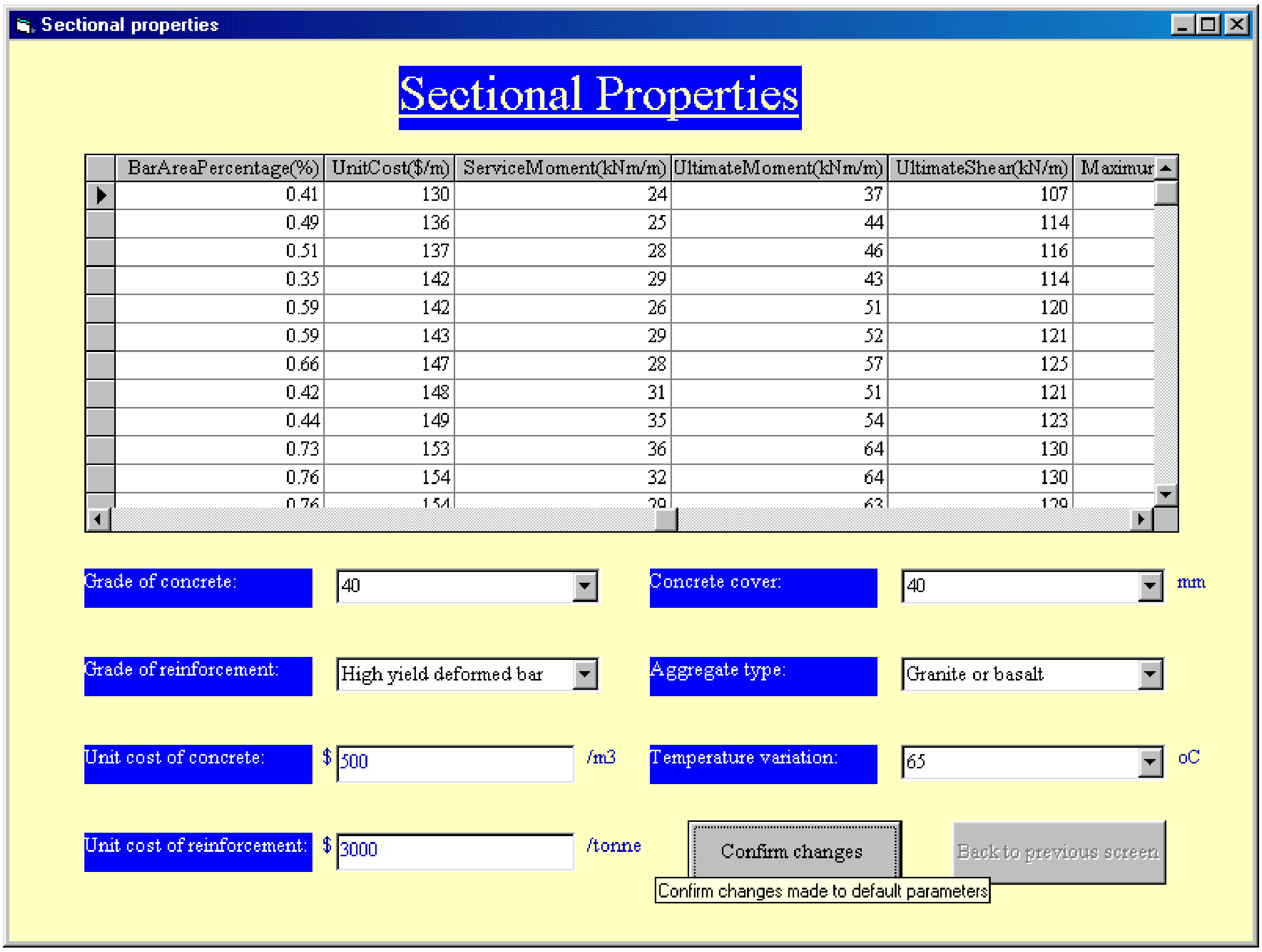


Figure 7 Screen showing sectional properties allowing changes to design parameters

Figure 8 Screen showing crack width checking in accordance to BS 8007