Application of expert systems to prestressed concrete bridge design

C. J. Burgoyne and S. H. R. Sham*

The potential merits of the application of expert systems to prestressed concrete bridge design are discussed. After identifying the areas in the design process to which expert systems can best be applied, attention is focused on the building up of a knowledge base in the domain of conceptual bridge design. The problems involved in the elicitation, validation and representation of the required knowledge are described. A design model is proposed for the selection of different bridge types. Implementation of the model to manipulate design knowledge is illustrated.

Keywords: expert systems, knowledge, prestressed concrete, Prolog

The present-day bridge engineer is confronted with a world of new challenges. In modern highway development, complicated road alignments dictate the geometry of the bridge structures. This results in unprecedented complexity in bridge design, analysis and construction.

The parallel developments in construction techniques, material usage and computer technology undoubtedly help engineers grapple with old problems. On the other hand, great strides in these fields have also encourages more complex structures to be attempted, which raise new problems in their turn.

A bridge engineer has to keep up-to-date with the changes in analytical techniques, the current thinking on good practice and the evolution of design philosophy. While it may be seem reasonable that more time should be devoted to the design of complex structures, the reverse is often true in practice.

With the decline of the road programme in the UK, the workload on bridge construction has correspondingly dwindled. While the overseas market provides some prospects for future work, different national standards, local conditions and construction practices tend to impose greater strains on design resources. The problem is further aggravated as the overseas work is being chased by an increasing number of bridge designers from the UK and other developed countries.

Are current design tools satisfactory?

The fierce competition for work has squeezed design fees and much reduced design time. The quest to automate and streamline the design process has led to significant advances in analysis software. There are aspects of the design however, which seem to defy an algorithmic approach used by procedural computer languages.

In the early stage of the design, viable alternatives are conceived and their merits compared and contrasted. An inappropriate solution often leads to abortive work, financial losses and even more severe consequences. At this stage, the requirements are often vague and hence the problems are ill-defined. A vast amount of experience and heuristic knowledge is demanded of the designer; analytical tools furnished by conventional software do not offer much help. Conceptual designs are therefore carried out by senior engineers who possess substantial experience.

As more precise information is gathered, the conceptual design will be used in analysis and detailing. While some of these tasks may be competently handled by junior engineers, with the aid of analysis software, support from experienced designers is still crucial and indispensable. The use of analysis packages leaves vital gaps in the design process and, at present, only good human judgement and decision can fill these gaps.

It takes years for an engineer to acquire his skill. Often, when he begins to be competent and cost-effective in design, he has to shoulder responsibilities in management, financial control, and in the supervision and training of junior engineers. It would be simplistic to assume that he can easily discharge his design duties in the diminishing time available. What is desperately lacking is a decision support system to assist the engineer to play his modern role.

A new approach to computer-aided design has long been needed. Expert systems have begun to show some promise in solving the problems and they can find wider applications in bridge design in future.

Application of expert systems

Experience possessed by generations of designers may be encapsulated in a knowledge base, and manipulated by an inference engine in such a way that the system serves as a powerful tool for decision-support. To the acceptical, involving computers in conceptual design may undermine the role of the engineer. This is far from the truth where expert systems are concerned. The workings of an expert system are much more transparent than those of traditional software. The user may question the system on how it arrives at a conclusion; he may also enquire why the system asks particular questions. The same approach, when used in dealing with a human expert, is quite likely to antagonize him and endanger a productive working relationship.

If the engineer disagrees with the recommendations made by the system he may update the knowledge base with new facts and rules. Obviously, to avoid novices

---

* Dept of Civil Engineering, Imperial College, London SW7 2BU, UK

(Received February 1987)

0263-0257/87/01014-06$03.00

contaminating the knowledge base, some measure of security is required. Although the rules are accessible for inspection, they should be protected from any additions or modifications except those performed by authorised experts.

It can readily be appreciated that on exposure to a multitude of expert users, the efficiency and credibility of an expert system will improve. The same benefits cannot be derived from 'blackbox' software which is prone to becoming obsolete as its rigidly coupled domain knowledge renders updating or adaptation extremely difficult.

It has been suggested[1] that people often prefer to deal with a machine which is impersonal and forgiving of their ignorance and uncertainties, besides being infinitely patient. This is obviously true in the case of an inexperienced designer seeking advice. A major hurdle in the learning process is the problem of asking the appropriate questions which will prompt useful answers from the experts. The instructional methods with which graduate engineers are accustomed through their education are hardly the norm in a civil practice. When faced with unfamiliar problems, the novice is unsure of the paths to be taken and their consequences, let alone knowing what alternative solutions there are. This points to a distinct advantage in a query-the-user strategy in an expert system whereby the user is prompted to participate in the problem-solving procedure and therefore gain an insight into the solution steps. This is not always possible when consulting a human expert.

Current research

Work in the area of the design of prestressed concrete bridges is being actively pursued at Imperial College, with the support of the Science and Engineering Research Council and the Ove Arup Partnership.

The aims of this project are:

- To identify the areas in the design procedure where expert systems may complement or surpass the present combination of human efforts and 'blackbox' software written in procedural languages.
- To review and rationalize the design process. Grey areas in the current methods will be scrutinised and attempts made to resolve them.
- To build knowledge bases in the English language such that they will not be made obsolete by the rapid development of computer languages.
- To test the rule base by implementing subsets of it in Prolog.
- To study the methodologies of knowledge elicitation and representation.
- To conduct interviews with experienced bridge engineers with a view to acquiring more knowledge and refining the system.

The design procedure

Fig 1 shows one potential procedure for the design of prestressed concrete bridges. The process is superficially sequential, in that most parts of the procedure lie on a single path. There are in fact a number of hidden loops, the formation of which is due to the complexity of the decision-making involved. At least three distinct areas can be identified where knowledge based expert systems may be used.

The first area will be in the initial, or conceptual, phase of the design process. The rules to be used here will be largely qualitative, with a considerable amount of subjective preference built in by the domain expert. It is quite possible that there would be substantial differences in rule bases established by different experts. The output of this part of the design process will be feasible structural forms. Only approximate limits on major dimensions are likely to be given, as an accurate sizing will call for refined analyses.

The second phase of the design process mainly concerns the numerical calculations, some of which require iteration, while others require the solution of sets of equations. These calculations will almost certainly be carried out by 'blackbox' routines written in a procedural language. The role of the expert system in this phase of the design will be to suggest the types of calculations to be carried out, and the order in which they are to be performed. More importantly, the expert system will make decisions using the results of the analyses. The output will be an economical cross section which satisfies the design criteria.

The final phase of the design process appears sequential; it involves determination of the cable profiles required, and checking of such items as parasitic moments, ultimate moment capacity and shear strength. These calculations would usually be performed by separate, procedural packages. If reasonable decisions have been made at the earlier stages, it should normally be possible to perform these tasks in a logical sequence. However, an expert system must be able to make rational decisions on the courses of actions to be taken when a trial cable profile proves unsatisfactory, or when any item fails on checking. This is not a trivial operation. In certain circumstances, small changes to the most recently made decisions will rectify the problem, while in other cases, the failure is severe enough to mean that the earliest conceptual decisions were invalid.

The received wisdom about the domain, as published in standard texts, makes little reference to decision making in the final stage of the design process. Standard texts tend to dwell on how the calculations required in the second and final phases may be carried out. In the event of a failure while performing the final phase, very little advice is available on the remedies. Specialist literature may present some limited information on how simple decisions can be made in the first and second phases; although the clues offered on conceptual design are far from being adequate to deal realistically with design problems. It is clear that considerable scope exists for a study of the design process, to provide the information required by an expert system, before work can start on building the expert system itself.

The present investigation concentrates at this stage on one aspect of the process: the selection of the type of bridge structure to be adopted, a subject domain which falls within the conceptual phase of the design procedure.

Constructing a knowledge base

A knowledge base of facts and rules governing the selection of different types of bridge decks has been developed. To clarify the rationale, Boolean algebra is used to edit clusters of heuristic rules and transform them into Horn clauses. The knowledge base encompasses

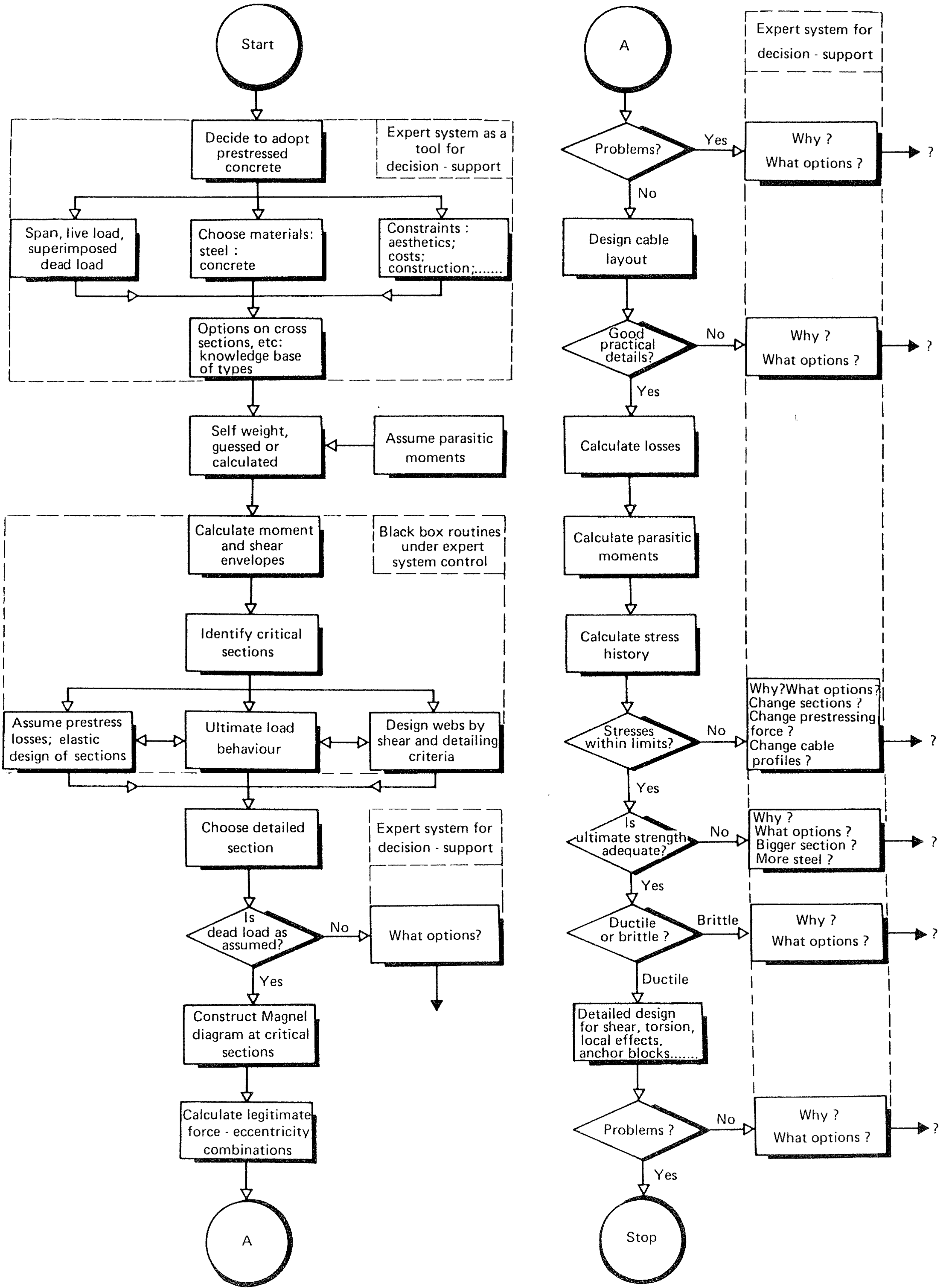
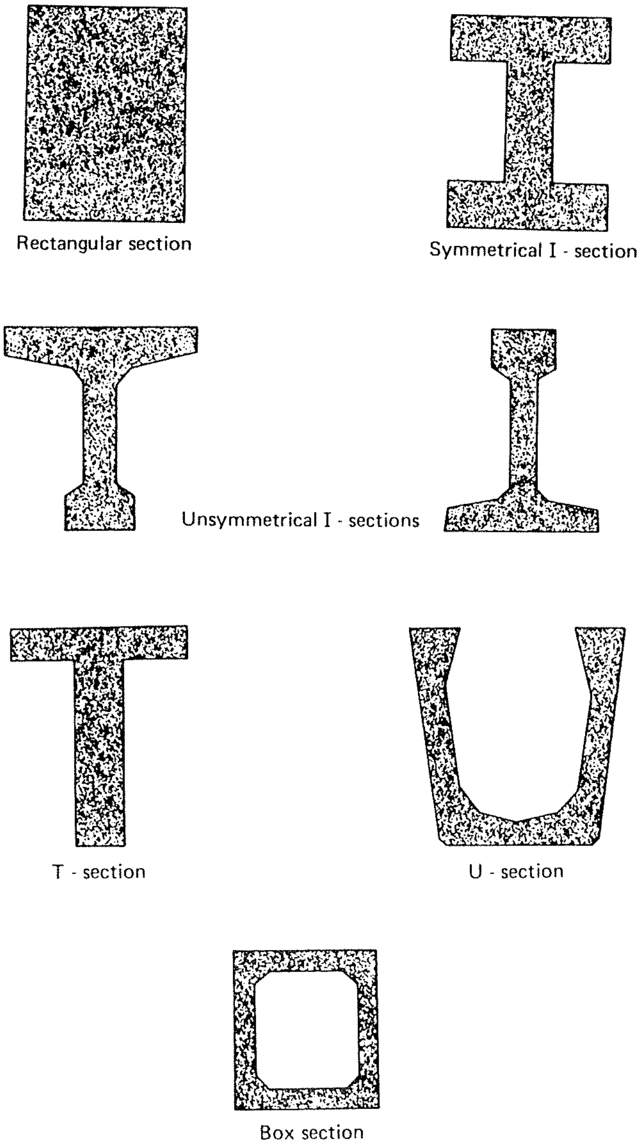


Fig 1 Design procedure

choices of prestressing methods, cross section shapes, structural forms, and methods of construction. In compiling the knowledge, a study is being conducted into the thinking behind the problem-solving strategies employed by competent designers. Progress has been made on identifying the conditions for which the use of I, T, U, rectangular, box and special sections (see Fig 2) are best suited. The work is being extended to include structural articulation and support systems. The next stage will involve parametric studies on the proportioning of cross-sections.

Validation of all rules is exceedingly important, as rigorous checking is always imposed on a structural design. Some expert system builders tend to assume that experts are infallible in their judgement, consequently little attention is paid to vetting the raw data elicited. The manipulation of such data is laden with problems, even if faultless experts did exist. It is not unknown for two experts to hold mutually exclusive views on the same subject, both of which can be supported by reasonable arguments. Mixing these views indiscriminately will almost certainly result in the system being crippled by inconsistencies. On the other hand, if a specific school of thought is pursued in a solution path while steering clear of contradicting opinions, a satisfactory answer can often be found. It is imperative, therefore, that clashes be identified. Once this has been achieved, irreconcilable approaches can be allowed to coexist, and any one of them may be used to search for a viable solution.

It is hoped that research in these areas will cast some light on what seems to be a mysterious art. In the past, attempts made to rationalise structural design did not meet with any significant level of success; this could be explained by the lack of appropriate tools to exploit the knowledge encompassed. The advancements in knowledge elicitation, representation and manipulation techniques may well assist us to prove what that knowledge actually is. A similar phenomenon has been observed in the evolution of structural analysis, in which the advent of Fortran and other procedural languages was instrumental in the rationalisation and development of analytical tools.


Rectangular section

Symmetrical I - section

Unsymmetrical I - sections

T - section

U - section

Box section

Fig 2 Shapes of beam cross-sections

Testing the rules

The validity of the knowledge base has been tested by trying out subsets of it on a Prolog program.

Since the inference engine is isolated from the knowledge base, new facts and rules can be incorporated without having to amend the control mechanism, a technique that is unachievable using procedural languages. Modules of knowledge can be built up to cater for different standards, variations in local conditions and disparities in design and contribution practices. Such decision-supports will enable designers to tackle schemes in different countries.[2]

In a consultative session, user-supplied information is prompted and then stored in a temporary workfile. The information may also generate transient data used in the evaluation of a goal state. The responses from the user are manipulated to trigger off viable rules by means of pattern matching. If the workfile is kept intact, its contents will obviously be utilised in subsequent sessions. Some of the facts will become default statements and the user will not be asked the same question more than once. This speeds up the selection procedure by pruning the search tree. Should the user decide to tackle a new problem in the light of different circumstances, or if it is desired to avoid contamination of the knowledge base by novice users, the temporary workfile may be edited or erased en bloc.

The set of all feasible bridge types is finite: and yet the rules for selecting an appropriate form are numerous. It is therefore preferable to adopt a backward chaining strategy[3] in the selection. A conclusion (goal) will be hypothesized; to prove whether this goal is valid, the problem is reduced to a chain of subgoals. The chain can be terminated by the user supplying information to satisfy the last subgoal or by using data which already exist in the knowledge base. This strategy is of special relevance to structural design, in which the designer often makes educated guesses at a feasible solution and attempts to substantiate the proposal by testing it with various design criteria. In trying to satisfy a design criterion, more interrelated conditions are often generated. A decision has to be made or information has to be supplied to arrive at a solution and hence terminate the chain of subgoals. Thus, problem solving is by means of a man–machine collaboration, a valuable improvement over the blackbox approach.

Although the order in which the subgoals is presented is immaterial to the logical representation of the problem, Prolog solves these subgoals in the order they are encoded and tries different ways of solving a subgoal in the sequence in which the conclusion-condition rules are programmed[4]. To achieve an efficient search strategy for the feasible bridge type, the hierarchy in which the subgoals appear has to be carefully arranged. If a solution exists in the knowledge base, the system will invariably find it; otherwise, negation as failure will result. Should there be more than one possible solution, the system can be designed to suggest the preferred solution, together with all the other satisfactory alternatives. This is not necessarily the way a human designer works. Unless given sufficient time and technical support, he is more likely to stop at the first workable answer.

An optimum search strategy would be one in which the expert system suggested firstly the best solution, with the non-optimal solutions being generated thereafter. It may well be impossible to configure the knowledge base in such a way that this is achieved, although a study of the strategy adopted in particular instances may lead to satisfactory results in at least some cases.

A bridge type may be represented by an *n*-tuple describing the various attributes of the bridge. These attributes include, for example, the type of construction, prestressing method, cross sectional shape and method of erection. The search for a feasible bridge type may be viewed as a goal that can be reduced into a chain of subgoals leading to the establishment of an attribute. There often exists a mutual exclusiveness among some attributes and the system must be designed to reconcile the conflicts.

Design criteria exists in many forms: they may be empirical, fundamental, codified or problem-specific. Invariably, they impose constraints in the search for a solution. A very useful guiding tool in the search procedure may be derived from an application of the celebrated Le Chatelier's principle in physical chemistry: upon application of a constraint, an equilibrium always shifts in such a direction as to counteract the effect of the constraint. As illustrations in this context, shortening the contract period induces a shift in favour of fast production and erection methods, possibly at the expense of economy, while difficult terrain necessitates launching techniques which obviate the need for falsework.

Provision is made in the Prolog program to display dynamically the generation of alternatives and the evaluation of the current goal state is made transparent. This is the essence of interactive computer-aided design.

It should be emphasized that the model of bridge type selection described is a valid and powerful tool in its own right. Future refinements to achieve a more elaborated representation of bridge types can be obtained by extending the *n*-tuple of attributes and adding the necessary goal-chains.

The logic of the bridge type selection procedure has been tried out on a Prolog program and the results have proved stimulating. To illustrate the ideas put forward in this section, a small sample of the rules is given below. These rules are used in solving the subgoal of evaluating suitable cross-sectional shapes (see Fig 2). They are presented both in natural language and in Prolog:

Rule 1

Rectangular sections are commonly used where the spans are less than or equal to 15 m. In Prolog:

```
X shape rectangle
if ( X span y ) is-told
& Y lesseq 15
```

Rule 2

Rectangular sections are suitable when simplicity of formwork is desired. In Prolog:

```
X shape rectangle
if ask-user( X simple-formwork Y )
& Y EQ true
```

Rule 3

Rectangular sections are suitable when ultimate strength is an important consideration and an ultimate design is adopted.[5] In Prolog:

```
X shape rectangle
if ask-user( X high-strength Y )
& Y EQ true
& ask-user( X ultimate-design Z )
& Z EQ true
```

Rule 4

I-sections are suitable when a high efficiency in the use of the concrete area is required. In Prolog:

```
X shape I
if clues( X high-efficiency true )
```

Rule 5

I-sections are suitable when serviceability criteria are of prime concern and elastic design is adopted. In Prolog:

```
X section I
if clues( X service-important true )
& clues( X elastic-design true )
```

Rule 6

Unsymmetrical I-sections with a bigger bottom flange are suitable if I-sections are recommended and M_t/M_s, the ratio between the moment at transfer to the total service moment, is small and there is sufficient compressive concrete area. (To avoid danger of overestressing the top flange at transfer.) In Prolog:

```
X shape I-big-bottom-flange
if X shape I
& clues(X t-s-moment-ratio-small true )
& clues( X sufficient-concrete-in-compression true )
```

Rule 7

Unsymmetrical I-sections with a bigger top flange are suitable if I-sections are recommended and the ratio M_t/M_s is large. In Prolog:

```
X shape I-big-top-flange
if X shape I
& clues( X t-s-moment-ratio-large true )
```

Rule 8

Tee-sections are suitable when I-sections are recommended and the ratio M_t/M_s is large and the web is wide enough to accommodate the tendons. In Prolog:

```
X shape Tee
if X shape I
& clues( X t-s-moment-ratio-large true )
& clues( X web-wide-for-tendons true )
```

Rule 9

Box sections have the same properties as the I-sections in resisting moment. They are suitable when a high torsional strength is required or when stability under temporary conditions is of prime concern or. In Prolog:

```
X shape box
if X shape I
& clues(X high-torsional-strength true )

X shape box
if X shape I
& clues( X high-construction-stability true )
```

Knowledge elicitation

When building the knowledge base, rules were initially compiled in a format commonly used by designers. It was soon discovered that in their unabridged version, some of the rules were difficult to apply and others could lead to fallacies in the reasoning. Negations are often used in place of assertions when designers talk loosely of their intentions and misuse terms in logic. Furthermore investigations are being made to seek a better way to identify and filter these pitfalls associated with knowledge gathering.

Although designers excel in solving problems in their specific domain, they do not always consciously examine the psychology behind their problem solving. With the exception of some experts engaged in education, people do not habitually practise expressing their knowledge in a coherent and explicit form. The methodology may be indistinct when problem solving becomes second nature. If forced into introspection, a designer may revise his rules to such an extent that they no longer represent what he actually uses in practice. One is then faced with the dilemma of whether to treat the modified rules sceptically as distortions of the genuine rules, or to accept them as a refinement to expert thinking. If a pragmatic approach is adopted, the dilemma is more apparent than real. An effective implementation is often a compromise between efficiency and fidelity in reproducing expert thinking.

The real problem however, is that one cannot model the unknown. Without some degree of knowledge in bridge design, the system builder will find it difficult to define the objectives, induce relevance from the experts and capture the crucial points. He may well be unaware of the avenues to be explored, or even more dangerously, he may innocently believe he has understood the solution paths. It would be naive to rely on the domain expert to take the initiative and spell out what the knowledge engineer should be looking for.

A technique that has been gauged successful in knowledge elicitation in other subject domains is to assign knowledge engineers to observe the expert in his work and to listen to his conversations with other experts. This is unlikely to achieve great results in the present investigation. Firstly, design is an introverted process; to probe into the designer's mental process through observation is virtually impossible. Secondly, since designers use pictorial constructs to reinforce their verbal communication, the obstacles involved in deciphering the protocols are considerable.

To overcome the problem, the researchers in this project have been choosen from a background of bridge design and construction. The knowledge base built up by the researchers will be discussed in interview sessions with the experts. In this way, specific problems and suggested solutions can be presented to the experts to seek their comments. Simulations can be made in which a researcher, armed with the rules, acts as a very slow-functioning computer and responds to an expert who plays the interrogating user. As a result of the consultations, the rules will be refined and more rules will be generated. A 'blackboard' technique will be used to resolve areas of conflict while persistent controversies will be retained as alternatives for the system user.

Conclusions

A new approach to computer-aided design has long been needed; expert systems show promise as an answer to many of the current problems. A research strategy has been formulated to try and overcome the real problems in the building of a practical system. The difficulties encountered in the construction of a knowledge base in the conceptual bridge design domain have been discussed and some solutions presented. A model based on the top-down, backward chaining strategy has been proposed for the bridge type selection procedure and its implementation has been explained.

Acknowledgements

The support afforded to this research by both the Science and Engineering Research Council and the Ove Arup Partnership is gratefully acknowledged. The Authors are particularly indebted to David Taffs and Angus Low of the Ove Arup Partnership for their constructive comments and helpful assistance.

References

1 **Taffs, D.** *Conf. Impact Comput. Technol. Constr. Ind., London, UK, 1984*

2 **Burgoyne, C. J.** Expert systems for structural design, *Constr. Ind. Comput. Conf., London, UK, 1986*

3 **Winston, P. H.** Artificial intelligence, *2nd ed. Addison-Wesley, Reading, MA, 1984*

4 **Kowalski, R.** Logic programming, *Byte, 1985,* **10***(8), 161–177*

5 **Lin, T. Y. and Burns, N. H.** Design of prestressed concrete structures, *3rd ed. John Wiley, Chichester, 1982*