Zeitschrift: IABSE reports = Rapports AIPC = IVBH Berichte

Band: 72 (1995)

Rubrik: Session 1: Knowledge support for structural design and construction

Session 1

Knowledge Support for Structural Design and Construction

Bases de connaissances pour le projet et la construction

Wissensunterstützung für den Tragwerksentwurf und die Bauausführung

Computational Decision Support For Preliminary Bridge Costing

Aide informatique à la décision pour le calcul du coût d'un pont

Computerisierte Entscheidungshilfe für die Vorkalkulation von Brücken

Carolynne J. MOORE
Dr Eng.
University of Wales
Cardiff, UK

Carolynne Moore, born 1966, obtained a Civil Eng. degree and Ph.D. at Univ. College, Cardiff. Her research concentrates on the development of innovative engineering decision support tools. She is currently a Lecturer at Cardiff Univ. and is joint leader of the Cardiff Decision Support Systems group.

SUMMARY
An innovative support system for the preliminary costing of bridges is based on the principle of heuristic substitution. It provides the designer with an efficient way of obtaining a preliminary bridge costing which can be easily amended and compared to other designs. This paper describes the system and the findings of the practical evaluation. The principle of heuristic substitution and the benefits of applying it in this instance are also discussed.

RÉSUMÉ
Un système innovant pour l'estimation du coût d'un pont est basé sur le principe de substitution heuristique et fournit au concepteur un moyen efficace d'obtenir cette estimation. Celle-ci peut être facilement modifiée et comparée à d'autres projets. L'article décrit le système, les résultats d'applications pratiques et les avantages de la substitution heuristique.

ZUSAMMENFASSUNG
Ein innovatives Unterstützungssystem für die Kostenvorkalkulation von Brücken basiert auf dem Prinzip der heuristischen Substitution und stellt dem Konstrukteur eine leistungsfähige Methode zur Verfügung, die auf einfache Weise ergänzt werden kann und auch das Vergleichen verschiedener Konstruktionen ermöglicht. Der Beitrag beschreibt das System, die Ergebnisse seiner praktischen Bewertung und die Vorteile der heuristischen Substitution.

Introduction

In recent years, there has been an increasing interest in the development of engineering design applications of Artificial Intelligence (AI), with a particular focus on expert or knowledge based systems [1,2,3]. However, practical applications of AI in engineering design are still very rare.

Cardiff has experience of developing innovative computer systems for civil/structural engineering, with particular reference to design [1,4,5,6]. Our research has always aimed to produce innovative computer systems which are of immediate use. To achieve this, close collaboration with the industry is essential, to ensure the applicability of the systems as well as providing new ideas and directions. This high level of industrial collaboration, combined with the experience gained when building early design systems, has produced an original approach to the research. We focus on building systems which can be implemented immediately; subsequently incrementally improving these systems with the help of industrial evaluation. The authors believe that this is in contrast to much other research which aims to build complex and powerful systems based on design theory and assumed industrial needs [7,8,9]. Both sides of the research spectrum are complementary and experience from both aspects will result in the development of innovative systems which are of real benefit.

This paper describes one system which has been produced at Cardiff as result of this 'bottom up' style of research. The idea relies heavily on previous and current associated work [10,11]. The system described is an enhancement of the knowledge base of Moore [1], assisting the designer with decision making in conceptual design. The underlying methodology is simple and yet has the potential to make a major impact on conceptual design processes.

Background Work

Work at Cardiff began with the development of 'standard' expert systems for conceptual design domains [1,12]: that is, associational, rule based systems which rely on a prescriptive question and answer format. These systems provided an insight into the KBS approach and using the accepted KBS approach resulted in restrictive and inadequate design systems. This work is detailed elsewhere [11]. Primarily, it was found that, because of their initial roots as diagnostic tools, associational KBS were unsuitable for most design domains, primarily because the demands of design are very different from diagnosis. For example, flexibility, innovation and creativity are only three of the criteria which are essential for design but which are less important or even undesirable for diagnosis. Also, there is rarely a single 'correct' design, making the examination of alternatives particularly important [11].

Our work has involved assessing the reactions of practising designers to our systems. This provides scientifically based knowledge of the utility of varying approaches and also is indicative of the importance of features which can be overlooked. For example the way in which information flows between the system and the user is very important [11,13]. Initial research has also shown that the domains which design systems originally tried to cover were very ambitious and hence tended to be complex and problematic. Experience has shown that it is preferable to break domains down into smaller, sub component systems which can be used individually or linked together. This has been implemented in our current research and the costing system described here is one component of a suite of such design systems [13,14].

The evaluation work also revealed that in some areas, the design KBS initially developed were trying to undertake tasks which designers were better equipped to perform. Designers are capable of many things which are difficult to emulate with current computing technology: most notably judgement, innovation and common sense. In other areas, the KBS were directly adopting simple heuristics obtained from expert designers [1], which were limited in terms of accuracy and efficiency. On examining these heuristics, it was found that they could often be replaced by more accurate computer based procedures thus producing a more reliable answers than was previously possible. The idea of heuristic substitution was therefore developed [10]. It is the development of this technique and the impact which it has had on the design engineers with whom we work, which forms the main topic of the rest of this paper. In the following section the reasoning behind and the concepts of heuristic substitution are introduced and the subsequent sections describe a practical application of heuristic substitution in bridge design.

Heuristic Substitution : What is it?

Despite the large number of design KBS reported in the literature, there are few examples of such systems being used by practising designers. As a part of our evaluation work, ways of making the systems more useful were investigated. On examining the knowledge bases, it became apparent that some of the heuristics elicited from expert designers could easily be replaced by more accurate methods which made better use of the computer power available. Further, it was found to be advantageous to classify heuristics into groups related to their knowledge and source [11,15]. For example, the heuristics could be used because no other more reliable estimate is available or because the underlying calculations are too lengthy to remember. Some basic classification groups are:

- Short Cut Heuristics
- Heuristics based on Background Knowledge
- Heuristics Based on Ill defined Concepts
- Heuristics based on Empirical Data
- 'Inherited' Heuristics

The groups are far from exclusive and for other domains, additional groups may be relevant. However, these groupings were found to be useful when dealing with engineering design and are described in detail elsewhere [11,15].

Fundamentally, the heuristics were classified empirically as this proved to be the best way of identifying the heuristics which could most profitably be replaced. This classification schema does not mirror the heuristic taxonomies derived by others working in psychological AI research [16]. These taxonomies tend to rely on breaking heuristics down according to their psychological function. The classification described has been developed purely through experience of the design process. In effect, current design practices have been developed to suit the capabilities of the human brain (large long term memory, small short term memory) and the above classification helps to identify areas of design where heuristics are adopted because the human cognitive processes fail to perform adequately. If this lack of performance is due to cognitive overload (fundamentally, the brain 'opting out' and using estimates because there are too many concepts to deal with at one time), then in some cases it is possible to devise new computer based design procedures to compensate this behaviour and hence provide more accurate answers than those given when heuristics are used.

When examining the type of heuristics where computer techniques can be used beneficially to supplant human heuristics, it is generally found that the heuristics have been developed because the original calculations are too complex or lengthy to conduct by hand. Consequently, short cuts or approximations are developed by the designer, hence inevitably introducing inaccuracies. When trying to derive a suitable computer based replacement, in some circumstances simply replacing the approximations with the underlying algorithms is not suitable, as the underlying calculations are still too lengthy for the computer to conduct in an acceptable time scale. It is then necessary to produce 'computer' heuristics: that is, new short cuts which can be used by the computer system to give a quick answer which is not theoretically complete but which is more reliable than the estimate originally used by the expert. Whether the full algorithms or new, expanded heuristics are used, generally heuristic substitution involves replacing heuristics with algorithms. While to some extent this may seem to transgress the earlier ideas of KBS which aimed to replicate human expert decision making, it is entirely in accord with more recent ideas on KBS, where more emphasis is placed on the quality of the input and consequent output as opposed to how the system achieves its goals [17]. It also complements the philosophy of the Cardiff group, which is to create support systems for areas in which people have difficulties as opposed to creating systems which emulate them in tasks at which they already perform well [11].

A parallel to heuristic substitution can be found in analysis, where such techniques as finite element analysis, which has enabled an accurate analysis of more complex problems than those which could be analysed by hand. This software succeeded purely because it enhanced human performance. Heuristic substitution offers a similar way of enhancing the performance of design systems.

The importance of heuristic substitution in practice has been shown by Hooper [5] at Cardiff, who developed a KBS for the strategic planning of sludge disposal. In this work, a genetic algorithm was used to find an optimal solution, replacing the previously used inaccurate hand calculation methods which were heuristically driven.

Applying Heuristic Substitution to Conceptual Bridge Design

During the development of the original conceptual bridge design system, one area which was found to be particularly problematic was preliminary costing, as this incorporates accurate costing difficulties as well as preliminary member sizing. Observations of design practice have shown that costing is an area which apparently relies on three types of estimate, offering completely different levels of accuracy. At the lower end when an overall price for a bridge is required, generally a price per m^2 costing is used to provide a quick estimate. This is based entirely on past experience. At a similar level of simplicity, when choosing an economic form of deck construction, experts use set ranges of spans to reach a decision. For example, reinforced concrete decks are generally thought to be economic for spans of up to 16m. However in contrast to these simple heuristics, when a more accurate costing is needed an almost complete design is performed and relevant quantities taken off. Typically this latter process would involve a minimum of one man week of work to cost two options.

There is no in-between form of costing, which provides the designer with a reliable estimate of the bridge cost but which is not too time consuming to be economic. Without such costing, comparison of alternative designs is currently both difficult and expensive. Our research has shown that a system which could provide accurate and rapid costs estimates would be beneficial in terms of time and money, facilitating not only better estimates but also enhanced comparison capabilities. Work has thus been conducted on the development of such a system which includes the creation of realistic, practically based costing models and the development of these is described in the following sections.

The costing process starts with an identification of the required components and materials and proceeds to member sizing. The level of accuracy with which the members are sized has a significant impact on the process. For example, a typical heuristic for sizing a bridge deck is to use a span:depth ratio (typically 20:1). However, for example, this can be substituted by using a grillage program which looks at all the possible loading combinations and hence provides a more accurate solution. Once member sizes have been fixed, it is a relatively simple process to take off the relevant quantities. From this, relative costs of the components for various options can be obtained by assigning current prices to the materials used. This is ostensibly a simple procedure, but there are several pitfalls in practice which are discussed below.

How Was The Information Obtained?

At the start of this project, some difficulties were faced as to how to obtain the necessary costing information. A knowledge base which contained many of the current 'expert' heuristics was available from previous work. These were focussed largely on the very approximate costing procedures described above. Simply moving to a full costing procedure based on a Bill of Quantities, as commonly used by a contractor when preparing a tender, was too cumbersome for our needs. Some way of obtaining an 'in-between' approach was needed.

In order to move away from Bill of Quantities style approach, an understanding of the components which comprised the bulk of the costing was required. Obviously contractors (particularly those involved in design and build style contracts) are better at reaching costing estimates than design consultancies and so two contractors were approached for help. One contractor provided a simplified costing system which his company used to initially analyse a Bill of Quantities. Instead of containing rates for hundreds of different items, as one would find in a typical bill, it contained just 10 rates, shown in Table 1. The contractor had found that this simplified costing method typically resulted in a price which was within a few percent of the final detailed estimate. This was acceptable for the aims of our system, as although it is important for the costs to be reasonably accurate, comparative costing is most important at the preliminary stage. However, this breakdown alone was not sufficient for our needs, as a greater level of detail was required.

Structures	Unit
Excavation	m^3
Bored Piling	*m*
Imported Fill	m^3
Insitu Concrete	m^3
Formwork (Horizontal)	m^2
Formwork (Vertical)	m^2
Reinforcement	*t*
Precast	m^3
Bearings	*no.*
Waterproofing	m^2

TABLE 1

Thus at this, the problem of how to cost the structure was largely solved. We still had to rapidly size the members and devise a system which allowed the user to easily look at options and take off quantities.

Using the preliminary costing breakdown provided, an outline of the proposed costing mechanism for the system was produced. From this, areas in need of further investigation were identified and these were used to structure a series of interviews with consulting engineers and specialist contractors These interviews researched costing methods currently used for relevant components.

The interviews were conducted on two different levels. Initially, the interviews concentrated on the costing of individual bridge components. First, the superstructure was examined. Designers were asked how a preliminary bridge cost would be determined for different superstructure types. The experts were prompted with simple diagrams of different span and deck types. The answers given ranged from consulting a manufacturer's precast beam catalogue to using a simple stress block analysis for cast in situ decks to using a grillage program for complex steel composite structures.

The next set of interviews focussed on the design and costing of the supports. Again a series of diagrams were used as prompts. The main finding was that the end supports should be designed as a retaining wall with various axial and lateral loads.

Finally, foundations, in particular piling, were examined by consulting a specialist piling contractor. These interviews initially aimed to elicit the type of pile which would be used for certain soil conditions. Once this had been roughly established, the price of the pile was developed by introducing hypothetical situations and asking the expert to build up a rough price for these cases.

The second stage of interviews were more general and involved the preliminary pricing of the overall structure. This included discussing such things as which were difficult aspects of the task, which were important, what could be neglected and what should be done in detail. This stage also incorporated working with the designers, producing preliminary estimates in their office and examining previous designs and costings.

By using the interviews and relevant Codes of Practice it was possible to build up a system which incorporated costing algorithms for the major contributing factors of a preliminary bridge design.

The System

The system has been developed using Microsoft's Visual C++ and the 'Windows' operating system. C++ was chosen because it offers a combination of benefits, namely: high flexibility, the ability to process numerical algorithms quickly and powerful graphical capabilities. By developing the system to operate in a Windows environment, an interface style has been adopted which is familiar to engineering designers. The decision to use C++ has proved to be beneficial in terms of both ease of programming and interface design.

Size: Currently the executable program is approximately 1.5 MB in size with the individual bridge designs needing less than 500 bytes. This gives the system the largest possible degree of implementation flexibility, allowing it to run on a 386 based PC. This was a project requirement as these machines are readily available in the smallest of regional engineering design offices.

Input: As the system aims to simplify and reduce the work load of the engineer, a simple input format was required. The philosophy of the system would be defeated if the designer had to spend hours inputting detailed dimensions. Hence, the minimum input which is accurate enough to give a realistic description of the bridge is required. There are approximately 70 different input variables consisting of numbers or strings given in list boxes. However, about half of these remain as default values and may not need altering for every design. For example some material properties will remain constant when comparing designs at a site. Typically, it takes 15 minutes to input a new design.

Calculations: The system aims to determine the preliminary costs which can be used to compare design options. As such, it does not perform a full structural analysis of the bridge but uses a mixture of heuristics and simplified design code procedures to reach a satisfactory estimate. An example of the different approaches used can be seen in the calculation of reinforcement areas. The areas in the abutment are calculated using design formulae from BS8110(1985) with highway loading and load cases from BS5400(1978). However the area of reinforcement in a pier is assumed as a percentage of the cross sectional area: an heuristic gained from the designers consulted. The difference arises because the reinforcement needed in a pier is fairly stable whilst there are many different parameters that can affect the load on an abutment and hence a more rigorous analysis is ideally needed.

Operating the system: In order to start a new design, data is input using approximately 9 different dialog boxes, which describe the entire bridge. The dialog boxes are controlled by conventional menu commands and toolbar buttons. These are listed below, together with a few examples of the variables which the dialog boxes include:

- *Summary*
  Allows the input of the name of the bridge, contract and a description. It also automatically updates the revision date and time.

Global Properties
Includes the number of spans, skew, curvatures and location.

Earthworks
Covers the type of end support, wingwalls and embankment or cutting dimensions.

Deck and Span
Dimensions the largest span and loading conditions. Also allows deck selection between cast, precast or steel types. It prompts the user with the recommended precast beam size.

End Supports
Allows input of the dimensions of various different support types and shapes. (Figure 1).

Piers
Inputs the size, location and shape of piers.

Foundation
Selection of foundation type and dimensioning.

Material properties
Input of any remaining properties e.g. soil and concrete parameters.

Prices
Input global prices e.g. reinforcement / tonne, vertical formwork / m^2, concrete /m^3 etc.

Once this has been completed, the structural stability of bridge is checked. The structure can then be redesigned, if it fails or is too conservative, by simply altering any of the dimensions or parameters input earlier. This can be repeated until the designer is satisfied.

The next step is to calculate the relevant quantities. This is achieved by simply selecting a toolbar button.

Finally the price of each item is allocated by the system and the total price is shown by selecting the summation toolbar button (Figure 2). The cost appears almost instantaneously.

The design can now be fine tuned by simply altering any of the input variables and recalculating the design. Any aspect of the design can be altered in any order and the whole process does not need to be repeated to alter one parameter. When changing the bridge to a completely different form e.g. by altering the number of spans, deck construction material and/or end support, the structural stability and cost can be checked and obtained in under a minute. This can be compared to possibly half a man week, per major design change, which this process would currently take by hand.

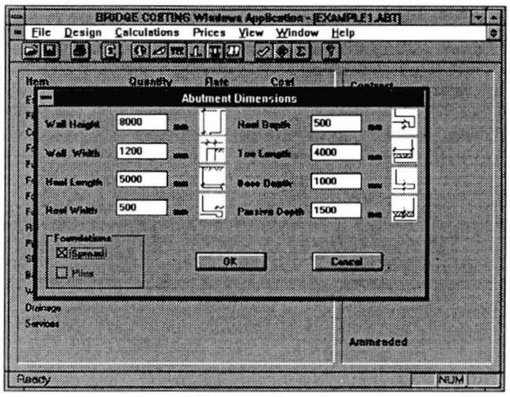


Figure 1.

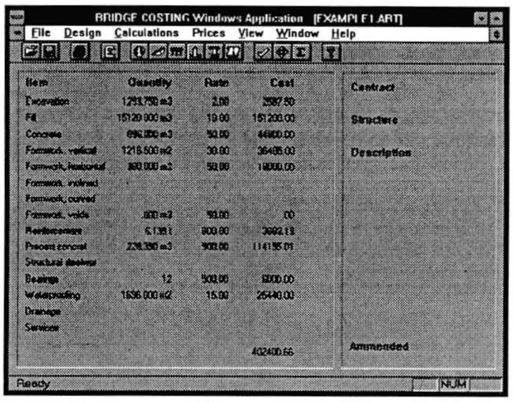


Figure 2.

Practical Experience with the System

The system is now undergoing preliminary evaluation in industry and to date the evaluation has proved to be very successful. The reviewers have stated that the system is both applicable and useful for practical bridge costing and have cited beneficial additional development work. For example, life costs, the overall weight of the bridge and the areas of reinforcement required could be included.

The evaluation has shown a number of things. Firstly, it has shown that the system does provide a better cost estimate than can currently be achieved by accepted approximate methods. In addition, it has shown that a considerable saving in time can be obtained. Also, once the initial design factors have been input, the system allows small changes to be rapidly made to the design enabling ready calculation of alternative costs. This level of flexibility facilitates the comparison of alternative designs and enables the designer to refine a design by making small changes to the chosen alternative, helping to ensure an optimal solution. It is anticipated from the reception which the system has received that this will encourage the designer to experiment and compare a greater number of alternative designs than would currently be considered possible. This should in turn result in better conceptual design as fairer assessments of economic alternatives will be possible [11].

An additional advantage stated by the engineers involved in the evaluation is the flexibility that the system offers. Currently, in design offices, one 'expert' is responsible for most of the costing, and their expertise allows them to conduct costings more efficiently than non-specialists. Using the same person can also help to ensure a degree of standardisation. However, reliance on a single person has obvious disadvantages The system evaluation has shown that designers feel that this software will provide a degree of standardisation and release the 'costing expert' from some of his/her duties.

The evaluators have also suggested that the enhanced preliminary costing techniques provided by the system could potentially be very useful for dealing with contractors. Frequently, contractors will suggest an alternative design which they claim is cheaper. The system would provide a quick and relatively easy mechanism for checking alternatives.

An additional benefit recognised by the evaluators is that as more designs are costed, they can be stored as files within the system. These files can then be retrieved if similar design costings are needed, effectively creating a database of design costs. By retrieving suitable past costings, small amendments could be easily made to assess their influence. This idea could be extrapolated to provide a case based structure which could retrieve similar designs on the criteria given and hence give a preliminary costing breakdown for that design.

Overall, the system has had a very good reception from the evaluators who suggest that the time saving aspects of the system combined with its enhanced flexibility will be invaluable.

Why is this System Different?

It is recognised that the development of a computer based costing system to aid design decision making is not unique. For example Retik et al [19] describe a probablistically based costing system for planning housing schemes and Syrmakezis and Mikroudis [20] describe a costing system for building design which costs solutions produced by an expert system. For bridges in the UK, there is a design costing tool called BRIDGET [21] which is based on database technology and uses specifically designed costing heuristics which are an improvement on the 'expert' costing heuristics normally used but nevertheless still incorporate a substantial degree of approximation. The costing model described here undertakes a considerable amount of detailed design analysis to provide member sizes which are very close to those provided by a full analysis. All this is achieved within less than a second of CPU time on a 486 PC. More important, however, is the route by which we arrived at the need for design costing models. This proves that heuristic substitution is beneficial for identifying areas where it is possible to provide enhanced computer based design procedures.

The system described here also differs in that the systems mentioned above do not provide a ready means of comparison nor do they incorporate heuristics actually used by designers. In addition, this system tries to be open to change and alteration, hence making it more suitable for use in practice.

Future Work

The system is currently undergoing further refinement, according to the comments of the reviewers. Further to this, a help system is being created. The system is also about to move on to the second stage of the evaluation: that is, testing the systems using case studies. The engineering companies involved in the evaluation have offered previous comparative designs which can tried on the system to provide a better indication of the system's performance. In addition to this, there is still a large amount of work to be done which will complement the system as it currently stands. The most important part of this work involves the development of 'risk' quantifiers. These are measures which will be incorporated in the system to give an indication of which criteria are most difficult to cost reliably. This will provide the designer with an appreciation of the most imprecise areas and hence enable him/her to make a better assessment of the cost provided.

Work is still needed to enable the system to fully interact with the other design systems being developed at Cardiff [14]. Once this work has been completed, the authors believe that the system will be useful and beneficial, both as a stand alone application and as part of a design suite.

Commercial development of this software is also being explored. Many of the approaches used in the costing system are largely generic and so alternative fields of application are being investigated.

Conclusions

A costing system which was initially developed as a result of findings of previous research has been developed. The system is based on the principle of heuristic substitution, and provides a preliminary costing estimate which can be used in the conceptual design process. The system, like all others developed at Cardiff, has been created in close collaboration with industry and hence is intended to be used as a practical design aid within the near future. This aim has influenced the style of development and it adheres to the philosophy that innovative systems should aim to enhance and support human design behaviour as opposed to supplant it.

The system evaluation has already shown that it is potentially a very useful tool which will enable designers to reach a better estimate for bridge costs more quickly and efficiently than is currently possible. It has also shown that enhanced comparative capabilities is one of the main strengths of the system and it is believed by the authors that the implementation of the completed system will enhance the preliminary costing process currently used in design offices, which will in turn improve conceptual design processes.

Acknowledgements: The authors would like to thank Sir Alexander Gibb and Ptnrs, Sir William Halcrow and Ptnrs, Taylor Woodrow and Davies, Middleton and Davies for their co-operation.

References:

1. **MOORE, C.J. (1991):** AN EXPERT SYSTEM FOR THE CONCEPTUAL DESIGN OF BRIDGES. Ph.D. thesis. University of Wales. February 1991. 350pp.
2. **MAHER, M.L. (1987):** EXPERT SYSTEMS FOR CIVIL ENGINEERING. ASCE, New York.
3. **FOO, H.C AND AKHRAS, G. (1993):** A KNOWLEDGE-BASED SYSTEM FOR THE DAMAGE ASSESSMENT OF TIMBER TRUSSES, Cohn, L.F (ed), Computing in Civil and Structural Engineering, V-ICCCBE, Anaheim Cal., 781-784.
4. **SOH, C.K. AND MILES, J.C. (1989):** THE DESIGN OF STEEL OFFSHORE STRUCTURES USING AND EXPERT SYSTEM. In: Topping, B.H.V (Ed.): AI Techniques and Applications for Civil and Structural Eng. Civil-Comp Press. 197-201.
5. **HOOPER, J.N. (1994):** A KBS FOR STRATEGIC SLUDGE DISPOSAL PLANNING, PhD Thesis, Cardiff School of Eng.
6. **MOORE, C.J., EVANS, S.N. AND MILES, J.C. (1994):** ESTABLISHING A KNOWLEDGE BASE FOR BRIDGE AESTHETICS. To be Published in Structural Engineering Review.
7. **GERO, J.S. and STANTON, R. (Eds) (1988):** AI DEVELOPMENTS AND APPLICATIONS. North Holland.
8. **HUANG, G.Q. (1990):** CO-OPERATING KBS FOR MANUFACTURING DESIGN. PhD. Thesis, Cardiff School of Eng.
9. **MCCARTHY, T. and NOUAS, Z. (1991):** A KNOWLEDGE REPRESENTATION SCHEME FOR THE DESIGN OF HYBRID STRUCTURAL STEELWORK CONNECTIONS. CIVIL-COMP 1991.
10. **MILES, J.C. AND MOORE, C.J. (1991):** CONCEPTUAL DESIGN: PUSHING BACK THE BOUNDARIES WITH KNOWLEDGE BASED SYSTEMS. In: Topping, B.H.V. (Ed.): AI and Structural Engineering. Civil-Comp Press.
11. **MILES, J.C. AND MOORE, C.J. (1994):** PRACTICAL KBS IN CONCEPTUAL DESIGN. Springer Verlag.
12. **SOH, C.K. (1990):** AN APPROACH TO AUTOMATE THE DESIGN OF OFFSHORE STRUCTURES. Ph.D. Thesis. University of Wales. 330pp.
13. **MILES, J.C., MOORE, C.J. , EVANS, S.N., LEHANE, M.S., PRICE, G. and REES, D.G. (1995):** INTEGRATED INNOVATIVE COMPUTER SYSTEMS FOR CONCEPTUAL BRIDGE DESIGN. IABSE Conference 1995, Bergamo.
14. **MOORE, C.J. (1994):** COMPLEMENTARY INNOVATIVE SYSTEMS FOR USE IN BRIDGE DESIGN. In: EG-SEA-AI Workshop on Structural Engineering and Artificial Intelligence.
15. **MOORE, C.J and MILES, J.C (1992):** IN DEPTH ANALYSIS OF HEURISTICS IN KBS DEVELOPMENT. In: IEEE Digest. London. January 1992. 1992/011 pp 6/1-6/4.
16. **CLANCEY,W (1985):** HEURISTIC CLASSIFICATION, Artificial Intelligence, 27, 289 -350.
17. **VAN DE VELDE, W. (1993):** ISSUES IN KNOWLEDGE LEVEL MODELLING. in David et al (eds), Second Generation Expert Systems, Springer-Verlag, Berlin, 211-231.
18. **MOORE, C.J. AND MILES, J.C. (1993):** THE DEVELOPMENT AND FUTURE OF A KBS FOR CONCEPTUAL BRIDGE DESIGN. In: Int. Assoc. for Bridge and Structural Engineering. Beijing. May 1993.
19. **RETIK, A., MARSTON, V. AND ALSHAWI, M. (1993):** DEVELOPMENT OF AN EXPERT SYSTEM FOR INTELLIGENT SIMULATION OF HOUSING MODERNISATION WORKS. in Topping, B.H.V.(ed), Artificial Intelligence and Civil Engineering, CIVIL-COMP Press, Edinburgh, UK,177-185.
20. **SYRMAKEZIS, C.A. AND MIKROUDIS, G.K. (1994):** AN EXPERT SYSTEM FOR THE EARTHQUAKE-RESISTANT DESIGN OF BUILDINGS, paper presented at a workshop, National Technical University, Athens.
21. **HORNER, M., MURRAY, M. AND MCLAUGHLIN, A. (1990):** BRIDGET - A COSTING ESTIMATING SUITE FOR HIGHWAY STRUCTURES. Highways and Transportation. May 1990. pp 14-18.
22. **BRITISH STANDARDS INSTITUTION. BS5400:** PART 2: 1978. STEEL, CONCRETE AND COMPOSITE BRIDGES. Part 2: Specification for Loads.
23. **BRITISH STANDARDS INSTITUTION. BS8110:** STRUCTURAL USE OF CONCRETE: PART 1. CODE OF PRACTICE FOR DESIGN AND CONSTRUCTION, 1985.

Supporting Tools for Evaluating Acoustic Building Performances at Early Design Stages

Outils d'évaluation des performances acoustiques de bâtiments lors de la conception

Beurteilungshilfen der akustischen Eigenschaften von Gebäuden im Frühstadium

M. KARROUM
Université de Savoie
Le Bourget du Lac, France

B. BAETZ
Université de Savoie
Le Bourget du Lac, France

G. ACHARD
Mcmaster University
Hamilton, ON, Canada

SUMMARY
The paper emphasises on the necessity to use a knowledge-based system for supporting acoustic design tools which allow designers the capability to evaluate acoustic performances of residential buildings at early stages of design, i.e., massing and sketching development.

RÉSUMÉ
La communication met en évidence la nécessité d'utiliser un système à base de connaissances pour implémenter des outils permettant aux concepteurs d'évaluer les performances acoustiques de bâtiments d'habitation dans les phases les plus précoces de la conception, soit lors de l'établissement du plan masse et de l'esquisse architecturale.

ZUSAMMENFASSUNG
Der Beitrag zeigt die Notwendigkeit für ein wissensbasiertes System auf, um Beurteilungshilfen zu erzeugen, die dem Architekten die akustischen Aspekte beim Bau von Wohnhäusern zu berücksichtigen erlauben. Besonders in den entscheidenden Entwicklungsphasen wie Konzeption, Statik und Rohbau sind solche Hilfsmittel sehr nützlich.

1. INTRODUCTION

Building performances in general and acoustic performances in particular result from the combination two types of performance : architectural and technical performances. The global performance of building depends closely on the decisions made at the early design stages [1].

In residential building, the acoustic design objective can be reduced, in first approximation, to the acoustic insulation [2 , 3]. This one aims at carrying out the appropriate elements (geometrical and technical) so that the sound level received in one room is located within a range depending on the nature and the origin of sound, room purpose,.. when an exterior noise source is emitted [4].

Acoustic design process can be considered as a sequence of decisions made at different levels of abstraction with several design stages were identified [2 , 3]. Each stage is more detailed and more specific than the former in terms of objective and description. Regardless of the design stage the main objective (insulation against noise) is the same but the way to achieve this objective is different

The evaluation of acoustic building performances can be made by using some tools like Qualitel's [5] and CSTB's [6] Methods. These ones are generally used for evaluating acoustic building performances at later design stages and can not be used at early design stages for a lack of information (neither the site and nor the building are completely described). Hence, the only plausible way to predict the acoustic performances of a building at these stages is the application of context-sensitive past experience in the form of rules of thumb (heuristics).

From the relevant points of view, we contend that significant benefits can be realised by developing a Computer Aided Design system for assisting the designer to determine the acoustic performances of a building particularly at early design stages, where the impact of decisions made at these stages are more significant than choices made at later design stages in terms of cost-effective issues. K.B.S has to be chosen due to the context of building acoustic design.

2. ACOUSTIC BUILDING STUDIES

Acoustic building insulation can be carried out in several ways [4 ; 7]. By relying upon sophisticated and expensive technical solutions, the designers can ensure the protection of building against exterior as well as interior noise sources. But this protection can be also achieved through a good choice of building placement and orientation, a good distribution of apartments and rooms and to some technical means which are more economic compared with the previous case [7]. In fact, with a judicious implementation of building (for example parallel to a surrounding road), it is possible to reduce the noise level, to which one façade of the building is subjected, of 15 dB (A) [4]. We would like to turn attention to the fact that, in order to increase a sound reduction index SRI of a single wall of 3 dB (A), the wall's unit surface weight has to be doubled.

The sophistication of technical solutions to propose and consequently the cost, in order to fulfil depends closely on the quality of architectural solutions. Hence designers must try, as much as possible, to exploit the characteristics of site where the building is to be constructed and to find out the optimal architectural configuration of rooms and apartments [8]. They must, for example, place the bedrooms and living away from the sources of noise such as service duct, elevator, energy central production, noisily roads,... However, trade-offs must be made to achieve a good overall balance. In fact, the acoustic aspect is not the only aspect to be assured by a building, and the designer must consider the overlapping of the different functions of building like thermal, structural ones. In any way, acoustic building aspects must consider from the formative design stages and solutions have to propose if the designers do not like to rely upon sophisticated and expensive technical solutions for the achievement of the acoustic objectives.

3. ACOUSTIC DESIGN PROCESS

Acoustic design studies are inherently complex because of the high degree of interdependencies between various design parameters like building orientation, rooms organisation, components characteristics and so on. In fact, the choice of one parameters limits the range of assignable values to the others parameters. For instance, the implementation of bedroom beside an elevator implies the use of a very heavy wall which the unit surface weight of the wall must be 550 kg/m^2 al least. If the reinforced concrete is used as constructive materiel, thus the thickness of the wall must be superior to 22 cm.

Acoustic design process can be viewed as a sequence of decisions made at different levels of abstraction with four design phases were identified [9]: (i) - "Massing Stage (M.S.)" : the objective is to study the relationship between the building and its environment. In fact, the acoustical engineer can assure the protection of the integrity of the building with a judicious location and disposition of the building in relation to the surrounding noise sources . (ii)- "Sketches Stage (S.S)" : the acoustical engineer attempts to ensure a complementary integrity protection against outside noise by using features as balconies, flat set backs, ... and to study the spatial distribution of rooms and apartments,... in order to ensure the protection of rooms against inside noise sources. (iii) - "Preliminary Design Stage (P.D.S.)" : if the desired acoustical protection is not achieved, the acoustical engineer relies upon physical elements such as walls and windows. This is the objective of the third design stage. The assessment related to this level is based on global values expressed in dB(A). (iv)- "Detailed Design Stage (D.D.S.)" : sometimes this assessment can give misleading results, which is why acoustical engineers are trained to make spectral analyses and this is the objective of the fourth design stage.

4. TOOLS FOR EVALUATING THE ACOUSTIC PERFORMANCE OF BUILDING

Some existing methods can be used to evaluate the acoustic performances of a building like CSTB's [5] and Qualitel's methods [6]. They give a rough estimation of acoustic performances of design alternatives at later design phases (P.D.S. and D.D.S.), but they require a detailed description of site as well as building components. Therefore, the methods can not be employed to evaluate the acoustic performance at the earlier phases where the most of building parameters have not yet been assigned specific values and where it is difficult to guide the decisions by these methods because of a lack of information. Evaluation is generally based on rules accumulated during experience, . Hence, the tools to used for the evaluation of acoustic properties of design alternatives, are to be founded on heuristic, rules of thumbs.

The development of such tools is described in [10]. They consist of some prototypes expressed as production rules, forming a small expert system. These rules translate the recommendations of some experts into prototypes related, on one hand, to the implementation of residential buildings beside airport and roads,... and, on the other hand, to the organisation of rooms and apartments.

Knowledge Based systems have to be chosen for supporting compute design tools allowing the prediction of acoustic building performances at the early design stages for their multiple advantages [11 ; 12]. In fact, these systems are very appropriate for implementing tools based on heuristic like the tools to be used for evaluating the acoustic performances of building at formative stages. In addition, the K.B.S allows an easier up-dating of knowledge, and powerful Man-Machine interface [11 ; 13].

5. CONCLUDING REMARKS

Our objective is to specify the computer environment for implementing a set of computer design tools which allow to evaluate the acoustic performances of design alternatives corresponding the early design stages. For this purpose, the acoustic building design process was analysed and different levels of abstraction are identified with. Then, the appropriate tools to be employed, for each design stage, is specified. Knowledge based systems are not only suitable but indispensable for supporting tools allowing to evaluate the acoustic building performances at early design stages because of the context which characterise acoustic building design at these stages (context are grossly described).

6. REFERENCES

[1] : P. Chemillier. - Sciences et bâtiments : La démarche scientifique appliquée à la construction. - Edition : Presses de l'école nationale des ponts et chaussées. - 1986, (448 pages).
[2] : G. Achard, M. Karroum, G. Sauce. - Acoustical building design in a multitechnical C.A.D. system. - ARECDAO'93, Barcelone, Espagne, 1993, (pages 307-318).
[3] : G. Achard, M. Karroum, G. Sauce. - Les aspects acoustiques lors de la conception des bâtiments. - Journal de Physique, Volume 4, n° 5, 1994, France. (C5, 163-166).
[4] : R. JOSSE. - Notions d'acoustique à l'usage des architectes, ingénieurs et urbanistes. - Edition : Eyrolles. - Paris, 1977, (292 pages).
[5] : G. Ifert, V. Sauval. - Guide Qualitel. - Association Qualitel, 1989, (173 pages).
[6] : Comment concevoir une protection satisfaisante contre les bruits extérieurs. - Cahier du Centre Scientifique et Technique du Bâtiment, N° 1855, juin, 1983.
[7] : H. P. Zeugin. - Protection acoustique contre les installations des bâtiments : Etude de projet. - Programme d'impulsions : Installation du bâtiment. - Edition : Office fédérale des questions conjoncturelles, Berne, 1989, (pages 47-66).
[8] L. Hamayon, C. Michel. - Guide d'acoustique pour la conception des bâtiments d'habitation. - Paris : Moniteur, 1982 (180 p.).
[9] M. Karroum. - Les performances acoustiques dans un système de C.A.O. multitechnique et evolutif. - Onzième rencontre universitaire en génie civil. - ITU Nîmes, France, Mars, 1993.
[10] M. Karroum. - Modélisation des études acoustiques au cours du processus de conception des bâtiments. Intégration dans un système de C.A.O. multitechnique. - Thèse de doctorat, Université de Savoie, 1994, (221 pages).
[11] S. J. Fenves. - IABSE colloquium on Expert Systems in Civil Engineering. - Bergamo, Italy, 1989,(p. 1-18).
[12] U. Flemming. - Artificial intelligence and design : a mid-term review. - International symposium on computer aided design in architecture and civil engineering. - AREDAO'93, Barcelona, Spain, 1993 (p. 21-36).
[13] : J. L. ERMINE. - Systèmes experts : théorie et pratique. - Edition : Lavoisier- Cachan, 1989, (221 pages).

General Purpose Expert System for Preliminary Structural Design

Structure générale d'un système expert pour l'avant-projet de structures

Struktur eines Expertensystems für die Vorbemessung von Tragwerken

Aldo Cauvin
Prof. of Structural Eng.
University of Pavia
Pavia, Italy

Aldo Cauvin earned his Dr of Eng. degree from Politecnico Milan, where he has been Professor since 1986. Author of papers on nonlinear analysis of reinforced and prestressed structures and on application of expert systems to structural design.

Giuseppe Stagnitto
Research Eng.
University of Pavia
Pavia, Italy

Giuseppe Stagnitto earned his Dr Eng. degree from the Univ. of Pavia and joined the Dept of Struct. Mechanics there. Author of papers on expert systems applied to structural design and structural restoration and strength assessment of Historical Buildings.

SUMMARY
An expert system for the preliminary design of structures is presented. Structural types are classified for a rational organisation of data and knowledge bases. Knowledge is organised in "models" establishing a relationship between structural types and behaviours. A simple "abductive" inference mechanism leads from given boundary conditions to non-univocal reasonable solutions, among which the designer can choose the most suitable one. The process is divided in a series of logical stages which lead to a "prototype" design, permitting an adequate evaluation of quantities and costs.

RÉSUMÉ
Un système expert est proposé pour l'avant-projet de structures. Les types de structures sont classés en vue d'une organisation rationnelle des données. La connaissance est organisée en "modèles" établissant une relation entre types de structure et leurs comportements. Un mécanisme inférentiel simple de type "abductif" conduit, à partir des conditions-limites à des solutions raisonnables et non univoques, et l'ingénieur peut choisir la mieux indiquée. La procédure est divisée en suite logique, conduisant à un projet-type, permettant une évaluation adéquate des quantités et des coûts.

ZUSAMMENFASSUNG
Vorgestellt wird ein Expertensystem für den Vorentwurf von Tragwerken. Die Klassifikation von Bautypen dient als Grundlage für die Datenorganisation. Das Wissen wird in "Modellen" organisiert, die die Bautypen und ihr Verhalten verbinden. Mit einem einfachen ableitenden Schlussverfahren werden aus geltenden Randbedingungen mögliche Lösungen entwickelt, aus denen der Ingenieur die geeignetsten auswählen kann. Der Prozess ist in logische Phasen gegliedert, die zu der Komposition eines "Prototypen" führen, für den hinreichend Massen und Kosten ermittelt werden können.

1.INTRODUCTION

The potential importance of Knowledge Based Systems in the preliminary phase of structural design in which qualitative choices of shapes and materials rather than quantitative evaluations based on repeated structural analyses prevail, was soon recognised[1] [2] especially for those structures,such as bridges and tall buildings ,which are both important and thus require a careful evaluation and selection of types ,and can be classified using limited and well defined structural schemes and components.
These projects ,in analogy with KBES prepared for other fields,and often using "general purpose" shells,use inference mechanisms based on deductive processes(forward and backward chaining).
In the approach which will be presented here a different approach was adopted,in which knowledge is described in the form of "models" and an abductive inference mechanism is used[10].
The resulting expert system,named EXSTRUCT, is described in the following paragraphs.
A particular version of it,devoted to the preliminary design of bridges and named EXBRIDGE2 is described in some detail in [8]

2.GENERAL CLASSIFICATION OF STRUCTURES

The first step in the preparation of an expert system for structural design is the definition of structural types among which a selection must be made in the preliminary stages of the design process.This definition is an uneasy task and,given the enormous variety of structural shapes and layouts which can be used in practice,could never be complete and entirely satisfying.
For some categories of structures this task can be made easier by the fact that in most cases the structural types are well defined and limited in number.Such is the case of bridge structures and tall buildings[1][2][3][4][5].
All this considered,it seems useful to determine a broad classification of structural types which can include most of those structures whose design requires a careful attention,especially in the initial stages,when critical decisions concerning selection of construction materials and structural types need be made.These structures,excluding some very specialized kinds such as dams,reservoirs,retaining walls and so on can be classified in three broad categories(see fig.1):
1)Structures whose most important elements lie essentially in a vertical plane;this is the case of bridges:the main structural elements to be designed are decks(horizontal layout) and piers(vertical layout)
2)Structures whose most important elements lie essentially in an horizontal plane or whose dimensions in the two horizontal directions prevail on the vertical one.This is the case of long span low rise buildings,such as industrial buildings,multistoreys parking lots,assembly and sports halls.The main structural elements in this case are floors ,roofs and theirs supporting beams(all horizontal or sub horizontal elements)
3)Structures whose elements must constitute a system capable of resisting strong horizontal actions.
This system is composed of frames,shear walls,tube elements,braced frames connected by floors and is contained in a 3d space.This is the case of tall buildings and seismic structures.
Of course there are some cases which lie somewhere between situation 2 and 3.
Collection of data pertaining to these categories is now under way as a part of this research project.

3.ADOPTED "MODEL ORIENTED" APPROACH TO EXPERT SYSTEMS IMPLEMENTATION

According to the definition given by Faltings[10] a model is a rule where the premise contains the cause and the conclusion the consequence of that cause(fig.2).In the case of design the cause represents the choice of a given structural component,layout or structural type,the consequence one of the predicted behaviours of that choice .
The relationships among causes and consequences can be represented in a given design space by a graph(fig.2) in which the joints Ci represent the choices to be made,the joints Rj the predicted behaviours,the arrows the models connecting choices and behaviours.
Therefore the knowledge base can be graphically represented by a network of joints and arrows as the one represented again in fig.2.
It is easy to observe that,while, given a choice,all the behaviours corresponding to that choice do occur ,given a behaviour,there are in general more than one choice which can alone produce it.

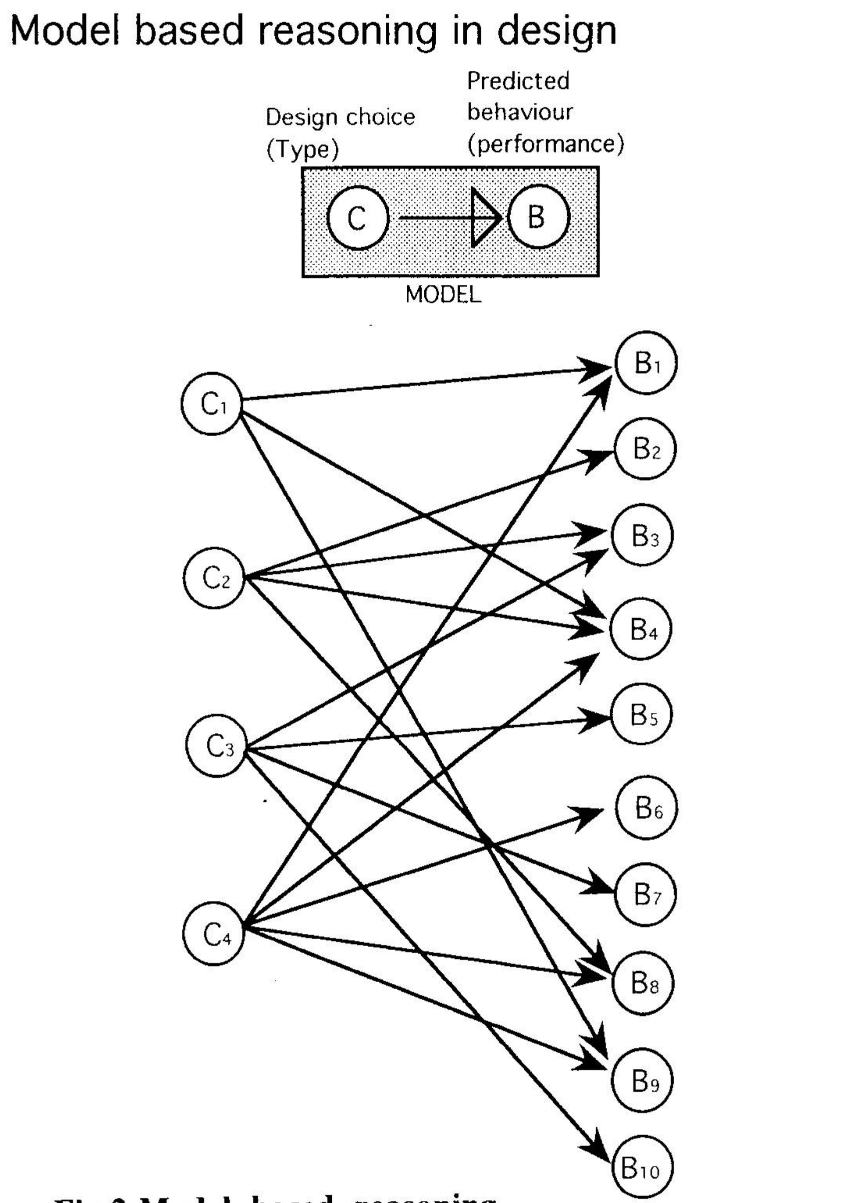


Fig.2-Model based reasoning

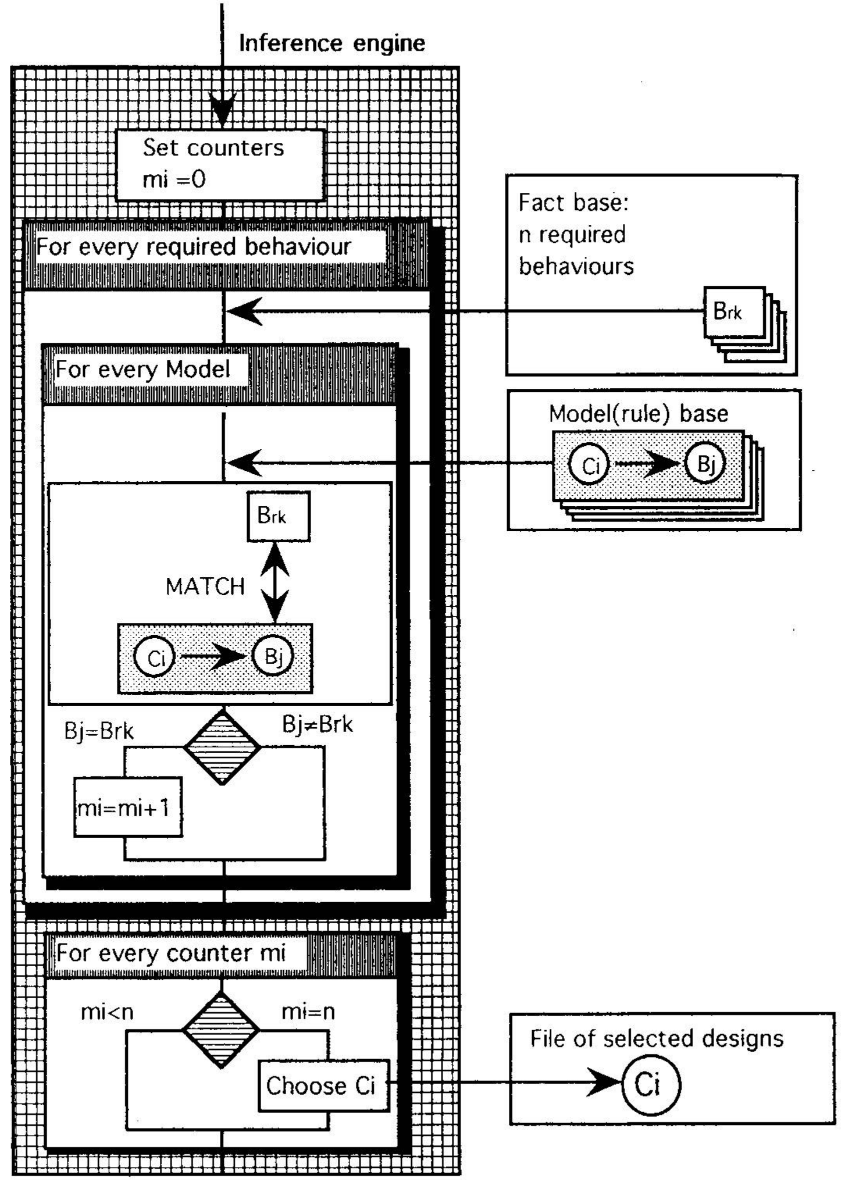


Fig.3-Abductive inference mechanism

The heuristic abductive problem which needs to be solved in the case of design is the following:
-Given a set of known required behaviours(performances)(F1,F2,.....Fk) derived by given boundary conditions ,and the corresponding functions to be fulfilled,determine the set of choices (C1,C2,...Cj) whose predicted behaviours match all the required behaviours and therefore permit to fulfill all the needed functions.
For what has just been said,the choices are not unique and therefore the non univocal,ambiguous nature of design is correctly simulated.
It is at this point up to the designer to choose,somewhat arbitrarily,one of the selected solutions,which are all reasonable as they comply to the boundary conditions.
The inference procedure,which is schematically represented in fig.3 can be briefly described as follows:
-Models are extracted from the knowledge base one by one.
-For each model predicted behaviours(conclusions) are matched against required behaviours(derived from boundary conditions)
-If the predicted behaviour of the model under consideration matches one of the required behaviours,a counter corresponding to the choice(premise of the model)is incremented by one.
-At the end of the process,all the choices whose predicted behaviours match all the required performances(that is whose counter matches the number of required preformances),are considered possible and reasonable designs among which the final selection can be made.
This procedure can be refined according to the procedure illustrated in[8] .
The design space is theoretically unlimited, being unlimited the number of possible structural types,components and layouts[1*] .
However,in the field under consideration, it is quite reasonable to reduce the possible choices to a limited number of well defined types,whose association,composition and sizing can generate a "prototype" design.
In this way the space to explore is limited and the type of design can be considered as a "routine" design according to the classification given by Gero[5] in the sense that both functions and possible types are well defined (It is not certainly however a routine design in the common sense of the word!).
The design space can be furtherly reduced by the fact the design process of a complex "artifact" such as a civil structure can be considered as an assemblage of well defined modules and must be executed in different steps which are,to some extent,independent from one another.
We can therefore divide the generic design space in a number of "specialized" design spaces according to a procedure which can be defined as"top down refinement plus constraint propagation"[11]as it will be explained in the next paragraph.

4.MODULAR NATURE OF STRUCTURAL DESIGN

The process of preliminary structural design can be divided in the following phases:
-Definition of basic boundary conditions
-Definition of general required behaviours(specifications)
-Definition of basic layout
-Definition of shapes
-Preliminary Sizing
-Preliminary quantities and costs evaluation.
The first two phases constitute the formulation of the problem and require the use of deductive processes.
The following three phases constitute the design part of the process and can in turn be divided in a number of "levels" as specified in fig.4
Each level can be processed in an independent way,with its own design space,proceding from top levels down to the lower ones(that is from more general design choices to increasingly more detailed designs)provided that specifications directly deriving from choices performed at the preceding

1

*collectively called "structures" in ref.[5].We won't however use this term to avoid confusion with the term structure,intended in the proper sense.We use the term "choice" or "type"instead.

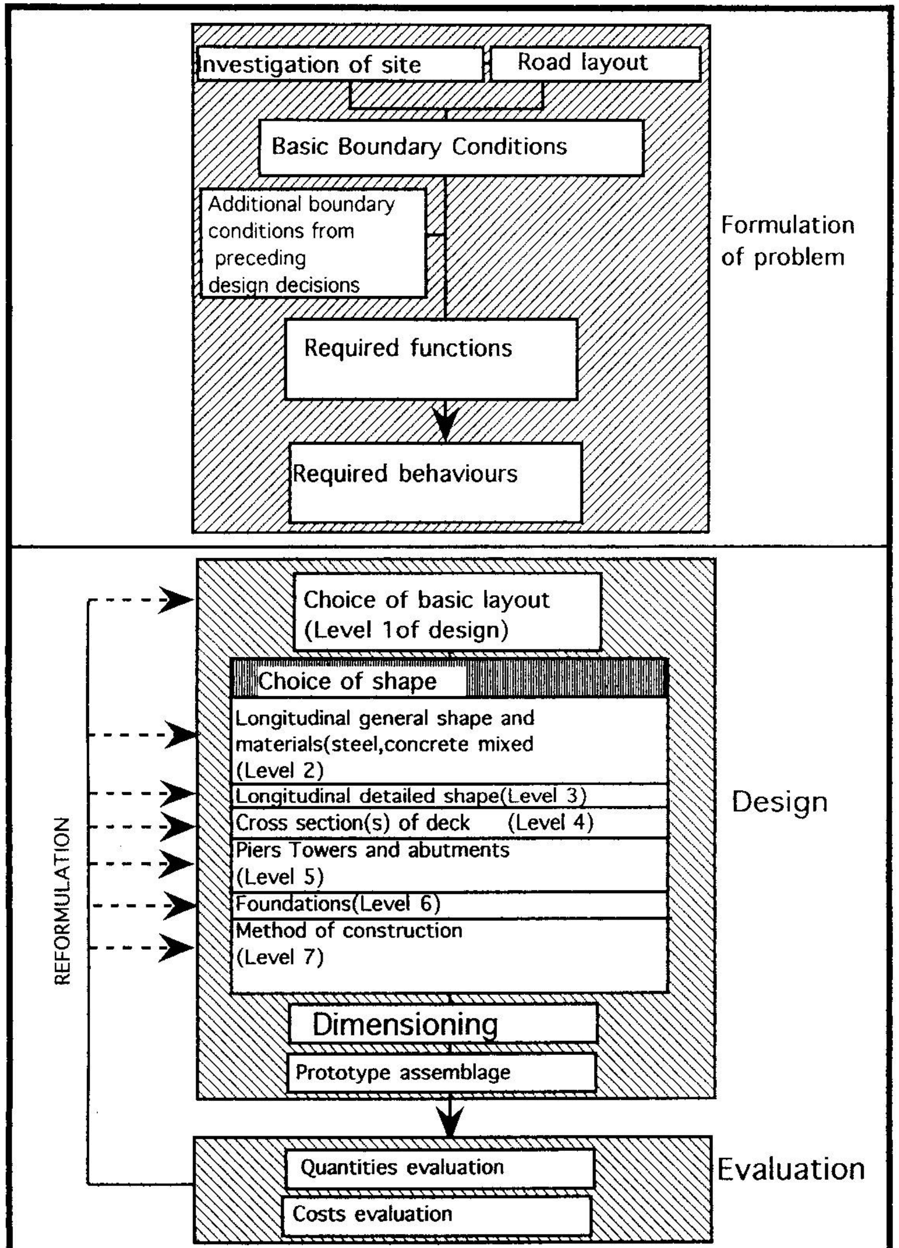


Fig.4-Levels of design in the case of bridges

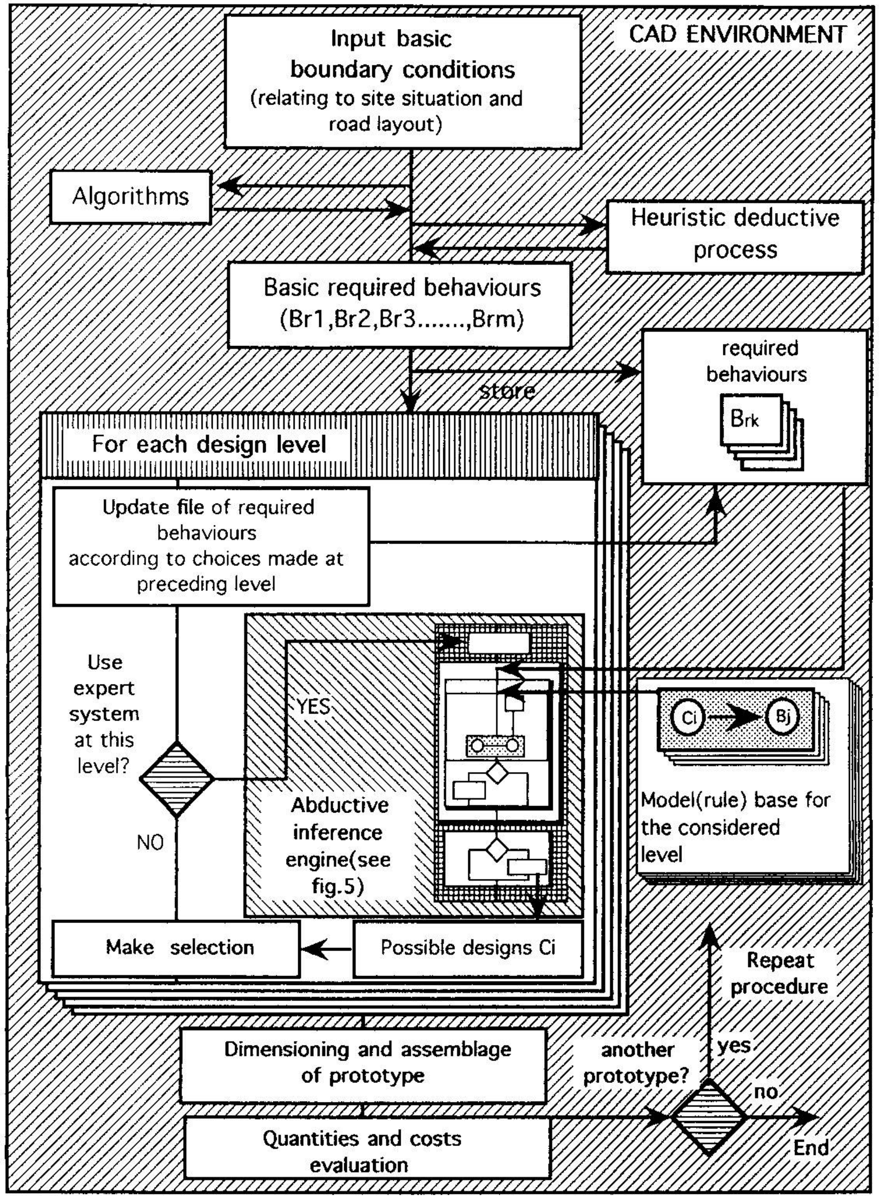


Fig.5-Macro flow-chart of the system

levels are added to the general specifications of the problem to be processed (by an abductive inference engine) at the level under consideration.
The preliminary sizing phase does not utilize procedures of Artificial Intelligence but rather statistical data transformed into suitable diagrams,and logical considerations related to the shape and compatibility of the chosen parts of the design,and also "rules of thumb" derived by common design sizing practice.
Once the various parts of the design are dimensioned they can be assembled in a "prototype" design which ,although generical,stereotypical and not detailed,should contain enough information for the evaluation process of the chosen structural type which must follow.
At last the quantities evaluation,obtained by using a spreadsheet associated with each part of prototype design which automatically update itself as dimensions are changed,provides an analytical mean of computing expected costs.A more syntetical mean would be provided by manipulation of parametric statistical data concerning costs extracted by a number of similar executed designs.
The choice and sizing of structural types and components,as well as their manipulation and assemblage in prototype designs is greatly facilitated if the system is programmed within a CAD system which can provide an enormous amount of graphic functions and facilities.This is made possible by the use of programmable CAD environments such as AUTOCAD™ and MINICAD™.
Let us at this point consider with some detail the nature and the tasks to be performed at each level,illustrating the process with an hypotetical design example.

5.DEFINITION OF BASIC LAYOUT

The first level of design concerns the determination of basic layout of the structure. In the case of bridges five types of basic layouts have been determined,in which most of the possible cases can be included.
The determination of the layout type and the values of spans can be determined taking as input data the basic required behaviours deduced from site situations and the road or railway layout.
In the given example,concerning the preliminary design of a cable-stayed bridge and schematically illustrated in fig.6, the inference procedure leads to the choice of two layouts between which the layout consisting of a large central span with two lateral smaller ones was selected.

6.DEFINITION OF SHAPE

The complete individuation of the bridge shape is accomplished in the next six levels:
The second level concerns the determination of the general longitudinal shape(structural scheme) ,which is associated closely with the choice of the construction material.
A simplified set of rules for this level is represented on fig.7,whose only purpose is to illustrate the logics of the process without any pretension of being complete.
To make these choices the specifications deriving from the preceding level are added to the basic specifications .
This procedure will be repeated in the following levels,every time a given choice increases the number of boundary conditions to be accounted for.

The third level concerns the refinement of the longitudinal shape .In the given example,this refinement consists in the choice of arrangement of the stay cables.
From this stage downwards the choices are being made among well defined components of the design which can be parametrized.
The fourth level concerns the choice of the type of deck cross section
The fifth level concerns the choice of vertical supports(towers,piers and abutments).
The sixth level concerns the choice of foundation types.
At last the seventh level concerns the choice of construction method
In the scheme of fig.6 the path which was followed for the given example,as well as the possible alternatives which could have been reasonably chosen are illustrated.

7.SIZING AND PROTOTYPE ASSEMBLAGE

At this point the general shape of the bridge and of its main components is clearly defined.

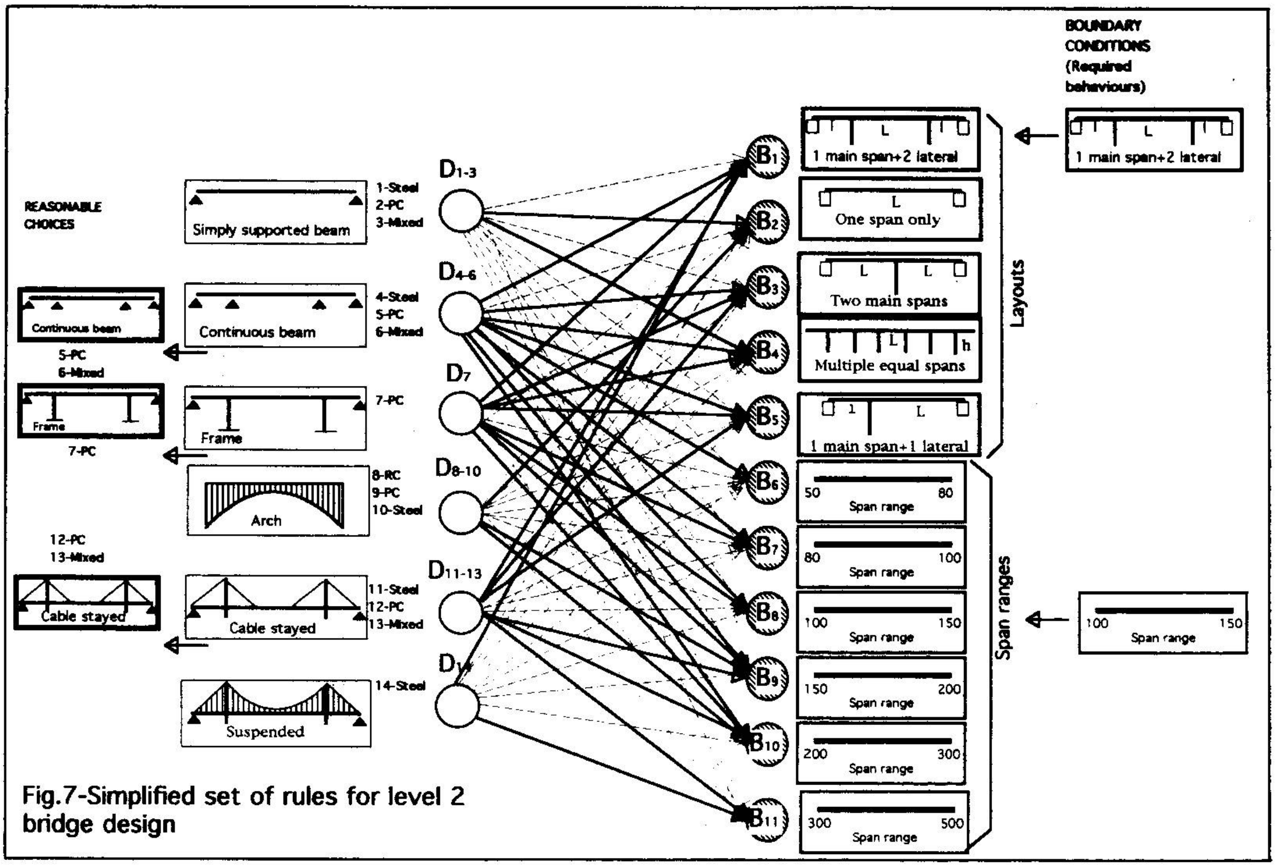


Fig.7-Simplified set of rules for level 2 bridge design

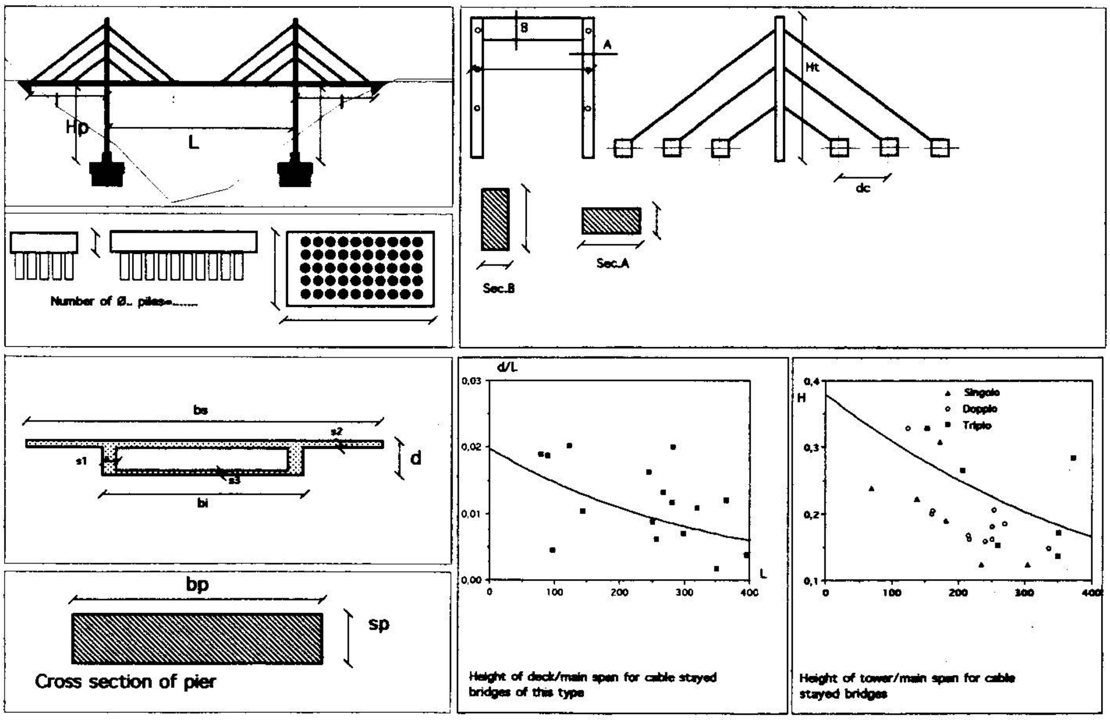


Fig.8-Preliminary dimensioning

To evaluate the obtained design and,in case,to perform preliminary structural analyses on it,it is needed to complete the definition of dimensions.
The sizes needed to univocally determine the project can be divided into four categories:
-Sizes which are defined from the beginning(for example the deck width) or as a consequence of the basic layout choice(the length of the spans,the height of piers)
-Sizes which can be defined statistically by comparison with designs of about the same shapes and span length.
-Sizes which can be determined using empirical "rule of thumb" rules in function of the already determined shapes.For example the depth of th web in a beam can be established in function of its height.
-Sizes which are established to insure compatibility between connected parametrized parts of the design.
Using these criteria the parts of the bridge are completely defined and can be assembled in a "prototype" on which the design evaluation can be performed .
The "sizing" process is schematically represented on fig.8

8.MACRO FLOW-CHART OF SYSTEM

The self explanatory macro flow chart of the system is represented on fig.5.
It can be divided into three parts according to the functions to be performed:formulation ,design and evaluation.
It is interesting to notice that,at each design level the use of the inference engine is not mandatory,but the design choice can be made directly in those cases where the knowledge base is not considered sufficient,reliable or it lacks completely;it is thus possible to use the procedure even when the collection of knowledge is far from complete.

9.CONCLUSIONS:ADVANTAGES OF CHOSEN APPROACH

The advantages of the chosen"top down refinement plus constraints propagation" approach can be summarized as follows:
-The division of design and therefore of the knowlege base in different stages both simplifies the formulation of rules and reduces the number of operations to be performed by the inference engine at each stage.
This approach also has drawbacks in the sense that the exploration of different and to some extent independent design spaces can lead to disregard some connections among the different stages;however the fact that design proceeds hierarchically"top down" that is,from general to more particular problems permits to conclude that the design spaces at the lower levels are in some way included in those at the upper ones (and therefore their rules do not normally contradict those at the upper levels).In other words it is assumed that "strategic" choices do influence more particular ones while it seldom happens that a particular decision can invalidate a strategic choice.Also the procedure is usually an iterative one,in the sense that it is repeated several time to get a number of possible design among which economical comparisons can be made.During this process possible contradictions can be detected and eliminated.
-The abductive approach to models elaborations does not eliminate artificially the inherent ambiguity of the design process and permits to obtain not a single but a range of possible designs.In this way the creativity of the designer is ,even at this stage, inhibited only to the extent that absurd or irrational choices can be avoided.
.The final decision stems from the comparison of these designs,which can be obtained quickly,with a sufficient degree of detailing to make these comparisons highly meaningful and reasonably objective.
-The inclusion of the process in a CAD environment which can be performed easily ,given the availability of programmable CAD systems,permits to interact graphically with the system in a very intuitive and clear way,avoiding lengthy and ambiguous descriptions of results.
-The modular nature of the process permits to skip some of the stages by directly making the decision at those stages.
In this way the system can be usefully exploited even if the model base is far from complete.
We could conclude saying that this framework works both for general purpose and for specialized Expert Systems devoted to design.

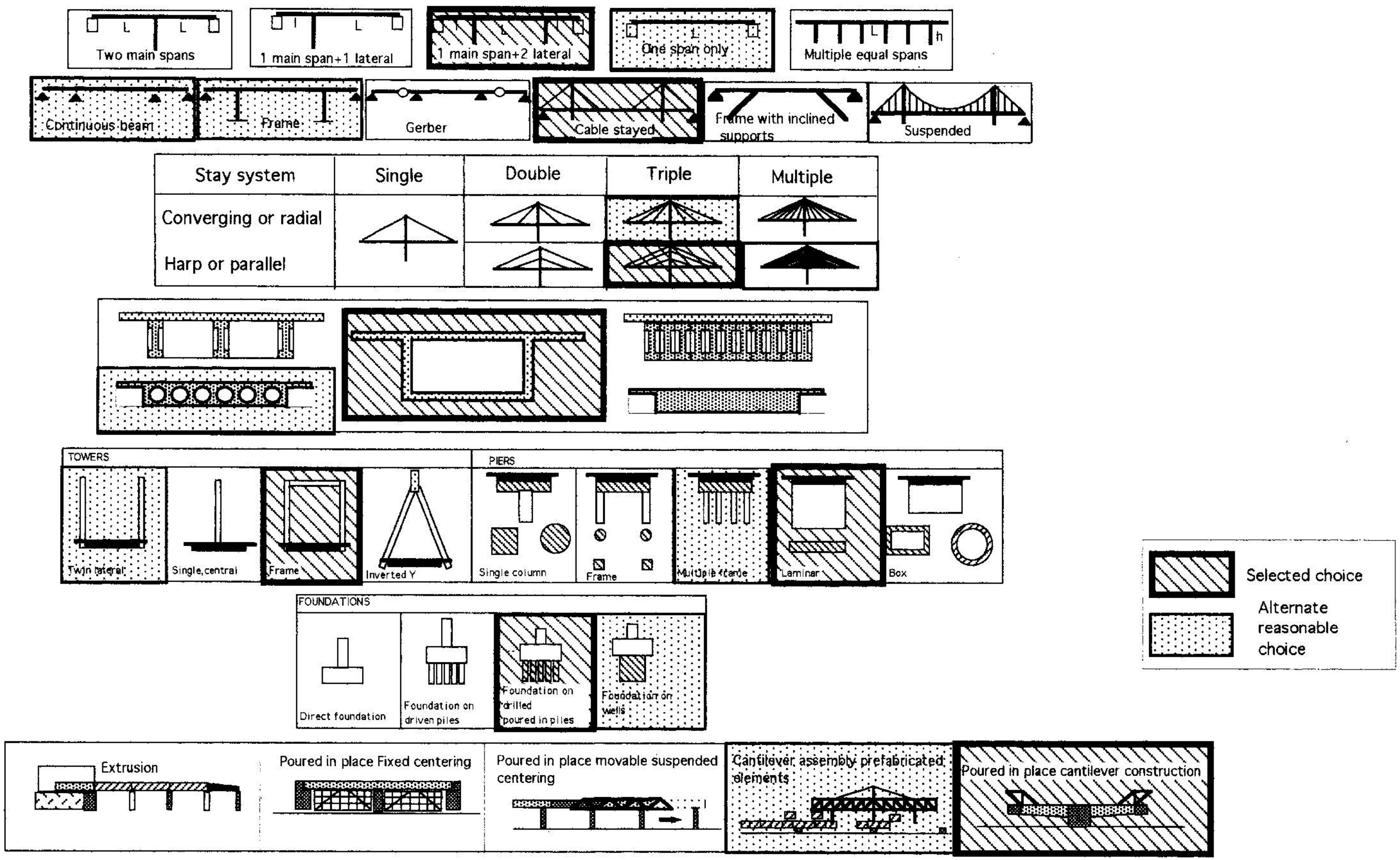


Fig.9-Example of bridge design

10. REFERENCES

[1]Maher M.L.,Fenves S.J.,HI-RISE:an Expert System for preliminary design of High Rise buildings,Technical Report,Dept of Civil Engineering,Carnegie Mellon University,1984
[2]Fenves S J ,Maher M.L.,Sriram D.,Knowledge based Expert Systems in Civil Engineering",IABSE Periodica,Number 4,1985
[3]Fenves S J,Expert Systems,Expectations versus realities,Proceedings of IABSE Colloquium "Expert Systems in Civil Engineering",Bergamo 1989.
[4] Moore C.J.Miles J.C.,Development of a KBS for Conceptual Bridge Design,Proceedings of IABSE Colloquium,Beijing 1993
[5]Fang et al.,Role of modular expert systems in planning,analysis and design of large structural systems,proceedings of 4th world congress"Tall Buildings,2000 and beyond",Council on Tall Buildings and Urban Habitat,Hong Kong,1990
[6]Gero J.S.,Design prototypes:a knowledge representation schema for design. AI magazine,winter 1990.
[7]Cauvin A,Stagnitto G.,Classification of Structures and organization of hypermedia data bases for Structural Design,First workshop of EG-SEA-AI,EPFL,Lausanne 1994.
[8]Cauvin A,Stagnitto G.,A "Model Oriented" Expert system for the preliminary design of bridge structures,First workshop of EG-SEA-AI,EPFL,Lausanne 1994.
[9]Tong C,Sriram,Artificial Intelligence in Engineering Design,Vol.1,chapt.1,introduction,Academic Press Inc.,1992
[10]Faltings B,Reasoning Strategies for Engineering Problems,Proceedings of IABSE Colloquium "Knowledge Based Systems in Civil Engineering",Bejing 1993.
[11]Steinberg L,Design as Top Down Refinement Plus Constraint Propagation,from Chapter 8 of book:Artificial Intelligence in Engineering Design,Vol.1,Academic Press Inc.,1992

AKNOWLEDGEMENTS

This research has been performed with the partial financial support of CNR(Italian Council for research)The Authors warmly express they thanks to Doctor Diego Stagnitto for his programming advice.

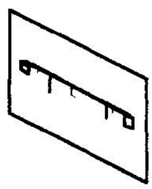
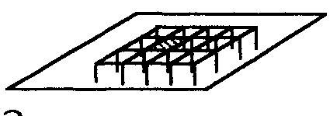
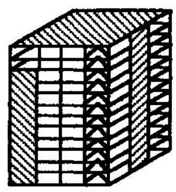
GROUP	MAIN STRUCTURAL ELEMENTS	TYPE OF CONSTRUCTION
 1	DECKS (and their supporting structures such as arches and stays) PIERS	BRIDGES OVERPASSES
 2	FLOORS ROOFS (and their supporting beams)	INDUSTRIAL BUILDINGS ASSEMBLY HALLS SPORTS HALLS MULTISTOREY PARKINGS etc
 3	ALL THE LATERAL FORCES RESISTING 3D SYSTEM	TALL BUILDINGS SEISMIC STRUCTURES

Fig.1-Groups of structures according to given classification

Representing Designs by Composition Graphs

Représentation de projets par graphes de composition

Die Wiedergabe von Entwürfen durch CP-Graphen

Adam BORKOWSKI
Prof.
IFTR
Warsaw, Poland

Adam Borkowski, born 1942, received his civil eng. degree at Techn. Univ. of Kaunas, Lithuania. He leads the Lab. of Adaptive Systems at the Inst. of Fundamental Technol. Res. (IFTR), Warsaw, Poland. His interests include applying AI-techniques to CAD and robotics.

Ewa GRABSKA
Assist. Prof.
Jagiellonian University
Cracow, Poland

Ewa Grabska, received her Ph.D. degree in math. from Jagiellonian Univ. in 1982. Since 1976 she has been with the Inst. of Computer Science of the same university. Her current research interests include graph transformations, computer graphics, CAD and computer aesthetics.

SUMMARY
The linguistic approach to the representation of knowledge about the shape and dimensions of the designed object is considered. A two-stage description is proposed: the composition graphs for representing topological features and the realisation schemes for generating instances of objects. Several examples of the conceptual designs of bridges demonstrate the ability of the proposed approach to support browsing through alternatives and adjusting geometrical properties.

RÉSUMÉ
Une approche linguistique est proposée pour la représentation de la connaissance de la forme et des dimensions d'un objet. Un graphe de composition représente la topologie et un schéma de réalisation produit les objets spécifiques. L'approche visuelle fournit des alternatives de représentation graphique pour chaque conception. Des exemples illustrent les différentes conceptions et démontrent le formalisme par le biais de propriétés géométriques ajustées.

ZUSAMMENFASSUNG
Der Artikel ist einer sprachorientierter Repräsentation des Wissens über Form und Abmessungen eines Entwurfsobjektes gewidmet. Eine zweistufige Beschreibung wiird vorgeschlagen: Die Mischdiagramme dienen der Festlegung von topologischen Eigenschaften des Objektes und die Entwurfsrepräsentationen generieren spezifische Austragungen. Ein paar Beispiele aus dem Gebiet des Brückenentwurfs zeigen, dass der vorgeschlagene Formalismus in der Lage ist, die Suche nach alternativen Lösungen und die Anpassung der geometrischen Eigenschaften zu unterstützen.

1. INTRODUCTION

Computer Aided Design systems were dedicated at the beginning of their history to pure geometrical modelling. Nowadays we expect much more from such tools: they evolve towards hollistic design environments that should facilitate an integration of the various aspects and phases of engineering design process [2, 11, 12]. Nevertheless, the geometric modelling remains the basis upon which all other elements of thinking about the designed object are built.

A possibility of applying the methodology of formal linguistic to the description of design objects was recognized early enough [10, 15]. An exhaustive discussion of that topic can be found in [5, 6], whereas the most recent contribution is probably the PhD-thesis [14].

Geometrical modelling can be discussed within the framework of graph transformations which constitute a relatively young discipline of computer science. Graph transformations are applied in many fields of computer science, because they are useful for modelling of a wide range of non-numerical computation [13]. The most interesting case in graph transformations arises when the effect of transformation is local, i.e., it can be specified by rules for transforming subgraphs of limited size. Allowing only a finite set of such rules leads to a syntactic model called a *graph grammar*. The process of transforming graphs by the iterative application of these rules is called *derivation* or *generating process* of the graph grammar.

A conventional approach to the shape grammar is to treat it as a generative system allowing the user to compose objects from geometrical primitives. The present paper develops further linguistic approach to the knowledge representation in engineering design. In the approach, the notion of graph grammar describing topological properties of the searched for objects is complemented by the concepts of *realization scheme* and *control diagram*. The realization scheme defines a specific instance of the generic object. Hence, it deals with the coordinates, transformations, dimensions and constraints. The control diagram selects specific productions from the grammar and defines the order of their application.

Given the limited length of the paper, we present in Section 2. only a brief summary of the proposed methodology. An exhaustive description of it can be found in [8]. Section 3. contains the discussion of the requirements that must be met by a system representing knowledge about the shape of the object to be designed. Several examples showing how the proposed formalism works in the design of bridges are given in Section 4..

2. CP-GRAPHS AND REALIZATION SCHEMES

In this Section the notions of *composition graph (CP-graph)* and *realization scheme* are presented. CP-graphs are directed labelled graphs which allow us to represent topological features of objects. Realization schemes are mappings which assign object instances to CP-graphs representing object topology [8].

Let us consider the bridge in Fig. 1a and its representation by means of the CP-graph in Fig. 1b. To describe all necessary relations between components of the bridge we would like to have a chance of describing on graph level the relations not only between the whole components (transformed primitives) but also between fragments of these components. To achieve this we equip graph nodes with two types of labels.

The first type of labels is the name of a primitive. The following prototypical parts: an abutment (ab), a beam (bm), a pylon (pl), a cable1 ($cb1$), and a cable2 ($cb2$) are distinguished for the bridge in Fig. 1a. The abbreviations of primitives ab, bm, pl, $cb1$, and $cb2$ are node-labels of the

CP-graph shown in Fig. 1b.

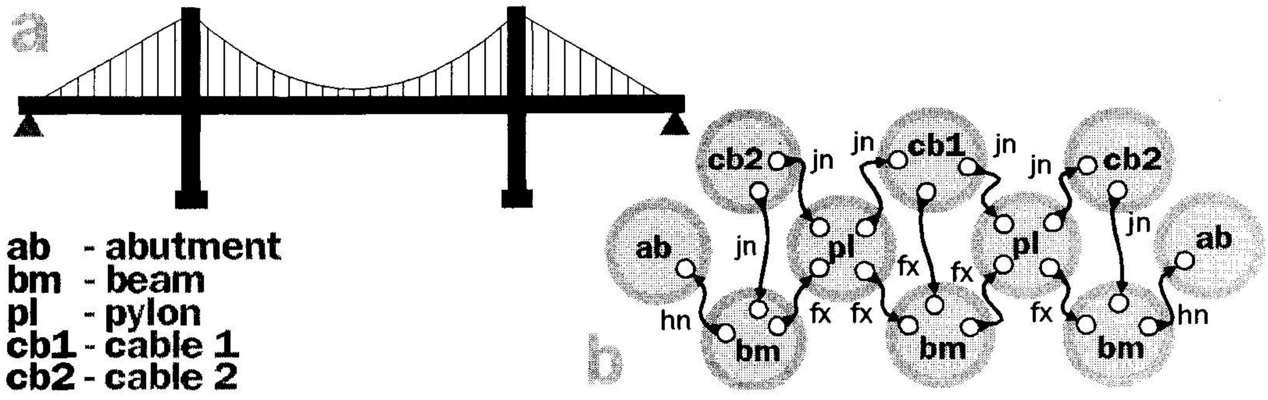


Fig. 1 The bridge and its CP-graph.

The second type of labels consists of node bonds. For a given node, in-bonds and out-bonds are defined. They correspond to fragments of the component representing by the node and express incoming and outcoming connections to and from the node.

We draw a picture of the CP-graph representing in-bonds and out-bonds as graphic symbols ȣ and ɵ, respectively, and edges as lines going from out-bonds to in-bonds.

Apart from labelling nodes of CP-graphs it is also useful to label their edges. Different edge labels can define different types of connections between the components of the designed object. In our example, the edge labels f_x and h_n denote fixed and hinged connections between components of the bridge. The edge label j_n denotes the join relation.

We show details of the bonding model for the part (Fig. 2a) of the bridge in Fig. 1a. Fig. 2b

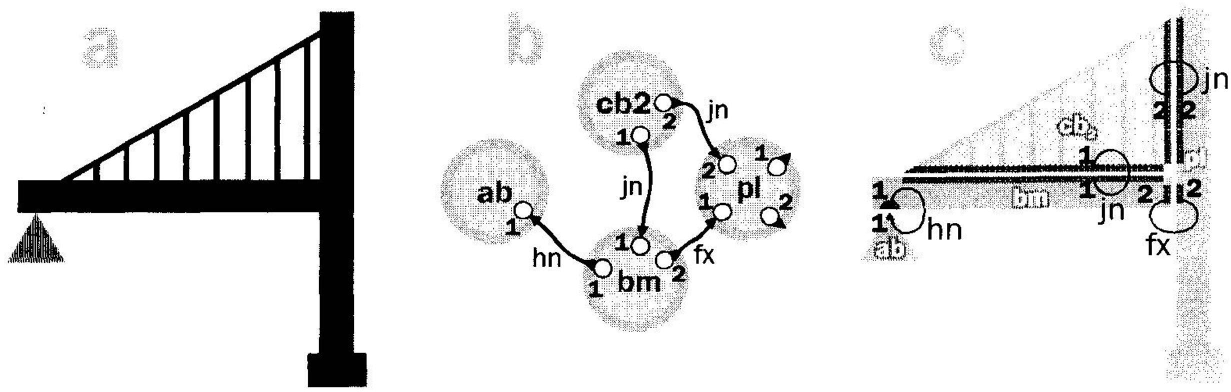


Fig. 2 The bonding model for the part of the bridge in Fig. 1.

presents the subgraph whose nodes with labels *ab*, *bm*, *cb*2 and *pl* correspond to the abutment, beam, cable2 and pylon of this part, respectively, and the bonds of the nodes represent the fragments of the components (Fig. 2c). To each pair of bonds connected by the edge in Fig. 2b corresponds the pair of fragments of two components, which are marked by a circle in Fig. 2c.

Nodes of CP-graphs play not only a static role as building components but also a dynamic role as potential places to locate convenient CP-graphs. In Section 4. we shall use context-free CP-graph grammar as a finite mechanism for producing CP-graphs which will represent relationships among components of bridges. A CP-graph grammar is similar to an ordinary context-free grammar. In the former, nodes of CP-graphs are rewritten by CP-graphs. In the latter, symbols of strings are replaced by strings. When using a grammar, rules of replacing nodes by other CP-graphs may be repeated and as a result of this similar parts of designs may appear [9].

Our approach to the knowledge representation in design makes it also possible to generate drawings of designs (graphical models). Ways of realizations of graphical models are defined by means of the so called realization schemes determined for CP-graphs.
A *realization scheme* for given set of CP-graphs consists of

1. a family of predicates which describe design criteria,
2. a set of admissible transformations in a Euclidean space,
3. a set of primitives in a Euclidean space,
4. a mapping, called the *prototype assignment*, which assigns primitives to the nodes of CP-graphs,
5. a mapping, called the *fragment assignment*, which assigns fragments of primitives to bond nodes of CP-graphs,
6. a function, called the *fitting function*, whose values are admissible transformations allowing to determine transformations which the primitives must undergo to transform them into design components.

Given a CP-graph and a realization scheme, a graphical model is obtained by a union of the transformed primitives, where the primitives directly and the transformations indirectly are specified by the realization scheme [9]. For the CP-graph shown in Fig. 3a three drawings of

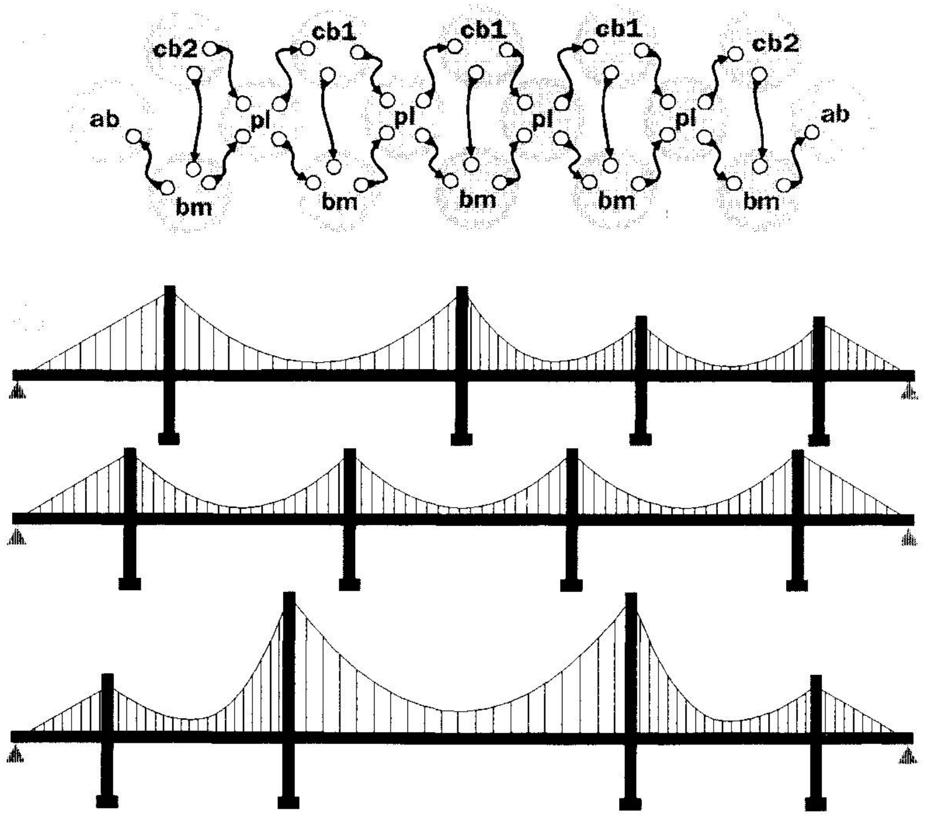


Fig. 3 A two stage description: a) the CP-graph representing topological features of potential bridges, b) three bridges specified by three different realization schemes for that CP-graph.

bridges presented in Fig. 3b are determined by three different realization schemes with the same set of primitives but different sets of admissible transformations and different fitting functions.

The fact that the structure of the object described by the CP-graph is independent of its realization may be useful in the process of designing. Due to it we can design an object applying one of two strategies: one that consists in changing of connections between nodes of the CP-graph, while primitives remain unchanged, or the other changing primitives or transformations, while the CP-graph remains unchanged.

3. REPRESENTING KNOWLEDGE ABOUT DESIGN

It is commonly agreed to distinguish two aspects of computer-aided design: the product modelling and the process modelling [12]. The former deals with more or less static representation of the knowledge about a specific class of artefacts. The latter should reflect the dynamic character of search for alternative designs. Our methodology conforms with such a dual description. The CP-graph generated by the grammar corresponds to the product model: it defines topological properties of the structure. The realization scheme and the control diagram describe the process model allowing the user to explore the space of admissible solutions.

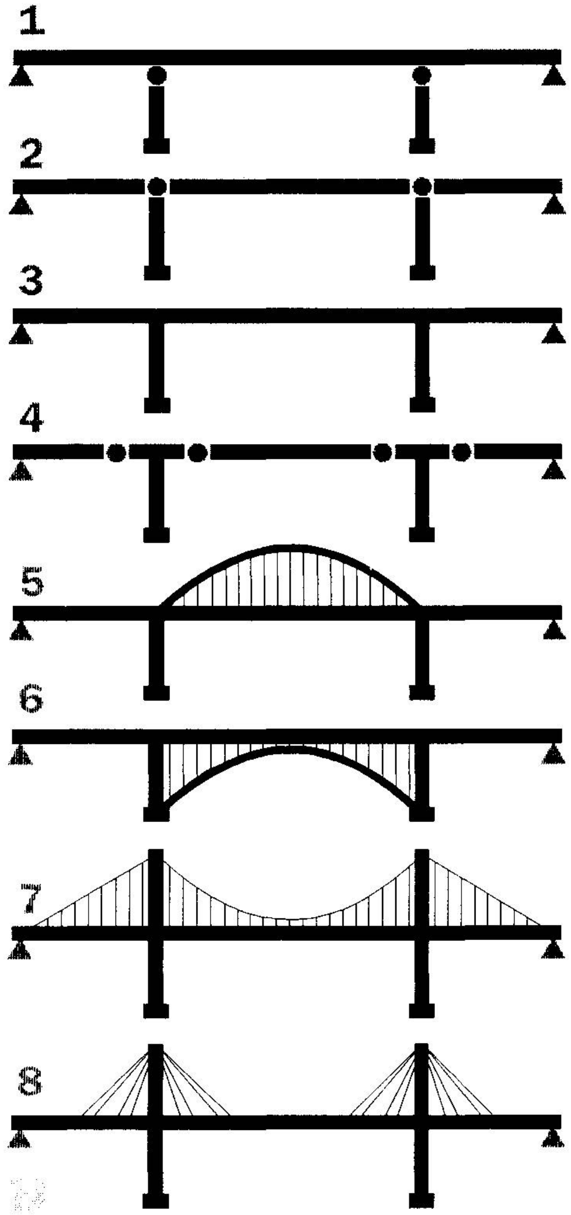
Each engineering object can be seen as an assembly of parts occupying specific portions of a Euclidean space. Such a decomposition can be performed recursively resulting in a hierarchical tree. The root of this tree corresponds to the entire object, intermediate nodes represent its subsystems and the leaves correspond to primitives.

Typically the design process follows the top-down pattern: at first the layout of the main subsystems is found, then each of them is considered separately [11]. If at a certain level of the decomposition tree an unsolvable situation is detected, then the process is turned back to the higher level in order to find an alternative solution. Thus engineering design can be seen as a repetitive transversal of the spiral path from the general idea of the artefact towards its detailed project. The trial-and-error nature of the search of optimum solution, inevitable due to complexity of real world problems, shows where assistance of the computer is most efficient. It should enable the human designer to explore the space of plausible solutions in a comfortable and efficient manner. It is also of great importance that modifications introduced at a particular level of the decomposed problem are properly propagated throughout the database in order to maintain it in a consistent state.

The old good rule says: never submerse into details before considering general goal to be accomplished. Thus, designing an engineering object should begin with a conceptual phase, where a notion of plausible solution is worked out. In our formalism this means establishing graph-production rules which express possible topological relations between the principal parts of the desired object. This can be done using a special purpose graph-editor.

After the user has defined the rules, the system knows the universe of plausible solutions. Upon request these rules can be browsed and edited by the user. Next the realization scheme is to be described. Here the user must enter data related to the coordinates of the designed object with respect to a specific reference frame, define characteristic dimensions of the object and write the predicates responsible for geometrical constraints.

Finally, the control diagram, defining which production rules in which order are to be applied, is to be specified. This completes the description of the object and the visualization module can be invoked in order to evaluate the result and to decide upon possible changes. The usage of the proposed system is easiest to explain on examples.



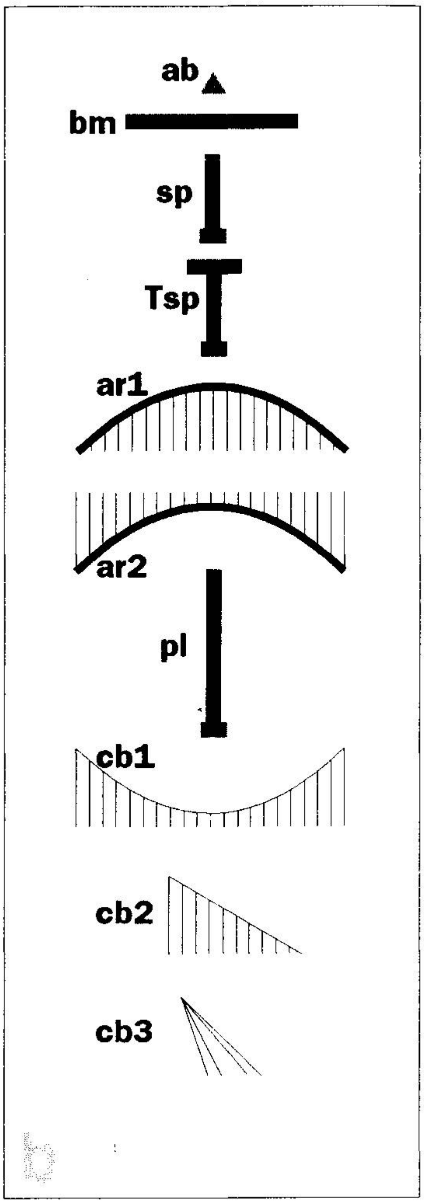


Fig. 4 Different types of bridges with their primitives: a) eight types of bridges, b) the set of labelled primitives for the bridges.

4. EXAMPLES OF APPLICATION

In this Section we give several examples to show how the proposed formalism works in the design of bridges. We define explicit syntaxes and semantics for computer modelling of bridges.

Fig. 4a presents different types of bridges which can be generated by CP-graph grammar. Fig. 4b shows the set of primitives with the abbreviations of their names. In Fig. 5 we present 6 syntactic rules $p_1 \div p_6$ with their typical interpretations $i1 \div i2$, which allow to generate the bridge numbered 7 depicted in Fig. 4a, and its modifications. The grammar for generation of all bridges shown in Fig. 4a contains 33 rules. As we can see, each rule is composed of two parts - its left-hand side (a CP-graph node) and its right-hand side (a CP-graph). The symbol $\Longrightarrow$ serves as a replacement operator which informs that the left-hand side of a rule is to

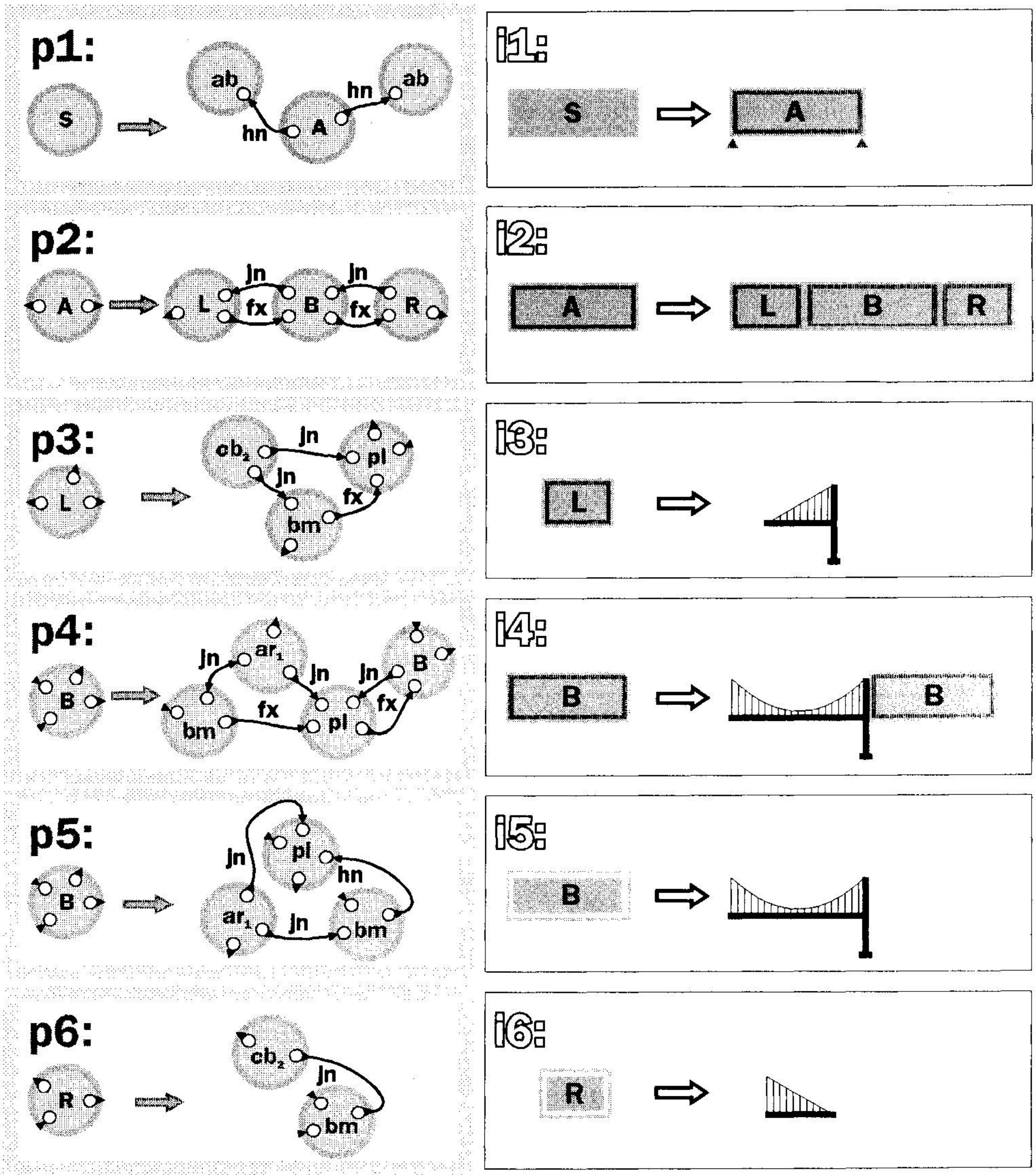


Fig. 5 Rules and their interpretations for the bridge numbered 7 in Fig. 4a and its modifacations.

be replaced by the right-hand side of the rule. The nodes of CP-graphs in rules, with labels being the abbreviations of names of primitives (Fig. 3b), correspond to these primitives and are terminal. The remaining CP-graph nodes are nonterminal. The edge labels f_x, h_n denote fixed and hinged connections between components. The edge-label j_n denotes the joint relation between components.

CAD requires knowledge about plans or sequences of rules rather than simply about individual rules. Therefore some means of representing and using control knowledge for CP-graph grammars are needed. The approach proposed in the following was originally introduced in [7]. We use a graphical notation for the control algorithm, which seems to be more illustrative [4].

A *control diagram* for a finite set P of productions is a connected directed labelled graph with the alphabet $P \cup \{I, F\}$ of node labels and satisfying the following conditions:

1. there exists exactly one initial node labelled with I,
2. there exists exactly one final node labelled with F,
3. there exists no edge terminating in the initial node, and
4. there exists no edge leaving the final node.

An example of a control diagram is shown in Fig. 6. With exception of the initial and the final

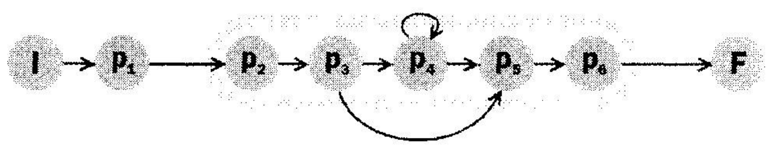


Fig. 6 The control diagram for the grammar rules in Fig. 5.

node, all other nodes are labelled with productions. Applying a derivation process according to the order stated in the control diagram, we start with a production which is the label of a direct successor of the initial node. The derivation process stops when the final node is reached. Each path from the initial node to the final node allows us to generate the CP-graph belonging to the CP-graph language under consideration. The control diagram shown in Fig. 6 allows to generate such CP-graphs which describe the structure of the bridges numbered 7 in Fig. 4a, and its modifications.

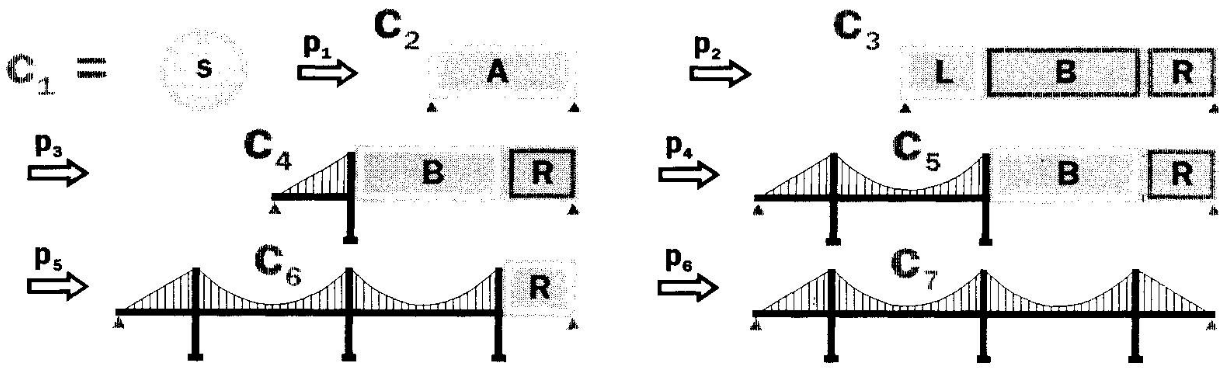


Fig. 7 Generation of a modification of the bridge 7.

When equipping terminal nodes of the generated CP-graphs with attributes being transformations we create drawing of bridges. Drawings corresponding to such CP-graphs are the unions of primitives determined by labels of terminal nodes and transformed according to the attributes assigned to these nodes. The primitives directly and the transformations indirectly are specified by the realization scheme. Each drawing can be formed step by step simultaneously with the derivation of the CP-graph. Fig. 7 presents fragments of the drawing and non-terminal parts obtained during the generation of the CP-graph describing the bridge being a modification of the bridge 7 in Fig. 4a. In this generation we start with the CP-graph node c_1 with label S and generate the sequence of CP-graphs c_2, c_3, c_4, c_5, and c_6, by means of the sequence of the productions p_1, p_2, p_3, p_4, p_5, p_6 determining one of the pathes of the control diagram in Fig. 6.

A typical session with the computer program based upon the proposed methodology is the last example which will be considered in the paper. Fig. 8a shows the initial situation faced by the designer: a profile of a river valley that should be transversed by the bridge. Choosing from the

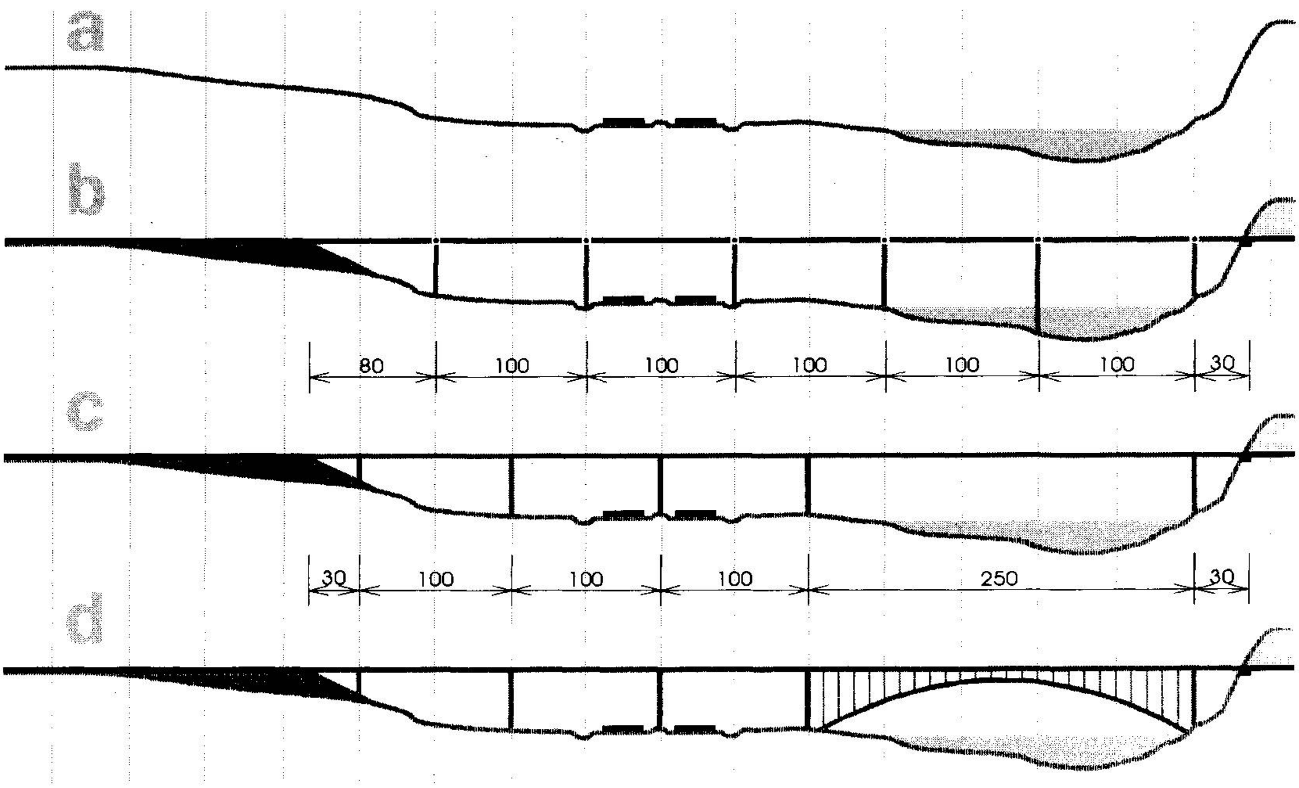


Fig. 8 Choosing the layout of a bridge: a) the cross-section of the valley, b) initial version - a bridge composed of the simple supported beams, c) a continuos beam solution - the support standing in the river removed, d) the arch added for aesthetical reasons.

alternatives proposed by the program the user takes the chain of 6 simply supported beams as the first alternative (Fig. 8b). It is easy to try different positions of the supports because this requires only the changes of few numerical parameters (the lengths of spans) in the realization scheme. Note that the knowledge about plausible span lengths is stored in the constraints of the realization scheme.

Assume that the user is not satisfied with the scheme shown in Fig. 8b. It presumes placing a support in the river itself which is expensive and might be opposed by the ship owners. An obvious alternative is the continuous beam bridge over the river shown in Fig. 8c. Removing a support demands removing an appropriate production in the production sequence of the control diagram and the changes of the lengths of spans in the realization scheme.

For aesthetical reasons one might consider putting an arch over the river (Fig. 8d). In order to generate such a bridge it is sufficient to add an appropriate production to the production sequence, which allows to add a new node (representing the arch primitive) to the CP-graph. Again the user can easily adjust the lengths or consider various support conditions for individual spans.

5. CONCLUSION

Preliminary experience gained so far indicates that the proposed format is quite convenient for storing and manipulating knowledge about the topology and geometry of the designed object. A clear cut separation of the topological backbone and of the specific realization allows the designer to investigate alternative solutions at two levels. Generating alternative objects amounts simply to the transversal of the control graph, whereas scaling and adjusting the dimensions of the object is performed by changing the predicates and parameters of the realization scheme.

Our further aims include merging together composition graphs related to the different perspectives of the design (the architectural layout, the load carrying structure etc., compare [1]). This can be accomplished by introducing a meta-grammar and developing hierarchical structure of the CP-graphs.

REFERENCES

1. BAILEY S. F, SMITH I., Structural and Architectural Integration in Building Design. Proceedings of the EG-SEA-AI Workshop on Artificial Intelligence in Civil Engineering, Lausanne, pp.327-341, 1994.
2. BOULANGER S., SMITH I., Models of design processes. Proceedings of the EGSEA Workshop, Lausanne, 1994, pp. 132-145.
3. BORKOWSKI A., JÓŹWIAK S., FLEISCHMANN N., On procedural and declarative aspects of optimum structural design. Progress in Structural Engineering, eds. D. Grierson, A. Franchi and P. Riva, Kluwer Academic Publishers, Dortrecht, 1991, 279-292.
4. BUNKE H., Programmed Graph Grammars. Lecture Notes in Computer Science, 1977.
5. COYNE, R. D., ROSENMAN, M. A., RADFORD, A. D., BALACHANDRAN, M. and GERO, J. S., Knowledge-Based Design Systems. Addison-Wesley, New York, 1990.
6. FENVES S. J., BAKER N. C., Spatial and functional representation language for structural design. Expert System in Computer-Aided Design, J.S. Gero (ed.), North-Holland, Amsterdam, 1987, pp. 511-530.
7. GEORGEFF M. P., Procedural Control in Production Systems. Artificial Intelligence 18, 1982.
8. GRABSKA E., Graphs and designing. Lecture Notes in Computer Science: Graph Transformations in Computer Science, 776, Springer-Verlag, Berlin, 1994, pp. 188-203.
9. GRABSKA E., Theoretical Concepts of Graphical Modeling, Part two: CP-graph Grammars and Languages. Machine GRAPHICS and VISION, 2(2), 1993, pp. 149-178.
10. MARCH L., STINY G., Spatial systems in architecture and design: some history and logic. Environment and Planning B, 12, 1985, pp. 31-53.
11. SAUSE R., POWELL, G. H., A design process model for computer integrated structural engineering. Engineering and Computers, 7, 1991, pp. 145-160.
12. SCHERER R. J., Integrated product and process model for the design of reinforced concrete structures. Proceedings of the EGSEA Workshop, Lausanne, 1994, pp. 132-145.
13. SCHNEIDER H. J., EHRIG H., Graph Transformations in Computer Science. Lecture Notes in Computer Science, 776, Springer-Verlag, Berlin, 1994.
14. SHIH S. G., The use of string grammars in architectural design. PhD Dissertation, Dept. of Architecture, ETH Zurich, 1994.
15. STINY G., Introduction to shape and shape grammars, Environment and Planning B, 7, 1980, pp. 343-351.

Representation of Concurrency in Object-Oriented Design Models

Convergences dans les modèles de projet orientés-objets

Zur Darstellung von Nebenläufigkeiten in objektorientierten Entwurfsmodellen

Dirk BRETSCHNEIDER
Civil Eng.
Ruhr University
Bochum, Germany

D. Bretschneider graduated in 1992, has worked on information technologies for tunnel rehabilitation. Since 1994, he is involved in research on computer-aided structural steel design. His research concerns realisation of concurrency aspects in object-oriented engineering design systems.

Dietrich HARTMANN
Prof.
Ruhr University
Bochum, Germany

D. Hartmann was Prof. for Struct. Mechanics & Optimisation at the Univ. of Dortmund, 1982-1987. Since 1987, he is Prof. for Computational Eng. at the Univ. of Bochum. His research fields are intelligent computation in CAE, in particular, object-oriented techniques, structural optimisation and parallel computing.

SUMMARY

The complexity of engineering design and construction processes is appropriately managed by applying object-oriented modelling techniques. The consideration of concurrency aspects, inherent to design processes, significantly improves the temporal behaviour and, thus, the acceptance of computer systems based on such techniques. A notation for the explicit representation of concurrency in object-oriented engineering processes is introduced. Subsequently, the power and expressivity of the notation is demonstrated by the description of concurrency phenomena in the process "design and code verification of steel structures according to the German Standard DIN 18800".

RÉSUMÉ

L'étude du projet et de l'exécution en génie civil sont des processus complexes, qui peuvent être maîtrisés à l'aide de techniques basées sur les modèles orientés-objets. La considération systématique des convergences permet d'améliorer considérablement la rapidité et l'acceptation, par l'usager, d'un modèle informatique orienté-objet. La puissance et la facilité d'emploi est illustrée dans le cas de la norme 'projet de calcul et contrôle pour les constructions métalliques selon la norme allemande DIN 18800'.

ZUSAMMENFASSUNG

Die Komplexität ingenieurmässiger Entwurfs- und Konstruktionsprozesse wird sehr gut mit Methoden der objektorientierten Modellierung bewältigt. Durch eine systematische Berücksichtigung inhärenter Nebenläufigkeiten in den modellierten Ingenieurprozessen kann sowohl das Laufzeitverhalten als auch die Akzeptanz eines objektorientierten Computersystems entscheidend verbessert werden. Es wird eine Notation zur expliziten Darstellung von Nebenläufigkeiten in objektorientierten Ingenieuranwendungen vorgestellt. Die Mächtigkeit und Ausdrucksstärke dieser Notation wird anhand der Beschreibung von Nebenläufigkeiten in einer typischen Ingenieuranwendung ("Nachweis und Bemessung von Stahlhochbauten nach DIN 18800") demonstriert.

1. INTRODUCTION

The process from planning and design to construction of structures implies a plentitude of cognitive and procedural activities each associated with a high degree of complexity. For this reason, a consistent integration of these acitivities into an appropriate computer system is very much desirable and benificial. At present, there are various CIM–research projects in civil engineering, funded by the European Union, dealing with integration in computational engineering. By way of example, in the EUREKA project CIMsteel [1] a computer integrated system applicable to the European constructional steelwork industry is developed. In particular, the entire constructional lifecycle from design to manufacturing is taken into account. In the ESPRIT project COMBI [2] emphasis is on the application of advanced information technology methods in order to establish a computer integrated environment for "cooperative" design work. In a further ESPRIT project entitled ATLAS [3] a generic platform for the incorporation of existing "large–scale engineering" application tools is created, also knowledge base representations for a computer integrated design system are examined.

From ongoing research at the ICE[1], it turns out that object–orientation is **the** key technique for future solutions in the CIM–area [4]. The object–oriented paradigm offers an exceptionally suitable platform for the representation of cognitive as well as procedural phenomena within engineering disciplines. Furthermore, it enables programmers to overcome severe difficulties occuring in conventional (procedural) computer science paradigms:

- The object–oriented model of a real–world engineering problem is more natural than the procedural one. This facilitates communication between programmers and application domain experts and provides "better" computer systems.
- The object–oriented approach leads to a much more "reliable" solution than procedural methods because the real–world problem is modeled using visisble and persistent entities of the natural problem domain (the objects with certain characteristics and specific behavior) instead of thinking in abstract and ephemeral procedures.
- Due to concepts like "information hiding" and "encapsulation" the application of object–oriented techniques ensures excellent software reusability compared to conventional solutions and, thus, improves the possibility of future adaptions to state of the art in engineering.

2. BACKGROUND AND PROBLEM DEFINITION

In a present research project, carried out at the ICE and funded by the DFG[2], the individual tasks within structural steel design and construction, such as preliminary design, structural analysis, design and code verification as well as structural detailing, are modeled by applying the object–oriented paradigm. Subsequently, the above four tasks are integrated into a holistic computer system based upon a central object–oriented database management system (OODBMS) in order to support engineers during the entire design as well as construction process. Consequently, the computer system consists of four distinct modules, each one representing one of the four tasks mentioned. All modules are implemented in C++ under the operating system UNIX. The graphical user interface of the computer sytem is based on OSF/Motif, while ONTOS [5] is applied as OODBMS.

In order to manage the complexity of the individual stages in the planning, design and construction process , and to ensure a reasonable response time of the computer system during execution it is obvious that a **concurrent** (parallel or distributed)[3] model is a must. This is a direct consequence of the requirements for integration and holism. Concurrency considerations are a key issue in the development of modern and innovative computer aided engineering systems if the natural way, in which an organized, systematic team work of engineering design experts takes place, is to be modeled: Similar to the human team work a team of computers concurrently performs distributed tasks of the engineering design process. The description and subsequent implementation of concurrency, therefore, is essential for modern and innovative computer aided engineering systems.

In the following an abstract notation for the representation of concurrency phenomena in engineering design processes is introduced. The notation serves (1) to illustrate basic process sequences as well

1. *ICE = Institute for Computational Engineering, Ruhr-University Bochum*
2. *DFG = Deutsche Forschungsgemeinschaft (German Research Foundation)*
3. *The terms "parallelism" and "concurrency" will be used as synonyms throughout the paper.*

as dependencies between single design activities and (2) to detect inherent concurrency aspects – and through this – establishing a framework for a subsequent implementation.

3. CONCURRENCY

Concurrency, as a generic solution concept, is a multi–layered phenomenon. From a programming point of view, four distinct levels of concurrency may be distinguished. Commonly, the following levels of concurrency can be identified: the intra–instruction level (focusing on concurrent operations in individual statements), the inter–instruction level (considering concurrency between program statements), the program–level (concerned with concurrent partial processes within a program) and the job–level (determining concurrent tasks from an overall system view).

A further source of concurrency lies in the various fashions in which instruction streams and data streams consisting of single or multiple items may be processed. Following the classification by Flynn [6] SIMD[4]– and MIMD[5]–hardware architectures may be applied for a computer realization. Recent research in computer technology, however, makes evident that the MIMD–paradigm in association with a message–passing mechanism offers the most powerful and general parallelization platform in engineering disciplines. Therefore, the MIMD–concurrency concept is taken into consideration for concurrent object–oriented modeling within the ICE group, exclusively, and with specific regard to the four parallelization levels in programs mentioned above.

4. OBJECT–ORIENTED MODELING (OOM)

To capture the details of concurrency in engineering applications an abstract notation for its object–oriented representation is an absolute must. An adequate notation has already been introduced by Rumbaugh et al [7]. This notation is based upon an object–oriented analysis (OOA) resulting in a total of three distinct submodels that are interrelated with each other (see Fig.1). Concurrency aspects are particularily represented within the dynamic model that usually follows the establishment of an object model but precedes the definition of the functional model.

Object model :	**Static structure of a real–world system** • Objects • Associations
Dynamic model :	**Control aspects and dynamic behavior** • States S • Transitions / Events E
Functional model :	**Data flow and data transformations** • Processes • Data

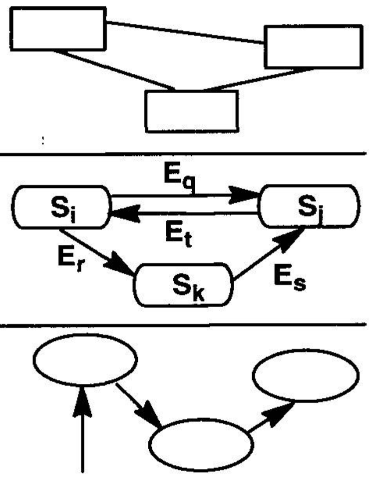


Fig. 1: Submodels of the object–oriented analysis

Since the interactions between the above three submodels are significant for concurrency they are briefly elucidated in the following chapter.

4.1 OOA–submodels

The representation of a real–world problem in terms of objects makes necessary a thorough analysis of the problem domain. To manage the inherent complexity of the real–world problem it is common

4. *SIMD = Single Instruction / Multiple Data*

5. *MIMD = Multiple Instruction / Multiple Data*

practise to decompose the original problem into surveyable portions and to consider only specified aspects of the entire problem in an orthogonal manner.

The object–oriented modeling follows this approach and primarily starts with the objects of the problem domain. This leads to the object model that represents the static structure of the problem by means of its characteristic entities and quantities (objects) as well as the relationships between them (associations). Each object created and defined has specific properties (attributes) and contains specific information on its behavior in terms of operational statements (methods). The object model, mathematically expressed, forms an indirected graph where object classes are represented by means of nodes and relations by edges or lines.

Secondly, the dynamic model is established with a direct reference to the object model. The dynamic model describes the temporal and, therefore, dynamic behavior as well as the control aspects of the problem solution. It is graphically defined through object state diagrams which are directed graphs. The nodes of the graphs are equivalent to states of a specified object while transitions between states are depicted in terms of arrows. Transitions may either be caused by events or happen automatically.

Third, a functional model is created. This model describes the data flow and transformations from object sources to object targets during the execution of the object–oriented computer system. In general, the functional model consists of various data flow diagrams each being a directed graph where nodes correspond with processes and lines (arrows) with data flows.

4.2 Representation of concurrency

As pointed out in the preceding chapter, two fundamental representation concepts are applied within the dynamic model: (object) states and transitions (between states). A state can be understood as an abstract description of a single object with regard to its attribute values and relationships to other objects. A state is valid over a period of time. Commonly, an object, therefore, has multiple states during its lifetime. The states are linked by transitions that have no duration over time. Both, states and transitions of an object are incorporated in the object's state diagram. The entire state diagrams of the different objects form the dynamic model of the system.

For the design of steel structures two characteristic features can be determined:

- most states represent sequences of specified computations , called **operations** in the following,
- transitions are primarily prescribed by the design process, and, therefore, happen **automatically**; only a minority of transitions is caused by explicit events created from direct user interactions.

Consequently, a key point is the detection of concurrency within the above operations. This problem is generally dealt with by Bernstein [8]: Considering two operations each one associated with individual input data and output data, both can only be carried out in parallel when the three conditions shown in Fig. 2 are simultanously satisfied.

$$
\begin{aligned}
& OUT(O_1) \cap IN(O_2) = \{\} \\
\wedge\ & IN(O_1) \cap OUT(O_2) = \{\} \\
\wedge\ & OUT(O_1) \cap OUT(O_2) = \{\}
\end{aligned}
$$

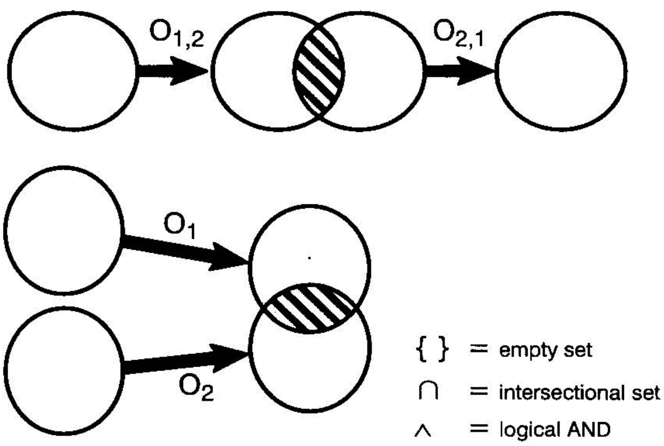


Fig. 2: Bernstein rules

Inherent to the object–oriented modeling are concurrencies due to aggregation. Aggregation is an abstraction principle most frequently used in object–oriented models of engineering disciplines. By way of example, the decomposition of a structure into single structural components along with a part–whole relationship represents an aggregation (see Fig. 4 in the following chapter). With respect to operations, also aggregate operations can be identified comprising several operations of **one** single object (Fig. 3).

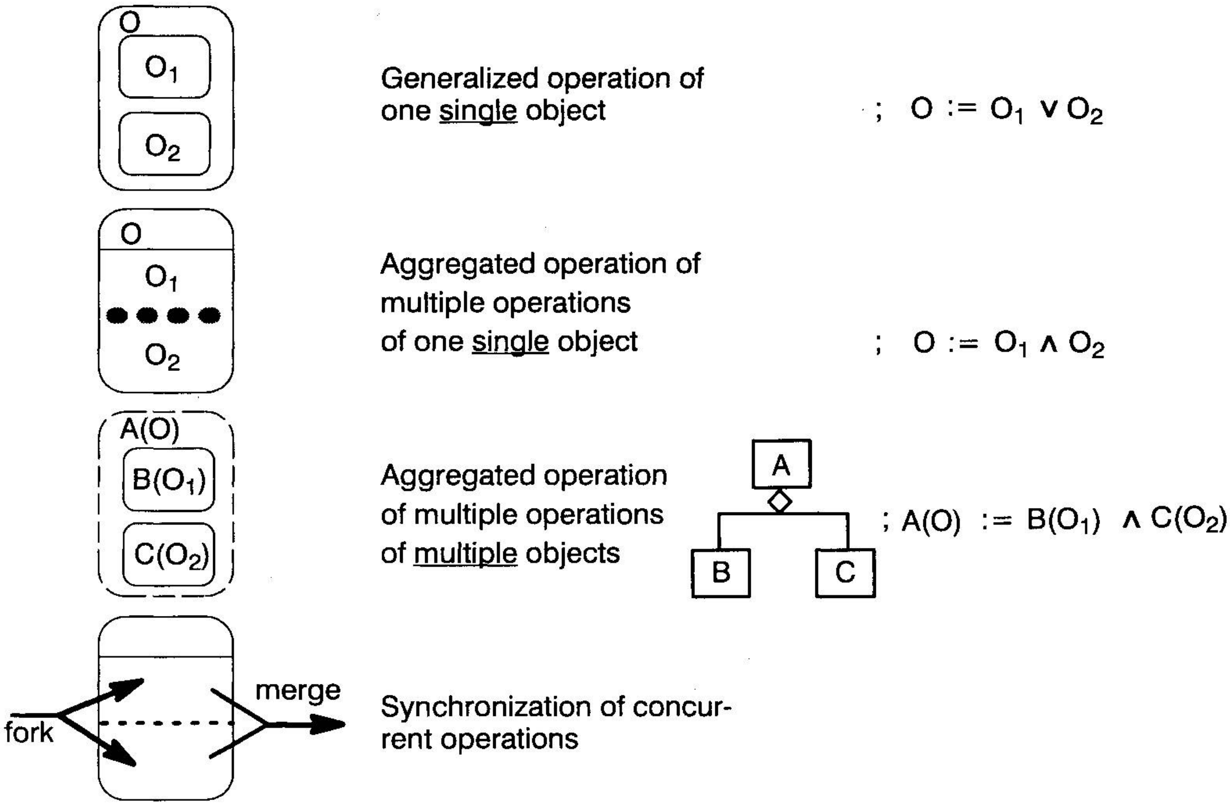


Fig. 3: Advanced concepts of the notation

Furthermore, an aggregate operation may be defined for not one but **many** objects. To prevent confusion in this case, the states have to be precisely identified by a state–name in harmony with the name of the involved object. In addition, the concept of a "generalized state" as well as two synchronization mechanisms for concurrent operations have been defined to provide additional elements for the notation (Fig. 3).

5. DESIGN OF STEEL STRUCTURES ACCORDING TO DIN 18800[6]

5.1 Object model

As mentioned above, the dynamic model encompasses multiple state diagrams, each of which is describing a network of states and transitions between these states for individual object classes. Within the dynamic model, solely classes with significant dynamic behavior patterns are considered. The corresponding classes are grey marked in the object class model of the application considered (see Fig. 4).

In the scope of this paper, exclusively, the dynamic model for "design and code verification of steel structures according to DIN 18800" [9] is examined. Design and code verification according to a given standard is a characteristic activity in the engineering design process and can be a time–consuming process depending on the size of the structure and the number of loading cases. As a consequence, the detection and realization of concurrency significantly improves the temporal behavior and effectiveness of a computer aided "design and code verification assistant" required to cope with future needs.

6. *DIN 18800 = German design code for steel structures [9]*

The dynamic model for "design and code verification" will be clarified through a specific example, where a three dimensional structure is considered that consists of two plane frames connected by two

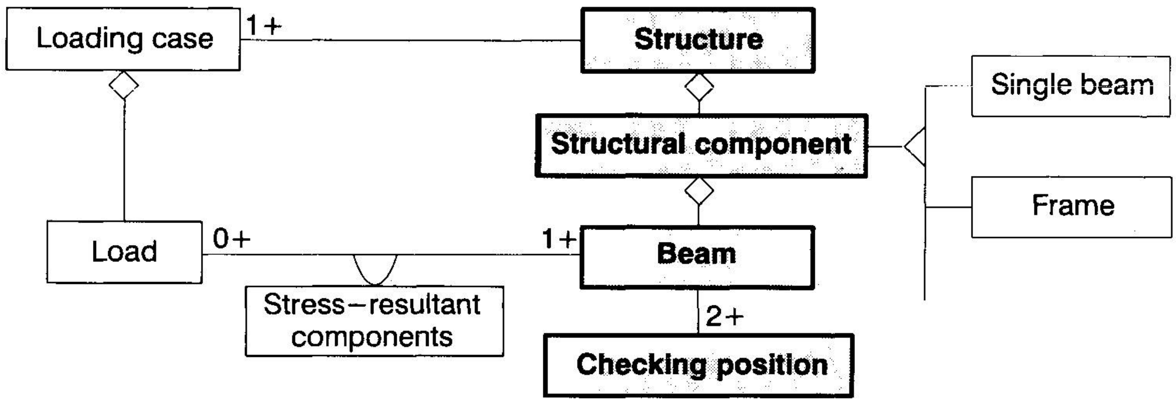


Fig. 4: Partial object model of the engineering application "Design and code verification"

bars perpendicular to the frame planes (Fig 5). The overall structure is represented through an aggregation of four structural parts called "structural components". Each of the structural components contains one or more beams. Usually, a beam has three checking positions where the local feasibility of stresses due to given loads is checked according to DIN 18800[7].

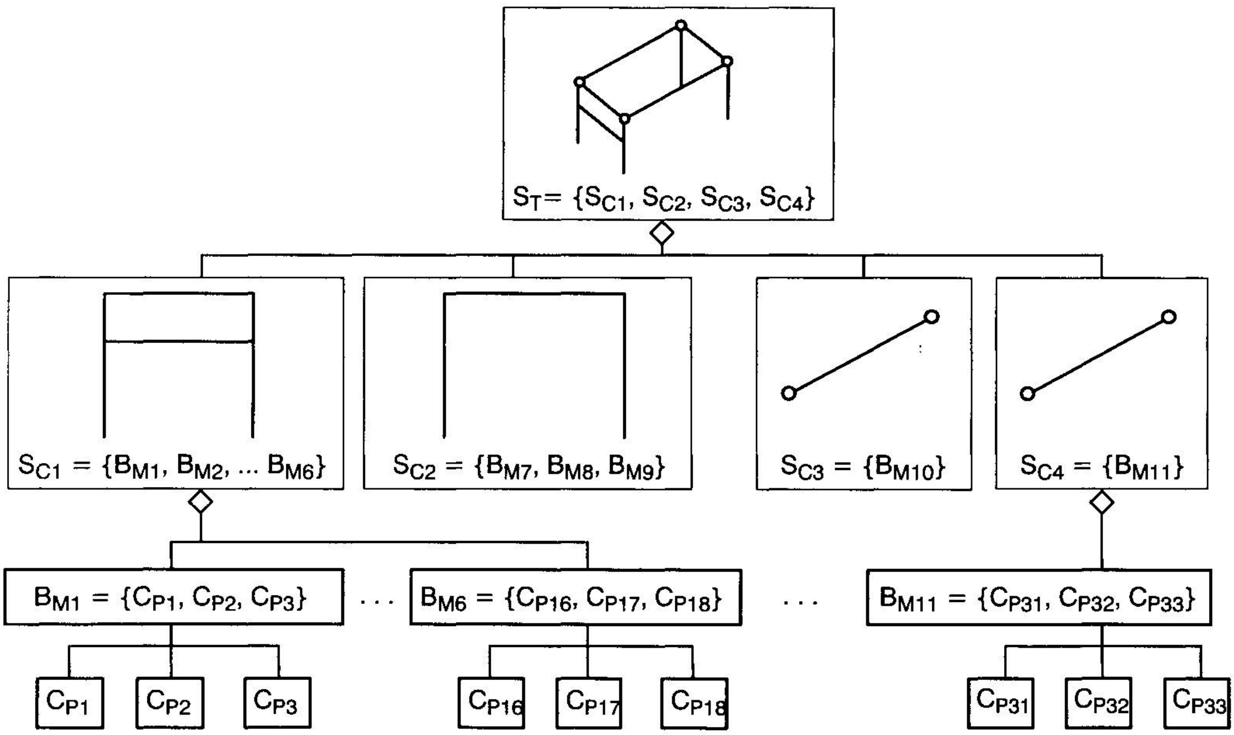


Fig. 5: Partial object instance model for the example considered

The object instance diagram in Fig. 5 used to represent concurrency follows the object class diagram in Fig. 4 when interrelating objects "Structure", "Structural component", "Beam" and "Checking position". In addition to the symbolic notation applied in Fig. 5, relationships between objects are expressed in a mathematical way using sets (e.g. $S_T = \{S_{C1}, S_{C2}, S_{C3}, S_{C4}\}$, indicating that object "Struc-

7. *In general, the number of checking positions in a beam is variable and depends on the number of loading cases as well as the shape and magnitude of different stress–resultant components associated with the loading cases. At present, the design and code verification only encompassses stability phenomena of single structural components but not of total structures.*

ture" (S_T) contains four objects "Structural component" (S_C), called S_{C1}, S_{C2}, S_{C3} and S_{C4}. The relations of the objects "Beam" (B_M) and "Checking position" (C_P) to other objects are represented in a similar way.). The sets serve as a reference to the object model when concurrency phenomena will be discussed in distinct state diagrams of the dynamic model, subsequently.

The object instance model in Fig. 5 has a tree-like structure. Following the tree from top to bottom three levels of concurrency can be identified: (1) concurrency in the structure layer (object S_T), (2) concurrency in the structural components layer (object S_C) and finally (3) concurrency in the checking positions layer (object C_P). All three levels will be examined separately with respect to potential sources of concurrency.

5.2 Representation of dynamic behavior and concurrency

In the overall structure layer the sequence of the three processes "structural analysis", "design and code verification" and "structural detailing" is obvious and demonstrated in Fig. 6. These processes are represented by states "S_T (Structural Analysis)", "S_T(Code verification)" and "S_T(Structural detailing)" of object "Structure", respectivly. All three processes can be integrated in the abstract state "S_T (Structural design)" of the object "Structure" (see Fig. 6).

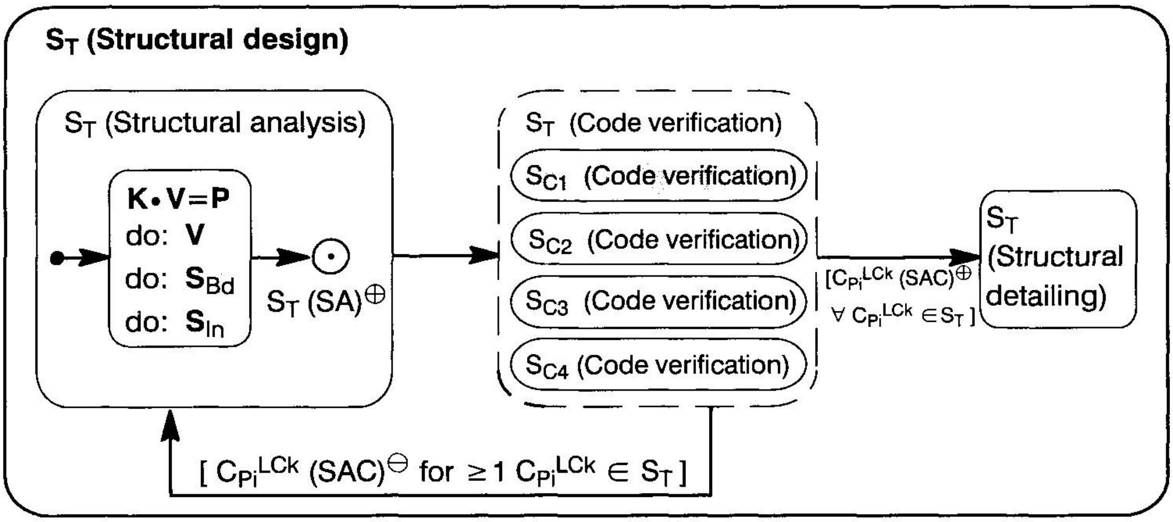


Fig. 6: Concurrency in the structure layer (object S_T)

The state "S_T(Structural analysis)" within state "S_T (Structural design)" is also an abstract state consisting of three substates[8]:

- an initial state that "Structure" automatically represents when entering the structural analysis operation (solid circle),
- a computation state where several activities (indicated by the keyword "do:") are performed, particularily the solution of the stiffness equations K•V=P is carried out and leads to the displacements V and the stress-resultant components S_{Bd} and S_{In} of the structure,
- subsequent to the computation state "Structure" automatically enters its final state (bull's eye), called "S_T(SA)$^\oplus$"; the plus sign indicates that the structural analysis (SA) of the entire structure S_T has been performed successfully.

Then the state "S_T(Code verification)" is entered. This state is an aggregate operation of several suboperations, each performed by a distinct object: The notation in Fig. 6 specifies that "design and code verification" of the whole structure S_T can be performed in parallel for every agggregated structural component $S_{Ci} \in S_T$. Hence, the state "S_T(Code verification)" is a "nested state" where substates may be expanded in separate state diagrams.

8. *Only those aspects within state "S_T(Structural analysis)" are pointed out that express dependencies to the state of main interest, "S_T(Code verification)". A more detailed consideration of "S_T(Structural analysis)" and "S_T(Structural detailing)" is omitted, because emphasis is laid on "S_T(Code verification)" within this paper.*

The state "S_T(Code verification)" is finished when the stress analysis has been performed at every checking position C_{Pi} for every load case LC_k. According to the verification results state "S_T(Structural analysis)" or state "S_T(Structural detailing)" is entered. This is graphically shown through arrows in Fig. 6 representing "guarded transitions". By that object "Structure" is transfered from state "S_T(Code verification)" to another state if the individual guard condition shown as a Boolean expression in brakkets is true. Thus, if the stress analysis is verified at all checking positions of the structure (indicated by condition "[$C_{Pi}{}^{LCk}$ (SAC) $^{\oplus}$ $\forall$ $C_{Pi}{}^{LCk} \in S_T$]" in Fig. 6) the structure can be designed in detail. Consequently, the state "S_T(Structural detailing)" is entered. By contrast, in the case that the stress analysis is not verified at one or more checking positions (indicated by condition "[$C_{Pi}{}^{LCk}$ (SAC)$^{\ominus}$ for ≥ 1 $C_{Pi}{}^{LCk} \in S_T$]" in Fig. 6), the corresponding parts of the structure have to be changed. This, of course, makes a re-analyzation of some parts or the entire structure necessary. Thus, state "S_T(Structural analysis)" is re-entered again (see Fig. 6).

As mentioned previously besides concurrency in the structure layer there also exists concurrency in the structural components layer. This is exemplified by expanding the state "S_{C1}(Code verification)" that is grey-shaded in Fig. 6. Examining a single structural component, basically three groups of operations can be parallelized when performing the stress analysis:

- computation of the effective length of all associated beams by determining the buckling factors β (state "Buckling factor (β)"),
- computation of the allowable stress-resultant components representing yielding in the individual cross section S_{pl} (state "Allowable plastic stress-resultant components (S_{pl})"),[9]
- execution of the stress analysis to verify that the collapse load is not exceeded at every checking position C_{Pi} and for every load case LC_k given (state "Stress Analysis against Collapse (SAC)").

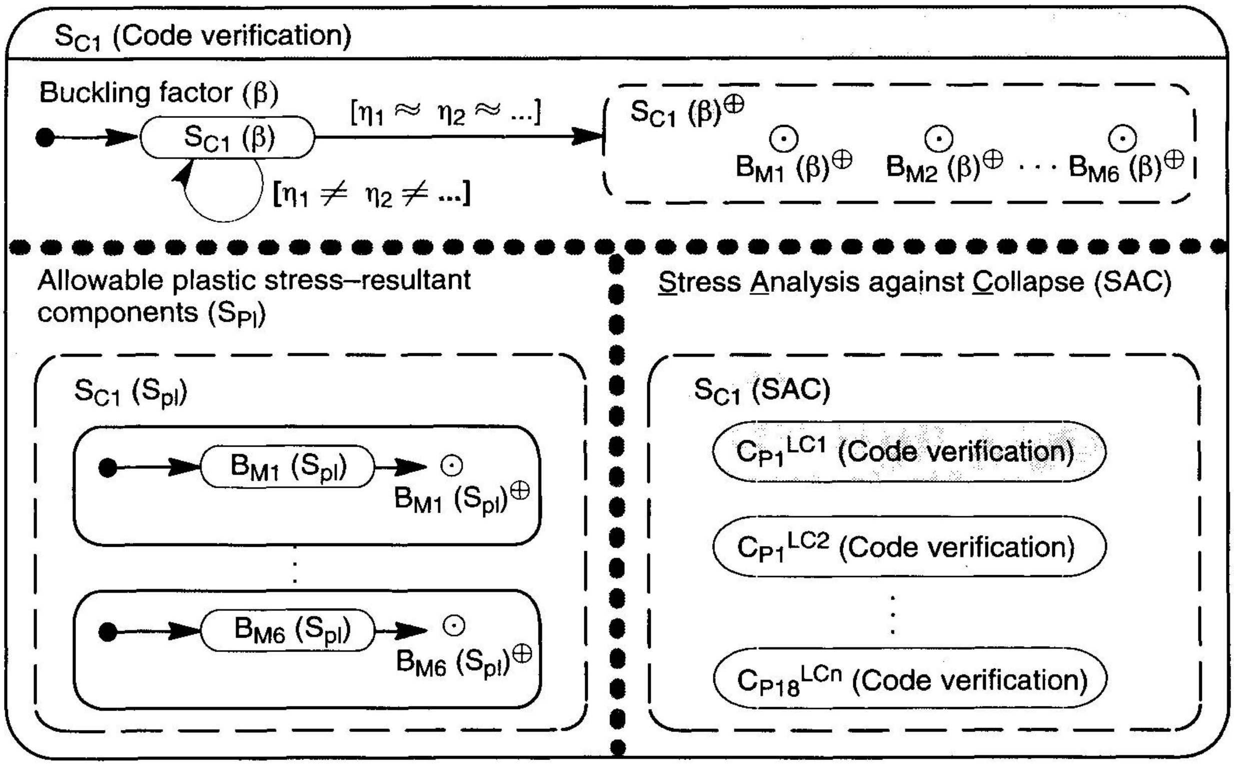


Fig. 7: Concurrency in the structural component layer (e.g. object S_{C1})

The concurrency incorporated between these three states is clarified by dotted lines inside state "S_{C1} (Code verification)" (see Fig. 7).

9. *The german code DIN18800 offers three different possibilities to compare loading and loading capacity, thus, to verify a structure: elastic-elastic, elastic-plastic, plastic-plastic. The considerations in this contribution are based on the method elastic-plastic: Loading, expressed in stress-resultant components, is based on elastic computations. Loading capacity, expressed in stress-resultant components for plastification in the cross section, is based on plastic computations.*

In addition, the three individual states themselves are again superstates entailing concurrent substates. These are either substates, describing operations of the **same** object (e.g. the computation of the buckling factor "$S_{C1}(\beta)$") or operations of **aggregated** objects expressed by rounded boxes with dashed lines (e.g. the computation of allowable yield stresses "$B_{Mi}(S_{pl})$" expressed in stress–resultant components for every beam associated, or the execution of the stress analysis for the specified checking position and load case "$C_{Pi}{}^{LCk}$ (Code verification)").

In some cases it is important to ensure that computations inside states are terminated, because the computation results are required in subsequent states. Thus, final states have to be defined, e.g. the state $B_{M1}(S_{pl})^{\oplus}$ indicating that all allowable yield stresses of beam 1 expressed in stress–resultant components have been computed (Fig. 7).

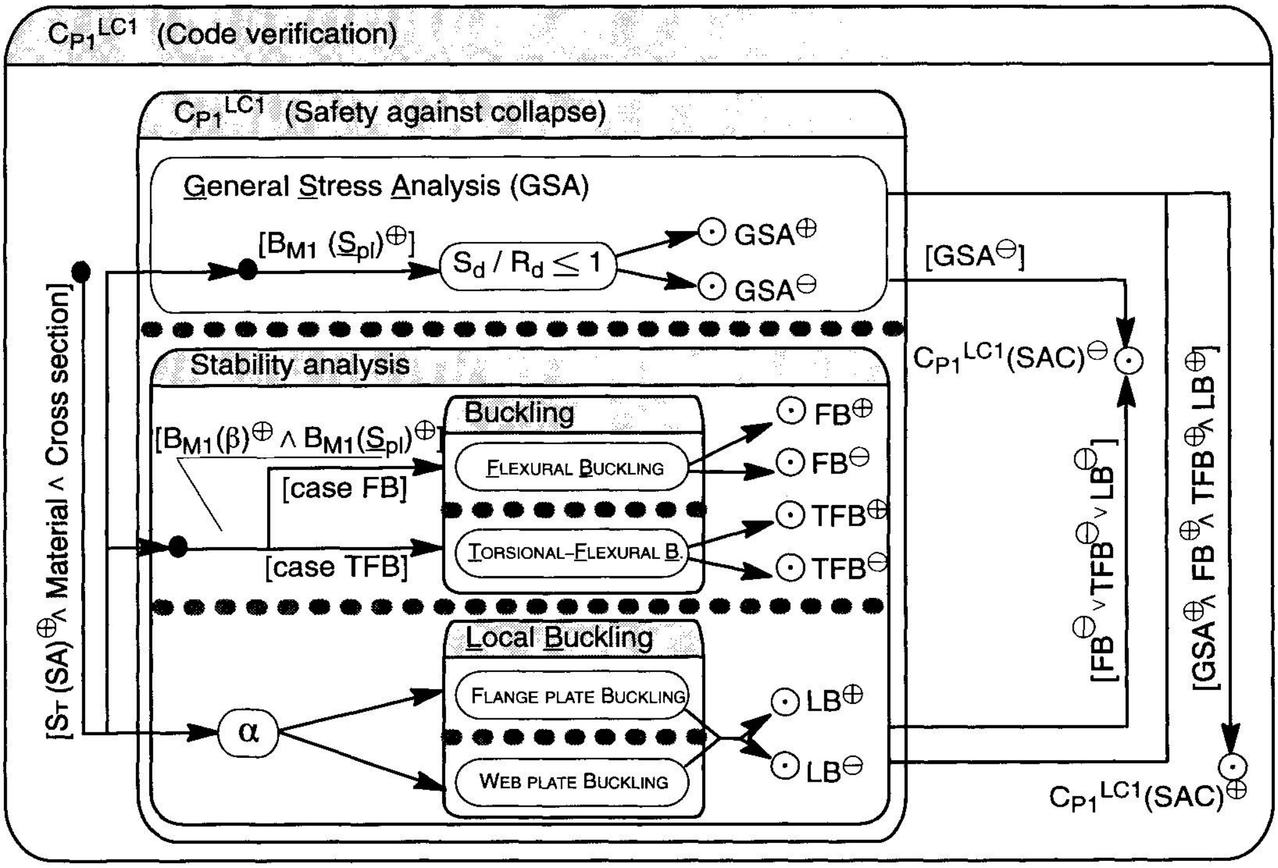


Fig. 8: Concurrency in the checking positions layer (e.g. object C_{P1})

Concurrency in the checking positions layers is examined by way of an example, too. Considering the state "$C_{P1}{}^{LC1}$ (Code verification)" this state is expanded as illustrated in Fig. 8. In this case, the stress analysis at specified checking positions can be described by means of three substates:

- an initial state that is finished automatically if structural analysis results (S_T $(SA)^{\oplus}$), material's and cross section's properties are known,
- a computation state through which safety against collapse with regard to different aspects (e.g. flexural buckling, torsional flexural buckling etc.) is determined (state "$C_{P1}{}^{LC1}$ (Safety against collapse)"),
- two alternative final states "$C_{P1}{}^{LC1}(SAC)^{\oplus}$" and "$C_{P1}{}^{LC1}(SAC)^{\ominus}$" entered automatically depending on the results of the code verification; the final state affects the process sequence within state "S_T(Structural design)" as shown in Fig. 6.

Inside state "$C_{P1}{}^{LC1}$ (Safety against collapse)" several code verification procedures for various categories of collapse are carried out. In detail these are a general stress analysis (GSA), verification of safety against flexural buckling (FB) and torsional–flexural buckling (TFB) as well as verification of stability against local buckling, separately for beam's web and flange plate (LB). All five code verifications can be examined in parallel. The concurrent portions are elucidated by dotted lines (see Fig. 8).

The transitions between the initial state and the stress analysis as well as the final states require measures for synchronization as introduced in chapter 4.2. State "C_{P1} ^{LC1} (Safety against collapse)" basically has three independent substates. If the guard condition at the guarded transition from the initial state to the stress analysis state "C_{P1} ^{LC1} (Safety against collapse)" is satisfied all three substates are active at the same time. This requires a fork of the control flow. Contrary to this, the transition from the stress analysis state to the final state "C_{P1} ^{LC1} (SAC)$^{\oplus}$" needs a merge of the control flow because the code verification at checking position C_{P1} for loading case LC1 can only be carried out if the constraints of **all** distinct code verification procedures are fulfilled. On the other hand, the transition from the stress analysis state to the final state "C_{P1} ^{LC1} (SAC)$^{\ominus}$" needs no synchronization because the violation of one single constraint leads to state "C_{P1} ^{LC1} (SAC)$^{\ominus}$".

6. CONCLUSIONS

The abstract notation introduced in this paper provides adequate expressive power along with a high representation density in describing concurrency of object–oriented models in engineering disciplines. Conceptionally, also parallelisms in the inter–instruction level between single statements can be captured using the concept of nested states.

In this contribution, particularily, all candidate categories of concurrency in the specific domain of "design and code verification of steel structures according to DIN 18800" are demonstrated. In a prototype computer system created at present, however, only portions of the overall concurrency concept are materialized in a coarse–grain solution. The prototype primarily models the natural concurrency of the team work of engineering design experts with sufficient success (see chapter 2). With respect to the concurrent implementation basically two categories of hardware have been applied at the ICE: (1) a highly parallel machine (transputer system) and (2) a heterogenous network of UNIX–workstations connected by a standard Ethernet. Subsequent to an evaluation of both hardware platforms the coarse–grain implementation model is realized on the workstation cluster applying PVM [10] as a message passing interface and using C++ as implementation language.

ACKNOWLEDGEMENTS

The authors would like to thank the DFG, Bonn, for funding this research work since 1992. The support by the DFG enabled the ICE research group to scrutinize a new and sophisticated field of research within object–oriented modeling.

REFERENCES

[1] CIMSTEEL, Computer Integrated Manufacturing for Constructional Steelwork, Information Brochure on the EUREKA 130 research project, Taylor Woodrow Construction Holding Ltd., 1994

[2] SCHERER et al., Architecture of an Object–Oriented Product Model Prototype for Integrated Building Design, Contribution to the 5th Int. Conference on Computing in Civil and Building Engineering, Anaheim, CA, USA, June 1993

[3] GARAS et al., Presentation of the EU ESPRIT research project "ATLAS", Session EP2 of the 1st European Conference on Product and Process Modeling in the Building Industry, Dresden, DE, Oct. 1994

[4] HARTMANN et al., Object–Oriented Modeling of Structural Systems, Contribution to the 5th Int. Conference on Computing in Civil and Building Engineering, Anaheim, CA, USA, June 1993

[5] ONTOS DB 2.2 Reference Manual, Vol. 1 "Class and Function Libraries", ONTOS Inc., 1992

[6] FLYNN, Some Computer Organizations and their Effectiveness, IEEE Transactions, Vol. C–22, 1972, pp. 948–960

[7] RUMBAUGH et al., Object–Oriented Modeling and Design, Prentice–Hall Int. Inc., 1991

[8] BERNSTEIN, Program Analysis for Parallel Processing, IEEE Transactions, Vol. EC–15, Oct. 1966

[9] DIN 18800 "Stahlbauten", Teile 1 und 2, Ausgabe November 1990

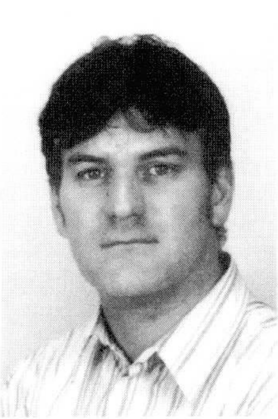
[10] GEIST et al., PVM 3 – Parallel Virtual Machine User's Guide and Reference Manual, Technical Report ORNL/TM–12187, Oak Ridge National Laboratory, USA, 1993

A Knowledge Support Component for a Site Optimisation Algorithm

Système à base de connaissances pour l'organisation optimale de chantiers

Eine wissensbasierte Komponente für einen Baustelleneinrichtungs-Optimierungs-Algorithmus

Kunibert LENNERTS
Dipl.-Wirtsch.-Ing.
University of Karlsruhe
Karlsruhe, Germany



Kunibert Lennerts, born 1963, received his diploma in computer science and business from Karlsruhe University, Germany. Since 1991, he is the project leader of the research project ESBE sponsored by the German research foundation.

Fritz GEHBAUER
Prof. Dr
University of Karlsruhe
Karlsruhe, Germany

Fritz Gehbauer, born 1945, received his M.S. degree in construction management from Stanford University and his Dr. of Eng. degree from Karlsruhe University. Since 1988 he is director of the Inst. of Construction Management and Machinery.

SUMMARY
ESBE is a hybrid object-oriented system to generate optimised construction site layouts. To keep the number of possible locations small for ESBE's mathematical optimisation algorithm, the expert system provides rectangular areas valued according to the given rules. This paper describes two methods to generate these suitable areas, namely the generating method and the grid method. Thorough evaluation proves a modified generating method to be best suited for ESBE.

RÉSUMÉ
ESBE est un système hybride orienté-objet destiné à optimiser l'organisation de chantiers. Un système expert crée des surfaces rectangulaires selon des règles données, de façon à limiter le nombre des solutions possible, à l'aide d'algorithmes d'optimisation. Cet article présente deux procédés générateurs de surfaces appropriées, à savoir la méthode de réseau et la méthode de génération. Après avoir comparé les deux, l'article conclut qu'une méthode de génération modifiée convient le mieux au système proposé.

ZUSAMMENFASSUNG
ESBE ist ein hybrides objektorientiertes System zur Generierung optimierter Baustellenlayouts. Um die Anzahl der möglichen Positionierungen beim mathematischen Optimierungsalgorithmus gering zu halten, liefert ein Expertensystem anhand der vorhandenen Regeln bewertete Flächen (Rechtecken). Dieser Artikel stellt zwei Verfahren zur Erstellung geeigneter Flächen vor, das Raster- und das Generierungsverfahren. Nach eingehender Bewertung stellt sich ein modifiziertes Generierungsverfahren als am besten geeignet heraus.

1. Introduction

The hybrid object-oriented system ESBE for the optimization of construction site layouts consists of three major parts: a construction site database which contains general and company specific data of the construction site facilities and the building data; a hybrid system with mathematical optimization algorithms and an expert system (knowledge based system) as well as a well-tailored user-interface with a 3D visualization component. Its task is to dimension and arrange the individual construction site facilities. The publications [1, 2] are giving an overview about the system ESBE and the related research.

This paper deals with a problem that occurred realizing the expert system component of ESBE: How to model suitable areas (rectangles) for the location of a specified facility. These areas serve as input to the mathematical algorithms, set up to optimize the location of the facility element inside the given suitable area considering the suitability value. The optimization algorithm with its objective function constraints and decision variables will be considered as a black box in this paper. The full description of the algorithm will be published soon.

The expert system component is based on a list of 161 original rules which have to be taken into consideration regarding the location of the site facilities. There are two different kinds of rules [3]:

- rules arising from regulations and laws with the relation ("must", "must not") (binding rules). In this case the suitability value for the areas is 1 or -1, and these areas must / must not be considered for the location of the facility element.
- rules derived from experience and expert knowledge with the relation ("shall", "can", "shall not", "shall if possible") (non binding rules). In this case a representation and aggregation of the knowledge is necessary to get one suitability value within (-1;+1) for the area and thus to rank the different areas.

Some of the original rules are not suitable for the rule processing system due to uncertain reference objects, missing weighting or redundancy. The residual rules (applicable rules) of which the system is constituted implement uncertain knowledge ("shall", "can", "shall not", "shall if possible") and vague knowledge ("at", "nearby", "far away" etc.). The vague and uncertain knowledge, that occurs in the given rules and the aggregation of both to get a suitability value within [-1, 1] for the areas have to be aggregated to get a ranking of these areas. This function in the paper called **agg** has been realized by a new-developed approach using a linear combination of fuzzy and probability functions and will be published soon [5].

2. HOW TO MODEL SUITABLE AREAS (RECTANGLES) - COMPARISON OF GRID- AND GENERATING METHOD

Independent of the choice of the knowledge representation manner, there are two different classes of methods for the expert system component to generate suitable rectangles serving as possible locations for the site facilities.

A) Grid Method

Rectangles are generated by the expert system component prior to the rule processing. This can be done by laying a grid over the construction site, so that every grid element is a rectangle. Subsequently the rule evaluation determines the suitability of these individual rectangles for the location of the site facility under consideration.

B) Generating Method

The rectangles are generated not before but by the rule processing. Thus the extension of the rectangles is not defined before the rule processing is finished.

An example for the generation of suitable areas to locate storage areas will illustrate the two methods. An evaluation of the following three rules for ascertaining of storage areas will be simulated with an actual construction site layout using both the grid and the generating method.

1. "storage area should be between site road and building" valuation: 0.5
2. "storage area should be within the crane's jib range" valuation: 0.7
3. "storage area should be located at the site road" valuation: 0.3

The following restrictions given by the mathematical optimization component must be taken into account:

- per call of the rule processing system suitable rectangles for only one site facility are ascertained.
- the shape of the construction site as well as that of every object located on the site is based on rectangles.
- the construction site is described by means of a coordinate system.
- all rectangular objects of the construction site are arranged right-angled in the coordinate system.

In the given actual layout, there are three objects located: a site road, a building and a crane (Fig. 1).

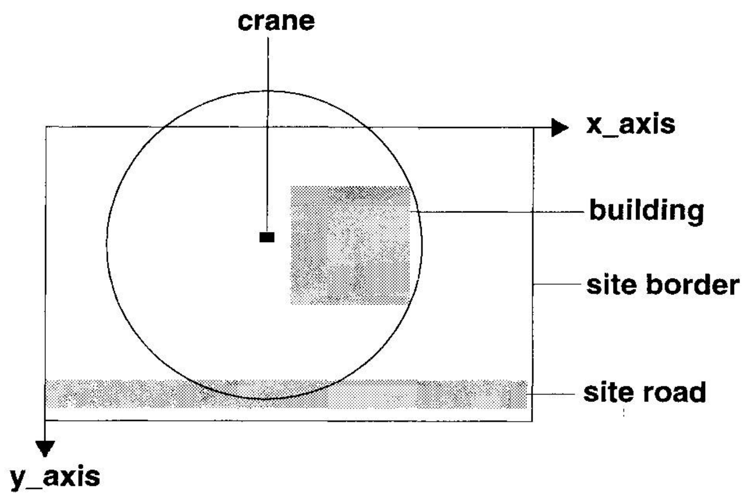


Fig. 1: Actual layout

2.1 GRID METHOD

A grid is laid over the construction site, so that every grid element is a rectangle. The size of the rectangles is arbitrary. A grid element shall be identified by its north-western corner. The size of the grid elements must be at least the extent of the respective site facility element. A coordinate system to describe any individual point of the construction site is given by the x-axis and the y-axis. In the following, the three rules mentioned above will be processed. Every rectangle ascertained a suitability value is assigned corresponding to the valuation of the rule's value. A rule is evaluated within the interval (-1,1). These values have no influence on the process itself. In every grid element fulfilling the rule the valuation of the processed rule will occur in the graphics.

A) 1. rule evaluation

"storage area should be between site road and building" (valuation: 0.5)

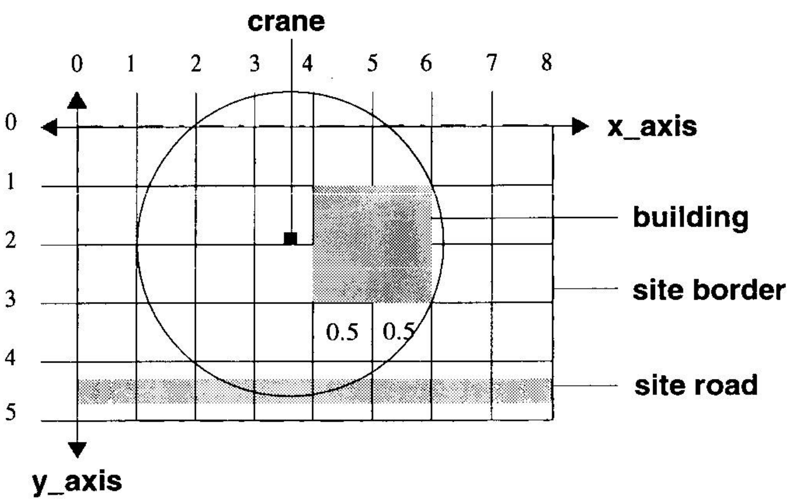


Fig. 2: Grid method 1. rule

According to this rule, the grid elements r(4,3) and r(5,3) are suitable rectangles with the value 0.5. The grid elements r(4,4) and r(5,4) are not appropriate because a part of the site road crosses them.

B) 2. rule evaluation

"storage area should be within the crane's jib range" (valuation: 0.7)

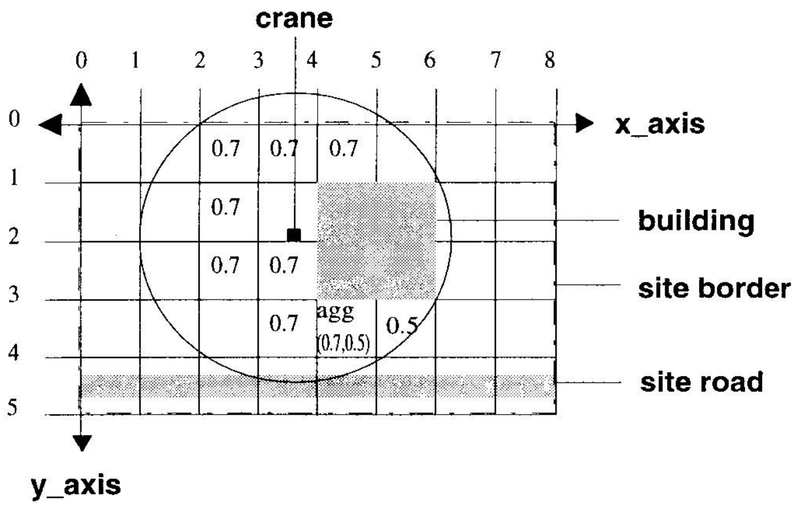


Fig. 3: Grid method 1. and 2. rule

All grid elements that are located completely inside the circle are assigned the rule valuation 0.7. Grid elements that cannot be covered completely by the crane's jib range do not obtain a suitability value. Further the abbreviation "agg" will be used for the aggregation of the suitable values (see r(4,3)). If a rectangle is ascertained by more than one rule and therefore more than one suitability value is assigned, the aggregation function derives one decisive suitability value out of the different ones. This is necessary, because the mathematical optimization algorithm can only take into account one suitability value for every grid element. Looking at Fig. 3, it can be observed that the grid element r(2,3) is not located completely within the crane's jib range and therefore it is not assigned the suitability value 0.7. On the other hand, the grid element r(4,3) which obtained the valuation 0.5 in the first step is assigned the value 0.7 by the second rule. Here the aggregation function has to derive a single suitability value from these two valuations.

C) 3. rule evaluation

"storage area should be located at the site road" (valuation: 0.3)

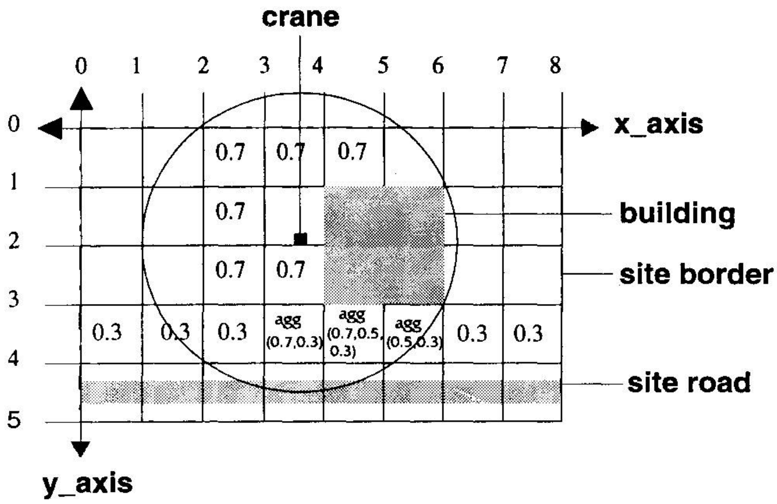


Fig. 4: Grid method 1., 2. and 3 rule

All rectangles located at the site road obtain the valuation 0.3, thus the rectangles r(3,3) and r(5,3) have two values, and r(4,3) even has three, which have to be aggregated to one suitability value. After processing all rules the positively valued grid elements are passed on to the mathematical optimization algorithm.

Concerning the exact location of a construction site facility within a grid element, the optimization algorithm examines all possible locations. The various possible locations are determined systematically by defined interval steps (Fig. 5). The site facility has to be located completely inside the grid element. In case of identical dimensions of the construction site facility and the grid element, the step above will be dropped due to the fact that the site facility fits exactly into one grid element.

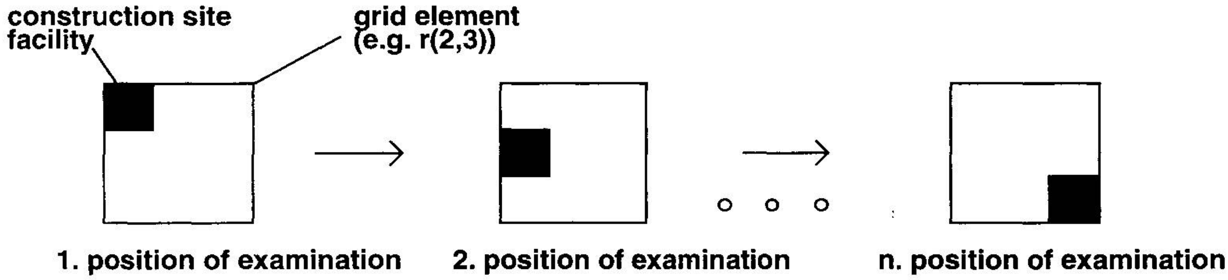
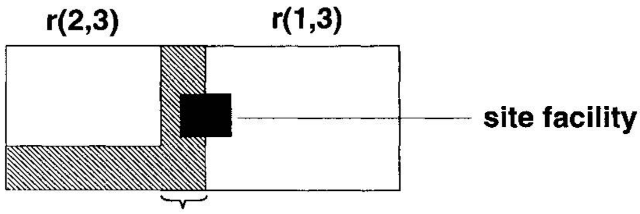


Fig. 5: Procedure of positioning a site facility inside a grid element

However, the mathematical optimization algorithm is not able to take into consideration locations for a site facility covering more than one grid element, even if the valuations of both grid elements were identical (e.g. r(1,3) and r(2,3): valuation 0.3). The layout algorithm does not examine the whole grid, but a given single grid element (rectangle). Therefore it is not able to split the base of the site facility and distribute it over different grid elements. Thus possibly appropriate areas for locating the facility element cannot be examined. For example the following Fig. 6 shows the hatched area of r(2,3) in which the site facility cannot be located due to the reasons described above. The size of the hatched area results from the length of the side of both the site facility and the grid element.



(length of a side of a site facility - ε, with $\varepsilon \rightarrow 0$)

Fig. 6: Problem of two overlapping grid elements

The sum of all hatched areas grows with the grid elements getting smaller as well as with the base of the site facility getting bigger. In case of identical sizes of the site facility and the grid element base, the hatched area amounts to the entire grid element minus its top left corner.

2.2. GENERATING METHOD

In this case every rectangle is generated by the rules. The size of the rectangle is determined by the minimum and maximum distance between site facility and its reference object given by the rule. The location of the rectangle results from the reference object's position and the rule's assignment instruction. Then a suitability value for the rectangle is deduced from the rule's valuation.

The generating method will be explained using the same rules as already done in the grid method. To compare both methods under equal conditions no overlapping of the rectangles generated by the same rule is allowed at this stage. A discussion of possible modifications concerning the overlapping of the generated rectangles will follow in the chapter dealing with the method's modification.

A) 1. rule evaluation

"storage area should be between site road and building" (valuation: 0.5)

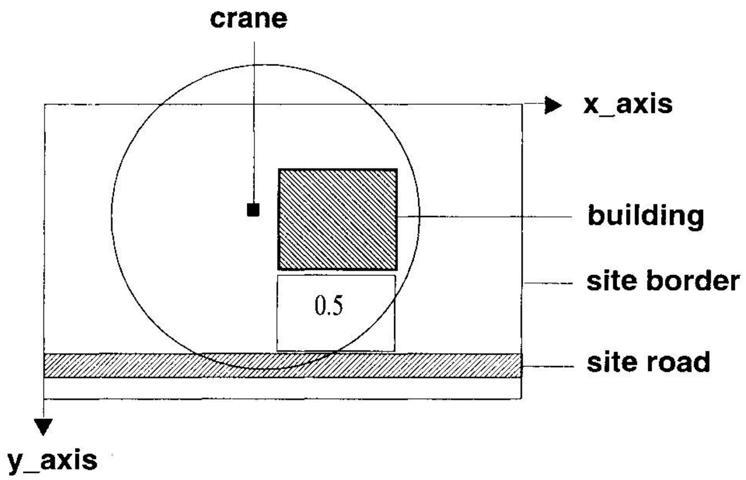


Fig. 7: Generating Method 1. rule

The generated rectangle is indicated by its suitability value 0.5. It is bounded by the site road and the building.

B) 2. rule evaluation

"storage area should be within the crane's jib range" (valuation: 0.7)

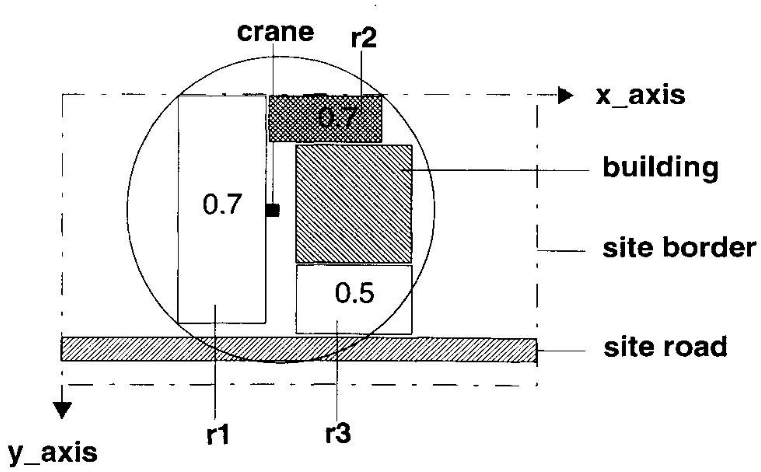


Fig. 8: Generating Method 1. and 2. rule

The two rectangles r1 and r2 are bounded by the building, the crane's jib range and the site border. Because of the equal conditions at this stage it is not allowed that the rectangles intersect with each other in order to prevent the ascertained rectangles from overlapping (like appearing in the grid method).

3. rule evaluation

"storage area should be located at the site road" (valuation: 0.3)

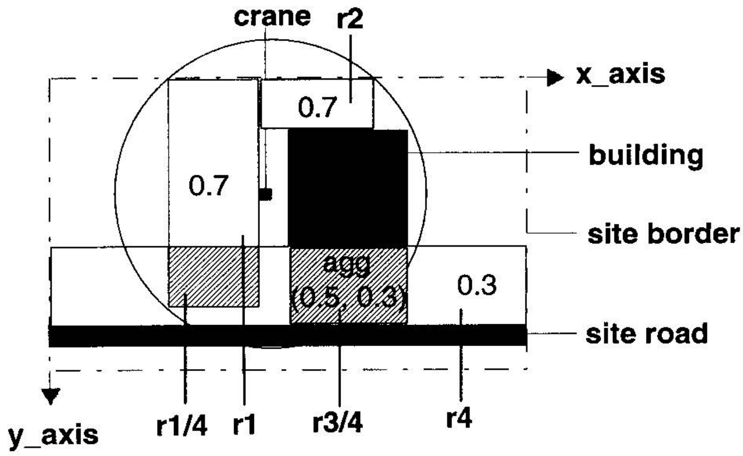


Fig. 9: Generating Method 1., 2. and 3. rule

The rectangle r4 with the valuation 0.3 is bounded by the building, the site road and the site borders. The hatched rectangles r1/4 and r3/4 represent the overlapping areas of the rectangles r1 and r4, and r3 and r4 respectively. They result from the combination of the 3rd rule with the 1st and 2nd one.

Now possible modifications of both methods are evaluated.

2.3 MODIFICATION OF THE GRID METHOD

The following faults occur when the grid method is realized as described in chapter 2.1.

A) 1. fault

If there are reference objects already located within a grid element, the remaining area cannot be considered as a suitable rectangle.

B) 2. fault

There are some grid elements partly covered resulting from the intersection of the crane's jib range with the grid elements, e.g. r(1,3) in Fig. 3. By not taking into consideration these grid elements the covered part of the rectangle gets lost as a possible storage area.

C) 3. fault

It is difficult to assign a valuation when only a part of a grid element obtains an additional valuation, as for example grid element r(5,3) in Fig. 3. For the complete grid element a valuation was already assigned and would have been passed on to the mathematical optimization algorithm, thus it is not clear if and how this additional value can be taken into account.

There are two kinds of modifications for the grid method: the first one retains the chosen grid and has no influence on the size of the rectangles whereas with the second the grid or single elements can be changed.

Modification concerning the first fault:

- with the size of the construction site facility in regard being chosen as the size of one grid element this fault cannot appear, but other disadvantages can.
- a rectangle is drawn into the remaining part of the grid element and is regarded as a suitable rectangle in case of sufficient size for locating the site facility inside.

Modification concerning the second fault:

- given a partly covered grid element, the whole grid element is regarded as a suitable area. Hereby it must be accepted that areas possibly not suitable will be considered as suitable.
- for the partly covered areas of a rectangle, that may not be rectangular any more, this suitable area will be approximated by one or several greatest possible rectangles.

Modification concerning the third fault:

- the valuation of the complete grid element will be changed in relation to the size of the affected area.
- the additional valuation will be aggregated with the old valuation and assigned to one or several rectangles approximating the suitable area.

2.4 MODIFICATION OF THE GENERATING METHOD

Apart from the first fault of the grid method, which can not occur using the generating method, the other faults of the generating method are identical to those of the grid method. This is why these faults are now called second and third fault, again.

B) 2. fault

Approximating the area covered by the crane's jib range with a rectangle means that possible suitable areas for locating the site facility get lost.

C) 3. fault

If only a part of a rectangle is assigned an additional valuation (see Fig. 9), it is uncertain if and how this valuation could be taken into consideration, because the complete rectangle has already got a valuation and would be passed on to the layout algorithm as a suitable area.

Modification concerning the second fault:

- the section resulting from the intersection of the crane's jib range with the rectangle will be approximated by a chosen number of rectangles.

Modification concerning the third fault:

- one possibility is to approximate the section by one or several rectangles. The new rectangle(s) is(are) assigned a new valuation resulting from the old one and the additional new one.
- another possibility is to assign the additional valuation to the complete rectangle in relation to the size of its covered area.

2.5 EVALUATION OF THE MODIFICATIONS OF THE GRID AND THE GENERATING METHODS

Both methods show some advantages compared to each other.

The advantages of the grid method are:

- neither the size nor the position of the rectangle has to be calculated.
- a calculation of the rectangle is not necessary, because there are no overlapping grid elements, given that the modification concerning the second fault has not been carried out.

- choosing the grid element very small, the resulting space is restricted in fact, but the mathematical optimization algorithm does not need to examine so many positions and, hence, a result can be achieved earlier.

The advantages of the generating method are:

- there is no first fault occurring because obstacles are considered, when the rectangles are generated.
- the generally bigger rectangles than in the grid method enable the mathematical optimization algorithm to examine more locations than in the grid method. Actually, the running time of the program can increase, but more locations can be examined, especially using the approximation of the crane's jib range with overlapping rectangles.

Fig. 10 shows the construction site after applying the three rules using the slightly modified generating method with overlapping rectangles of the same rule.

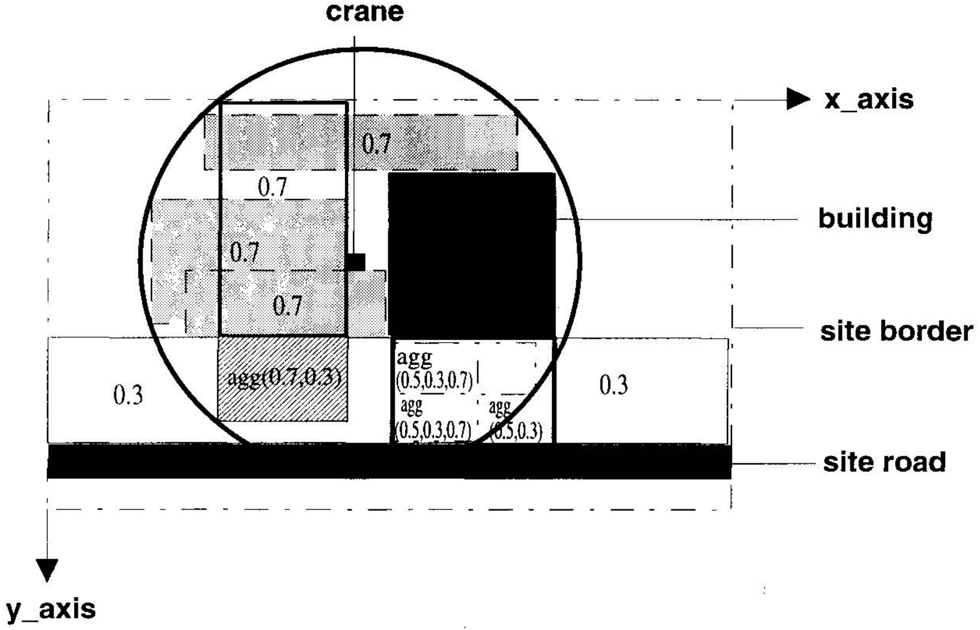


Fig. 10: Modified generating method rule 1.,2. and 3.

3. CONCLUSION

Even after the modifications concerning the faults in the grid method, the basic problem still exists, namely the disability to take areas at the borders of the grid elements into consideration even if the adjoining grid elements are identically valued. A further modification might be to unite identically valued grid elements to form the biggest possible rectangles. This results in rectangles similar to those obtained by the generating method. The definition of the grid and therefore the size of the single grid elements in the beginning of the procedure naturally still has influence on the size of the suitable rectangle. Here the grid method still differs from the generating method.

After evaluating the two methods a modified version of the generating method with the following modifications will be carried out by ESBE:

- several rectangles, which may overlap, are generated to approximate the crane's jib range.
- if the section of two suitable areas does not result in a rectangle, this section will be approximated by several rectangles (see the third fault of the generating method).

Fig. 11 shows the result of the expert system program using the modified generating method. Especially the approximation of the crane's jib range by overlapping rectangles can be seen very well.

The expert system has been realized using the object-oriented language Eiffel [4].

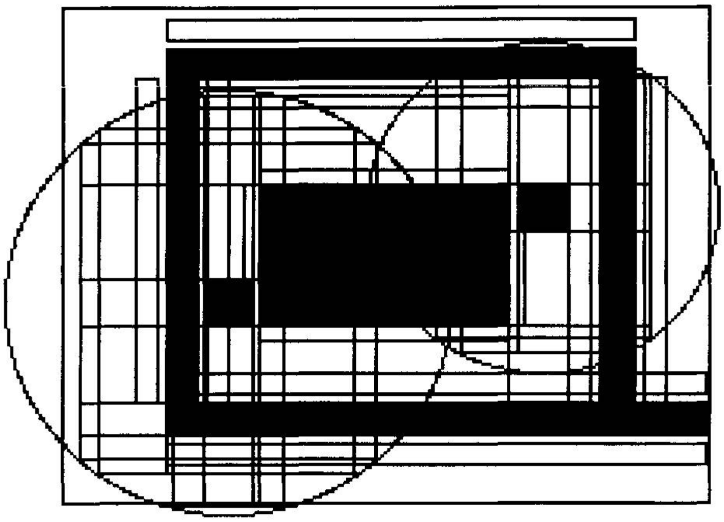


Fig. 11: Output of the expert system using the modified generating method

4. REFERENCES

1. Lennerts, K., Gehbauer, F.: The hybrid solution approach of ESBE, ASCE Proceedings, Anaheim, 1993
2. Lennerts, K., Gehbauer, F.: The User Interface/3D-Visualization Component of ESBE, IABSE-Colloquium, Peking, 1993
3. Lennerts, K., Viesel, K.: A rule processing system for ESBE, Computer Methods in Civil Engineering, Warsaw, 1992
4. Meyer, B.: Eiffel: the language, Prentice Hall International (UK) Ltd., 1992
5. Lennerts, K.: A new approach aggregating vague and uncertain knowledge, unpublished paper.

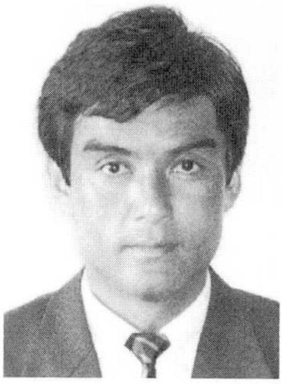
System for Launching Erection Method of Steel Box-Girder Bridges

Système expert pour le lancement de ponts-caissons métalliques

Ein wissensbasiertes System für das Taktschieben von Stahlhohlkastenbrücken

Takayuki NISHIDO
Dr Eng.
Kawada Techno-System
Tokyo, Japan

Yoshito ITOH
Assoc. Prof.
Nagoya University
Nagoya, Japan



Takayuki Nishido, born 1957, received his Dr. of Eng. degree from Nagoya Univ. in 1992. He is currently manager of the Development Div., Kawada, Inc. His research interests includes knowledge advisory systems and economy and safety of bridges.

Yoshito Itoh, born 1952, received his Dr. of Eng. degree from Nagoya Univ. in 1985. He is currently Assoc. Prof. of Civil Eng. at Nagoya Univ. His research includes knowledge-based systems and ultimate strength of structures.

SUMMARY

The authors have proposed a new knowledge approach for the launching erection method to automatically decide the erection steps, decreasing the number of stiffeners required to avoid web buckling. The knowledge of expert design engineers is embedded into the system. The approach considers the stress conditions of a bridge as well as the geometrical conditions, for example, a yard length where blocks of a bridge are joined. The usefulness of the proposed knowledge-support system is demonstrated in application examples.

RÉSUMÉ

Les auteurs proposent un système expert pour la construction de ponts par lancement, afin de déterminer les phases de construction minimisant le besoin en raidisseurs d'âme. Les règles appliquées dans le système sont basées sur les connaissances acquises par des ingénieurs expérimentés. Le système tient compte des conditions de contrainte dans le pont ainsi que des conditions géométriques, comme par exemple la longueur des éléments poussés. La valeur du système expert est démontré par quelques résultats d'application.

ZUSAMMENFASSUNG

Für das Taktschiebeverfahren wird ein wissensbasiertes System vorgeschlagen, das die Verschubzustände im Hinblick auf eine Minimierung notwendiger Stegaussteifungen bestimmt. Die verwendeten Regeln wurden aus den Kenntnissen erfahrener Entwurfsingenieure gewonnen. Sie berücksichtigen die Spannungszustände der Brücke und geometrische Bedingungen wie z.B. die Schusslängen mit der Lage der Konstruktionsfugen. Die Brauchbarkeit des wissensbasierten Hilfsmittels wird anhand einiger exemplarischer Anwendungen demonstriert.

1. INTRODUCTION

In the launching erection method of bridge construction, the bridge superstructure is fabricated in successive segments at one end of the bridge. The completed section of the bridge is launched one stage forward and the procedure is repeated. Fig.1 shows one stage of the erection method. As the support condition in the girder always changes during the erection, the stress condition in the bridge also change.

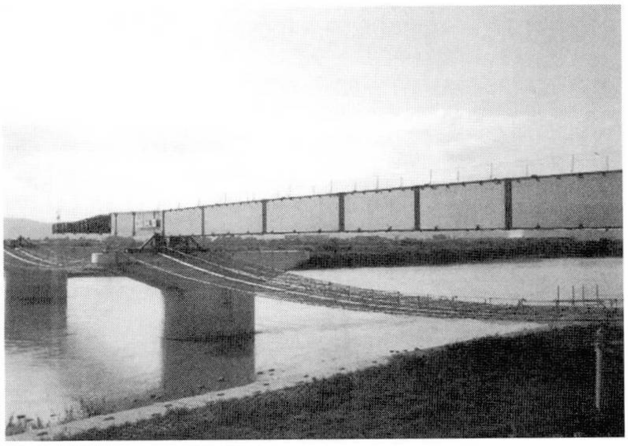


Fig.1 One stage of the launching erection method

Therefore an expert design engineer has to perform many calculations for safety checks in order to adopt this erection method. This erection method is often applied to steel box girder bridges[1], when it is difficult to employ other erection methods because of lack of space in the site.

A convenient program, which is called the checking program here, has been developed for the erection method to easily check the safety of steel box girder bridges. The checking program can consider support conditions during all erection steps. It can automatically make new nodal points for the positions of all supports at calculations. It is written in FORTRAN language and works on a personal computer.

Webs are usually stiffened with horizontal stiffeners to avoid buckling during the erection. The expert design engineer should decrease the number of the stiffeners for the web buckling by selecting the optimum erection steps. The optimum erection steps can not easily be determined because the procedure is very complicated. Expert systems in construction have been developed to resolve unknown or complicated factors[2]. Therefore the authors proposed a knowledge approach method to automatically decide the erection steps decreasing the number of stiffeners required to avoid web buckling. The method considers the stress conditions of a bridge as well as the geometrical conditions, for example, a yard length where blocks of a bridge are joined.

The proposed knowledge support system that is now applied to actual design problems as a prototype uses production rules to embedded the knowledge from expert engineers. The knowledge support system is consists of the checking program and the production rules and works on a personal computer. This paper describes the procedure and features of the proposed method and its usability is demonstrated by some examples.

2. OUTLINE OF THE CHECKING PROGRAM FOR THE LAUNCHING ERECTION METHOD

The launching erection method uses a thin teflon plate and a stainless plate on each support as shown in Fig.2[3]. The teflon plate is put in between a bridge and all supports except a roller one. The bridge is launched using a center-hole type jack and a steel bar at the speed of about 3 mm/sec. The friction coefficient between stainless steel and teflon plate is less than 0.05. Therefore the bridge can be launched with low horizontal force. This is one of

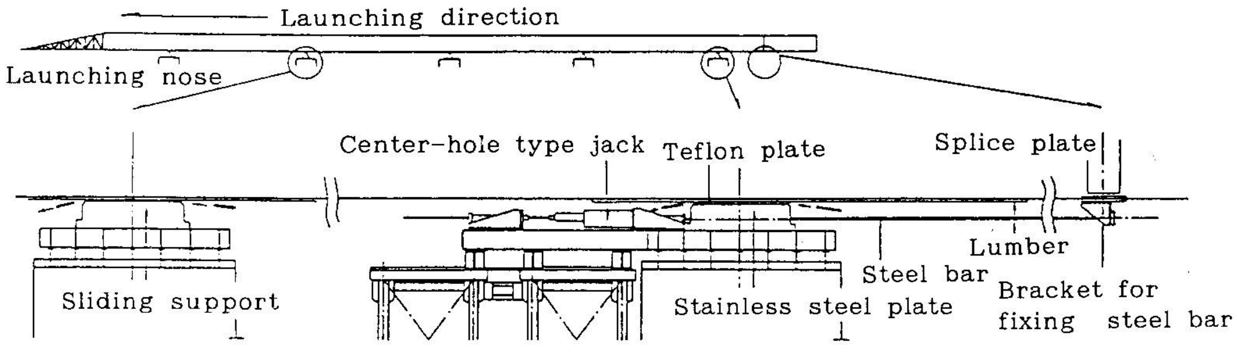


Fig.2 Launching erection method by using teflon plate

the main advantages for this erection method.

In this erection method, the support conditions always change during the erection because the support location moves and the girder length changes. Expert design engineers have to calculate reaction and internal forces at all erection steps by modifying the skeleton data in order to assure bridge safety during the erection. The authors have developed a checking program to determine automatically the whole process.

This program has the following features:

(1) A box girder bridge is idealized by beam elements.
(2) The locations of supports and block joints during the erection steps are automatically made as nodal points in the stress analysis.
(3) Maximum and minimum internal forces at every nodal point through all erection steps are automatically divided.
(4) Safety checks for bending moments and shear forces are done by using the maximum and minimum internal forces.
(5) Required stiffnesses of vertical stiffeners at upper and lower flanges are checked[4].
(6) Webs are checked against buckling.

These procedures are automatically done after inputting the data. Input data include length, weight, sectional properties, support locations and block joints locations at each step. Fig.3 shows the stress patterns of σ_{xb}, σ_{xc}, σ_{yc} and τ by acting forces on the web. The possibilities of buckling are checked at each panel of a web by using Eq.(1)[5].

$$\left(\frac{\sigma_{xb}}{\sigma_{xbcr}}\right)^2 Fs^2 + \left(\frac{\sigma_{xc}}{\sigma_{xccr}}\right) Fs + \left(\frac{\beta_{yc}}{\sigma_{yccr}}\right)^2 Fs^2 + \left(\frac{\beta\,\tau}{\tau_{cr}}\right)^2 Fs^2 = 1.0 \qquad (1)$$

Where, β is a coefficient considering the unbalance in reaction forces on the left and right webs of a box girder bridge. σ_{xbcr}, σ_{xccr}, σ_{yccr} and τ_{cr} are critical buckling stresses obtained by multiplying the buckling factors K_b, K_x, K_y and K_τ by the basic elastic buckling stress σ_e. The Eq.(1) is follows DIN[6]. The web is safe against buckling when the safety factor Fs is more than 1.36 according to the Japanese specifications[5].

3. RULES FOR AUTOMATIC DECISION

Rules are used to automatically decide the erection steps. Fig.4 shows an example of launching erection by expressing the rules. These rules reflect expert design engineers' knowledge and experience. A procedure for automatic decision of erection steps is added to the checking program.

Some of the main rules are as follows:

Rule 1 : As many blocks as possible of the bridge are joined in the yard.

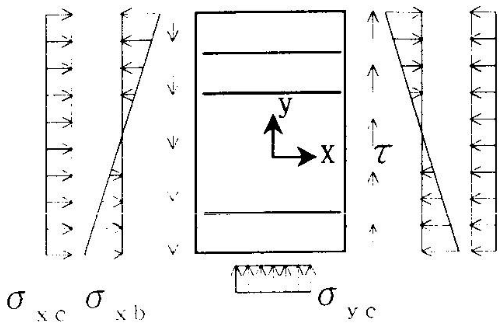


Fig.3 Stress patterns on a web plate

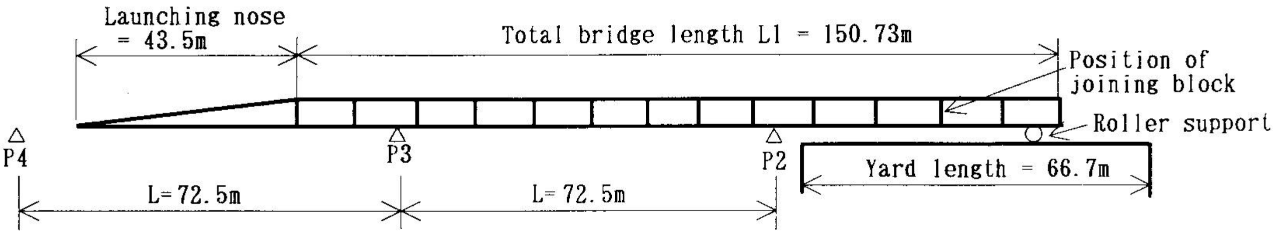


Fig.4 An example of launching erection - Case 1

This erection repeats launching and joining within the yard. The total time of joining blocks takes half of the erection. Of course, a truck crane and impact wrenches need to join blocks. If the bridge is always launched after joining one block, this erection work is inefficient. The collected launching erection can be done according to this rule.

Rule 2 : The roller support is basically located below the diaphragm of the last block.

Only the location of the roller support to the bridge does not change till the joining of new blocks. This rule is considered to avoid the web buckling over the roller support.

Rule 3 : The roller support is replaced when the length to move in a new position is more than 10m.

The reaction force of the first sliding support P3 may decrease by changing the position of the roller support. When the roller support is replaced, the last block must be suspended by the truck crane. If the work of replacing the roller support increases, this erection procedure is not appropriate.

Rule 4 : The roller support is removed within the yard and the distance from the support P2 to the roller one is less than 0.1 times of the completed span length L.

The roller support can not be removed out of the yard. The reaction force of the forward support P2 may be negative when the distance from the forward support to the roller one is too short. The reaction force of the first sliding support may decrease by removing the roller support, but the reaction force of the forward support P2 is critical for its web against buckling when the distance of cantilever is too long.

Rule 5 : Erection steps are always made exactly before and after joining blocks.

The reaction forces of all supports always change after joining blocks. These reaction forces must be calculated to check the safety of the bridge.

Rule 6 : Only the heights of the P4 that is the first sliding support at the final erection step and the roller supports can be adjusted.

The sliding supports except the first one and the roller support are difficult to jack up and down during erection because these reaction forces are large. On the other hand, the first sliding support and the roller support are easy to jack up at joining blocks.

Rule 7 : Erection steps may be made at the intervals of 0.1 times of the completed span length when the bridge total length L1 and the number of supports do not change.

This rule is established from the result of trial calculations. Useless calculations can be removed firing this rule.

These rules are embedded as production rules into the proposed support system. The number of rules is about 30 in the system. The part of knowledge base exists as a subroutine in the system.

4. AUTOMATIC DECIDING METHOD

Figs.5(a) and 5(b) show the flow-charts of automatic decision for the basic steps and for the erection steps except the basic steps respectively. The flow-charts are described by using Fig.6(a) as an example. Initial data for automatic decision are yard length, the possibilities of adjusting the heights of the first sliding support and the roller support, the possibilities of using blocks as counter weights and the position of a diaphragm at each block, and the data for the checking program.

For deciding automatically basic steps as shown in the flow-chart of Fig.5(a), a condition is considered where the tip of the launching nose is immediately before the first support P3 as shown in Fig.6(a). This erection step is called *basic step* here. During this step the web over the forward sliding support P2 is most critical to buckling. The basic step is also most critical for over turning of the cantilevered launching nose.

The checking program has two functions. one is to check against buckling of the web, and another to predict the stability of over-turning of launching nose.

The number of the basic steps is found from the number of all the supports. For example, there are two basic steps in Figs.6(a) and 6(b). At the basic steps, the number of blocks is fixed according to the Rule 1. The position of a roller support is changing within satisfying the Rule 2 and the condition C2 in Fig.5(a). The suitable position of the roller support is that the reaction force of the first sliding support is smallest. Reaction forces at all supports are quickly calculated by using the checking program. The first step from which the launching erection starts is decided in the same way.

If the basic step is decided at this stage, remaining steps would be decided following the rules according to the flow-chart as shown in Fig.5(b). There are some rules in the flow-chart except from the Rule 1 to the Rule 7. Here, L is the completed span length. At joining blocks and setting the roller support, the condition C1, namely the Rule 1, and C2 that is described in Fig.5(a) are used.

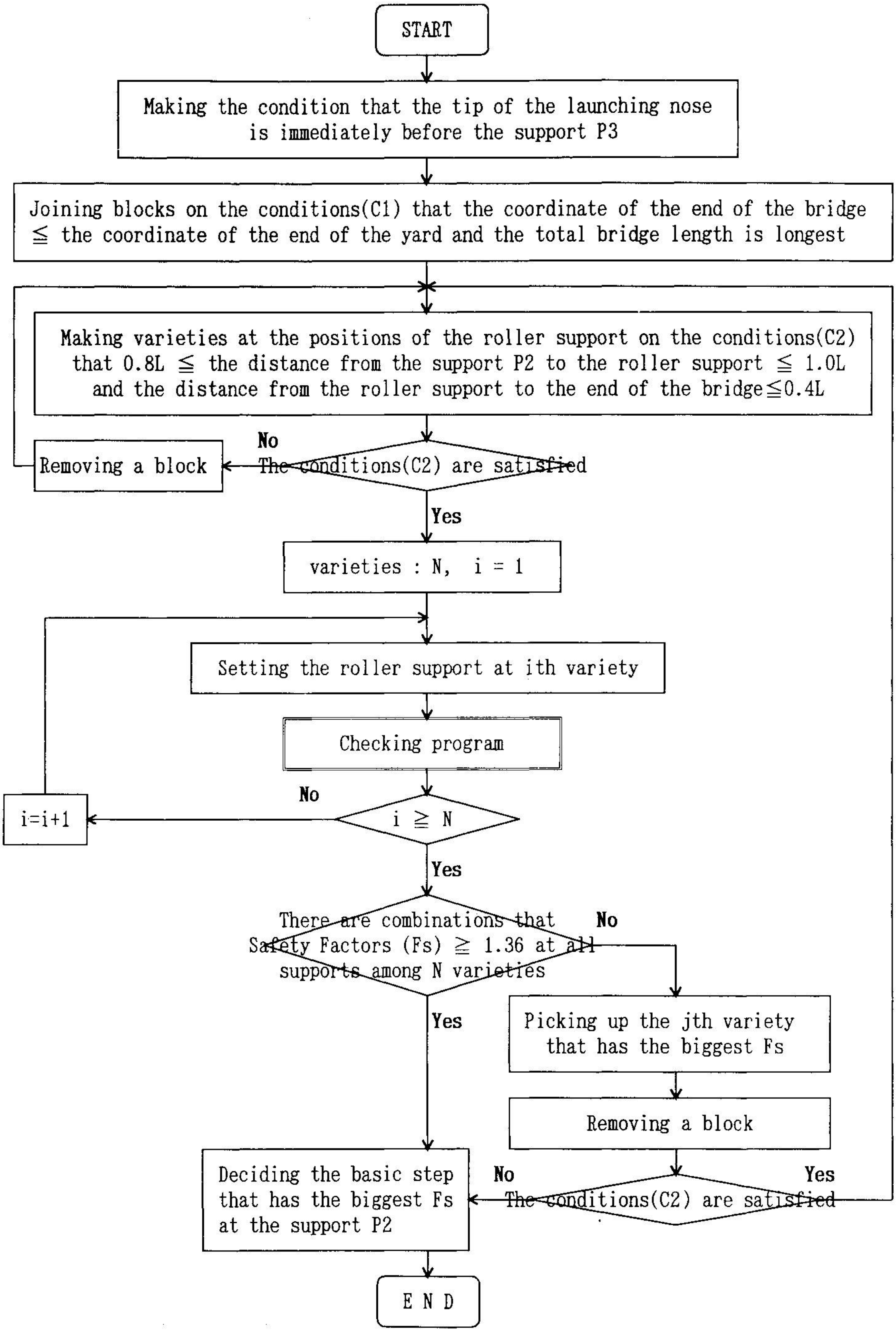


Fig.5 (a) Flow-chart of automatic decision for basic steps

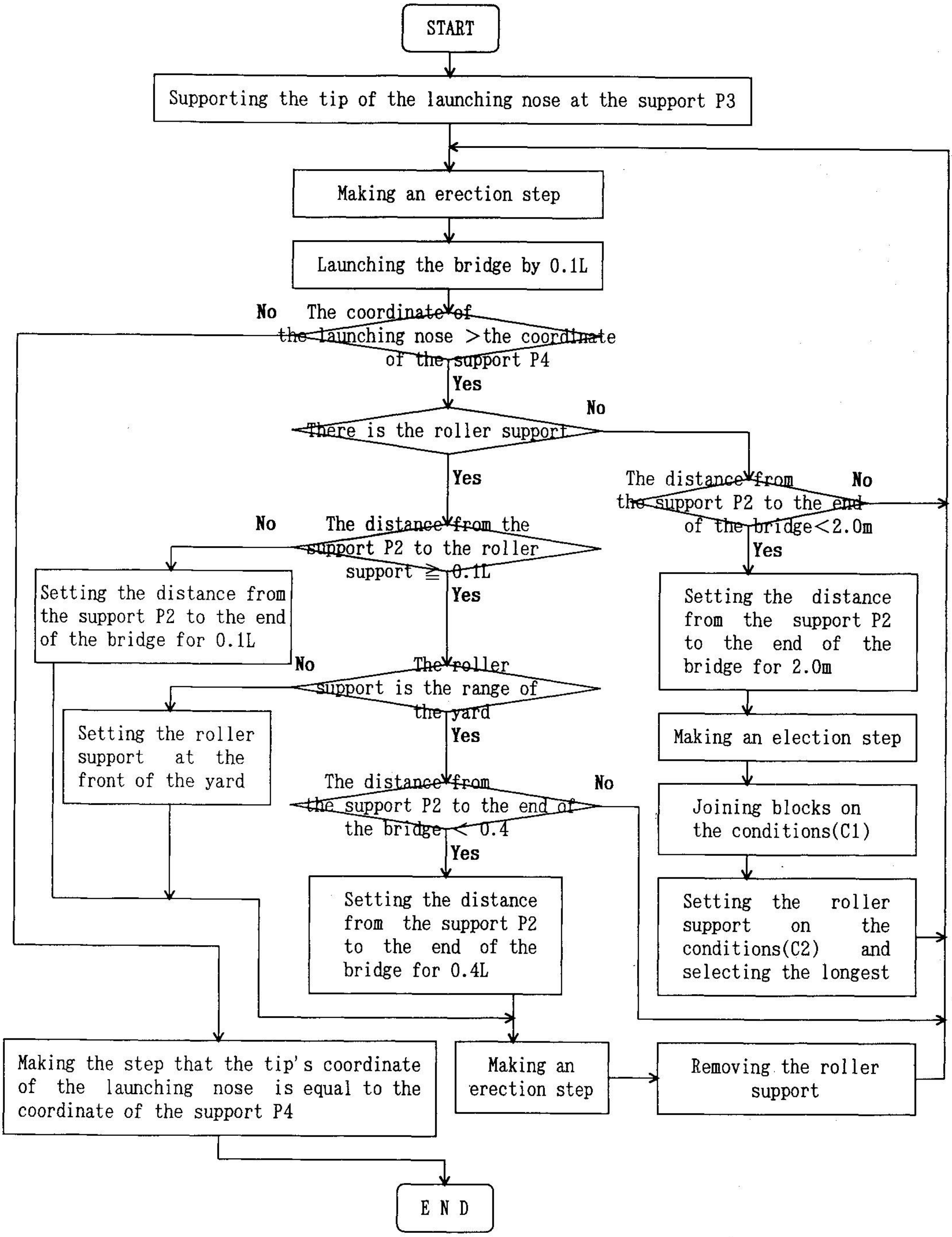


Fig.5 (b) Flow-chart of automatic decision for erection steps except basic steps

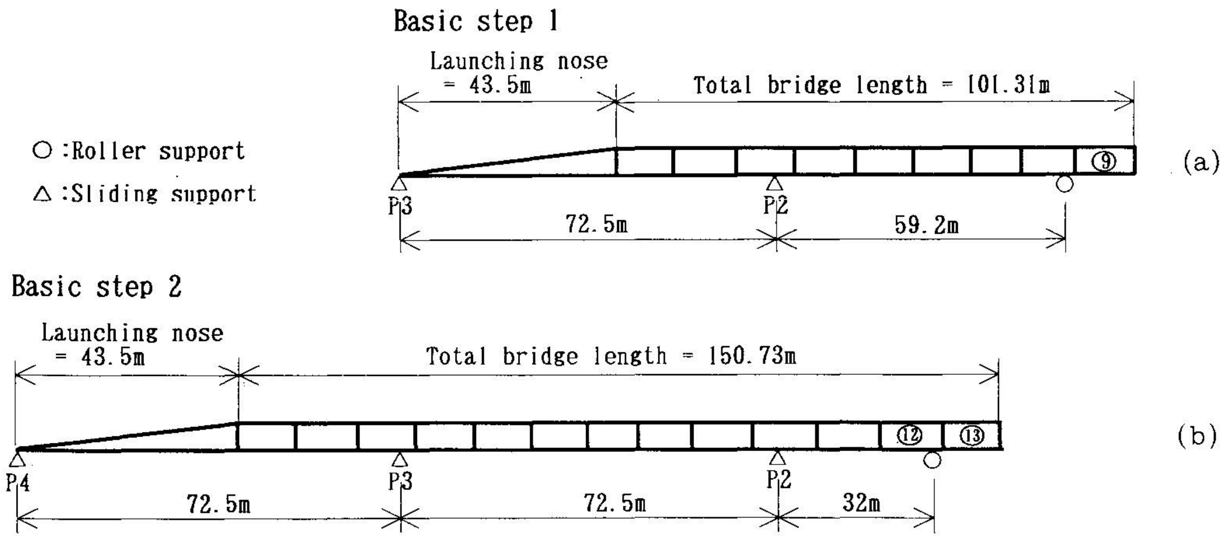


Fig.6 Basic steps in Case 1

Then, the erection steps except the basic steps and the first step are modified to decrease the number of the stiffeners for the web buckling by using the results of buckling analysis. The modification of erection steps ends when the number of stiffeners for the web buckling at the erection steps of the ith variety is less than the number of stiffeners at the erection steps of the (i-1)th variety. The part of knowledge base that is a subroutine uses FORTRAN language because almost production rules can be expressed with the flow-charts in Figs.5(a) and 5(b) by elaborating the procedures of expert designer engineers.

The reaction force of the support before the roller one can be decreased by using a block as a counter weight and increasing the length of cantilever at one end of the bridge. If the first sliding support is jacked up, the safety factor of the web buckling on the second support can increase. On the other hand, if the roller support is jacked up, the safety factor of the web buckling on its forward support can decrease. The reaction force of the first sliding support would be less when the number of spans is odd. Moreover, if the roller support is jacked down, the reaction force of the first support can decrease when the number of spans is even.

5. APPLICATION

Here, the comparisons of the number of stiffeners necessary to avoid web buckling from the obtained proposed knowledge support system and from expert design engineers are done on the two launching erection cases. Case 1 is shown in Fig.4. The erection conditions are as follows: the yard length is 66.7m, blocks are not used as counter weights and the first sliding support P4 and the roller support can be jacked up by 10cm.

Fig.6 shows two basic steps. At the basic step 1, the maximum block number is 9 considering the yard length. The reaction force of the first sliding support is minimum at this position of the roller support by applying the Rule 3. At the basic step 2, the maximum block number is 13. The suitable position of the roller support is on the block number 12 firing the Rule 4.

Table 1 shows the safety factors of the web. It is clear for panels which step's reaction forces that influence the buckling of the web panels can be

Web No.	Coordinates of web(m) Left	Right	Maximum reaction Coordinates(m)	Value(tf)	Support point of left side of web Step No.	Reaction(tf)	Support point of right side of web Step No.	Reaction(tf)	Safety factor Fs
1	42.850	43.762	42.850	58.247	15	55.923	15	55.923	1.38
2	43.762	44.886	44.886	58.560	15	55.923	16	65.710	1.32
3	44.886	46.010	46.010	60.131	15	55.923	16	65.710	1.28
4	46.010	47.133	47.133	61.702	15	55.923	16	65.710	1.24
5	47.133	48.558	48.558	63.694	15	55.923	16	65.710	1.04
6	48.558	49.983	49.983	65.687	15	55.923	16	65.710	1.45
7	49.983	51.408	51.408	67.365	16	65.710	16	65.710	1.41
8	51.408	52.833	52.833	69.040	16	65.710	18	73.936	1.30
9	52.833	54.258	54.258	70.714	16	65.710	18	73.936	1.32
10	54.258	55.683	55.683	72.389	16	65.710	18	73.936	1.28

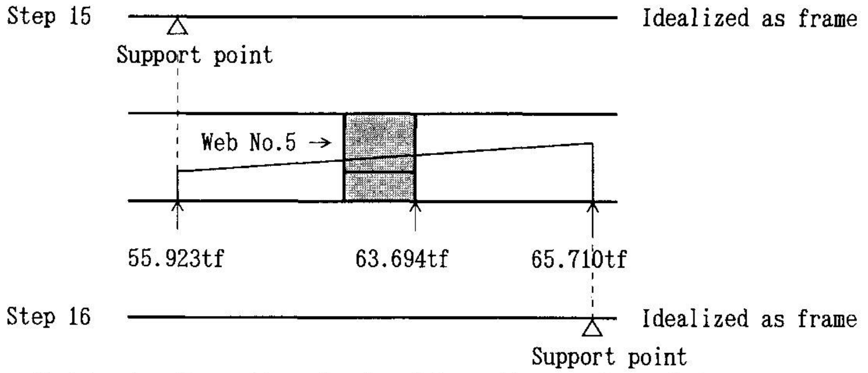


Table 1 Results of checking the web safety

found from this table. For example, at the fifth web, there is no support point in the web. Therefore, the reaction force is calculated to interpolate by using reaction forces outside of both vertical stiffeners, namely, 55.9tf at the step 15 and 65.7tf at the step 16. Although the safety factor of the web is less than 1.36, the factor may be improved by rearranging the position of the roller support and the number of joining blocks on the step 16 according to the rules.

Table 2 shows the number of stiffeners required to avoid web buckling. The number of erection steps is also shown in the table. The number of stiffeners decreases by using the present method even if the heights of the first sliding support and the roller support are not jacked up. The present method can also reduce the number of total erection steps.

Items	An expert planning designer	Present knowledge support system The height of supports No adjusting	Adjusting
Number of stiffeners for the web buckling	44	37	34
Total number of erection steps	36	27	27

Table 2 Number of stiffeners and steps for the web buckling - Case 1

Case 2 is shown in Fig.7. The erection conditions are as follows: the yard length is 130m, blocks are not used as counter weights, and the first sliding support P6 and the roller support can not be adjusted. As shown in Table 3, similar conclusions can be drawn from this example. The present knowledge support system iterates very complicated calculation and many judgments.

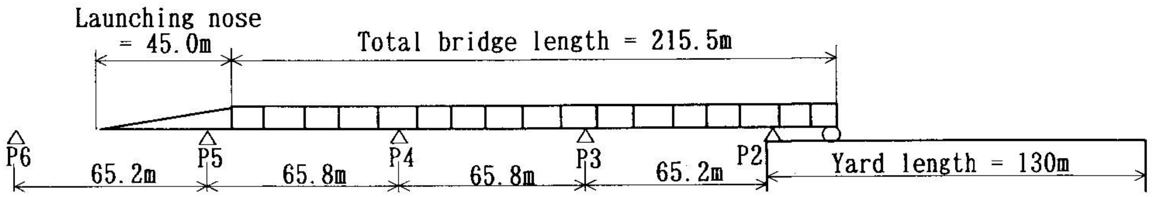


Fig.7 An example of launching erection - Case 2

Items	An expert planning designer	Present knowledge support system The height of supports	
		No adjusting	Adjusting
Number of stiffeners for the web buckling	33	23	—
Total number of erection steps	61	40	—

Table 3 Number of stiffeners and steps for the web buckling - Case 2

6. CONCLUSION

The following main conclusions can be drawn by applying the proposed knowledge approach to the two erection examples.

(1) The proposed knowledge support system including checking program is useful for the actual planning of the launching erection method to steel box girder bridges which requires numerous calculations to assure safety.

(2) Not only the number of horizontal stiffeners required to avoid web buckling, but also the member of total erection steps can be reduced by using the proposed knowledge support system.

(3) Although the weight of all stiffeners to avoid the web buckling at the launching erection is very small compared with the total weight of a bridge, the economy achieved by eliminating the complicated work of attaching horizontal stiffeners is considerable.

REFERENCES

1.Durkee J L., Railway Box-Girder Bridge Erected by Launching. ASCE, ST, No.7/1972.
2.Maher M L., Expert Systems for Civil Engineers - Technology and Application. ASCE, 1987.
3.Maeda K., Nishido T. and Honda S., Study on Reaction Force Control of Launching Erection Method for Steel Bridges by Using Concentrated System. Proc. of JSCE No.415/1990 (in Japanese).
4.Japan Road Association, Specifications for Highway Bridges - Superstructures. 1980.
5.Japan Road Association, Steel Road Bridge Erecting Manual. 1985, (in Japanese).
6.DIN 4114 Blatt 2, Stahlbau, Stabilitätsfälle(Knickung, Kippung, Beulung), Berechnungsgrundlagen, Richtlinien, 1953.

Bridge Fabrication Error Solution Expert System

Système expert tenant compte des erreurs de fabrication dans les ponts

Ein Experten-System zur Fehlererkennung in der Brückenfabrikation

W. M. Kim RODDIS
Assoc. Prof.
University of Kansas
Lawrence, KS, USA

Michael R. HESS
Bridge Eng.
HNTB Corp.
Overland Park, KS, USA

Dr W. M. Kim Roddis is Assoc. Prof. of civil eng. at the Univ. of Kansas. She is involved in research in expert systems and application of artificial intelligence to engineering and steel structures behaviour . Member of expert systems committees and author of papers in the area.

Michael R. Hess earned his M.S. degree in civil engineering from the Univ. of in Kansas in 1994. His research work involved developing an expert system dealing with bridge fabrication errors. He is a bridge design engineer with HNTB Corp. consulting engineers.

SUMMARY

The Bridge Fabrication Error Solution Expert System was developed to help designers and inspectors determine the severity of fabrication errors on steel bridge members and specify the necessary repair. The scope of the system focused on tolerance, drilling and punching, cutting, and lamination fabrication errors that do not have a codified repair procedure. The knowledge acquisition methodology focused on collecting actual cases of past fabrication errors and successful repair. It provided the correct repair in two-thirds of the test cases. This system has been in use at the Kansas Department of Transportation since January 1994.

RÉSUMÉ

Le système expert basé sur la connaissance des erreurs de fabrication des ponts a été établi pour aider projeteurs et vérificateurs à déterminer la sévérité des erreurs dans la fabrication d'éléments métalliques de ponts et à proposer les réparations nécessaires. Le système prend en compte les erreurs de fabrication, la tolérance, le perçage, la perforation, le découpage, et le laminage, qui n'ont pas de procédure codifiée de réparation. L'acquisition des connaissances résulte de cas réels, d'erreurs précédentes de fabrication et de réparations réalisées avec succès. Le système présente une réparation correcte dans 2/3 des cas test. Ce système expert est en usage au Département des Transports du Kansas depuis janvier 1994.

ZUSAMMENFASSUNG

Das Experten-System zur Fehlererkennung in der Brückenfabrikation wurde entwickelt, um dem Konstrukteur und dem Prüfer im Auffinden von ernsthaften Fabrikationsfehlern bei Stahlbrückenteilen zu unterstützen und gegebenenfalls notwendige Nachbesserungen vorzuschlagen. Das System ist auf Toleranz-, Bohrungs-, Stanz-, Schneide- und Laminerungsfehler in der Fabrikation ausgerichtet, die keiner systematischen Nachbesserung unterworfen sind. Die Methode basiert auf dem Festhalten vorhergegangener Fabrikationsfehler und deren erfolgreiche Nachbesserung. Das System schlug eine richtige Besserung bei zwei Dritteln der Fälle vor. Es wird seit Januar 1994 im Verkehrsministerium von Kansas angewendet.

1. INTRODUCTION

Errors arising during the steel fabrication stage may have a catastrophic effect on the performance of a completed highway bridge. More commonly, fabrication errors can cause delays in the fabrication process. All the information needed to support a good decision may not be available at the right time and in the right place to solve the problem in the restricted time necessary to keep the job on schedule. The Bridge Fabrication Error Solution expert system [1] was developed to help design engineers and materials inspectors determine the extent of damage due to fabrication errors and specify the necessary repairs. The development goal was to provide the most suitable repair solution in the most timely manner. The development methodology used a case approach during both the knowledge acquisition stage and the validation and verification procedure. Using the case methodology consisted of gathering cases from actual supporting cases and through interviews with experts (including sample problem-solving protocols). The completed expert system was delivered to the Kansas Department of Transportation (KDOT) in January of 1994.

2. SYSTEM DEVELOPMENT

BFX is a practical example of the successful development of a knowledge-based expert system using modest resources. Approximately 16 person-months of effort were expended on system development, testing, delivery, and training. This effort was largely performed by graduate students under the direction of their supervising professors. BFX was jointly developed by a team from the University of Kansas and a team from Kansas State University [1]. The system was developed and delivered using the Level5 Object shell [2], chosen as a standard for KDOT, running on PC 486 machines.

A specific development methodology with sequential tasks was defined at the beginning of the project consisting of: (1) panel formation and feasibility analysis, (2) conceptual design, (3) knowledge acquisition and engineering, (4) integration and development of pilot delivery application, (5) validation and verification, (6) project evaluation and documentation, and (7) delivery and maintenance. The project development methodology was designed to develop an expert system for any type of domain using the strategies presented in the specified tasks.

Successfully developing a viable expert system required access to and the cooperation of experts in the problem domain. The success of BFX was highly dependent on establishing interaction with target users at an early stage of the project and maintaining contact throughout the development cycle. To meet these requirements, both a panel of experts and a panel of users were assembled, each consisting of six individuals, including design engineers, materials inspectors, and a fabricator. Having participants from all three areas of bridge construction – design, inspection, and fabrication – allowed more interaction and broader input on conditions of errors and repair solutions. Total time spent by all panel members combined was between 2 and 3 person-months. This includes panel meetings, collection of cases, knowledge acquisition interviews, evaluation of the system, and training.

The focus of the system is on fabrication errors that do not have standard code specifications for repair. The scope of the system consists of errors relating to tolerances (dimensional), drilling and punching, cutting, and lamination. The tree graph of the knowledge domain is shown in Figure 1. The errors covered by this program can be classified into four major modules, which are listed below. The tolerance module deals with fabrication errors relating to dimensional tolerances, including hole-boring errors, incorrectly attached members, incorrectly cut members, and stress fractures. The drilling and punching module covers fabrication errors that are the result of incorrect boring procedures, hole sizing, and partially drilled holes. The cutting module covers fabrication errors that result during the cutting process – specifically, gouges and nicks. The lamination module covers fabrication errors that result from edge, internal, or surface lamination defects. It was very important to establish a well bounded scope during development of the expert system so that the design criteria could be applied effectively and in more detail.

The knowledge acquisition occurred in different stages. The first step was to gather case examples directly from fabrication shops, state inspectors' field notes, and bridge project documents. Next, individual interviews were conducted using case studies and hypothetical data case examples based on variations of the actual data cases gathered and interview sessions. Using actual and hypothetical cases, the solution sets for multiple types of errors were determined. Finally, the repair solutions generated were approved by design engineers and inspectors and verified by certified design procedures.

Data cases were gathered from KDOT inspection diaries, fabrication shop quality control records, and bridge design records. These cases were further checked against technical specifications and documentation of current procedures. These case examples were collected from (1) experts' questionnaires to KDOT bridge engineers, fabrication personnel, and inspectors, (2) historical records such as case studies, maintenance data bases, and inspection reports, and (3) simulation results that were generated internally. Actual data cases were cataloged and checked for completeness, then from these actual data cases, hypothetical data cases were created by the knowledge acquisition team to be used during individual interview sessions. The collection of actual cases was partitioned into development examples to be used for knowledge acquisition and test cases to be used for validation and verification.

The personal interviews included one-on-one sessions and, in some cases, two panel members per interview session. These interview sessions were used to gather specific information about certain data cases provided by panel members and also answer hypothetical variations of these data cases. In addition, these sessions were used to discuss the rationale of certain repair solutions associated with problem types described in the data cases. These data cases provided by panel members were actual errors that had occurred during fabrication and were resolved at the fabrication shop. These cases described the errors and their repair solutions.

More data from the interviews were gained by structuring the interviews around developing repair solutions for prepared actual cases and hypothetical cases. Information from actual data cases was also verified by panel members during the interview sessions. Secondary interviews were used to finalize clarification of synthetic data cases and information on technical specification requirements. Interview sessions began by covering actual data cases and clarifying any incomplete information needed for specific data cases. Hypothetical data cases were then presented and repair solutions completed with corresponding information. The documented actual data cases were modified to be hypothetical to collect more information and get as complete coverage of error cases as possible. These hypothetical cases were used to address issues arising from the knowledge base development. The documented data cases were also reviewed during the interviews for confirmation on the repair procedures given. These hypothetical cases included minor and major changes in actual data cases. Repair solutions given for these hypothetical data cases were checked by presenting the cases at subsequent interview sessions with other panel members. Once completed, these cases were included in the prototype development system. Data cases were then transformed into rules for the system program and assisted the design team in understanding the experts' problem-solving techniques.

A Fabrication Error Record document was created to record these cases and information needed for the development system. This document covered all the information needed for the knowledge acquisition of the actual data cases in the development of BFX. The data sheet was distributed to panel members to be referred to when gathering data cases and included the type of documentation required for the individual actual data cases. Initial information gathered at the first panel meetings also helped construct the Case Attribute Value Sheet [3]. This document helped in amassing case data and input necessary for the program development. This document also established a set of classes, attributes, and values that were used consistently for the various program modules.

A dual track was pursued for transformation of the acquired expert knowledge into rules for the knowledge base. One approach was to use inductive learning [4]. Another approach was to extract domain knowledge from literature, documentation of current procedures, and interviews of the experts (including sample problem-solving protocols) [5]. Both approaches made use of the gathered data cases using 77 cases for development and reserving 33 cases for testing. The inductive approach did not prove satisfying for this application due to the incompleteness, irregularity, nonuniformity, and limited number of error cases. Explicit domain knowledge extraction was thus the approach used for development of the BFX knowledge base.

3. SYSTEM ARCHITECTURE

The architecture of BFX consists of a main menu module with six major modules. Two of these major modules – the Help module and the SI units module – are for reference and assistance. These major modules consist of display screens that are available to the user for reference. The Help module has an error index and program module tree graph. This error index includes an alphabetical listing of the errors covered by the system and the submodule in which each can be found. The program module tree graph is an interactive screen that allows the user to select a module or submodule by clicking on the screen, with a text description of that program becoming visible on the screen. Both of these areas of the Help module are able to be modified as additional information becomes available or necessary. The SI units module has three screens which show reference materials on unit conversion and SI unit standards relating to bolt sizes. This module also has the ability to be modified in the future as more detailed information becomes available or necessary.

The other four major modules each consist of submodules that contain specific knowledge areas. The submodules were developed from the scope of the system and are the smallest, most manageable areas that allowed useful development. The user moves from the main menu module to one of the four major modules and then calls the submodule that represents the error type needing to be solve. Once a submodule is called and loaded, all of the necessary knowledge for that particular submodule is resident in the computer's memory. The system allows the user to move among the major modules, submodules, and main menu at different times.

The architecture of BFX was developed using agendas for each submodule. An agenda represents a numbered, hierarchical outline of goals representing the desired hypotheses that can be concluded by a backward-chaining knowledge base [2]. The outline is developed to divide the goals of the knowledge base into logically ordered repair states. The goals are ordered so that the initial goals in the outline require the least information to determine the goal state. Additional goals in the outline are ordered so as to build on the information required of the user. It was important to order the goals in the outline so that a goal state would not be reached before all necessary information from the user was checked for repairs that could occur using the given information. This was done by defining any hierarchical subgoals within any primary set of goals.

Backward chaining was used in BFX to reach the individual goals of the agenda. The system checks the goals by firing individual rules corresponding to the order of the goal statements. The user is prompted for information to prove these rules. As input from the user is gathered by the system, the goal is either proved or disproved. Once a goal has been disproved, the system then selects the next goal state in the hierarchy of the outline. Forward chaining was used once a goal state was successfully proved. The forward-chaining rules fire when the goal state has been proved and cause the corresponding conclusion text to be displayed for the user. Repair recommendations were placed in an array of the system's domain. The hierarchy of the goals does not cause a conclusion to be reached before all necessary information has been entered into the system. During the development and addition of rules to the system, continuous checks of the goal hierarchy were made. The system's repair recommendations were tested with the actual and hypothetical data cases received during the interview portion of the explicit domain knowledge extraction process.

BFX was developed as an expert system that can be maintained and kept current to accommodate new fabrication errors introduced to the system. The activities of the maintenance phase include processing of system modifications and the continual evaluation of the system. Modification may be necessary on the operation, logic, interface, or knowledge base of the system. BFX was designed to allow addition of knowledge to the system and increases in the scope. The system was segmented into individual submodules to allow easier modification and maintenance, with each submodule corresponding to an individual scope area of the system. Strong emphasis was placed on the rule ordering and hierarchy to cover all ranges of responses, allowing the user to answer questions on individual display screens without concerns about the order of the answers. The better the system is maintained, the more comprehensive and useful it will be to KDOT; therefore, it was important that KDOT personnel be trained in procedures and methods of modifying the system.

To address the issues of maintenance and modification, a training seminar was established on BFX for KDOT personnel [6]. The seminar's purpose was to familiarize KDOT personnel with the technical specifications of the knowledge base and provide sufficient instruction for them to perform basic maintenance on the program without outside assistance. Basic maintenance includes direct changes and additions to items in the rule base; however, it does not include fundamental changes in system capabilities or major restructuring of the knowledge base.

4. SYSTEM PERFORMANCE

System performance testing results are summarized in Table 1. Validation and verification of the system was based on two methods. The first method was the actual running of BFX by the panel of experts and the panel of users. Panel members ran a total of 18 hypothetical cases on the system. The hypothetical cases were based on actual cases that the panel members had experienced. The total 18 panel test cases resulted in 11 correct solutions, 6 no solutions, and 1 incomplete solution. The second method was checking the performance of BFX using 33 actual cases provided by panel members. These cases had not been used in system development and met the scope of the system. After running the 33 test cases, 21 of the cases gave the correct repair solution for each case. Twelve of the cases did not match the contents of the knowledge base during runs of the system. When a fabrication error case is run on the system and no match between that particular type of error and the knowledge base occurs, the system will inform the user and suggest that the error case be implemented into the system. No match between the test cases and the knowledge base occurs when these particular types of errors have not been found during development of the knowledge base.

Module	Development Cases		Validation Cases						Results			
Class	Distribution		Panel Cases		Actual Cases		Total Cases		Correct	No Match	Incom-plete	Wrong Solution
Tolerance	56	72.8%	13	72.2%	25	75.8%	38	74.4%	21	17		
Drill/Punch	7	9.1%	2	11.1%	5	15.2%	7	13.7%	6	1		
Cutting	9	11.7%	2	11.1%	2	6.1%	4	7.8%	3		1	
Lamination	5	6.5%	1	5.6%	1	3.0%	2	4.0%	2			
Total	77		18		33		51		32	18	1	0
Percent									63%	35%	2%	0%

Table 1 Case Distribution

No logic errors occurred during any testing stage of the system, which shows that in terms of reliability, the system performed very accurately. This is very important in building user confidence; it is much better to receive no answer than an incorrect one. The distribution of development cases by module roughly matches the distribution of validation test cases by module. The percentage distribution of the development cases may be assumed to give a rough measure of distribution of error types encountered in practice by KDOT, since the development cases were collected from past KDOT experience. Combining both validation methods, BFX reached the correct solution in 63% of the cases, determined that the case did not match the contents of the knowledge base and therefore not making a recommendation in 35% of the cases, and provide an insufficiently detailed recommendation in 2% of the cases.

5. EXAMPLE

One operational case involving several uses of BFX is presented to demonstrate BFX's capabilities. This example deals with mislocated holes at a plate girder flange splice. Several holes were misdrilled in the bottom flange of a plate girder, as shown in Figure 2. The hole mislocations resulted in a variety of fabrication errors. First, the specified splice plate will no longer fit the hole locations in the bottom flange. This problem was entered into the tolerance module of BFX with the mislocated hole submodule selected. The input described the lack of fit problem. BFX's recommended solution was to leave the existing hole in the main member and make a new splice plate to match the existing hole pattern. The repair specified in Figure 2 does indeed use this approach.

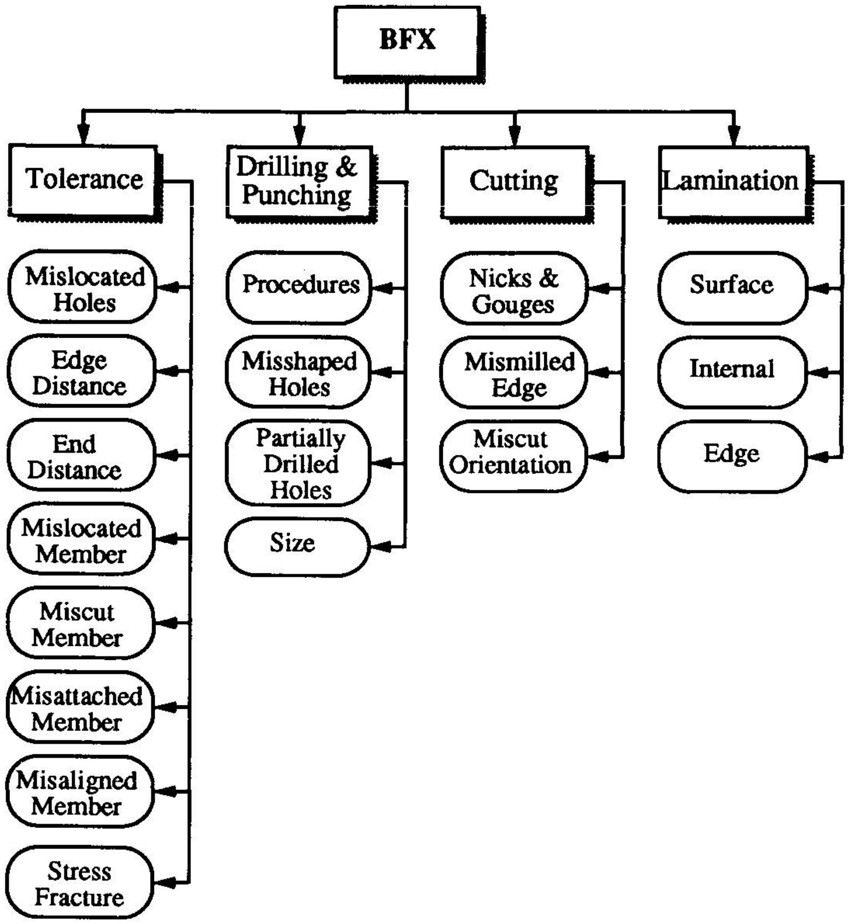


Figure 1 BFX Knowledge Tree

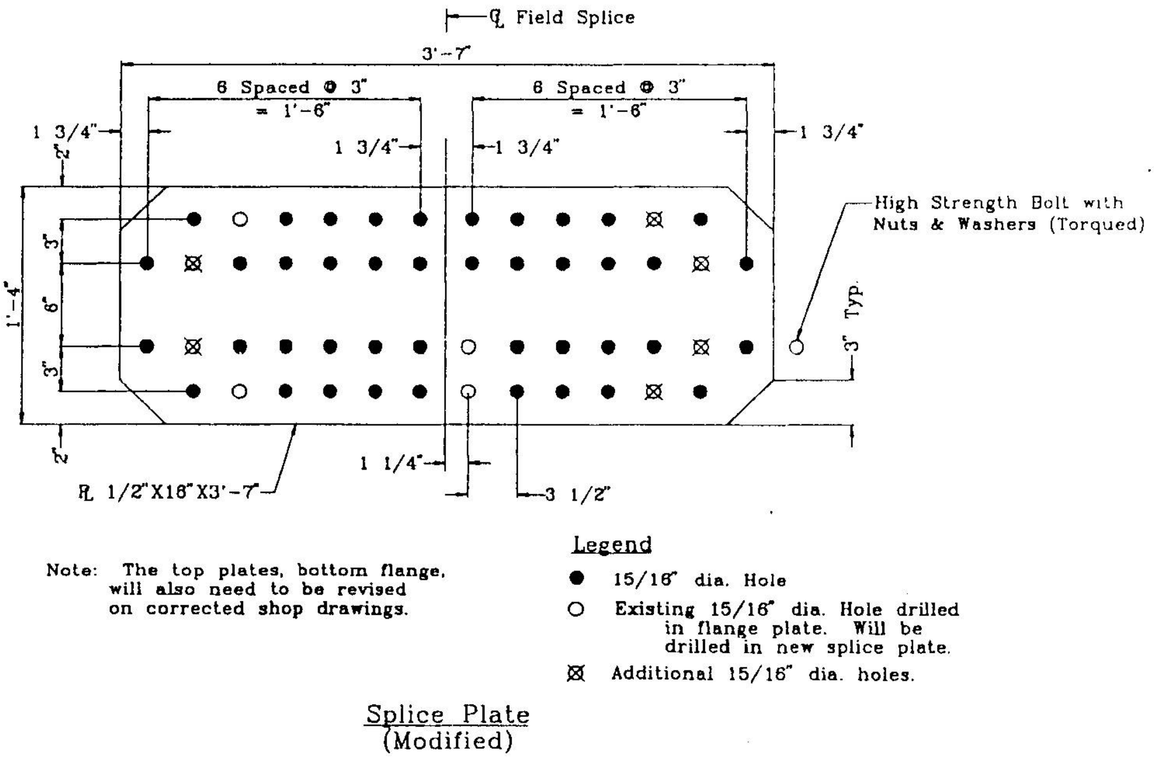


Figure 2 Operational Example: Mislocated Holes at Flange Splice

Second, the mislocated hole on the extreme right is superfluous since it begins an additional row beyond those specified. This problem was entered into the tolerance module of BFX with the mislocated hole submodule again selected. The input this time described the extra bolt line problem. BFX's recommended solution was to leave the existing splice in the specified location and then take one of the following options: 1) extend the splice plate to cover the mislocated holes and drill to match, or 2) place bolts and washers in the additional holes and leave the splice plate as designed. The repair specified in Figure 2 takes the second approach. Third, the two mislocated holes immediately to the right of the splice centerline violate end distance requirements. This problem was entered into the tolerance module of BFX with the end distance submodule selected. BFX's recommended solution was to add additional bolts in the bolt line if possible or cut and replace the member if not possible. The repair specified in Figure 2 takes the first approach. The total repair specified in Figure 2 is thus a superposition of the three approaches recommended by BFX for the three individual problems generated by the hole mislocations.

6. CONCLUSIONS

The development of BFX has resulted in the following conclusions:

- BFX achieved the performance expectations desired by KDOT.
- BFX achieved the desired scope and accuracy established by KDOT. The knowledge domain was very suitable for development.
- The development methodology of using panels and experts was successful for this project.
- Explicit domain extraction was the best method of knowledge acquisition, given the domain and knowledge available.
- Modular development of BFX allowed easier development and will make maintenance and modifications by KDOT less complicated.

ACKNOWLEDGEMENTS

This work was partially funded by the Kansas Department of Transportation Cooperative Research Program (K-TRAN), an ongoing, cooperative and comprehensive research program addressing transportation needs for the State of Kansas. The cooperation and contributions of Dr. Hani Melhem, Mr. Srinath Nagaraja, and Ms. Amy Moran of Kansas State University are gratefully acknowledged.

REFERENCES

1. HESS, MICHAEL R., RODDIS, W. M. KIM, SRINATH NAGARAJA, HANI G. MELHEM, and AMY MORAN, “An Expert System for Fabrication Error Solutions,” Report No. K-TRAN/KU/KSU-93-2, Kansas Department of Transportation, Topeka KS, August 1994.
2. INFORMATION BUILDERS, INC., "Level5 Object: Object-Oriented Expert Systems for Microsoft Windows," Information Builders, Inc., New York, 1993.
3. HESS, MICHAEL R., and RODDIS, W. M. KIM, "Bridge Fabrication Error Solution Expert System User's Manual," *SL Report* No. 94-1, University of Kansas Center for Research, Inc., Lawrence, KS, 1994.
4. NAGARAJA, SRINATH, and MELHEM, HANI, “An Inductive Knowledge-Based System for Mitigation of Bridge Fabrication Errors,” Computing in Civil Engineering Proceedings of the First Congress held in conjunction with A/E/C Systems'94, June 20-22, 1994, Washington DC, Edited by Khalil Khozevich, ASCE, New York NY, pp 1083-1090.
5. RODDIS, W. M. KIM, and HESS, MICHAEL R., "Bridge Fabrication Error Solution: Developing a Knowledge-Based Expert System for a State Department of Transportation," American Society of Civil Engineers Computing Conference, Washington, 1994.
6. HESS, MICHAEL R.; RODDIS, W. M. KIM; MELHEM, HANI; and MORAN, AMY, "Bridge Fabrication Error Solution Expert System Training Manual," *SL Report* No. 94-2, University of Kansas Center for Research, Inc., Lawrence, KS, 1994.

Leere Seite
Blank page
Page vide

Knowledge-Based Connection Design in Steel Structures

Projet et calcul, basé sur la connaissance, d'assemblages en construction métallique

Wissensbasiertes Entwerfen von Anschlüssen im Stahlbau

J.T. DE GELDER
TNO Building & Constr. Res.
Delft, The Netherlands

C.M. STEENHUIS
TNO Building & Constr. Res.
Delft, The Netherlands

A.P. HUIJSING
TNO Building & Constr. Res.
Delft, The Netherlands

SUMMARY

A complex task in the design process of steel structures is the design of the connections between the members. To support the structural engineer in the connection design process, software has been developed that is able to calculate the capacity of the connection based on its layout. This software, however, does not support the design of the layout itself. The structural engineer has to enter and modify the layout using his own expert knowledge and experience. Due to lack of knowledge, this often leads to connections which are more expensive to manufacture than necessary. This paper describes an approach to develop a knowledge-based system that supports a structural engineer in designing more efficient and cheaper connections in steel structures.

RÉSUMÉ

Le projet et le calcul d'assemblages des cadres métalliques est une activité complexe. Afin d'assister le projeteur, des programmes ont été développés, lesquels permettent de calculer la résistance d'un assemblage sur la base de sa forme. L'ingénieur doit cependant décider de la forme sur la base de ses propres connaissances et expériences. Un manque de connaissances conduit souvent à des assemblages trop coûteux. L'article décrit le développement d'un système de connaissances permettant à l'ingénieur de concevoir et réaliser des assemblages de cadres métalliques plus efficaces et moins chers.

ZUSAMMENFASSUNG

Der Entwurf von Anschlüssen im Stahlrahmenbau ist eine komplexe Tätigkeit. Um den Konstruktionsingenieur bei dieser Aufgabe zu unterstützen, wurden Programme entwickelt, die aufgrund des Layouts die Tragfähigkeit eines Anschlusses berechnen können. Der Ingenieur muss jedoch immer noch aus Fachwissen und Erfahrung heraus das Layout eingeben und gegebenenfalls abändern. Mangelndes Wissen führt oft zu Anschlüssen, die unnötig teuer in der Fertigung sind. Der Beitrag beschreibt die Entwicklung eines wissensbasierten Systems, das dem Ingenieur hilft, effizientere und daher billigere Stahlbauanschlüsse zu entwerfen.

1. INTRODUCTION

Connection design is a knowledge intensive task in the design process of steel structures. Knowledge is required about mechanical and economical aspects of connections. The connection design process is usually performed by the steel fabricator, who's knowledge about connection design is limited. This paper describes how knowledge technology can help to complete this limited knowledge. To explain the specific problems concerning connection design the process of designing steel structures and connections in particular is explained in chapter 2. Chapter 3 describes how conceptual modelling of knowledge is performed by the Centre of Knowledge Based Systems at TNO Building and Construction Research (TNO-KBS). Chapter 4 describes a prototype knowledge based system for connection design developed at TNO-KBS. Finally, chapter 5 presents some concluding remarks.

2. CONNECTION DESIGN IN STEEL STRUCTURES

2.1 Design process of steel structures

In general, the design process of a steel frame consists of 7 steps (see figure 1). In the modelling phase (step 1), the engineer models the joints as pinned or rigid. Pinned joints are capable of transmitting the forces calculated in design, without developing significant moments which might adversely affect the beams or columns in the frame. A rigid joint has no influence on the distribution of internal forces and moments in the frame, nor on its overall deformation.

1.	Mechanical modelling of the frame in the building.
2.	Estimation of loads.
3.	Pre-design of beams and columns.
4.	Determination of forces and displacements in the frame.
5.	Check of beams and columns in limit state conditions.
6.	If required, adjustment of beams and columns (continue with step 4).
7.	Design of joints.

Fig. 1 Design process of a steel frame

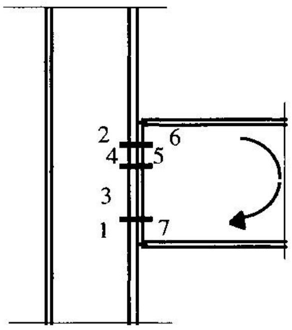
Normally global frame analysis is carried out with first order elastic analysis [1]. Input for this analysis are the loads and the stiffness of beams and columns. Output are the deflexions of the frame and the force distributions. The member sizes and the forces which should be transmitted by the joints are the starting point of the design of joints (step 7). The purpose of this design is to find a layout capable of transmitting the forces between the beam and the column (figure 2). This design process is commonly used in various European countries. It follows a practice in which the engineer designs the members and the steel fabricator the connections.

Fig. 2 Design task of steel fabricator

This paper focuses on moment joints and assumes that the rigidity of the joints is anyhow sufficient, and that a joint should be designed for strength only. The here given approach of conceptual modelling of connection design could be extended to rigidity checks. This is, however, not essential for the approach and therefore omitted in this paper.

2.2 Connection design with Eurocode 3 and CASTA/Connections

Eurocode 3 [2] gives rules for the determination of strength, stiffness and rotation capacity of beam-to-column joints. These rules are based on a so-called component-approach. A joint is divided into components. Figure 3 shows the components for a beam-to-column end plate joint.



- column web in compression (1), tension (2) and shear (3);
- column flange in bending (4);
- end plate in bending (5);
- beam web in tension (6);
- beam web and flange in compression (7).

Fig. 3 Components in a beam-to-column end plate joint.

The mechanical properties of the components can be determined with the rules given in Eurocode 3. The properties of the joint should be determined based on the properties of the components. Eurocode 3 provides freedom to choose whatever layout of a joint is the best. The drawback of this freedom is that many potential failure modes in a joint need to be checked. The determination of connection properties is consequently a comprehensive task. Programs have therefore been developed for the determination of joint properties according to Eurocode 3 [3].

One of the programs developed is CASTA/Connections [4]. Joint-types covered are: bolted end-plate, cleated and base plate connections between I and H shaped sections. In this paper we focus on its potential for end plate beam-to-column joints. Figure 4 shows some possible alternatives CASTA/Connections can deal with.

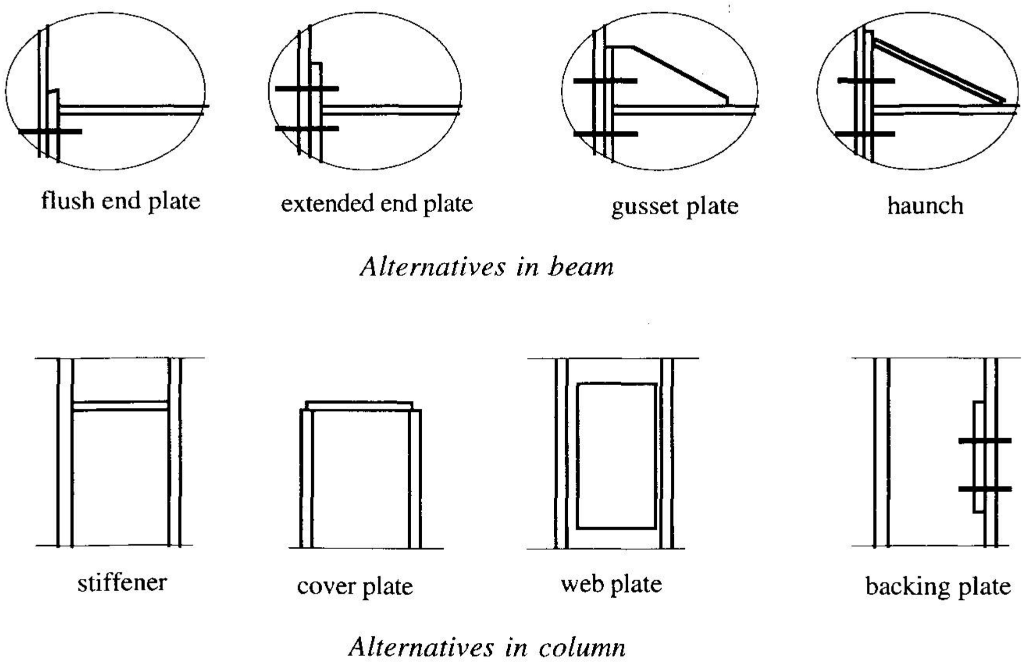


Fig. 4 Some alternatives CASTA/Connections can deal with

CASTA/Connections calculates a moment-rotation curve of an end plated joint, or checks if a joint is capable to transmit the applied forces. It can be used as follows (figure 5): (1) A designer inputs the layout of a joint into the program. (2) After acceptance of the layout, the program calculates the mechanical properties. (3) The program reports the mechanical properties and the first and second component that will collapse.

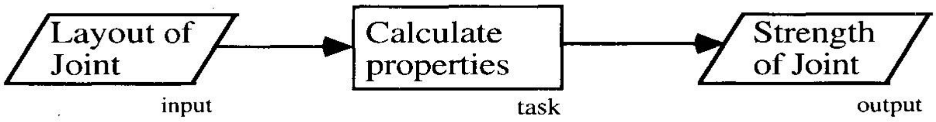


Fig. 5 Design task supported by CASTA/Connections

The task supported by CASTA/Connections is only a subtask of the design task of the steel fabricator. In figure 6 the design task of the steel fabricator (see figure 2) has been decomposed into its subtasks. It shows that the design task is an iterative process. In this process CASTA/Connections supports the subtask *Calculate Properties*. The steel fabricator has to perform the subtasks *Initialize Layout Joint* and *Modify Layout.* This task requires knowledge of mechanical and manufacturing aspects, and is hard to automate because of its creative nature and large number of alternative solutions. A risk is, however, that sub-optimal solutions are achieved due to the limited design knowledge of a practitioner. A solution to this problem is a knowledge based system that can be used by a practitioner to complete his knowledge and that helps to support the design of cheaper connections. In chapter 4 a prototype of such a knowledge based system is described.

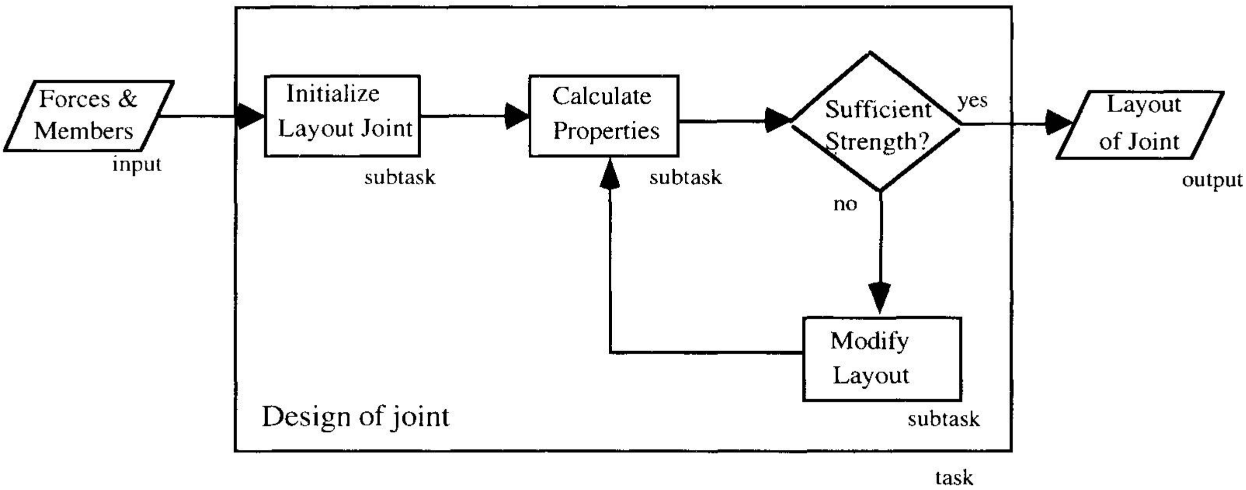


Fig. 6 Design task of a steel fabricator divided into subtasks

3. CONCEPTUAL MODELLING OF KNOWLEDGE

The application of Knowledge Based Systems in practice is growing and will continue growing into the next decades [5]. In the development of knowledge based systems conceptual modelling of knowledge is the main activity [6]. It is of vital importance for the success of the knowledge based system that the knowledge in the system reflects reality and is modelled correctly, completely and consistently. The development of conceptual models requires:

(1) a conceptual modelling theory
(2) a conceptual modelling language that is compatible with the conceptual modelling theory
(3) a conceptual modelling tool that supports the development of conceptual models using the conceptual modelling language

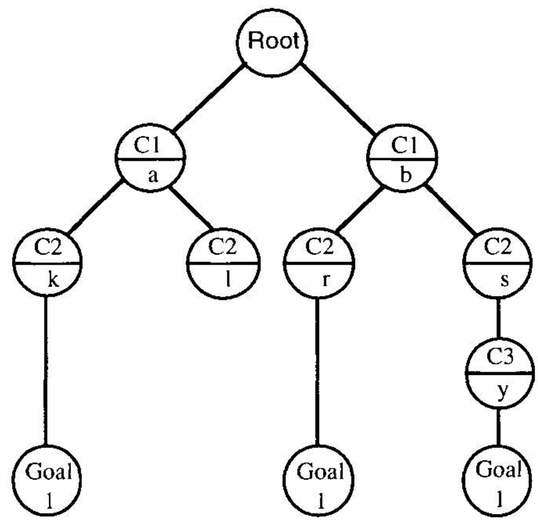
3.1 Conceptual modelling theory: Theory of functional classifications

At TNO-KBS knowledge is modelled following a theory, called the theory of functional classifications [6,7,8,9]. In this theory conceptual modelling of knowledge is viewed as a process in which concepts are modelled following a goal-oriented approach. Only knowledge which is necessary for the goal should be part of the conceptual model.

Often there are several alternatives to reach a specific goal or to describe a concept. In the theory of functional classifications this is called functional equivalence. In a conceptual modelling process a knowledge engineer should try to find functional equivalent solutions in order to get a complete conceptual model. Functional equivalence can appear in three different ways. This is illustrated by figure 7.

3.2 Conceptual modelling language: Decision Tables and Prolog

In the development of knowledge based systems often the language of production rules is used to represent knowledge. This is mainly because most of the commercially available knowledge based system shells are using production rules. Production rules are very powerful to represent knowledge. However, production rules have the drawback that their correctness, completeness and consistency is hard to validate, especially in case of large knowledge bases containing hundreds or thousands of rules. Another drawback is that knowledge bases of production rules are hard to maintain.



1. Under certain conditions additional descriptors may become important to reach the goal. If C1=b and C2=s then C3 becomes a descriptor. If C1=a C3 is not relevant at all.

2. Categorisations of descriptors influence each other. If C1=a then the classification of C2 is (k,l). If C1=b then the classification of C2 is (r,s). This phenomenon is called conceptual interaction between descriptors.

3. Within a goal-constructed category descriptors may have different values. If the values k1 and k2 fall within the category k then a value k1 for C2 is equivalent to a value k2 for C2 if C1=a.

Fig. 7 Appearances of functional equivalence

At TNO-KBS a combination of Decision Tables (DT's) and Prolog is used as a Conceptual Modelling language [9]. Figure 8 shows an example of a DT. A DT is divided into four components separated by double lines. The component left of the vertical double line is called the stub. Above the horizontal double line the stub contains condition parameters (descriptors) and below the horizontal double line action parameters (goals or concepts). The components right of the vertical double line contain the condition categories (above the horizontal double line) and action categories (below the horizontal double line).

In contrast with production rules correctness, completeness and consistency of knowledge represented in DT's can easily be validated because of the structured representation of knowledge. Another benefit is that the expert in the knowledge domain can easily read the model and validate its correctness and completeness [10]. Further DT's have the same power as production rules, because DT's actually are structured production rules. The DT in figure 8, for instance, represents 5 production rules.

Goal 1					
C1	a		b		
C2	k	l	r	s	
C3	-	-	-	not y	y
Goal 1	X	-	X	-	X
	R1	R2	R3	R4	R5

(1) IF (C1 = in category a) AND (C2 = in category k) THEN (Goal 1 is reached)
(2) IF (C1 = in category a) AND (C2 = in category l) THEN (Goal 1 is not reached)
(3) IF (C1 = in category b) AND (C2 = in category r) THEN (Goal 1 is reached)
(4) IF (C1 = in category b) AND (C2 = in category s) AND (C3 = not in category y) THEN (Goal 1 is not reached)
(5) IF (C1 = in category b) AND (C2 = in category s) AND (C3 = in category y) THEN (Goal 1 is reached)

Fig. 8 A Decision Table as a set of structured production rules

The Decision Table language is compatible with the theory of functional classifications. The use of DT's commands a goal oriented approach. The goal is normally represented as an action parameter in a DT. Condition parameters and categories have to be found which are necessary to reach the goal. Functional equivalence can easily be represented in DT's. The DT in figure 8 contains all three appearances of functional equivalence. Rule 4 and rule 5 illustrate that C3 becomes important as an additional descriptor under certain conditions. Conceptual interaction is represented in the influence of C1 on C2's categorisations. And the notion that a value of s1 or s2 for C2 doesn't affect the reaching of the goal illustrates that different values may fall within one goal-constructed category.

Knowledge is usually represented in an hierarchy of DT's. A condition parameter in one table can be an action parameter in another table. In this case the condition parameter is actually a subgoal that has to be reached first in order to reach the ultimate goal. In Artificial Intelligence this process of finding a value for a goal parameter (i.e. the action parameter of the main table) is known as a backward chaining process.

There are some types of knowledge which are not easily represented by DT's [11]. Examples are recursive processes, procedural functions, unconditional decisions and database facilities. For these types of knowledge Prolog is a powerful language. Actually Prolog is a language that is able to represent knowledge in a way compatible to all aspects of the theory of functional classifications, but validation of knowledge represented in Prolog is much more difficult than validation of knowledge in DT's. However, in combination Decision Tables and Prolog is a powerful language to represent knowledge.

3.3 Conceptual modelling tool: Advanced Knowledge Transfer System (AKTS)

At TNO-KBS a knowledge modelling tool has been developed called Advanced Knowledge Transfer System (AKTS) [9]. In AKTS knowledge is modelled in DT's and Prolog. AKTS has graphical editing facilities to build an hierarchy of DT's. Further, Prolog statements can be used anywhere and programs in Prolog or other languages can be called from various places in a DT.

Knowledge included in AKTS can be consulted if one of the parameters is defined as the goal parameter. In a consultation process AKTS tries to find a value for the goal parameter by trying to find values for subgoals, sub-subgoals, etc. The system tries several consequent steps to find a value for a specific parameter:

(1) it starts a WHEN NEEDED demon (if defined for the parameter) that might calculate a value for the parameter or read a value from a database. Besides delivering a value for the parameter a WHEN NEEDED demon can also be used to execute a command, for instance, to show a picture or to set initial values of other parameters;
(2) if a WHEN NEEDED demon isn't defined or doesn't deliver a value for the parameter, the system looks for a table having the parameter as action parameter;
(3) if no table is found having the parameter as action parameter, the system reads the default value for the parameter (if defined);
(4) if no default is defined finally the user is asked to provide a value. If the parameter is defined as ASK FIRST the user is asked to provide a value first before the system tries the other strategies.

4. PROTOTYPE KNOWLEDGE BASED SYSTEM FOR CONNECTION DESIGN

The aim of the Knowledge Based System for connection design is to support a designer in performing the subtask *Modify Layout* in the design process of end plate beam-to-column joints (see figure 6). This aim differs from the aim other researchers using Decision Tables for building applications have [10,11,12,13,14]. They use Decision Tables to represent the design standard. Their systems support the design subtask *Calculate properties* and the *Sufficient Strength* check (see figure 6). Although we agree that representing design standards into Decision Tables has great advantages in terms of clearness, flexibility and maintenance [10,12,13], we didn't do this for the practical reason that we already had a computer program available in which the design standard was hard-coded: CASTA-Connections, Therefore we could concentrate on supporting the real creative design task. For a designer this task is complex because the design standard Eurocode 3 provides freedom to choose whatever layout of a joint is the best. Only the requirements defined in the design standard need to be satisfied. This implies that in a specific situation a lot of alternative solutions can be valid. The prototype system supports the designer to find among the alternatives a solution that satisfies the design standard and is optimal in terms of economy. To reduce complexity the knowledge in the prototype is limited to only 3 of the 7 components defined in chapter 2: end plate in bending, column web in shear and column web in compression. Further the number of alternative elements to select is limited as well. Only haunch, stiffener and web plate are included. Figure 9 displays the system of DT's.

4.1 Decision Table *Design Layout of Joint*

The parameter *layout of joint* in DT *Design Layout of Joint* has been assigned the property MAIN GOAL (see figure 10). This means that the goal of the system is to find a value for this parameter. In order to find a value for the goal parameter *layout of joint* the system first executes the WHEN NEEDED demon defined for this parameter (see description of consequent steps to find a value for a parameter in 3.3). The demon defines an initial simple layout: a thin endplate to connect the column and the beam and no other elements. It represents the subtask *Initialize Layout Joint* in the design process (see figure 6). Since no value for the goal parameter has been traced yet the system looks for a DT having the goal parameter as action parameter. The system finds DT *Design Layout of Joint* and starts tracing the first condition parameter *sufficient strength.* This parameter becomes a subgoal. The parameter *sufficient strength* doesn't have a WHEN NEEDED clause, so the system looks for a DT having *sufficient strength* as action parameter. It finds DT *Sufficient Strength.*

Design Layout of Joint

C1	sufficient strength	yes	no	
C2	layout modified	-	yes	no
A1	COMMAND		repeat_table	repeat_table
A2	layout of joint	{!thickness end plate!, !number of web plates!, !height haunch!, !stiffener!}		
		R1	R2	R3

Sufficient Strength

C1	first moment capacity	X < !required moment capacity!	X >= !required moment capacity!
A1	sufficient strength	no	yes
		R1	R2

Modify Layout

C1	first collapse mechanism	end plate in bending		column web in shear		column web in compression		
C2	element to be modified	thickness end plate	height haunch	number of web plates	height haunch	number of web plates	height haunch	stiffener
A1	COMMAND	reset('thickness end plate')	reset('height haunch')	reset('number of web plates')	reset('height haunch')	reset('number of web plates')	reset('height haunch')	reset('stiffener')
A2	layout modified	yes	yes	yes	yes	yes	yes	yes
		R1	R2	R3	R4	R5	R6	R7

Modification Advice

C1	first collapse mechanism	end plate in bending		column web in shear			column web in compression			
C2	second moment capacity	X < !required moment capacity!, X/!first moment capacity! < 1.5	ELSE	X < !required moment capacity!, X/!first moment capacity! < 1.5		ELSE	X < !required moment capacity!, X/!first moment capacity! < 1.5		ELSE	
C3	second collapse mechanism	-	-	end plate in bending	column web in compression	-	end plate in bending	column web in shear	-	
C4	number of web plates	-	-	-	-	-	-	-	0	1 OR 2
A1	advised to modify	height haunch	thickness end plate	height haunch	number of web plates	number of web plates	height haunch	number of web plates	stiffener	height haunch
		R1	R2	R3	R4	R5	R6	R7	R8	R9

Fig. 9 Decision Table System of prototype

4.2 Decision Table *Sufficient Strength*

In DT *Sufficient Strength* the system starts tracing parameter *first moment capacity*. In order to find a value for this parameter the system first executes the WHEN NEEDED demon (see figure 10). This demon represents the subtask *Calculate properties* in the design process (see figure 6). The system is calling CASTA/Connections which returns the first and second moment capacity and the first and second collapse mechanism.

PARAMETER	: layout of joint
WHEN NEEDED	: ?thickness of end plate?=5, ?number of web plates?=0, ?height haunch?=0, ?stiffener?=no
TYPE	: set
MAIN GOAL	
PARAMETER	: first moment capacity
WHEN NEEDED	: calculate_properties (!thickness of end plate!, !number of web plates!, !height haunch!, !stiffener!, ?first moment capacity?, ?first collapse mechanism?, ?second moment capacity?, ?second collapse mechanism?)
TYPE	: real
PARAMETER	: element to be modified
PROMPT	: What element do you like to modify? (advised: !advised to modify!)
DEFAULT	: !advised to modify!
TYPE	: text
ASK FIRST	

Fig. 10 Parameter properties of some parameters

The first moment capacity of the joint indicates the maximum moment which can be transferred through the joint. The occuring mechanism at this moment level is called the first collapse mechanism. It refers to the weakest component in the joint. The second moment capacity is the capacity which can be reached theoretically if the first collapse mechanism is prevented (for example by a stiffener). The occurring collapse mechanism is indicated by the second collapse mechanism. It refers to the one but weakest component in the joint.

If the first moment capacity is less than the required moment capacity the designed connection hasn't sufficient strength yet (parameter *sufficient strength* =no) and the system continues tracing a value for the parameter *layout modified* in DT *Design Layout of Joint.* This parameter doesn't have a WHEN NEEDED demon, so the system looks for a DT having this parameter as action parameter and finds DT *Modify Layout.*

4.3 Decision Table *Modify Layout*

This DT describes which modifications affect the strength of the joint in case of a particular collapse mechanism. This is important knowledge for a designer. Novice designers usually don't have this knowledge. They often apply alternatives which only have limited effect on the strength of the joint. Often stiffeners are applied whatever collapse mechanism occurs.

When the system enters DT *Modify Layout*, parameter *first collapse mechanism* already has a value (calculated by CASTA/Connections; see figure 10: WHEN NEEDED demon of parameter *first moment capacity*), so the first parameter to trace is *element to be modified.* This parameter is defined as ASK FIRST (see figure 10), so the user is asked to provide a value. The PROMPT defined for this parameter (see figure 10) contains a reference to a parameter that hasn't been traced yet: parameter *advised to modify.* This parameter doesn't have a WHEN NEEDED demon, so the system looks for a DT having this parameter as action parameter and finds DT *Modification Advice.*

4.4 Decision Table *Modification Advice*

DT *Modification Advice* gives advice on which of the alternative modifications modelled in DT *Modify Layout* will have most effect on the strength of the joint. The advice primarily depends on the first collapse mechanism, but in some situations also knowledge about the second moment capacity and collapse mechanism influences the advice. DT *Modification Advice*, for instance, describes that in case of end plate in bending as the first collapse mechanism the alternative modifications are applying a haunch (when a haunch is already present, increase its height) or applying a thicker end plate. A novice designer will probably decide to apply a thicker end plate in all situations. However, applying a thicker end plate is only an effective choice, if the second moment capacity exceeds the required moment capacity or the second moment capacity is 1.5 times greater than the first moment capacity. If the second moment capacity is close to the first moment capacity an expert designer knows that applying a thicker end plate will help to increase the first moment capacity of the connection, but it will definitely not be sufficient to reach the required moment capacity. If a novice designer still decides to apply a thicker end plate he will surely need an additional element in the next step. In this situation an expert designer will therefore apply a haunch or increase the height of the present haunch, because a haunch is not only effective to increase the moment capacity if end plate in bending is the collapse mechanism, but is also effective in case of other collapse mechanisms (see DT *Modify Layout.*). By applying a haunch the designer will definitely reach the goal in less steps than by applying a thicker end plate. In many cases it also leads to a cheaper layout, because the novice designer will probably not go back to a thinner end plate again because he has the feeling that he made progress by applying the thicker end plate.

The structure of DT *Modification Advice* shows the importance of functional classifications. All three appearances of functional equivalence are present in this DT: (1) in some situations an additional parameter *second collapse mechanism* or *number of web plates* becomes relevant, (2) depending on the value of the parameter *first collapse mechanism*, parameter *second collapse mechanism* has different categories and (3) the presence of 1 or 2 web plates is equivalent for the advice to modify height haunch in case of column web in compression.

4.5 Continuation of the process

After tracing Decision Table *Modification Advice* the system has a value for the parameter *advised to modify* and shows this value as an advice when prompting the user for a value of the parameter *element to be modified* in DT *Modify Layout*. The user is not forced to follow the advice. The advised element is default (see figure 10), but it's still allowed to choose another alternative. After choosing a value for the parameter *element to be modified*, the system resets the current value of the element and returns to DT *Design Layout of Joint* and executes the Prolog-statement *repeat_table*. DT *Design Layout of Joint* is entered again and a value for the parameter *sufficient strength* is traced followed by a tracing of the parameter *first moment capacity*. As described before a value for this parameter is calculated by executing its WHEN NEEDED demon, i.e. running CASTA/Connections. The difference this time however is that the input parameter, that represents the element to be modified, doesn't have a value yet, because its value was reset in DT *Modify Layout*. So, the system first has to find a value for this parameter before the demon can be executed. Since all parameters representing the joint elements are defined as ASK FIRST (not mentioned in figure 10) the user is asked to provide a value. After giving a value CASTA/Connections calculates new values for the first and second moment capacity and collapse mechanism. The first moment capacity is compared to the required moment capacity etc. This process continues until the first moment capacity is greater than or equal to the required moment capacity. Then parameter *sufficient strength* receives value "yes" and the system halts and reports the final layout.

5. CONCLUSION

This paper described the development of a prototype Knowledge Based System for connection design in steel structures. Although the knowledge represented in the prototype is limited, the approach showed that, following the theory of functional classifications and using the conceptual modelling languages, Decision Tables and Prolog, a practical application can be developed. It turned out that by using Decision Tables and the tool AKTS the expert could easily validate the knowledge and give suggestions for improvement. Consequently this approach will lead to knowledge based systems which better reflect reality and better supports the solution of problems in practice. Hopefully this is an important step towards a situation wherein knowledge based systems will really be accepted as useful tools by people working in the building industry.

6. REFERENCES

1. ECCS, Analysis and Design of Steel Frames with Semi-Rigid Joints, ECCS-TC8, publication no 67, Brussels, Belgium, 1992;
2. EUROCODE 3, ENV - 1993-1-1, Design of Steel Structures, Commission of the European Communities, European Prenorm, Brussels, Belgium, April 1992;
3. BROZZETTI, J., Design of Connections in the EUREKA "CIMSTEEL" project, Proceedings of the Second International Workshop on Connections in Steel Structures, edited by Bjorhovde R., Colson A., Haaijer G. and Stark J., AISC, Chicago, Illinois, USA, 1992;
4. STEENHUIS, M., DOL C. & VAN GORP, L. Computerised calculation of force distributions in bolted end plate connections according to Eurocode 3, to be published in Journal of Constructional Steel Research, 1994;
5. HAYES-ROTH, F. & JACOBSTEIN, N. The State of Knowledge Based Systems, Communications of the ACM, Vol. 37, No. 3, pp. 27-39, March 1994;
6. GELDER, J. T. DE., & LUCARDIE, G. L., Reconstructing Conceptual Models of Structures, Proceeding of IABSE Colloquium Knowledge-Based Systems in Civil Engineering, pp. 65-76, Beijing, China, 1993;
7. LUCARDIE, G.L., A Functional Approach to Realizing Decision Support Systems in Technical Regulation Management for Design and Construction, Design & Decision Support Systems in Architecture (Kluwer Academic Publishers), pp. 97-109, 1993;
8. GELDER, J. T. DE., & LUCARDIE, G. L., Knowledge and Data Modelling for CAD/CAM Applications, Design & Decision Support Systems in Architecture (Kluwer Academic Publishers), pp. 111-121, 1993;
9. LUCARDIE, G.L., GELDER, J. T. DE, HUIJSING, A.P., DTSS: A Functional Application of Decision Tables and Prolog as a Conceptual Modelling Language for Knowledge Based Systems, Proceedings of the 1st Workshop of the European Group for Structural Engineering Applications of Artificial Intelligence (EG-SEA-AI), pp. 192-204, Lausanne, Switzerland, 1993;
10. FENVES, S.J., Tabular Decision Logic for Structural Design, Journal of the Structural Division, 92(6), pp. 473-490, 1966;
11. GARRETT, J.H.& HAKIM, M.M., Object-Oriented Model of Engineering Design Standards, Journal Computing in Civil Engineering, Vol. 6, No. 3, pp. 323-347, 1992;
12. GARRETT, J.H.& FENVES, S.J., A Knowledge-Based Standards Processor for Structural Component Design, Engineering with Computers, Vol. 2, No. 2, pp. 219-238, 1987;
13. GARRETT, J.H.& FENVES, S.J., Knowledge-Based Standard Independent Member Design, Journal of Structural Engineering, Vol. 115, No. 6, pp. 1396-1411, 1989;
14. CRONEMBOLD, J.R. & LAW, K.H., Automatic Processing of Design Standards, Journal Computing in Civil Engineering, Vol. 2, No. 3, pp. 255-273, 1988.

Taking Advantage of Design Process Models

Tirer profit de modèles du processus de la conception

Ausnützung von Entwurfsprozessmodellen

Sylvie BOULANGER
Civil Eng.
Swiss Fed. Inst. of Technol.
Lausanne, Switzerland

Esther GELLE
Computer Scientist
Swiss Fed. Inst. of Technol.
Lausanne, Switzerland

Ian SMITH
Dr Eng.
Swiss Fed. Inst. of Technol.
Lausanne, Switzerland

SUMMARY

This paper describes explicit representations of design processes for improving knowledge acquisition, implementation and user-interface design. A model for conceptual design of bridges uses assumptions and physical principles as well as design criteria and design strategies for incorporating several starting points, directed trial-and-error and multiple solution traces. The implementation is non-monotonic and uses the constraint-propagation paradigm. Through integrating attributes of other models and several processes, an extended maze provides an intuitive mapping of reasoning and knowledge for conceptual structural design.

RÉSUMÉ

Cet article présente une description explicite du processus de la conception pour faciliter l'acquisition de la connaissance, l'implantation informatique et la conception de l'interface utilisateur. Le modèle retenu traite la conception préliminaire des ponts et est fondé sur des hypothèses et des principes physiques, ainsi que sur des critères et des stratégies de conception pour incorporer différents points de départ ainsi que des cheminements de solution multiples. L'implantation est non-monotone et s'appuie sur le paradigme de la propagation des contraintes. En assimilant des attributs propres à d'autres modèles et à plusieurs processus, un labyrinthe étendu permet un rapprochement plus intuitif du raisonnement et de la connaissance utilisés lors de la conception des ouvrages.

ZUSAMMENFASSUNG

Dieser Artikel beschreibt explizite Repräsentationen des Entwurfsprozesses mit dem Ziel der Verbesserung der Wissenserfassung, der Implementierung und der Benutzerschnittstellen. Ein Modell für den Vorentwurf von Brücken benützt neben Annahmen und physikalischen Prinzipien ebenso Entwurfskriterien und -strategien zur Behandlung von verschiedenen Ausgangssituationen, gerichtetem "trial-and-error" sowie alternativen Lösungswegen. Die Implementierung erfolgt nichtmonoton und benutzt das Fortpflanzungsparadigma für Nebenbedingungen. Durch Anpassung der Eigenschaften weiterer Modelle und Einbeziehung von Produkt- und Prozessmodellen entsteht ein Labyrinth aus intuitiven Uberlegungs- und Wissensmustern für den Vorentwurf von Bauwerken.

1 Introduction

No two artifacts are designed in the same way and therefore it is difficult to develop useful models of design. Different assumptions are made in light of incomplete information. When decisions regarding the refinement and the direction of the design are made and when conflicts occur, they are resolved using different criteria and strategies.

Given the complexity of design tasks, models of design processes often propose a procession from general to specific, little support for incomplete information, a rigid structure for conflict resolution, and few possibilities for deviating from a global plan. The need for more realistic models has been recognised for many years (Simon, 1981, Gero, 1993, Fenves, 1992, Holgate, 1986), and many computational models have been proposed which partially support such characteristics. Ganeshan, Finger and Garrett (1991) provide an environment for capturing the intent of a decision. Sause and Powell (1991) maintain two levels for the development of design steps. Many researchers have recognised the need to incorporate more than one paradigm : Bañares-Alcántara's (1992) two hypotheses, Bowen and Bahler's (1992) multiple perspectives, Tong and Tueni's (1990) control and domain levels, Soo and Wang's (1992) qualitative and quantitative reasoning, and Zhao and Maher's (1992) analogy and mutation.

Our goal is not to automate conceptual design but to augment human designers' creativity and to provide an explanatory trace of their steps. Representation in design is knowledge-intensive and reasoning is dynamic and temporarily inconsistent (Gero, 1993). We begin by observing experts during sketching in order to extract a set of desirable features for recognising various starting points, trial-and-error approaches, and end points during conceptual design of bridges. We propose traces of solutions with multiple uses of design knowledge, applying design models as conceptual frameworks and including explicit representation of assumptions, design criteria and design strategies for developing alternatives and for resolving conflicts. Thus we aim to provide support for design exploration as proposed by other researchers (Petrie, Cutowsky and Park, 1994, Logan, Corne and Smithers, 1992, Brazier, van Langen, Ruttkay and Treur, 1994). We emphasise the desirability of working with a model, as distinct from am implementation version of it, see Figure 1. Models become the focal point for the iterative process used during development, providing a common platform for interacting with experts (a) and the implementation team (b) and for integrating the user-interface (c). A model can be very different than the cognitive map of an expert's mind and also different than the implementation.

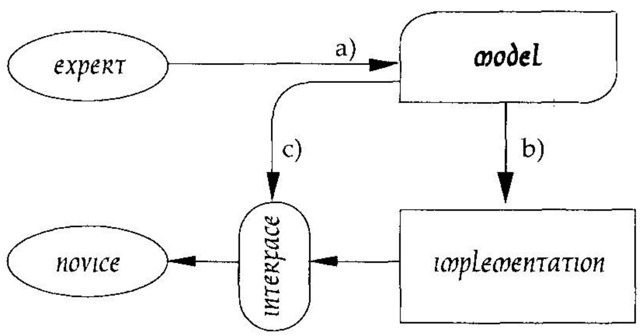


Figure 1 — The role of design models during development. A model should assist knowledge acquisition (a), system implementation (b), and user interface design (c). A model acts as a focal point for interacting with experts and the implementation team, and provides the basis for development of the user interface.

2 Models of design processes

A model, in our context, is a non-exhaustive description of key elements of the design process and provides a framework of important design aspects in order to facilitate knowledge acquisition, implementation and interface design. Models of design processes help capture initial conditions, state transformations and final specifications. Transformation implies a change from one state to another, such as from function to structure, from abstract to concrete concepts, and from qualitative to quantitative attributes. Models represent problem solving, search, decision-making and exploration which englobe major design tasks. A global environment needs to provide multiple opportunities for advancing the design. In Figure 2, several models are presented, linear, tree and semi-lattice models as well as more complex network and maze models.

A **linear model** assumes that a design problem can be segregated into successive tasks resulting into one set of specifications. Initial models of the design process were mostly linear. It presupposes the existence of the one-best-solution. However, that idea is incompatible with many design tasks where a high number of solutions are plausible. Designers use *satisficing* methods (Simon, 1981) which cannot be embraced by a linear model. Since in design, an interdependency of parameters meshes quickly during exploration, the linear model cannot be used to represent a global design process.

A **tree model** is an improvement since more than one alternative can be elaborated. It allows the solution space to be decomposed into subgoals and provides an environment for the 'generate-and-test' method. It is commonly used to represent decision nodes. The tree model typifies conventional decomposition methods or hierarchical approaches provided that weak interactions exist (Stefik, 1981, Maher, 1989, Mittal and Araya, 1990, Topping and Kumar, 1989). They may also represent AND/OR graphs (Sause and Powell, 1991, Bédard and Ravi, 1991). The **semi-lattice** is an enlargement of the tree model as it offers accommodation for dependent subgoals. During her analysis of empirical design studies, Visser (1991) observed that design activities deviated so regularly from a decomposition approach that it could not be representative of a global design control strategy.

A **network** is another improvement as it can model interdependencies between subgoals and provide support for backtracking to previous decisions. Non-monotonic systems can partially be modelled with networks. Many researchers have included the network as a successful representation. Zhao and Maher (1992) use a network-based prototype where the links represent dependencies as well as domain-independent relations such as qualitative, quantitative, and inequality relations while global operations consist of a blend of mutation and analogical reasoning. Garcia and Howard's (1992) ADD (Augmented Design Documentation) is based on design and decision network models. The design network model provides local (or microlevel) relations which consist of activities such as : generate, constrain, evaluate and select while the decision network model provide global relations such as sequencing, composition and dependency. Network models have evolved significantly to the point where they have become a preferred model for many researchers as they offer a better support for exploration.

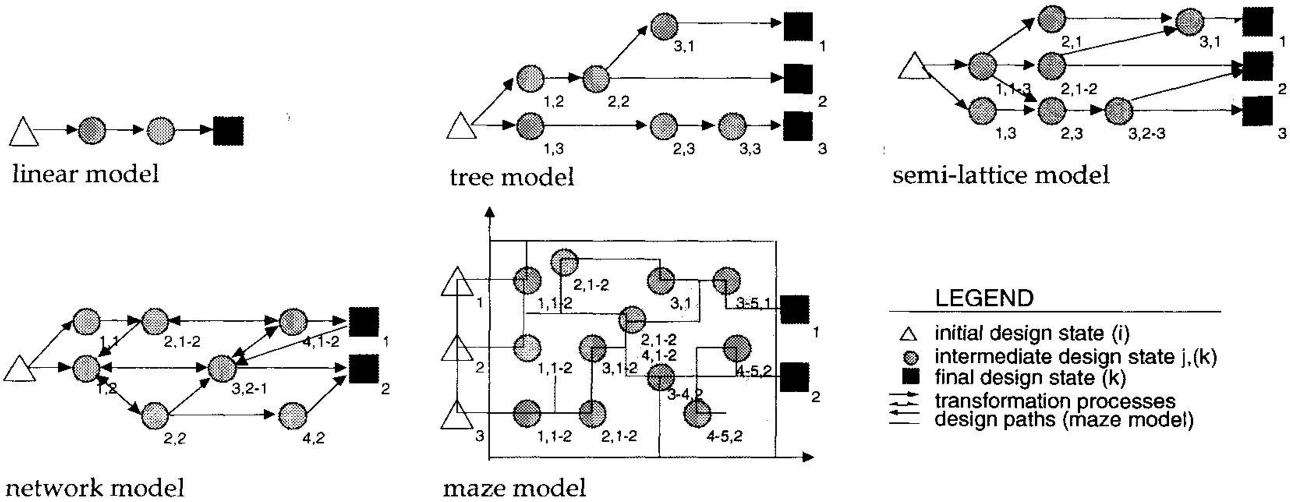


Figure 2 — Design process models inspired from Holgate (1981) and Bañares-Alcántara (1991)

The **network** and the **tree** models are often used to model *and* implement design systems. Their implementation strengths are not questioned. However, as Gardiner (1987) points out, strict hierarchical conceptualizations may be convenient for mimicking human performance but there are important problems associated with them for explaining and for describing human performance. These models, she claims, tend to assume too much knowledge on the part of the human. On the other hand, it is difficult to imagine a model which is completely hierarchy-free. A compromise for a model would be to represent a product loosely and redundantly, and allow a dynamic and fluid creation of the design path. The maze model is proposed to help satisfy these requirements.

A **maze** facilitates the integration of typical design tasks such as imitating false-starts, backtracking, lateral searches, the presence of more than one point of entry and exit, and the strong influence of the selection of the starting point on the end result (Holgate, 1986). Newell and Simon (1972) describe the GPS (General Problem Solver) for selectively searching through a large environment as moving through a large maze and reducing it to manageable proportions. In the maze, Visser's (1991) concept of *cognitive cost*, that is, as soon as other actions are more interesting, the engineer deviates from a global plan in favour of these actions, can be integrated. The maze then, appears suitable for representing a global control process, with the possibility of introducing local plans which may take on the form of another maze, a network, a tree or a linear model. Another distinction is the presence of axes; for example, the horizontal axis may represent parameters while the vertical axis indicates the range of each parameter. The dynamic creation of the path makes it less inhibiting for designers to proceed. The intricacy of the paths, in a maze, models more realistically exploration and multiple models of design processes.

3 Observations from conceptual design sketching

Sketching provides a rich medium for observing processes and collecting knowledge, and hence it was used as the basis for prescribing model requirements. Sketching is an effective informal method for starting, developing and communicating a design. Sketching is dynamic and constraint-free. Although imprecise, sketching is both concise and realistic. Some researchers have recognised sketching as such an important activity during conceptual design that computer-support is under development. For example, Jenkins and Martin (1993) have partially completed a system for automatic sketch input, called Easel. Gross and Zimring (1994) are adding a link between diagrams in Archie III, a case based design aid, and an 'electronic cocktail napkin' program which tags a designer's conceptual sketch, in order to explore alternatives quickly.

For illustration purposes, a small sample of sketches produced by four experienced Swiss engineers is presented in Figure 3. The four resulting products for spanning a 300 m long and 70 m deep unsymetrical valley are very different. Expert 1) used constant-depth beams and focused on shorter (economical) spans, while adding aesthically pleasing diminishing spans up the long slope on the right of the valley. This complicates construction as launching is more practical when all spans are equal. Expert 2) indicates a preference for cantilevered construction and longer spans, and provides haunched beams which have a higher aesthetic rating. The light diagonals drawn indicate an evaluation of the spaces enclosed which he found satisfying. In a symmetrical environment, an even number of spans would not be recommended. Expert 3) is not influenced by the complexity of the foundations for such an arch, and exploited the strong effect of the long span and symmetry of the arch. Cost was not his first concern. Expert 4) is highly influenced by his area of expertise which is cable-stayed bridges. This environment, as other experts have noted, does not initially lend itself to this bridge type. His first reaction was also to provide a symmetrical bridge, as experts 2) and 3) did, but was dissuaded after a second glance. A non-symmetrical single-mast cable-stayed bridge pleased him, as this reduced the height of the mast and provided a dominant span.

In summary, two aspects of the processes used during design stand out. First, experts can distinguish many **levels of importance** and attribute different priorities and values to parameters, criteria and strategies in order to help them refine designs. Second, they are able to manage **change**, prevalent during sketching, by deciding when to iterate, when to use intelligent trial-and-error, and when to compromise. Although additional sketches and calculations performed on the side are not shown in Figure 3, they provided a strong indication that experts rarely followed a strict hierarchical approach and felt strained to describe a precise plan of their tasks (Gruber, 1991). They adapt as they design. Visser's (1991) detailed study of programmers confirms this informal observation. In those side sketches, one expert would dwell on a cross-section, thinking of a good transfer between pier and beam, one would dabble on a construction sequence to reduce doubt on its feasibility, or one would look at a mast more closely in order to make it more slender and discrete. An opportunistic approach (Hayes-Roth and Hayes-Roth, 1979) is a partial explanation. However, other factors have influence : each expert responds uniquely to incomplete and competing information in order to fill the gap between specifications and product description; they each have their own priorities regarding cost and aesthetics; each one is lightly or heavily influenced by previous designs; and at least one prefers generating many partial solutions rather than commit quickly to a more detailed solution.

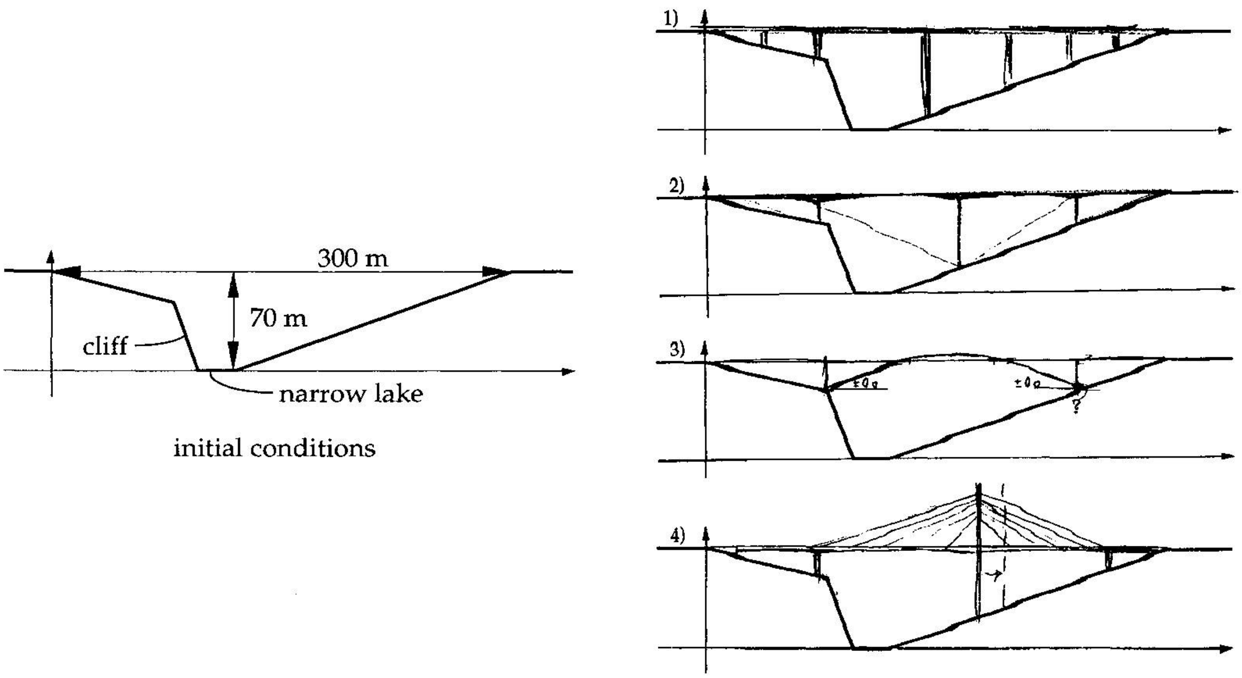


Figure 3 — Sketches from four experts based on same initial conditions

4 Adoption and adaptation of the maze model as a framework

An extended maze is presented in Figure 4 for meeting the design flow requirements observed during sketching sessions with experts. More specifically, the extended maze comprises three features which support the three activities mentioned in Figure 1. The primary task of assignment and refinement of parameters is modelled with *assumptions and physical principles;* the main maze (middle maze of Figure 4). It delimits and classifies the search environment. In general, assumptions act as magnetised zones of positive or negative intensities within the maze while physical principles indicate rigid barriers against penetration for a particular context. *Design criteria* represent factors such as social acceptance, viability, feasibility and economics of an artifact. These criteria influence exploration of the design space in order to reflect an order of importance and an order of use of knowledge. Different orders change the development of a solution set by allowing the designer to alter the emphases of the criteria on the sub-goals and final specifications. *Design strategies* deal with the problem-solving approaches of designers. Four modes enclosing eight strategies are used. They are described in more detail later. Essentially, a designer starts at a more abstract level and proceeds to the specifications. Designers employ these strategies to suit their style and switch from one strategy to another during the search of a suitable alternative. The three mazes are linked in a three-dimensional diagram to provide an interaction schema. The design criteria maze and design strategies maze behave as exploration guides for assisting the identification of design spaces, expressed in terms of parameters and their range of feasible values.

Constraints related to *assumptions and physical principles* are the founding labels of a design space. Assumptions are context-dependent and defeasible whereas physical principles must be satisfied in a final alternative. In structural design, assumptions are made continuously from the initial conception and during the iterative and refinement processes, since hypotheses are needed in the absence of complete and exact information. In fact, the ability to determine reasonable boundaries in situations of incomplete knowledge is one of the most valuable assets of experienced engineers ; an asset which distinguishes experts from novices who are accustomed to viewing design problems in closed worlds.

Design criteria indicate the many facets a product must satisfy before it is considered an acceptable alternative. It is always tempting to generalise domain-dependent features in knowledge-based systems in order to provide a "generic design procedure" applicable to most domains. Although this approach presents advantages, a completely domain-independent approach holds unrealistic expectations. An encompassing

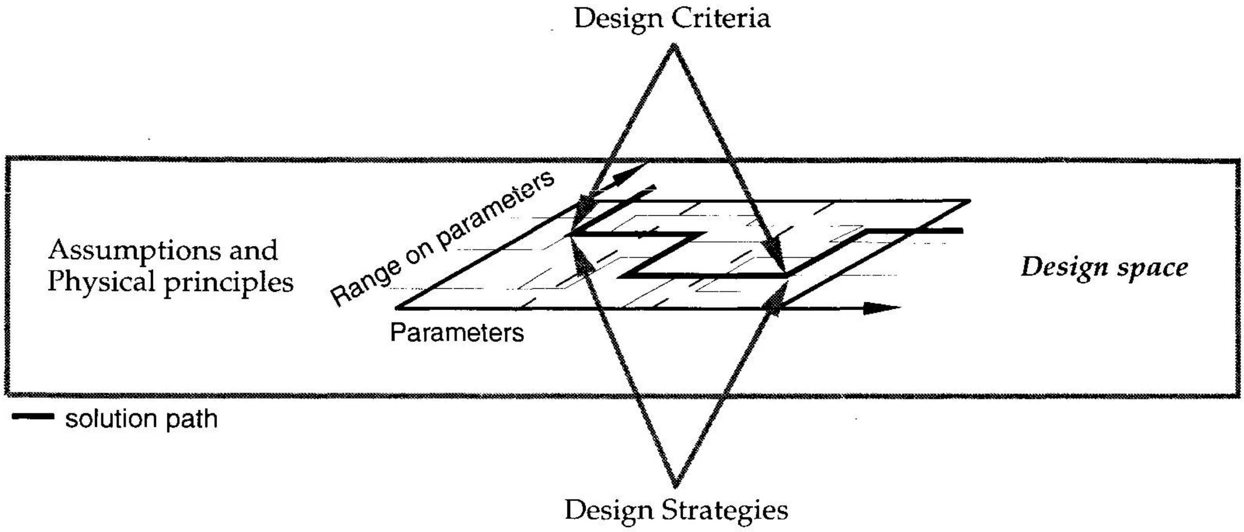


Figure 4 — Design criteria and strategies guiding search through the design space

procedure for designing electronic boards and urban bridges is not feasible, especially given their significantly different lifespans. An intermediate approach is to consider general design criteria labels which characterise a project, such as feasibility and then apply it locally to a domain, such as construction for bridge design, and manufacturing for board design.

Definition of general design crieria —
social acceptance : how will the artifact integrate into its environment ?
viability : how will it withstand its environment ?
feasibility : how will it be assembled and from which sub-assemblies ?
economics : how much will it cost initially and during its lifespan ?

General design criteria applied to bridge design —
I social acceptance : **aesthetics** (harmony, accessibility, integration)
II viability : **resistance** (statics, strength, stability, fatigue, serviceability, durability)
III feasibility : **construction** (fabrication, transportation, erection)
IV economics : **cost** (material, labour, maintenance)

Designers rarely consider one criterion only but attribute different importance to each one, which may differ at each main phase of design. For example, the alternating importance might affect whether they simplify construction by compromising on aesthetics, or whether they provide minimum resistance in order to save on short-term cost.

A model of multiple *design strategies* makes explicit knowledge assimilation processes and decision-making skills employed during synthesis. The motivation for the development of multiple design modes stems from informal observations of a dozen experienced (expert) engineers over the course of five years. Designers' activities are described according to four modes. Within each mode, engineers can apply two strategies. Examples of each strategy are provided below.

Modes and strategies employed by designers —
I paradigms : **derivers and retrievers**
II granularities : **generalisers and detailers**
III medias : **visualisers and verbalisers**
IV metaphors : **lateral thinkers and extrapolaters**

Designers are rarely in one mode exclusively. However, one mode more than another dominates for a particular sub-space of the solution. Also, these modes are not entirely independent and they are grouped according to types of strategies. Altering dominant modes affects the elaboration and commitment to different solution paths thereby creating a complex web of possibilities which can be captured in a maze environment.

6 Comparison of the model with the design of an existing bridge

The simulation example is based on a constructed bridge in Germany: the Kochertal viaduct in Geislingen (Figure 5). The topology consists of a 1128 metre long gap, with a maximum depth of 185 metres. The slopes are relatively gentle and the surroundings are a peaceful blend of farm land and scattered forests. During the bidding process, many solutions were proposed. Three are schematised in Figure 5.

The basis for deciding on number-of-spans and span-distribution is rarely recorded . However, such considerations have an important influence on the remainder of the design, including details. An Italian bridge design system reduces this aspect by a harsher categorisation of span distribution (Cauvin, 1992) because of the complexity represented by geometrical interpretation of the surroundings and the subjectivity of the knowledge associated to this phase. A Japanese bridge system (Nishido, Maeda and Nomura, 1990) also simplifies this problem by limiting their system to simple river-crossing bridges although they still dedicate more than half of their rule-base to "geometry".

This example is limited to initial decisions regarding number-of-spans, span-distribution, beam and pier types and cross-sections, and erection-methods (Figure 5). Initially, coordinates for valley-profile and bridge-alignment are specified. There is then an enrichment of this environment with terms such as V-shaped profile, symmetrical-distribution and viaduct-use. Since there is a small river, a large and deep valley, and a gentle slope, a beam bridge is selected. Aspect-ratios, design-ratios and static-limits help choose satisficing systems. Member dimensions are attributed opportunistically during design. Three alternatives are obtained with the same knowledge by varying the emphasis on design criteria, and hence their order of importance and order of introduction during a session.

Alternative 1) :	1) cost; 2) constructability; 3) statics; 4) aesthetics
Alternative 2) :	1) aesthetics; 2) cost; 3) statics; 4) constructability
Alternative 3) :	1) aesthetics; 2) constructability; 3) statics; 4) cost

Global evaluations of each alternative are summarised for aesthetics and cost criteria. The heaviness of alternative 1) and the high cost of alternative 3) tilted the decision towards alternative 2). The latter alternative is retained for further investigation. Figure 6 provides a design simulation of alternative 2), in Figure 5. It is illustrated by the dark line (the solution path) in the maze and thirteen numbered nodes. Each node represents a decision which is guided by design criteria and/or design strategies when assumptions and physical principles do not constrain the design space sufficiently. Decisions involve refining parameter ranges, selecting other parameters or managing conflicts. It is an explicit application of the design space maze in Figure 4 with implicit reference made to the criteria and strategies. The navigation between parameters, criteria and strategies show deviations from a hierarchical plan.

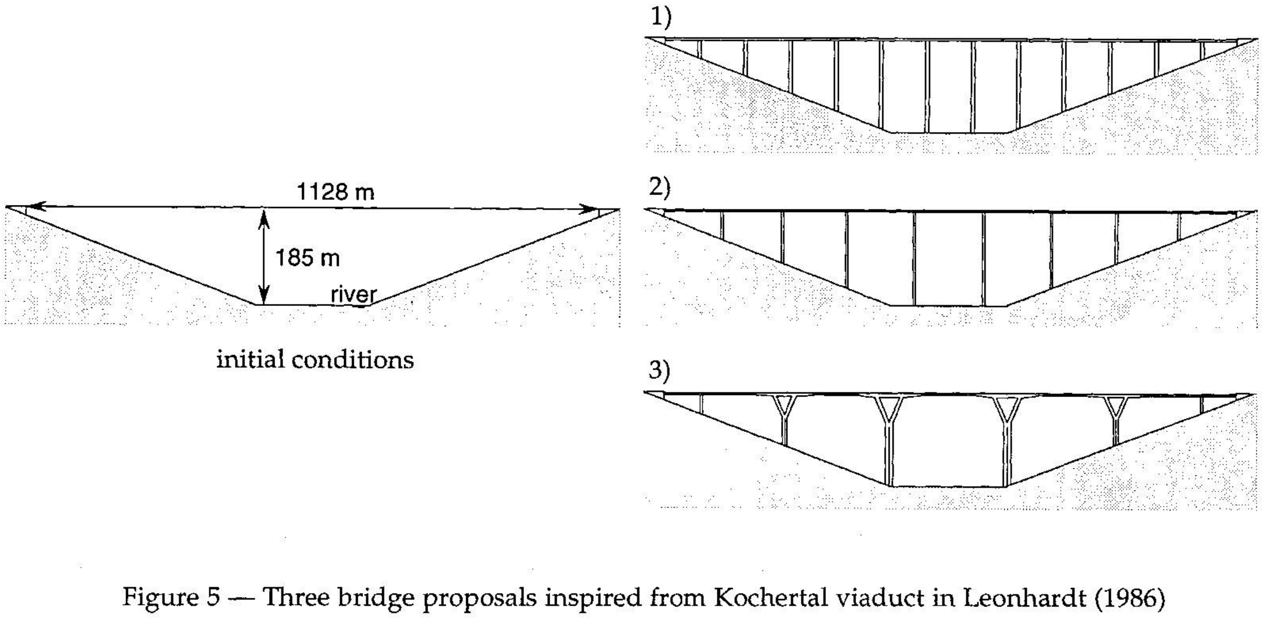


Figure 5 — Three bridge proposals inspired from Kochertal viaduct in Leonhardt (1986)

The entry into the maze (**node 0**) begins with 6 spans (bridge-length/valley-depth). Other possible entry points contain initial attributes such as equal-spacing, constant-depth I-beam, rectangular wall piers and erection by-crane. An aesthetic rule for that valley-type, suggests an odd number-of-spans (**node 1**) for avoiding placement of a pier in the middle of the valley. This rule is introduced early because of its ranking importance. It expands exploration with the generaliser mode and is based on the deriver mode i.e. it is not directly inspired by specific projects. Number-of-spans 6 is overridden by a set of odd numbers from 1 to 19. This set is reduced to {7, 9, 11}(**node 2**) by physical principles and a design-ratio reflecting reasonable pier-to-span distribution. The design continues with 7 spans (**node 3**) and retains equal-spacing and constant-depth I-beam (**node 4**) for cost reasons although a local evaluation based on the retrieval of other projects with equal-spacing reveals low-aesthetic quality. Since a static-consideration indicates that a span-limit is exceeded, the I-beam is replaced by a box-girder. After applying equal-spacing in more detail (**node 5**), the importance of aesthetics, previously evaluated as low, surfaces and causes a switch to graduated-spacing. A graduated-spacing, or a gradual decrease in span values, is especially recommended by experts in the presence of long, gentle slopes. This means that the central (main) span increases and the end spans decrease. A more comfortable number-of-spans (**node 6**) to satisfy the requirements of the longer span is 9. Graduated-spacing is then reapplied in more detail (**node 7**), as well, a plain girder is replaced by a reinforced girder to increase slenderness and cantilever resistance. Piers change from a wall-type (**node 8**), after retrieving examples of other projects, to a column-type to increase compatibility with beam slenderness. A rectangular cross-section is initially accepted (**node 9**), the erection-method is overridden from crane to launching (**node 10**) given the depth of the valley. Additional aesthetic refinement propose inclined, slightly-curved piers (**node 11**). The design session terminates (**node 12**).

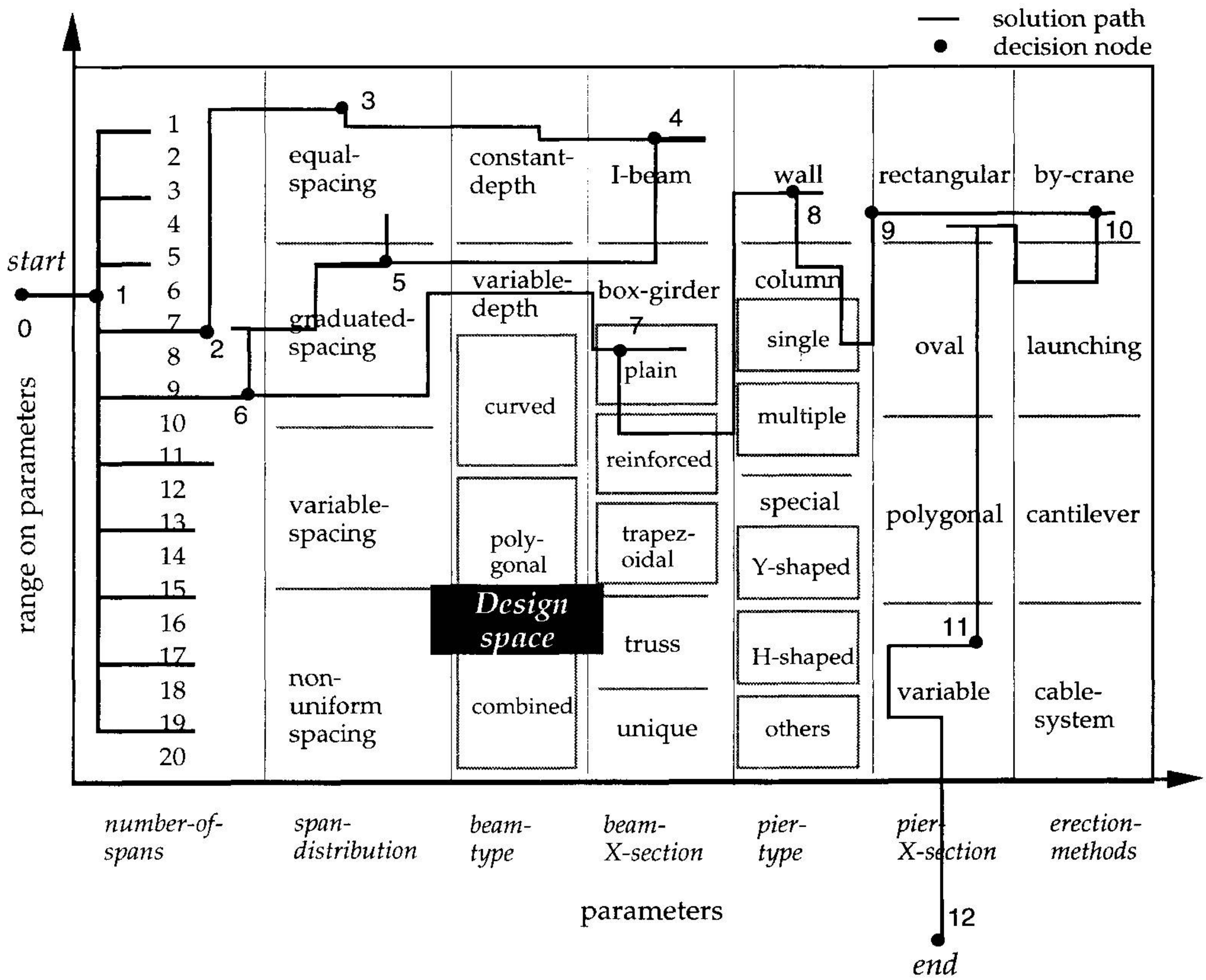


Figure 6 — Possible design path for alternative 2) in Figure 5, using schema of Figure 4

5 Overview of implementation in PRELIM

Some of the concepts described in earlier sections are implemented. The maze model provides a framework for making control knowledge explicit, thereby assisting system developers during implementation.

For processing knowledge, PRELIM incorporates a forward chaining rule engine for activating constraints, a justification-based truth maintenance system, a constraint processing framework for checking consistency and a conflict resolution module. The system can treat symbolic knowledge as well as continuous variables. These are represented as intervals indicating a range of feasible solutions. During constraint propagation each new interval inferred is justified by the justification-based truth maintenance system label expliciting links between design variables and constraints.

Conflict resolution is handled non-monotonically by overriding a default, weakening a preference or backtracking. Weakening preferences is a type of partial backtracking allowing retraction and reinstatement of previous decisions. Other domain-dependent and variable-oriented information contribute secondary help for conflict resolution. More details are provided by Haroud, Boulanger and Smith (1994). The algorithms treating the process explicitly in PRELIM are written in LISP, on Sunworkstations.

For representing knowledge, PRELIM uses assumptions (defaults and preferences) and physical principles (rigid rules), design criteria and design strategies to label rules. Objects represent physical, conceptual and relational properties. Internally knowledge about the artifact is represented in a flat structure as a constraint network enabling consistency checks. ICAD, an intelligentCAD system, provides the user-interface with graphical representation and a product model written in IDL, the object-oriented language of ICAD.

Assumptions have labels which are used to guide instanciation and assign values. They are further divided into defaults and preferences. Defaults can be included in objects as an attribute or in rules. Defaults impact on the initial stages of design, i.e. the entrance into the maze, and are highly defeasible. A context which consists of defaults is most probably inconsistent as each parameter evaluated via a default is set according to different contexts. As the design space progresses,these inconsistencies may dissipate without a formal mechanism. Preferences represent "expert" knowledge and are directly recognizable heuristics. They are the most difficult to manipulate, particularly in conflicting situations. The labels on preferences and the activity of weakening together supply a form of redundancy as unexplored paths are maintained in the objects and retrieved when a temporary impasse during the exploration occurs. A thorough treatment of explicit representations of assumptions in PRELIM can be found in Smith and Boulanger (1994).

Limits of the current implementation

Navigation within the maze model is partially influenced by the order in which rules are introduced in the system. In our implementation the user can fix rule order by manipulating design criteria before the session is started. Dynamic reordering during the session is not possible. Different strategies reflecting the user's way of tackling conception are not yet implemented. Although the system behaves non-monotonically, controlling weakening and backtracking interactively with the user is not yet stable. For constraint satisfaction, we are currently developing methods to improve reliability and ensure consistency of constraint sets that include both equalities and inequalities.

7 Conclusions

Although several researchers currently propose a network model for design, a maze description provides additional modelling potential; for *representing key expert behaviour* including exploration of several design paths from several starting points, for *specifying implementation requirements* such as non-monotonicity, decision networks and temporarily inconsistent contexts, and for *developing user interfaces* by distinguishing between different types of information. Observations from sketching assisted the selection of a maze model. The basic maze structure is extended for representing design criteria and design strategies. The maze is the design space containing information related to parameters and ranges on parameters. Two additional elements, formulated using design criteria and design strategies, provide control methods for directing search and for conflict management. A system for preliminary design of bridges uses these descriptions as a conceptual framework for design process support.

8 Acknowledgement

The authors wish to thank Manfred A. Hirt and Jean-Paul Lebet of ICOM (Steel Structures) for discussions related to civil engineering as well as Boi Faltings and Djamila Haroud of LIA (Artificial Intelligence) for participation in the development of this research. This research has been funded by the Swiss National Science Foundation.

9 References

Bañares-Alcántara, R. (1992). Representing the engineering design process : two hypotheses, *Proc. of Artificial Intelligence in Design '92*, Kluwer Academic Publishers, Netherlands.

Bédard, C. and Ravi, M. (1991). Knowledge-based approach to overall configuration of multistory office buildings. *Journal of Computing in Civil Engineering*, Vol. 5, No. 4.

Bowen, J. and Bahler, D. (1992). Supporting multiple perspectives, *Proc. of Artificial Intelligence in Design '92*, Kluwer Academic Publishers, Netherlands.

Brazier, F.M.T., van Langen, P.H.G., Ruttkay, Zs., and Treur, J. (1994). On Formal Specification of Design Tasks. *Proc. of Artificial Intelligence in Design '94*, Kluwer Academic Publishers, Netherlands.

Cauvin, A., Stagnitto, D. and Stagnitto, G. (1992). Expert Systems in design of Structures : An application to Bridges. *Progress in Structural Engineering*, Kluwer Academic Publishers, Netherlands.

Fenves, S.J. (1992). Status, Needs and Opportunities in Computer Assisted Design and Analysis, *Structural Engineering International*, 2/91, International Association For Bridge and Structural Engineering.

Ganeshan, R., Finger, S. and Garrett, J. (1991). Representing and reasoning with design intent. *Proc. of Artificial Intelligence in Design '91*, Kluwer Academic Publishers, Netherlands.

Gardiner, M. (1987). Principles from the psychology of memory; Part II: Episodic and semantic memory, *Applying Cognitive Pwsychology to User-interface design*, John Wiley & Sons, Chichester.

Garcia, A.C.Bicharra and Howard, H.C. (1992). Acquiring design knowledge through design decision justification. *AI EDAM*, 6(1), 59-71.

Gross, M., Zimring, C., Do, E. (1994). Using diagrams to access a case base of architectural designs, *Proc. of Artificial Intelligence in Design '94*, Kluwer Academic Publishers, Netherlands.

Gruber, T.R. (1991). *Interactive Acquisition of Justificaations: Learning "Why" by being told "What"*, Knowledge Systems Laboratory Technical Report KSL 91-17.

Haroud, D., Boulanger, S., Gelle, E., Smith, I. (1994). Strategies for Conflict Management in Preliminary Engineering Design. *Wkshp Proc.. of Artificial Intelligence in Design '94*, Kluwer Academic Publishers, Netherlands.

Hayes-Roth, B. and Hayes-Roth, F. (1979). A Cognitive Model of Planning, *Cognitive Science*, 3, 275-310, 1979.

Holgate, A. (1986). *The art in structural design*, Clarendon Press, Oxford.

Jenkins, D.L. and Martin, R.R. (1993). The importance of free-hand sketching in conceptual design: automatic sketch input, *Proc. of the 5th International conference on design theory and methodology*, ASME, New York.

Leonhardt, F. (1986). *L'esthétique des ponts*, Presses Polytechniques Romandes, Lausanne.

Logan, B., Corne, D.W., Smithers, T. (1992). Enduring support, *Proc. of Artificial Intelligence in Design '92*, Kluwer Academic Publishers, Netherlands.

Maher, M.L. (1989). Structural Design by Hierarchical Decomposition. *Computing in Civil Engineering*, American Society of Civil Engineers, New York.

Mittal, S. and Aray, A. (1990). A knowledge-based framework for design. *Proc. of the AAAI90*, American Association of Artificial Intelligence.

Newell, A. and Simon, H.A. (1972). *Human Problem-Solving*, Prentice-Hall Inc., Englewood Cliffs, New Jersey, USA,

Nishido, T., Maeda, K. and Nomura, K. (1990). Study on practical expert system for selecting the types of river-crossing bridges", Journal of Structural Engineering/Earthquake Engineering, `Japan Society of Civil Engineers, Vol. 7, No. 2, 239s-250s.

Petrie, C.J., Cutowsky, M., Park, H. (1994). Design space navigation as a collaborative aid. *Proc. of Artificial Intelligence in Design '94*, Kluwer Academic Publishers, Netherlands.

Smith, I. and Boulanger, S. (1994). Knowledge representation for preliminary stages of engineering tasks, *Knowledge-Based Systems*, Butterworth-Heinemann Ltd, Vol. 7, No. 3, Sept. 1994.

Simon, H.A. (1981). *The Sciences of the Artificial*. Second Edition. The MIT Press, Cambridge.

Soo, V.-W. and Wang, T.-C. (1992). Integration of qualitative and quantitative reasoning in iterative parametric mechnical design. *AI EDAM*, 6(2), 95-109.

Stefik, M. (1981). Planning with Constraints (MOLGEN : Part 1), *Artificial Intelligence* 16, 111-140.

Tong, X. and Tueni, M. (1990). CARMEN : A platform for building 2nd generation expert systems. *Proc. of the Tenth InternationalWorkshop on Expert systems and their applications*. Avignon '90, EC2, Nanterre, France.

Topping, B.H.V. and Kumar, B. (1989). Knowledge representation and processing for structural engineering design codes. *Engineering Applicaton of AI*, Vol. 2, September, Pineridge Ltd.

Visser, W. (1991). The cognitive psychology viewpoint on design : examples from empirical studies. *Artificial intelligence in design '91*, Butterworth Heinemann, Oxford, Great-Britain.

Zhao, F. and Maher, M.L. (1992). Using Network-based prototypes to support creative design by analogy and mutation. *Artificial Intelligence in Design '92*, Kluwer Academic Publishers, Netherlands.

Integrated Innovative Computer Systems for Conceptual Bridge Design

Systèmes innovants et intégrés pour la conception des ponts

Integrierte innovative Computersysteme für die Konzipierung von Brücken

John C. MILES
Dr Eng.
University of Wales
Cardiff, UK

John Miles, born 1950, graduated from Manchester University in Civil Engineering and then spent some time in industry before receiving a PhD on numerical simulations of ground water flow at Birmingham University. Since the middle 1980s he has been the joint leader of the Cardiff Decision Support Systems Group.

SUMMARY

The development of a suite of innovative, complementary systems for bridge design is described. The suite consists of four systems which cover conceptual design, a case-based database of existing designs, decision support for costing and aesthetics. The systems are linked through a dynamic database and all adopt a user-centred approach. The four systems are briefly described and their utility in practical bridge design is discussed. Also, the proposed method of interaction is discussed. The system's development is based on the authors' experience of the needs of practising designers.

RÉSUMÉ

Le développement d'une suite de systèmes innovants et complémentaires en matière de conception de ponts est décrit dans cet article. Cette suite est constituée de quatre systèmes qui couvrent: la conception, une banque de données de réalisations existantes, l'aide à la décision en matière d'estimation des coûts et l'esthétique. Les systèmes sont reliés entre eux grâce à une banque de données "dynamique" et chaque système a une démarche centrée sur l'utilisateur. Les quatre systèmes sont présentés brièvement et leur utilité pratique dans la conception de ponts est discutée. Le développement du système est basé sur l'expérience des auteurs et les besoins des praticiens.

ZUSAMMENFASSUNG

Die Entwicklung einer Gruppe innovativer, sich ergänzender Systeme für die Brückenkonstruktion wird beschrieben. Die Gruppe umfasst folgende vier Systeme: Konzeption, eine Datenbank bestehender Konstruktionen, sowie Entscheidungshilfen für Kostenberechnung und Ästhetik. Die Systeme sind durch eine dynamische Datenbank verknüpft und von benutzerfreundlichem Design. Die vier Systeme, ihr Nutzen in der Praxis und die angestrebte Methode des Zusammenwirkens untereinander und mit dem Benutzer, werden diskutiert. Die Entwicklung aller vier Systeme basiert auf den Erfahrungen des Autors mit praktizierenden Brückenkonstrukteuren.

1. INTRODUCTION

Many concepts and ideas have resulted from Artificial Intelligence research. How best to use these ideas to assist designers has grown into a major area of research. Initial efforts to achieve these aims were inflexible and fragile but more recent work is starting to overcome these problems; for example, the generic spatial reasoning system of Coyne and Subrahamin [2]. The authors have concentrated on the development of innovative systems for conceptual design which are of more immediate benefit, ([17],[11],[5],[8]). The research has progressed from the development of relatively large expert systems which covered the entire domain, to more flexible systems which aim to cover smaller, more focused sub-domains. Although the aims of these latter systems are more pragmatic and therefore easier to achieve, they could be criticised for being too limited in their coverage. To overcome this, the systems currently being developed are highly interactive, both with other complementary systems and with the user. The merits of this approach form a major part of the discussion in this paper.

The provision of sophisticated decision support software for designers is generally accepted as a desirable goal. A number of large scale projects whose aim is to create comprehensive design environments which incorporate CAD, various KBS, databases and analysis capabilities are either in progress or have been attempted. These systems offer the advantage of compatibility. However, to date, the success rate of these projects has been disappointing. Generally throughout the software industry, it is recognised that large systems are difficult to develop, demanding a disproportionately high number of man hours compared with the development of smaller systems [10].

The alternative to large complex projects is to develop separate, readiliy compatible systems which can be successfully linked. This leads immediately to the concept of linking technologies such as product models [18] which facilitate the transfer of information between different systems. However, the development of product models is still in its infancy and furthermore their development is a relatively involved process. These difficulties have hampered the acceptance of product models by the construction industry. Hence, funding is difficult to obtain and it is likely that progress will be slow. In the absence of such sophisticated methods of linking, it is pragmatic to develop methods which exploit the available technology and can therefore be more immediately employed. One such method is described here.

This paper briefly describes four complementary sub-domain design systems and the mode of interaction chosen by the developers. The application domain is the conceptual design of bridges. The subsequent discussion describes the underlying philosophy of the work, the linkages between the systems themselves and also between the system and the user.

2. THE SYSTEMS

Four systems associated with different aspects of bridge design are currently being developed. These are described below. The first three systems are being developed in Microsoft Visual C++. A Case Based Reasoning system is also being developed, currently using Remind but it is anticipated that the final system will also be written in C++. C++ was chosen instead of a conventional AI language due to the greater power and flexibility available, albeit at the expense of increased programming effort for certain parts of the work. In particular, the version of C++ used offers considerable control over the user interface which past work [15] has shown to be important. In the original system of Moore [11], because of the limitations of the software used, it was not possible to incorporate graphics. Instead the user interface was text based. Tests of this software showed that designers prefer to reason about designs in a more pictorial way, presumably because such information can be more readily handled in short term memory. Thus, wherever possible, the systems provide information in a graphical format.

The domain of the work is small to medium span road bridges, as this is where the experience of the research group lies [11],[6]. In addition, as these are currently the most common form of bridge built in Britain there are large volumes of accessible data.

As with all of the authors' work, these systems are all being developed by collaborating closely with practising designers who are used to evaluate the work. This helps to ensure relevance as well as providing and ideas and impetus for further research.

2.1 System One: A Non Prescriptive Conceptual Design System

The original conceptual bridge design system [11] underwent extensive industrial evaluation. This revealed that the users did not feel in control of the design process because the original system was too prescriptive. Like many other KBS, the system controlled the decision making process . Also this form of reasoning resulted in a system which was inflexible, particularly when the user wished to incorporate non-standard, case specific information.

Despite these criticisms, the original system was, in principle, well received as it provided correct answers and useful advice. This was because, despite the style of user interaction being flawed, the knowledge base was sound. Based on an analysis of the reactions to this initial system, it was decided to replace some of the heuristics incorporated with more sophisticated forms of reasoning (as described in Systems Two and Three below) and for the initial conceptual design a far more flexible, user driven, knowledge-based system has been developed. This system solely undertakes conceptual design and does not venture into preliminary costing or member sizing as was the intention of the original system. These areas are now catered for by separate systems.

The new system has been developed from the initial knowledge base, although this has been rewritten in an object oriented format. In addition the entire structure of the knowledge base has been altered so that the "rules" are clustered in small groups with no more than 10 rules per group. Each group is associated with a daemon which only fires when the user violates certain constraints. Constraint violation can occur for a number of reasons, for example when the user chooses an uneconomic structural form or when there is a locational clash say between a water main and a foundation. Thus rather than controlling the design process, the knowledge base observes and interacts only when necessary via a message on the screen. On receiving a message, the user is left to decide what action (if any) to take. The initiative is left with the user because given the impossibility providing of knowledge bases to cope with every situation, one has to allow human judgement and common sense to be included in the design procedure. This obviously permits the user to make mistakes but is a vast improvement on current procedures where no checks are made. This style of user interaction we call non-prescriptive because the system does not prescribe an answer; it only suggests alternatives [7]. By leaving the user in control of the design, the benefits of computers are maximised (i.e. memory, computational power and reliability) without stifling the capabilities of human beings (creativity, flexibility and innovation) [8].

A further facility allows the user to access and amend the knowledge base through a purpose built knowledge manager. This is possible because the knowledge base has been fragmented into short and separate constraint trees, which is in turn possible because of the non-prescriptive nature of the interface. This facilitates access by expert bridge designers who are not familiar with the system, hence allowing design consultancies to modify the knowledge base to suit their own practices. Further work is in progress on allowing users to add to (rather than to amend) the knowledge base and progress to date is encouraging.

At present the system is undergoing its first design office trials. Initial findings show that the designers see a distinct role for the system, providing a quality assured design procedure. They particularly like the knowledge manager and those people who had used the previous system [11] are appreciative of the improved style of interaction.

2.2 System Two: A Preliminary Costing System for Bridges

Following the evaluation of the first bridge design system, it became apparent that engineering consultants found the costing of alternative bridge designs a major problem. Currently, bridge designers use very simple heuristics (typically a cost per m^2 of deck). Obviously such a method is very crude. When a more detailed search of the design space is required then the usual procedure is to design a limited number of bridge options in some detail, and take off quantities in order to cost them. This can take several man weeks, the amount of work involved effectively prohibiting a proper search of the design space. To overcome these problems, a design costing system has been developed. This provides an cost estimate which is far more accurate than that reached by using current heuristics by sizing the components of the bridge to a level of accuracy which is close to that achieved by using a full analysis. The system is then used to take off appropriate quantities to obtain a preliminary cost.

By combining improved estimating techniques, approximate contingency factors and heuristic replacement [8] with the power and speed of computers, an effective system which can rapidly and accurately cost a bridge has been produced. The system enables bridge costings to be produced in few minutes, which in turn provides the designer with a tool which can rapidly cost alternatives and assess the impact of small changes, thus enabling the design to be 'fine tuned'. This means that the designer is able to search the design space for an optimum solution. The principle of heuristic substitution and the developed system are described elsewhere [7] [14].

2.3 System Three: An Advisory System for Bridge Aesthetics

Another decision variable in bridge design is aesthetics. This is an area of engineering design which is highly subjective and therefore difficult to investigate. However, the need to elicit information and provide assistance to designers in this area was identified during the evaluation of Moore [11]. The knowledge base for this project is being developed with the help of a number of expert bridge designers and architects. The opinion of the general public is also being included, via sophisticated questionnaires. Some work in this area has already been undertaken, notably by Crouch [3] and the work of the authors has extended this.

The system does not aim to provide a definitive set of rules which must be adhered to for all road bridges. Instead, it provides advice and assistance with the benefits of visualisation. As with the costing system, this system enables the designer rapidly to evaluate options. It is important that the use of such a system should not lead to the standardisation of road bridge design, therefore it only provides suggestions for improving the aesthetics of a bridge and, as with the other systems in the suite, control of the design stays ultimately with the designer.

Initial evaluation has again showed that the system is liked by the reviewers and that they see a worthwhile role for the system.

2.4 System Four: A Case Based Reasoning System for Bridge Design

Case Based Reasoning (CBR) is an important new technology which allows the more effective use of databases; initiating problem solving techniques which are based on the utilisation of past, recognised solutions [16]. Engineering design is a complex task. However, previous research has shown that much of the design conducted on a day to day basis consists of modifying past designs; as this approach is economic. In the construction industry, there are large collections of design data which are traditionally stored on paper or more recently in computer aided design packages. When considering a new scheme a designer will typically want to locate previous designs which can be used as the basis for a new design. Currently, this process is done manually. Many design offices recognise the need for more efficient search and retrieval techniques and CBR presents a possible solution to these problems.

The project does not aim to compile detailed design information on all types of bridges. Nor does it aim to develop a design standard which removes the creative side of design. Inevitably, bridge designs are complex and to aim to store all information about them would be unrealistic. Therefore, a sub section of bridge designs are considered (small to medium span road bridges as mentioned above). In addition, the CBR system aims to capture conceptual design decision information as well as specific design criteria.

The project is still in its early stages, but industrial collaborators have already given a number of suggestions and bridge designs, which have provided the basis for the formulation of a prototype system which incorporates a preliminary case breakdown [13].

3. THE INTERACTION MODEL

The four systems outlined above are being developed as independent systems. However consideration has been given to their interaction to form a complementary set of design systems. The mode interaction of the systems is shown in Fig.1. Also shown is the proposed interaction with AUTOCAD. Further work has taken place to investigate linkages to packages such as MOSS which could be used to input directly topographical data. The initial findings are that information can easily be passed using file protocols such as DXF but obviously this involves a substantial loss of information. It is also recognised that Fig.1 does not show any links to analysis software. However previous work [17] has shown the feasibility of such links and no problems are anticipated in providing them.

It can be seen that the user can enter or exit the system at any point. For example, the user can enter the system at the conceptual design stage, create a design, cost it, obtain advice on its aesthetics and leave the system with the completed conceptual design, with the option of storing the new design in the CBR system. Alternatively, the user could retrieve a design from the CBR system and check its cost, receive advice on its aesthetics or both. These are only two examples of many possible modes of consultations. The proposed connections (shown by the arrows) have been deliberately limited, (that is, they are one directional in places), so that the prototype architecture is realistic in its aims.

This suite of systems will be supported by a dynamic 'database' of the information accumulated during a consultation (Fig.1). This facilitates the transfer of information between the interacting systems as well as providing an easy to access record of previous user input, enabling them to 'tweak' the input data to see the affect of changes in the design criteria. From this it can be seen that the systems will operate on an equivalent basis, with the user maintaining control over the entire consultation. The architecture adopted is reminiscent of a blackboard principle [1]. It also exhibits some properties of agents [4]. However, it differs markedly in that there is no central controller or system 'manager' as is the common case with these alternative architectures. Instead, the user takes control of the consultation and acts as the suite manager. The aim of the arrangement is to provide decision support for the user, with the user controlling the interaction and the design process. This maintains the sense of user control, which previous research has shown to be important [8].

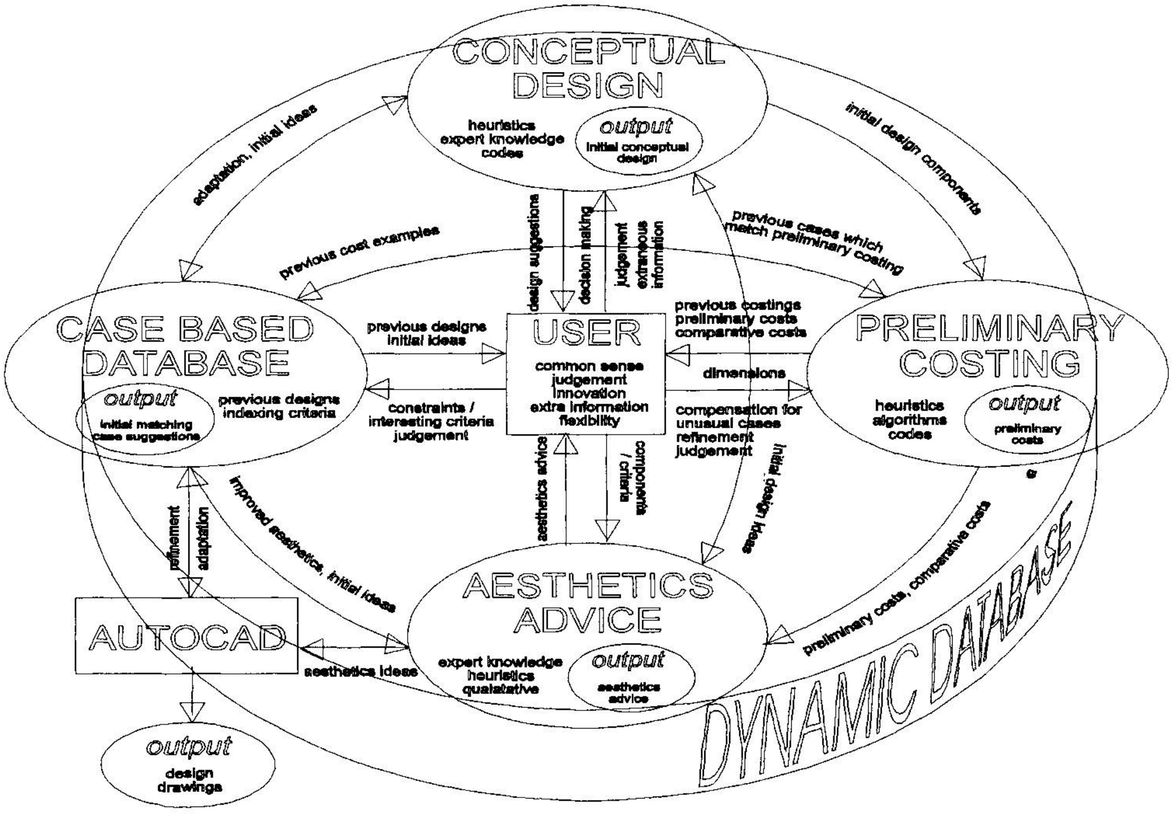


Figure 1

The mode of ineraction adopted allows the user to supplement the areas in which the computer systems currently deficient, such as common sense, judgement, innovation and flexibility: all qualities which computers currently find hard to emulate and which are recognised as being important in design . As discussed above, the philosophy of the research at Cardiff is to support designers in areas which they find difficult, leaving them to cover tasks which come naturally and which are difficult to incorporate in computer programs. Using the user/designer as a system component enforces this philosophy and ensures that the systems operate successfully whilst maintaining their support role. This user centred approach is felt to be vital for the future success of design systems. The authors believe that there is the temptation in AI to "over-automate".

There is a potential disadvanatge to the above approach because the user exhibits such human failings as inconsistency but the supporting systems have been designed to help to mitigate such problems and it is believed that the gains far outweigh the losses.

4. TRANSFER OF INFORMATION BETWEEN THE SYSTEMS

The manner in which information is transferred between the various systems in Fig. 1 requires careful thought to ensure flexibility whilst avoiding excessive complexity. Given the inadequate development of suitable advanced coupling technologies such as product models, it was necessary to devise a schema which satisfies current needs and allows for future expansion. The importance of the inter-system linkages and the user interaction is such that a great deal of effort has been expended in planning and forethought. It is not possible to present all this work here but an outline of the approach can be given.

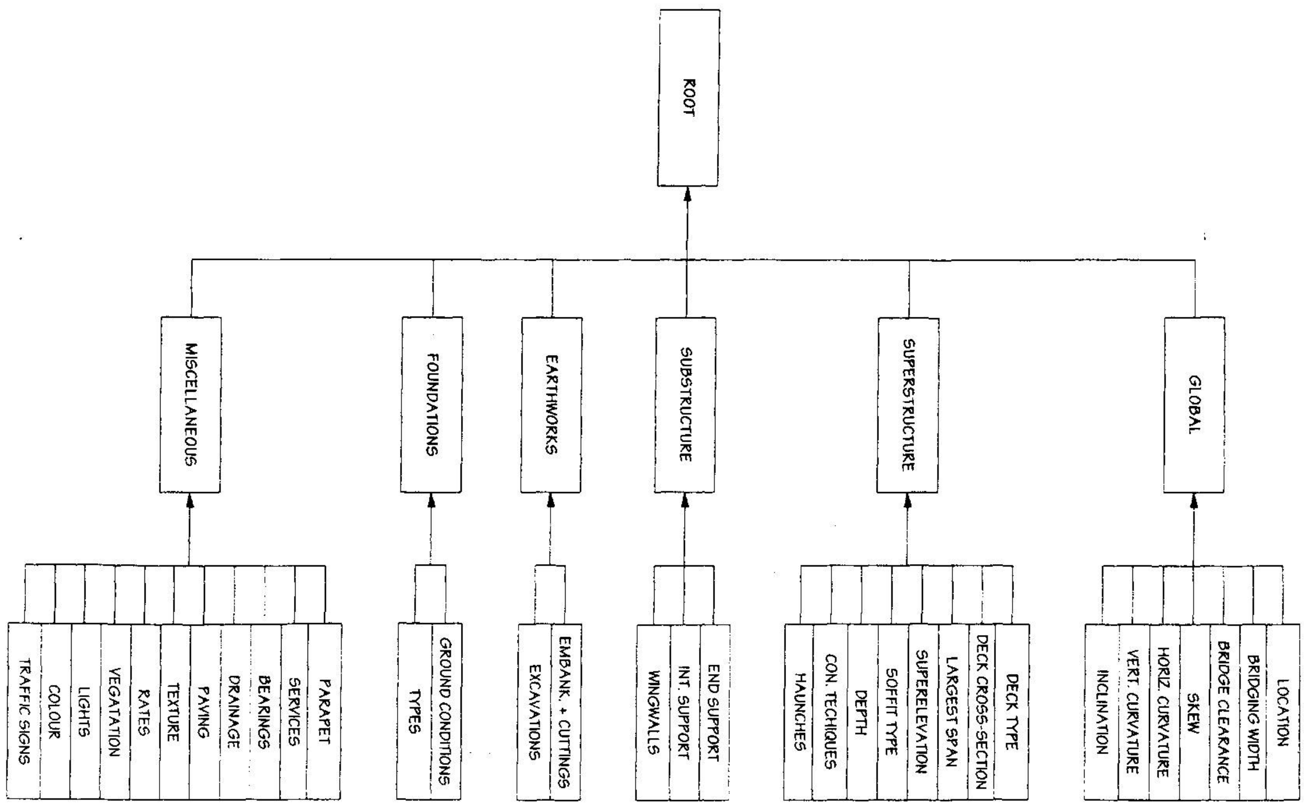


Figure 2

CATEGORY	INPUT	SYSTEMS	TYPE OF INPUT
Topological	Location	1,(2),3,(4)	*descriptive*
	Bridging Width	1,3,(4)	*descriptive*
	Clearance	1,3,(4)	*numerical*
	Skew	1,2,3,(4)	*angle*
	Inclination	(2),3,4	*angle*
	Horiz. Curvature	1,2,3,4	*radius*
	Vert. Curvature	(2),3	*radius*
Super Structure	Deck Type	(1), 2,(4)	*descriptive*
	Largest Span	(1),2,3,(4)	*numerical*
	Deck Depth	1,2,3	*numerical*
	Width	(1),2,3,4	*1,4 - descriptive* *2,3 - numerical*
	Constr. Technique	1,2,(3),4	*descriptive*
	Deck X Section	2,(3)	*descriptive/numerical*
	Soffit Type	1,(3)	*descriptive*
	Superelevation	(2),(3),4	*angle*
Sub Structure	End Support Type	(1*),2,3,4	*1,3,4 - descriptive* *2- descriptive/ numerical*
	Wing Wall Type	(1*),2,(3),4	*1 - angle* *2 - descriptive/ angle* *3,4 -descriptive/angle*
	Int. Support	1,2,3,4	*1,4 - descriptive* *2,3-descriptive/numerical*
Earthworks	Embank / cutting	1,(2),3,4	*1 - descriptive* *2,3,4-descriptive / numerical*
Foundations	Ground Conditions	1,2	*1 - descriptive* *2 - numerical*
	Type	(1*),2,4	*descriptive*
Miscellaneous	Bearings	1,2,4	*descriptive*
	Services	(1),(2),4	*descriptive*

Table 1: Summary of Input Study

The information which needs to be transferred between the systems has been studied using a variety of techniques to show the types of information that are involved and the possible modes of interaction between systems. Firstly, the input and output of each system was listed. The outcome of the study of the input is summarised in Table 1. Numbers 1,2,3,4 represent the systems as described in the previous sections. Only those data used by more than one system are shown. The numbers in brackets indicate that the input is optional as opposed to essential (again enhancing the flexibility of the systems). The numbers showing an asterisk (*) indicate that the user can choose whether to input his/her own criteria or to let the system make the choice.

Many of the items in the input column are single facts or datum but others are more complex. For example deck type can describe the material(s), the form of construction and the shape of the cross section.

Table 1 is also interesting as it shows the overall data requirements for the domain. The data which are common between the systems are, as one would expect, topographical and basic dimensions. There are some variations in format between the different systems but these are fairly minor and should be easy to cope with. In addition to table 1, a series of Venn diagrams showing the overlap of information between various system combinations have been created. Space limitations preclude their inclusion but they have proved to be useful for planning modes of interaction and also they provide a new insight into the design domain structure. To further clarify the workings of the linked systems, a number of tree diagrams (or basic semantic networks) have been created to represent the hierarchy of the domains studied and their inter-relationship. An example of these is shown in Fig. 2.

5. FUTURE WORK

In the immediate future, work will concentrate on developing the individual systems and with an increasing emphasis on the dynamic database. The industrial evaluation of the systems has commenced and it is anticipated that this will produce some changes and new ideas which will be included in the development of all the systems. As well as evaluating the systems, opinions will be sought on the interaction between the systems. Depending on these findings, and in conjunction with the development of the component systems, the interaction architecture shown in Fig. 1 will be further developed.

As has already been mentioned, further additions to the above four systems are planned and work has already started on knowledge acquisition for a foundation design system. Further systems are also planned but as yet funding for these is not available. Further work on linkages to external software is also planned but as the research element in such work is minimal, it is not planned to go beyond feasibility studies.

6. CONCLUSIONS

A suite of systems for the design of bridges has been described. The systems all deal with sub-domains of the overall design problem which is essentially the conceptual stage of the design process. The interaction between the systems is being designed in such a way that the user effectively forms an integral part of the set up. This is felt to be of paramount importance, offering greater flexibility and maximising the strengths and abilities of the designer and the current computing technology.

A study of the data requirements of the various systems has been outlined. This is part of ongoing work into the formulation of a dynamic database which will be used to control and facilitate the interaction between the current systems and future systems. Linkages to other common design software have been investigated and it is not anticipated that these will present any significant problems although with current technology the types of data which can be exchanged are somewhat limited. However it is felt that despite these limitations, such linking has much to offer and to wait until more sophisticated methods are available would waste the benefits of current technology.

7. REFERENCES

1. CORBY, O., Blackboard architectures in computer aided design. Civ Eng Systems. Vol. 4. pp 14-19/1986.
2. COYNE, R.F. AND SUBRAHAMIN, Computer supported creative design: a pragmatic approach, In Gero, J.S. & Maher, M.L. (eds) , Modelling creativity and knowledge-based creative design, Lawrence Erlbaum, USA, 1993.

3. CROUCH, A.G.D., Bridge aesthetics - a sociological approach, Civ. Eng Trans, Inst. Engs. Aust., CE16(2), 138-142/1973.
4. HUANG, G.K. AND BRANDON, J.A., Agents: Object Oriented Prolog for Co-operating Knowledge Based Systems. Knowledge Based Systems 5, (2), 125-136/1992.
5. HOOPER, J.N., A Knowledge Based System for Catchment Planning. Ph.D. Thesis. University of Wales, 350pp, 1994.
6. MILES, J.C. AND MOORE, C.J., An expert system for the conceptual design of bridges. in Topping, B.H.V. (ed), AI Techniques and Applications for Civil and Structural Engineers, CIVIL_COMP Press, Edinburgh, UK, 171-176, 1989.
7. MILES, J.C. AND MOORE, C.J., Conceptual Design: Pushing Back the Boundaries with Knowledge Based Systems. In: Topping, B.H.V. (Ed.): AI and Structural Engineering. Civil-Comp Press/1991.
8. MILES, J.C. AND MOORE, C.J., Practical Knowledge Based Systems in Conceptual Design. Springer Verlag,250pp, 1994.
9. MILES, J.C., MOORE, C.J. AND EVANS, S.R., Deriving Rules for Bridge Aesthetics. Civil-Comp '93. Edinburgh. August 1993.
10. MONTGOMERY, A., Combining SE Discipline With AI Creativity, in Software Engineering and AI, IEE (London), Digest 1992/087, 1/1-1/5/1992.
11. MOORE, C.J., An Expert System for the Conceptual Design of Bridges. Ph.D. thesis. University of Wales. February 1991. 350pp/1991.
12. MOORE, C.J., EVANS, S.N. AND MILES, J.C., Establishing a Knowledge Base for Bridge Aesthetics. To be Published in Structural Engineering Review/1994.
13. MOORE, C.J., LEHANE, M.S. AND PRICE, C.J., Case Based Reasoning for Decision Support in Engineering Design. IEE Colloquium on Case Based Reasoning. London. March/1994
14. MOORE, C.J., MILES, J.C. AND PRICE, G.E., Computational Decision Support and Heuristic Substitution in Preliminary Bridge Costing, In Smith, I.A.C.,(ed), IABSE Colloquium, Bergamo, 1995.
15. PHILBEY, B.T, MILES, C. AND MILES, J.C., The development of an interface for an expert system used for conceptual bridge design. In:Topping, B.H.V(Ed.): AI and Struc Eng, Civ-Comp Press, 87-96/1991.
16. SLADE, Case Based Reasoning: A Research Paradigm. AI Magazine 1991. pp 42-55.
17. SOH, C.K. (1990): An Approach to Automate the Design of Offshore Structures. Ph.D. Thesis. University of Wales. 330pp/1991.
18. WATSON, A. AND BOYLE, A.,Product Models And Application Protocols, in Topping, B.H.V. (ed), Information Technology for Civil and Structural Engineers,121-129/1993.

An Integrated Tool for Designing Space Trusses

Système de conception pour le projet de structures tridimensionnelles

Ein integriertes Entwurfssystem für Raumfachwerke

Adam BORKOWSKI
Prof.
IFTR
Warsaw, Poland

Adam Borkowski, born 1942, received his civil engineering degree at the Technological University of Kaunas, Lithuania. From 1966-1986 he conducted research in structural mechanics. At present he leads the Laboratory of Adaptive Systems at the IFTR, Warsaw.

Nikolaus FLEISCHMANN
Civil Eng.
Ed. Züblin AG Contractors
Stuttgart, Germany

Nikolaus Fleischmann, born 1959, received his diploma in civil engineering from the Stuttgart University. Currently he works as a project manager at Ed. Züblin AG, Civil Engineering Contractors, Stuttgart.

SUMMARY

The integration of specialised programs for the design of space frames has been realised in a Windows-based environment. The design system provides support for all phases of design, i.e., for conceptual design and detailed design as well as for archiving and retrieval of projects completed. In order to enable a fast cost estimation, a neural network application has been added. Thus, this program is able to learn from past projects in order to predict the cost of new structures.

RÉSUMÉ

L'intégration de programmes spécifiques pour le projet de structures tridimensionnelles est réalisé dans un environnement Windows. Le système de conception apporte une aide à tous les stades du projet aussi bien pour la conception et le détail que pour l' archivage et la recherche de projets achevés. Une application de réseau neuronal est incluse en vue d'une rapide estimation des coûts. Ce programme est en mesure de tirer les leçons de projets réalisés pour estimer les coûts de nouvelles structures.

ZUSAMMENFASSUNG

Die Integration von Spezialprogrammen für den Entwurf von Raumfachwerken wurde innerhalb einer Windows-Umgebung verwirklicht. Das System unterstützt in allen Entwurfsphasen, d.h. sowohl beim konzeptionellen und detaillierten Entwurf als auch beim Archivieren und Suchen bereits abgeschlossener Projekte. Um eine schnelle Kostenschätzung zu ermöglichen, wurde eine neuronale Netzwerk-Anwendung hinzugefügt. Dieses Programm ist in der Lage, aus bereits abgeschlossenen Projekten zu lernen, um damit die Kosten für neue Tragwerke voraussagen zu können.

1. INTRODUCTION

Space frames like the ZÜBLIN space frame system are regular steel structures made of prefabricated spheres and rods, connected by bolts. Such filigree spatial or plane bearing structures are frequently used to cover large areas without introducing intermediate supports (Fig. 1).

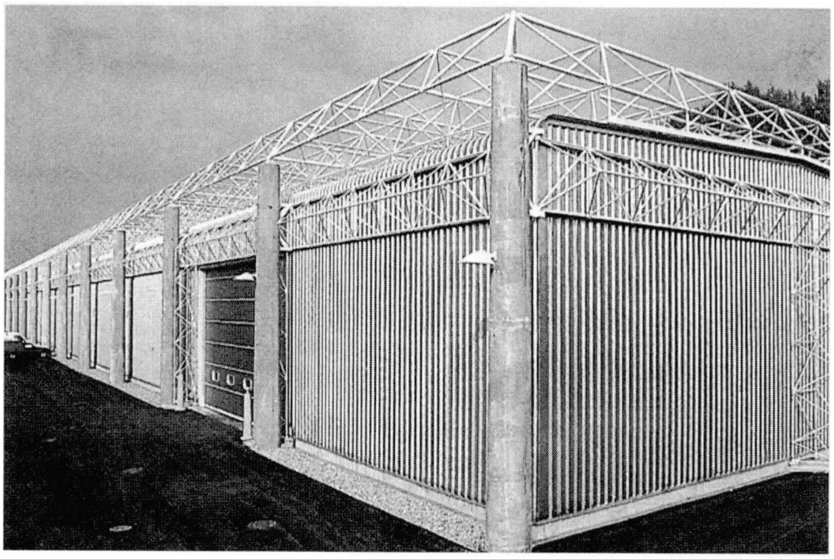


Fig. 1 Space frame roofing of a factory in Kempten, Germany

Design and manufacture of space frames belong to a field of engineering design where substantial computer assistance is available at different stages:

- modelling
- structural analysis
- construction
- cost evaluation
- manufacturing.

Specialized computer programs are available for each phase [1]. However, integration of these programs poses trouble in practice. In the course of designing and comparing alternative solutions, a great deal of routine and time-consuming work like generating, editing, transferring, evaluating, compressing and archiving of data is involved. In order to separate necessary design decisions from this routine work, the integrated design system SpaceFrame has been developed. This system manages the automated data flow between the specialized programs.

The tendency towards developing integrated tools prevails nowadays in the software industry in general and the Computer Aided Design is no exception to that trend. Most systems appearing contemporarily are based on the Object Oriented Programming paradigm (see [2], for example). Using object oriented languages (C^{++}, Smalltalk) together with CAD and database systems, complex structures are modelled according to classification and aggregation concepts. In a similar way, object-oriented languages can serve for the representation of design knowledge, e.g. codes and standards [3]. The object oriented concept will lead to new development of design applications. The integration of existing analysis and CAD programs, however, can be difficult because most of these programs have been realized without respect to the object oriented concepts.

Therefore, the integration of design software in complex fields of design in a heterogeneous environment can be achieved by so-called agents. Agents are computer programs which perform the communication between several applications. In facility engineering, the development of an agent-based framework based on interface standards is used to integrate different users (architects, engineers), different design software and different hardware [4].

2. DESCRIPTION OF PACKAGE

The integrated system SpaceFrame is based on a design model following the process model paradigm [5]. Since representation of the design structure is embedded in the application modules, the object oriented paradigm [6] is of less importance in this design model. Input and output of the specialized programs TRIMAS G, STARA, ZUBER, TRIMAS A, ZEICON are modelled as data flows.

TRIMAS G	graphic preprocessor, generation of structures, e.g. spatial frames, folded plates etc. [7]
STARA	linear and nonlinear analysis of spatial frames [7]
ZUBER	dimensioning of members, construction and manufacturing charts [1]
TRIMAS A	graphic postprocessor, evaluation of results, e.g. displacements and stresses [7]
ZEICON	CAD system for structural engineering [7]

As weights and stiffnesses of the members are not known a priori, STARA (structural analysis) and ZUBER (dimensioning of members) are usually run in several iterations. This is illustrated in Fig. 2 and 3. The structural analysis is performed with initial member sections, each member having the same section. The structural analysis leads to member forces which serve as input for the dimensioning of members. The member sections are chosen from a member catalogue and serve as an input for a new STARA run. The iteration is repeated as long as previously assumed member sections and member sections found by ZUBER are the same.

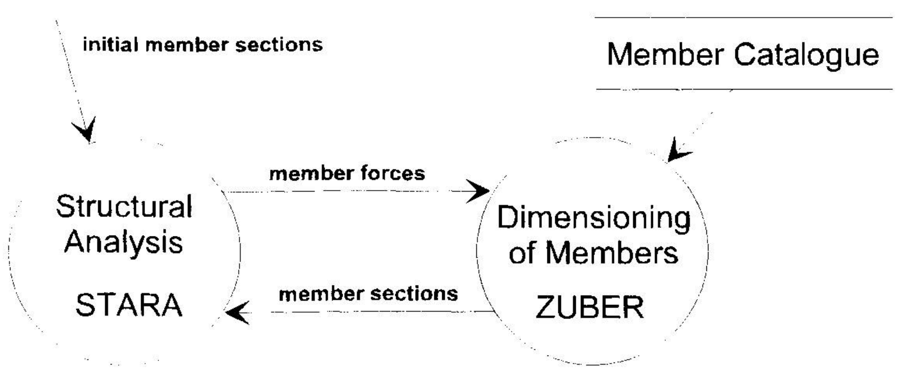


Fig. 2 Data Flow Diagram of the STARA-ZUBER Iteration

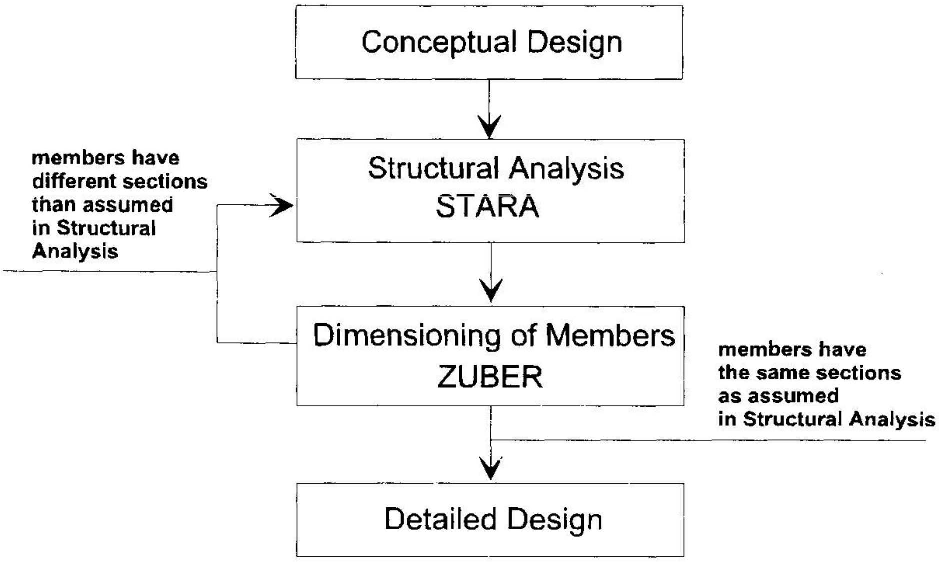


Fig. 3 State Transition Diagram of the STARA-ZUBER Iteration

Emphasizing the information flow, the approach of SpaceFrame is similar to the agent-based approach. The existing programs for analysis and CAD are embedded in one environment which allows the transfer of information from one application to the other. However, the complexity of knowledge and data is manageable using straightforward interface formats. E.g., the output list of the dimensioning program ZUBER serves as input for the CALCULATOR module and the CALCULATOR list can be directly read by the FILTER module. The main focus lies in capturing the design process from the top to the bottom, i.e. from the early design stage to the archiving of completed projects.

The goal of SpaceFrame is to cover all activities of structural engineering in a single environment. In the early design stages, i.e. for conceptual design, the GENERATOR and ARCHIVER moduli are mostly used. They support the user in checking alternative solutions, e.g when a proposal for bidding is to be prepared. In order to gain flexibility, these moduli are complemented by the LEARNER and ESTIMATOR modules.

LEARNER	creation and update of a cost estimation data base
ESTIMATOR	estimation of cost and weight based on a few input parameters, e.g. length and width

SpaceFrame combines the above mentioned moduli in a windows-based environment, adding the following features:

GENERATOR	quick generation of space frames out of a few input parameters, e.g. length, width, height, type of support, loading
CALCULATOR	calculation of prices for manufacturing, assembly, etc.
FILTER	compression of results for characteristic parameters, e.g. weight, manufacturing cost, assembly cost, etc.
ARCHIVER	storage and retrieval of compressed data from past projects

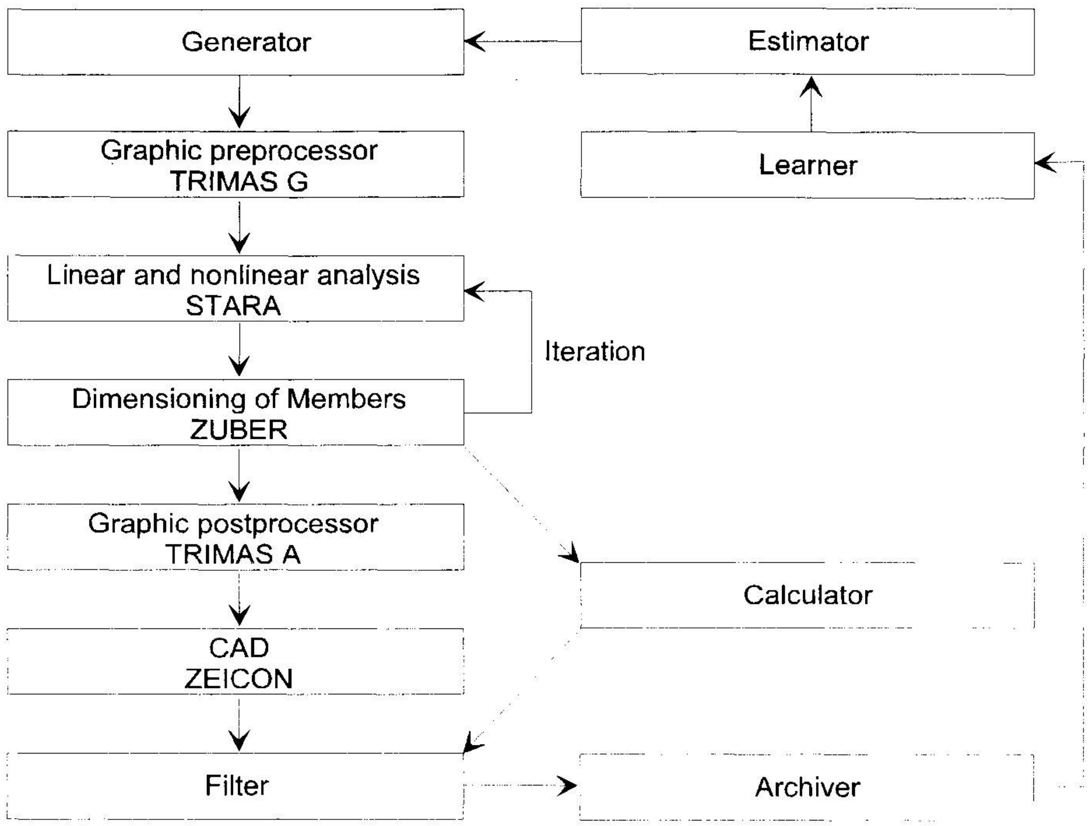


Fig. 4 Block diagram of SpaceFrame system

3. COST ESTIMATION MODULE

In order to be able to evaluate the total cost of the structure quickly, it was decided to use simulated neural networks. Their ability to learn from examples seems to be well suited for the present case, where a sufficiently large set of structures of known total cost is either available from past experience of the user or can easily be prepared by applying the analysis-design part of our system. Our previous experience in applying neural networks in structural engineering ([8, 9]) confirms that they can efficiently complement conventional AI-tools, like symbolic reasoning systems and heuristic search algorithms.

Our aim is to learn a continuous mapping $f: \boldsymbol{A} \subset \boldsymbol{R}^n \to \boldsymbol{B} \subset \boldsymbol{R}^m$ from a bounded subset of design attributes $\boldsymbol{A}$ to a bounded subset of evaluation attributes $\boldsymbol{B}$. There are several models of neural networks which could be considered as candidates for solving such a problem [10]. The most frequently used is the multilayered feedforward network with backpropagation of error (the BP-network). It admits continuous real-valued input/output data but suffers from local minima of the error surface and requires considerable time-consuming learning. Hence, we committed ourselves to the more efficient Fuzzy ARTMAP paradigm proposed by G.A. Carpenter and S. Grossberg [11].

The antecedent of the Fuzzy ARTMAP was the ART-1 model able to categorize binary coded patterns in an unsupervised manner. This rather complicated model stored learned categories in the feed-forward and feed-back weights linked to the connections between input and output layers. After a new pattern was presented to the network, a multiphase processing started. First, the winning output was determined by lateral inhibition similar to that of the Kohonen layer. Then the winning node was subjected to the similarity test (vigilance test). This test consists in comparing the similarity measure:

$$S = |(\mathbf{x}\ and\ \mathbf{t})| \ / \ |\mathbf{x}|$$

where $\mathbf{x}$ is the input vector, $\mathbf{t}$ is the vector of feed-back weights for winning node and $|\,.\,|$ means norm, to the user defined vigilance threshold.

The Fuzzy ART model is simpler since it uses only feed-forward weigths. Its input can be real-valued because the logical operators *and*, *or* applied to binary patterns in the ART-1 were replaced by their fuzzy counterparts *min*, *max*. Thus each category is treated now as a fuzzy set [12] and the vigilance threshold controls the level of fuzzinesss of the classification preferred by the user.

The next step was to combine two Fuzzy ART networks with an associative memory called map field (Fig. 5). Such a 3-layered structure is able to learn in a supervised manner, like feed-forward networks with the backpropagation of error. During learning process the Fuzzy ARTMAP receives a large number of training pairs ($\mathbf{a}^k$, $\mathbf{b}^k$). The input $\mathbf{a}^k$ is categorized by the network a and the desired output $\mathbf{b}^k$ by the network b. The correspondence between a-categories and b-categories is coded in a kind of associative memory, called the map field. The size of an individual category is governed by the vigilance parameter ρ. During learning the user selects such a value ρ_b that each $\mathbf{b}^k$ is assigned a separate category. The network automatically adjusts the vigilance parameter ρ_a: if current $\mathbf{a}^k$ and $\mathbf{b}^k$ do not match each other on the map field, then ρ_a is slightly increased and either another a-category is found or a new a-category is generated. As a result of fine tuned dynamics, the details of which can be found in [11], the network minimizes the predictive error and maximizes generalization. It is important for practical applications that Fuzzy ARTMAP indicates unrecognized cases, contrary to the BP-network.

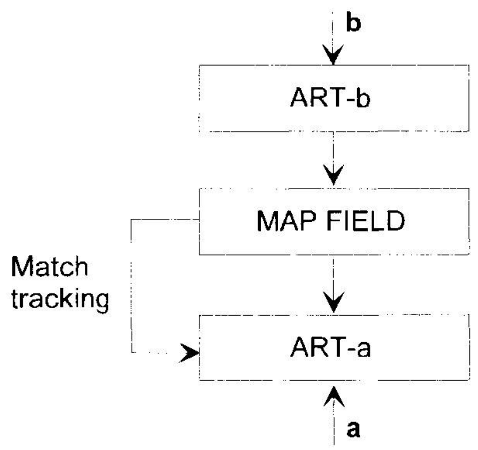


Fig. 5 Conceptual scheme of the Fuzzy ARTMAP network

4. EXAMPLE

Since archiving of the (approx. 200) completed space frame projects realized by Züblin is not yet finished, the generation of the required training set was carried out using the GENERATOR module. Fig. 6 shows the user interface for input. The user has to set the main input parameters, i.e. length *(Länge x)*, width *(Länge y)*, height *(Höhe z)*, grid *(Teilung x, Teilung y)*, load *(Last obere Ebene, Schnee)*. Default values, e.g. for cross-section, can be changed if desired. After having specified the input parameters, analysis is commenced with the tool panel (Fig. 7), providing all options according to Fig. 4.

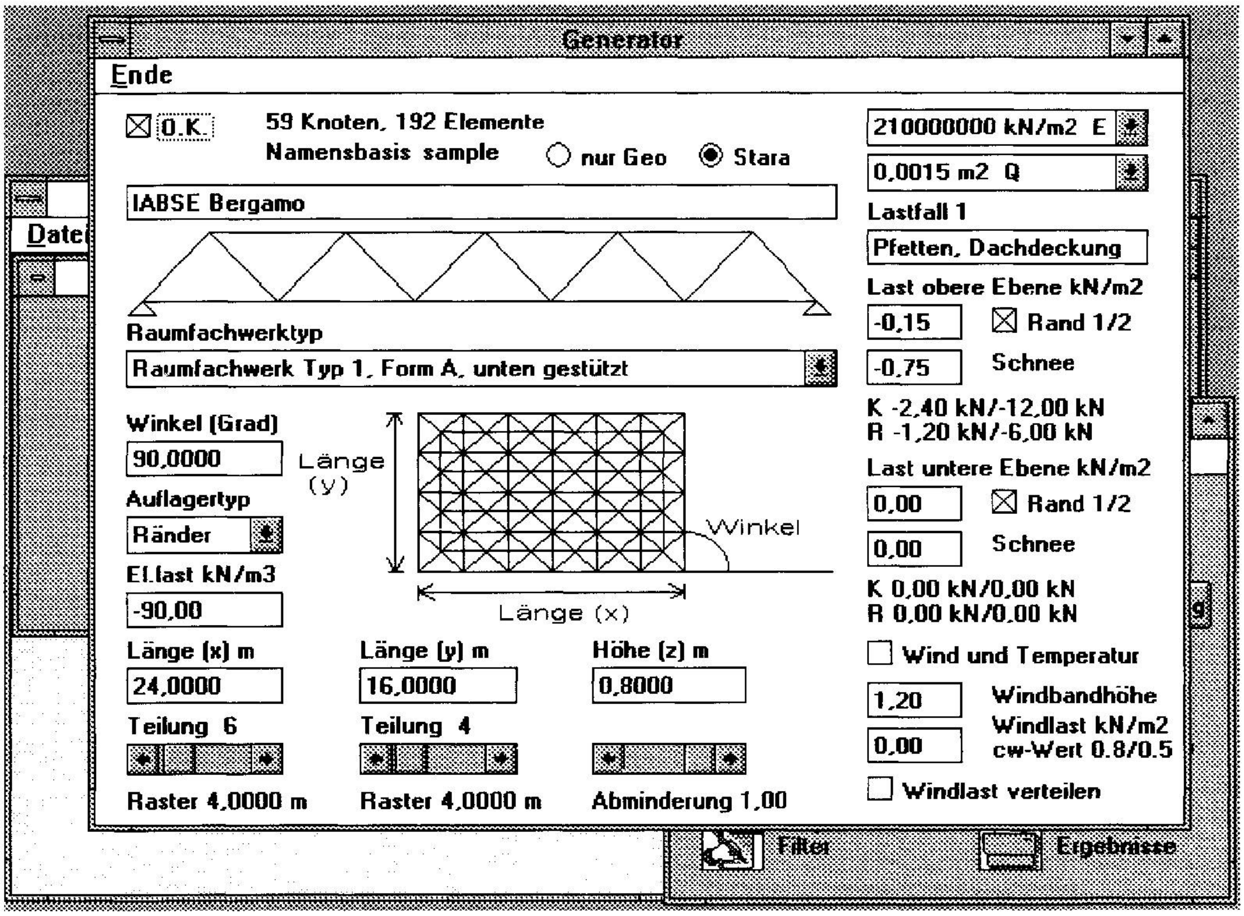


Fig. 6 GENERATOR user interface

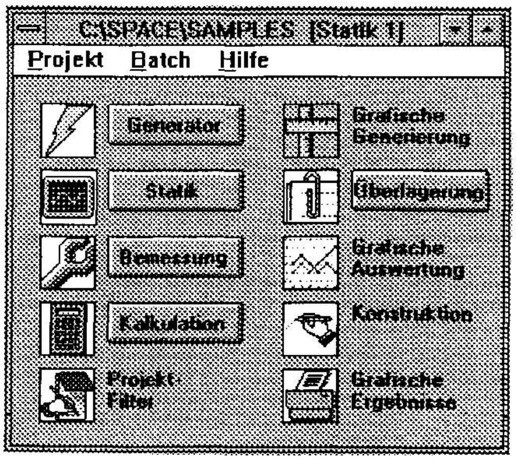


Fig. 7 SpaceFrame Tool Panel

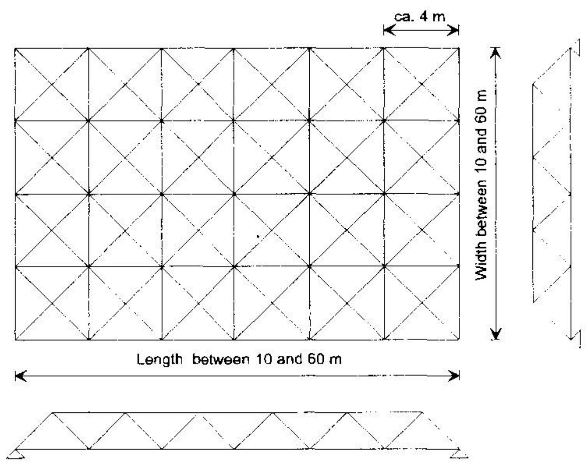


Fig. 8 Layout of the example

In order to evaluate the accuracy of cost estimation, the series of 200 roofings were generated randomly within the following constraints (Fig. 8):

- both length and width were taken between 10 and 60 m
- loading equal to 0.90 kN/m^2 (dead load, snow)
- height equal to 1/20 of the smaller dimension
- grid cell was kept to approx. 4 X 4.00 m

Each roofing was completely analyzed, optimized and dimensioned by SpaceFrame. After that the total cost of the structure was calculated according to the specification of required parts and the list of their prices. In this manner, a data set consisting of 200 triples (L_x, L_y, *cost*) was obtained.

According to the usual procedure of supervised learning, the data set was split into 2 parts: a training subset and a testing subset. The ratio between their cardinalities was taken as 2 : 1 . The first part was presented to the Fuzzy ARTMAP network 5 times allowing it to adjust the weights. These weights were subsequently frozen and the network was used to estimate the cost of each of the members of the testing set given the dimensions L_x and L_y of the structure. The following results were obtained:

Error (%)	Sequence 1	Sequence 2	Sequence 3
Mean global	5.6	7.5	5.6
Max local	17.6	15.1	15.2

The mean global error was calculated for the entire training set. The maximum local error was observed for a particular space frame. In order to check whether the learning depends upon a sequence of training examples, the experiment was conducted for 3 randomly generated sequences of data.

The quality of prediction depends mainly upon the completeness of the training set. Despite the modest number of 140 examples presented to the network, the mean cost estimation accuracy turned out to be in the order of 6%, which is quite sufficient for practical purposes. It is worth noting that the computational cost of learning is small (5 epochs) and that the response of the trained network is immediate.

5. CONCLUSION

Design integration is achieved by combining specialized programs under a windows-based environment. Following the process model paradigm, the SpaceFrame system could be realized within a few months. Intensive discussion with practitioners was a main element during work, leading to a good acceptance. Further work is aimed at the specification of the design process of structures in a formal way, using methods proposed in [5, 6].

Experience gained so far using the cost estimation module confirms our decision of applying the Fuzzy ARTMAP network for solving this particular task. Sufficient accuracy for the stage of tendering and preliminary design was achieved with 5 training cycles (approx.) involving 200 examples. It is of considerable advantage that the Fuzzy ARTMAP network learns incrementally. Consequently, knowledge gained from new cases can be incorporated easily.

ACKNOWLEDGEMENTS

The assistance of Janusz Racz and Piotr Wojan of the Laboratory of Adaptive Systems (IFTR) in performing calculations is greatfully acknowledged.

Support for developing SpaceFrame was provided by Ulrich Stuible, Department Systembau, Ed. Züblin AG, Civil Engineering Contractors, Stuttgart.

REFERENCES

[1] BURMEISTER, A., FLEISCHMANN, N., HILBER, H.M., KÖNIG, D.: Durchgehende CAD/CAM-Bearbeitung von räumlichen Stabwerken. In: Eibl, J., Obrecht, H., Wriggers, P. (Ed.), *Finite Elemente - Anwendungen in der Baupraxis*, Berlin: Ernst & Sohn, 1992.

[2] SCHERER R.J.: Integrated product and process model for the design of reinforced concrete structures. In: *Proceedings of the EG-SEA-AI Workshop*, Lausanne, 1994, 132-145.

[3] HARTMANN, D., FISCHER, A., HOLEWIK, P.: Object Oriented Modeling of Structural Systems. In: Cohn, L.F. (Ed.), *Proceedings of the Fifth International Conference on Computing in Civil and Building Engineering (Anaheim, California, June 1993)*, American Society of Civil Engineers, New York, 1993.

[4] KHEDRO, T., GENESERETH, M.R., TEICHOLZ, P.M.: Agent-Based Framework for Integrated Facility Engineering. *Engineering With Computers* 9 (1993), 94-107.

[5] YOURDON, E.: Modern Structured Analysis. Prentice-Hall, 1989.

[6] RUMBAUGH, J., BLAHA, M., PREMERLANI, W., EDDY, F., LORENSEN, W.: Object Oriented Modeling and Design. Prentice-Hall, 1991.

[7] RIB: RIBTEC User Guide, RIB Bausoftware GmbH Stuttgart, 1994.

[8] ARCISZEWSKI, T., BORKOWSKI, A., DYBALA, T., RACZ, J., WOJAN, P., Empirical performance comparison for symbolic and subsymbolic learning systems. *Proceedings of the First ASCE Congress on Computing in Civil Engineering* (Washington, DC, June 1994), ASCE Press, Washington, DC, 1994.

[9] BORKOWSKI, A., Neural networks and structural optimization. In: Burczynski, T. (Ed.), *Proceedings of the Advanced TEMPUS Course on Numerical Methods in Computer Aided Optimal Design (Zakopane, May 1992)*, Silesian Technical University of Gliwice, Gliwice, 1992.

[10] HECHT-NIELSEN, R., Neurocomputing, Addison-Wesley, 1990

[11] CARPENTER, G.A., GROSSBERG, S., REYNOLDS, J.H., ART-MAP: Supervised real time learning and classification of non-stationary data by a self-organizing neural network. *Neural Networks* 4 (1991), 565-588.

[12] Zadeh, L.A., Fuzzy Sets. *Information and Control* 8 (1965), 338-353.

Decision Support for the Fabrication-Led Design of Tubular Trusses

Aide pour le dimensionnement de fermes à sections creuses pour un coût minimal de fabrication

Entscheidungshilfe zur fabrikationsbeeinflussten Bemessung von Fachwerken aus Hohlprofilen

Walid M. K. TIZANI
Lecturer
University of Nottingham
Nottingham, UK

Walid Tizani obtained his BSc, MSc and PhD from the Universities of Mosul, Iraq; Leeds, UK; and Bradford, UK. His main area of research is in the application of Artificial Intelligence techniques in civil engineering. He is now a lecturer at the Dept of Civil Eng. at Nottingham Univ.

David A. NETHERCOT
Prof.
University of Nottingham
Nottingharn, UK

David Nethercot obtained the degrees of BSc, PhD and DSc from the University of Wales in Cardiff. He has more than 25 years research experience in structural engineering, having specialised in the area of steel structures. In 1993 he was elected to the Royal Academy of Engineering, UK.

Nigel J. SMITH
Senior Lecturer
UMIST
Manchester, UK

Nigel Smith, BSc, MSc, PhD, is a Senior Lecturer in the Project Management Group at the Univ. of Manchester. His current research interests include construction as a manufacturing process and the applicability of expert systems in construction management.

SUMMARY

This paper is concerned with the development of a knowledge-based system for the economic design of tubular trusses. The system interfaces to structural analysis and design packages and includes a module for checking joint capacities and a model for estimating the cost of fabrication of these trusses. Using this system, designers can analyse and design the structure and then judge the optimality of their solution based on the predicted overall cost of production and not on material cost only. The cost model is used to predict the relative cost of fabricating a design option compared to its feasible alternatives.

RÉSUMÉ

Cet exposé concerne le développement d'un système fondé sur la connaissance pour le dimensionnement économique des fermes à section tubulaire. Le système présente un interface pour l'analyse des structures et des logiciels de dimensionnement. Il comprend un module pour la vérification de la capacité des joints et un modèle pour l'évaluation des coûts de fabrication de ces fermes. Utilisant ce système, les ingénieurs peuvent analyser et dimensionner la structure, juger de l'optimalité de leur solution basée sur le coût global prévu de production. Le modèle d'évaluation de coût est utilisé pour juger le coût relatif de fabrication d'une solution comparée à ses alternatives.

ZUSAMMENFASSUNG

Die Arbeit beschäftigt sich mit der Entwicklung eines wissensbasierten Systems zur wirtschaftlichen Bemessung von Fachwerken aus Hohlprofilen. Das System verbindet Berechnungs- und Bemessungsprogramme, ein Modul zur Ueberprüfung der Knotenkapazität und ein Modell zur Schätzung der Herstellungskosten eines Fachwerks. Der Gebrauch dieses Systems ermöglicht dem Planer die Berechnung und Bemessung eines Tragwerks; darüber hinaus gibt es ihm die Möglichkeit, die Qualität seiner Lösung auf Basis der ermittelten Gesamtkosten und nicht ausschliesslich Materialkosten zu beurteilen und so eine optimale Lösung zu finden.

1. INTRODUCTION

At the conceptual stage of design of steel structures, decisions are made which influence the overall cost of the construction. This is mainly due to these decisions imposing constraints on the later stages, namely the fabrication and erection of the structure. Designers generally attempt to produce economic solutions by concentrating on producing efficient structures with minimum weight. It is a well-known fact, however, that minimising the weight of the structure is not only limited in scope but can also be counter-productive [1]. In many such cases an efficient structure would require stiffening at the joints, which is a costly fabrication operation likely to out-weigh the saving made from optimal weight. Often, though, this is the only option available to designers, since it is unlikely that they would have had extensive experience in the fabrication of structures in addition to their expertise in design. More importantly, they do not have the information about which of the options available to them is more costly with regard to fabrication and erection.

This situation is more relevant in the construction of tubular trusses than when using open sections. Tubular construction is relatively new and is not as widely used as open sections. Consequently, less expertise exists among designers and fabricators. In addition, variations in the joint detailing in tubular structures tend to produce high cost differentials for fabrication, especially when reinforcement is involved. This is mainly due to the complexity of the intersections between tubular sections, more specifically circular hollow sections, and to the large amount of welding involved in their fabrication. Therefore, the effect of decisions taken at the conceptual stage of design will have an even higher effect on the cost of construction of tubular structures than on that of open sections.

It is clear that more economical solutions would result if the information regarding the likely costs of design options was readily available. Designers would then be able to make better decisions being aware of the effect of these decisions on the overall cost of the structure. In a sense, this will widen the scope for economy since designers are then able to incorporate factors that influence the cost of fabrication in conjunction with minimising the cost of material.

This paper concentrates on describing the way in which an integrated system for the design of tubular trusses makes use of a model for the cost of fabrication in order to arrive at fabrication-led economical solutions for tubular structures. The development of the system is being funded by the Engineering and Physical Sciences Research Council (EPSRC) UK. It includes structural analysis and design modules and a module for the analysis of joint capacities of tubular joints and an economic appraisal module. The economic appraisal module advises on available options for design aspects and employs a cost model that estimates the likely cost of fabrication of tubular trusses. A more detailed description of the integrated system can be found in reference [2].

2. THE COST MODEL

The major factor influencing the variations in the cost of fabrication is the complexity of joint detailing. The simpler the details the less costly they are to fabricate. Conversely, the more complex the details, especially those which include stiffeners, the more costly they become. One of the major objectives of the model is to be able, for a fabrication content, i.e. the amount of cutting, profiling, welding, etc., to estimate the likely cost of fabrication and to be able to compare this cost to that of alternative details. Hence the aim is to predict the relative cost and not the absolute cost of the structure.

The cost model has been developed using knowledge acquired from expert designers and estimators working in the construction of tubular trusses. This knowledge was then interpreted and represented within a computer program capable of manipulating knowledge, encompassing rules of thumb and heuristics, as well as algorithms, here termed knowledge based system [3]. The techniques used in acquiring the knowledge, the way the knowledge was interpreted and the way it was represented in a computerised form is described elsewhere [4].

The model was implemented using the object oriented methodology [5]. The process of estimating the cost of fabrication is carried out through message-passing between objects representing the various entities and relationships of the model. The main objects adopted in this representation are: fabrication-machine, fabrication-operation, section, member, joint, and truss. An instance of the fabrication-machine object represents a specific machine with associated specifications. A fabrication-operation object, eg WELDING, estimates the cost of this operation, in minutes, given relevant data such as the details of the operation, eg weld length and thickness, and the fabrication-machine to be used. A joint object, composed of a number of member objects, will divide its fabrication content into the various fabrication operations and request a cost estimate for each by communicating with the relevant fabrication-operation objects. A truss object, composed of a number of joint objects, will add together the cost of fabricating each of its joints.

The model also allows for the concurrent costing of a number of instances of objects. Hence, the costing of joint detailing and its alternatives can be carried out concurrently. Similarly, the cost of a number of trusses can be obtained.

3. FABRICATION-LED DESIGN PROCESS

The fabrication-led design process applied to tubular trusses can be carried out using the integrated system. This is achieved by providing within a single system all the tools required for this purpose. In the following the dynamics of this process are explained in more detail while concentrating on the way the cost model is used to achieve the aim of this process.

At the conceptual stage, a designer might propose a number of conceptual solutions for the structure. These proposed solutions will most likely be based on experience gained from previous design cases. The designer may choose to structurally analyse and design only one or a number of these conceptual solutions. The likely uses that can be made of the system in order to produce economical solutions are given below.

3.1 Joint capacities

The conventional process requires designers to produce structurally adequate solutions where the geometrical configurations, the section sizes, and the welding requirements are specified. Detailed design of joints will not usually be carried out by designers. It is assumed then that the resulting joint details are capable of transmitting the forces adequately and that it is the responsibility of fabricators to ensure that the joint capacities fulfil this function.

It is often the case, however, that the resulting joint capacities are inadequate and joint stiffening might be required. This is especially true when dealing with tubular joints, where adequate member stiffness does not automatically mean adequate joint capacities even if full strength welds were specified. In tubular structures the joint capacities depend, among other factors, on the geometrical properties of the joint, eg for circular hollow sections they depend on the ratios of the diameters of the section sizes and on the diameter to thickness ratio of the chord member. This is in addition to limiting geometrical requirements such as minimum overlapping distance of bracing members. Therefore, if costly joint stiffening is to be avoided, the section sizes should be chosen with joint capacities in mind as well as the strength of members.

For this purpose a module for checking joint capacities is integrated within the system. Having achieved a structurally adequate solution for the members, the designer can then carry out joint capacity checks. Future enhancements to the model will provide advice on appropriate modifications in order to improve the capacities of unsatisfactory joints. Of course a designer can insist on the use of a certain configuration despite this resulting in the need to reinforce the joint.

3.2 Fabrication cost

Having arrived at a structurally adequate solution for members and joints, the solution can then be assessed in terms of the likely cost of fabrication. The designer can obtain a summary of the cost of fabrication of the whole of the truss. The cost, in this case, consists of the total time required to carry out all fabrication operations (in minutes), the total weight of the truss (in tonnes), and the total surface area of the truss (in square metres). The estimated time taken to carry out one type of fabrication operation is also available, eg cost of all welding and cost of profiling all members.

Four ‘global’ parameters are provided to convert costs in minutes, tonnes and square metres into monetary figures, i.e. cost per hour for manual-intensive fabrication, cost per hour for equipment-intensive fabrication, cost of steel per tonne, and cost of paint per square metre. Consequently the designer can obtain the likely cost of the structure in monetary values. However, this is not the primary use of the system. All the information and rules provided in the cost model, although they are obtained from up to date sources, are ‘tuned’ to give relative costs. Therefore the intended use of the model is to compare the obtained figures for the various alternatives. A number of feasible solutions for the structure could then be compared by fabrication content, weight, and surface areas. Monetary values can be used to compare a number of feasible solutions, where the global parameters should be calibrated to reflect the current cost balance between material and production costs.

Designers are also able to inspect the cost associated with individual joint detailing. In addition a *what-if* scenario can be followed by modifying details and requesting cost assessments. For example, the designer can assess the sensitivity of the cost of a detail to changes to the joint geometry, welding specification and/or member thickness. Consequently, a designer can appraise the likely saving that can be made from changes to one or a number of joint details before making final decisions. A breakdown of the cost of joints detailing between individual fabrication operations, and the total cost of certain operations, eg cost of assembly of joints, is available for inspection.

Costs can also be grouped in terms of individual members and in terms of individual fabrication operations. Thus, the cost sensitivity of fabrication operations to certain parameters can be obtained. For example, the cost of welding two specific items can be inspected and its sensitivity to parameters such as welding thickness, welding length resulting from geometry and welding type, can be assessed.

3.3 Decision support

The integrated system is not intended to automate the process of the design of tubular trusses but to aid in the making of informed decisions. The main benefit of the cost model is as a decision support tool for designers. The model makes available information that can be used to appraise the effects of decisions on the overall cost of the structure. The information can be used in many ways depending on the priorities and constraints imposed on the aesthetic appearance and the function of the structure. The designer can modify certain parameters, within the constraint imposed, in an attempt to optimise the solution based on the criteria of economy and practicality. For example, if the designer is restricted to a geometrical configuration and a certain range of diameters for section sizes, the types of connections, section thickness, and fabrication specifications can be calibrated to arrive at optimised solutions within these constraints.

4. SUMMARY AND CONCLUSIONS

A system for the economic appraisal of tubular truss designs has been developed; it is aimed at aiding designers in obtaining economical solutions. The system includes, in addition to analysis and design modules, a joint analysis module and an economic appraisal module. The joint design module calculates joint capacities. The economic appraisal module advises on feasible alternatives and uses a cost model to estimate the cost of fabrication of tubular trusses.

Economical solutions are achieved by indicating where joint reinforcement would be required and by estimating the relative costs of a number of feasible alternatives to the design. Therefore designers will be able to avoid costly reinforcement and to make informed decisions on which alternative to choose based on knowledge about the effects of these decisions on the cost of fabrication. This approach to design is termed fabrication-led design.

The main benefits of this system are that it provide designers with an integrated system containing all the tools necessary to produce economical designs covering all the stages of design and production and that it provides them with information not usually available to them, thereby, allowing them to consider all factors affecting cost and not only the weight of the material.

REFERENCES

1. STEEL CONSTRUCTION INSTITUTE, Economic design and shorter lead times, *Steel Construction Today*, UK, Vol. 6, No. 5, 1992, p. 228.
2. TIZANI W M K, DAVIES G, McCARTHY T J, NETHERCOT D A, AND SMITH N J, A knowledge-based approach to construction-led structural design, In Information technologies for construction, civil engineering and transport (eds: Powell J A and Day R). Brunel University with SERC, 1993.
3. ADELI H, Expert systems in construction and structural engineering, Chapman & Hall, London, 1988.
4. TIZANI W M K, DAVIES G, NETHERCOT D A, AND SMITH N J, Fabrication-led design of tubular trusses using a cost model: knowledge acquisition and representation. Submitted to the Sixth International Symposium on Tubular Structures, Melbourne, Australia, 1994.
5. GRAHAM I, Object Oriented Methods, Addison-Wesley Publishing Company, 1991.

Definition of a Knowledge Base for Structural Design

Définition d'une base de données pour le projet de structures

Definition einer Wissensbasis für den Tragwerksentwurf

Aldo CAUVIN
Prof. of Structural Eng.
University of Pavia
Pavia, Italy

Aldo Cauvin earned his Dr of Eng. degree from Politecnico Milan, where he has been Professor since 1986. Author of papers on nonlinear analysis of reinforced and prestressed concrete structures and on application of expert systems to structural design. Member of the European Group for Structural Engineering Application of Artificial Intelligence.

Rinaldo PASSERA
Partner
Passera & Pedretti
Lugano, Switzerland

Rinaldo Passera earned his Civil Eng. degree from ETH Zurich and has since acquired much experience in structural design and analysis. He is author of several important designs in Switzerland and abroad. Member of the European Group for Structural Engineering Application of Artificial Intelligence.

Giuseppe STAGNITTO
Research Eng.
University of Pavia
Pavia, Italy

Giuseppe Stagnitto earned his Dr Eng. degree from the Univ. of Pavia and joined the Dept of Struct. Mechanics of the same Univ. Author of papers on expert systems applied to structural design and structural restoration and strength assessment of historical buildings. Member of the European Group for Structural Engineering Application of Artificial Intelligence.

SUMMARY

The criteria adopted in the preparation of the knowledge base to be used in an expert system specialised for the preliminary design of civil structures are briefly illustrated. In particular the organisation of that part of the knowledge base which is concerned with long span floors and roofs is described. The importance of the use of these criteria in the preparation of a "non-idiosyncratic" knowledge base is emphasised.

RÉSUMÉ

L'article décrit les critères adoptés dans la préparation d'une base de données destinée à être utilisée dans un système expert aidant à l'avant-projet de structures de génie civil. Il traite en particulier l'organisation de cette partie de la base de données relative aux planchers et toitures de grande portée. Il souligne l'importance de ces critères dans la préparation d'une base de données aussi peu influencée que possible par des critères subjectifs.

ZUSAMMENFASSUNG

Es werden die Kriterien zur Vorbereitung einer Wissensbasis erläutert, die in einem Expertensystem dem Vorentwurf von Ingenieurbauwerken dient. Insbesondere wird die Organisation des Teils beschrieben, der sich auf weitgespannte Decken und Dächer bezieht. Es wird betont, wie wichtig diese Kriterien für eine Wissensbasis sind, um sie so wenig wie möglich subjektiv zu beeinflussen.

1. INTRODUCTION.STRUCTURE OF EXPERT SYSTEM

An Expert system shell for the preliminary design of structures is now being prepared according to the method explained in ref.[2].The system is based on two basic criteria:
-Knowledge is organized in "models";by "models" we intend a rule where the premise(cause) is the design choice and the conclusion(effect) is the performance(predicted behaviour) of that choice.
-The design space is divided in a number of separated,hyerarchically organized sub-spaces,to which separated knowledge bases correspond.
The design proceeds from general to particular sub-spaces according to a "top-down" refinement plus constraint propagation process[6]. This is nothing else that a simulation of the procedure which is naturally adopted by every designer,who makes basic decisions first and then proceeds to more particular decisions taking into account the constraints which are consequences of "strategic"choices already made.

2. SUBDIVISIONS OF DESIGN SPACE

The first subdivision of design space is made according to the definition of structural types corresponding to different categories of structures. This definition is an uneasy task and,given the enormous variety of structural shapes and layouts which can be used in practice,could never be complete and entirely satisfying.
For some categories of structures this task can be made easier by the fact that in most cases the structural shapes are well defined and limited in number. Such is the case of bridge structures and ,to some extent,also of tall buildings.
A broad classification of structural types which can include most of those structures whose design requires a careful attention has been proposed in ref.[3]. Among the three types which were individuated(the other two are bridges and tall buildings) we shall consider briefly the following(defined as type B in ref.[3]):
-"Structures whose most important elements lie essentially in an horizontal plane" or whose dimensions in the two horizontal directions prevail on the vertical one. This is the case of long span low rise buildings,such as industrial buildings,multistoreys parking lots,assembly and sports halls. The main structural elements in this case are floors ,roofs and theirs supporting beams(all horizontal or sub horizontal elements).
This is maybe the most "crowded" structural type and also the one with the greatest number of layout and shapes and therefore the most difficult to classify.
A trial partial classification of layout and shapes for square grids is represented on fig.1.
The second subdivision of design space concerns the definition of structural layout. Given the almost unlimited possible plan layout of buildings the classification of layouts can only be made in our opinion according to the geometry of the structural grids. We can thus individuate square and rectangular grids with different span ratios,isolated and multiple spans, equal or unequal spans.
In the scheme of fig.1 only square grids ,isolated and multiple equal spans are considered.
The third subdivision of design space concerns the definition of structural shapes.
Some possible shapes suited for square grids are indicated on fig.1.
At last,the final purpose of the expert system is the definition of basic structural dimensions.

3. PREPARATION OF KNOWLEDGE BASES FOR THE INDIVIDUATED DESIGN SUB-SPACES

The knowledge bases for the defined design sub-spaces should be based on careful examination of a great number of existing structures more than on the "practical"experience of single designer.
In other words the knowledge base should be "extracted" in an as much as possible objective way from a great number of valid and well tested designs,so that the decisions can exploit the much broader practical experience of a great number of engineers in different countries and environments.
If the data base is organized in the way that has been described the preparation of rules can be performed in a rather "mechanical" way,possibly using "perceptron type" neural network as is explained in ref.[8]

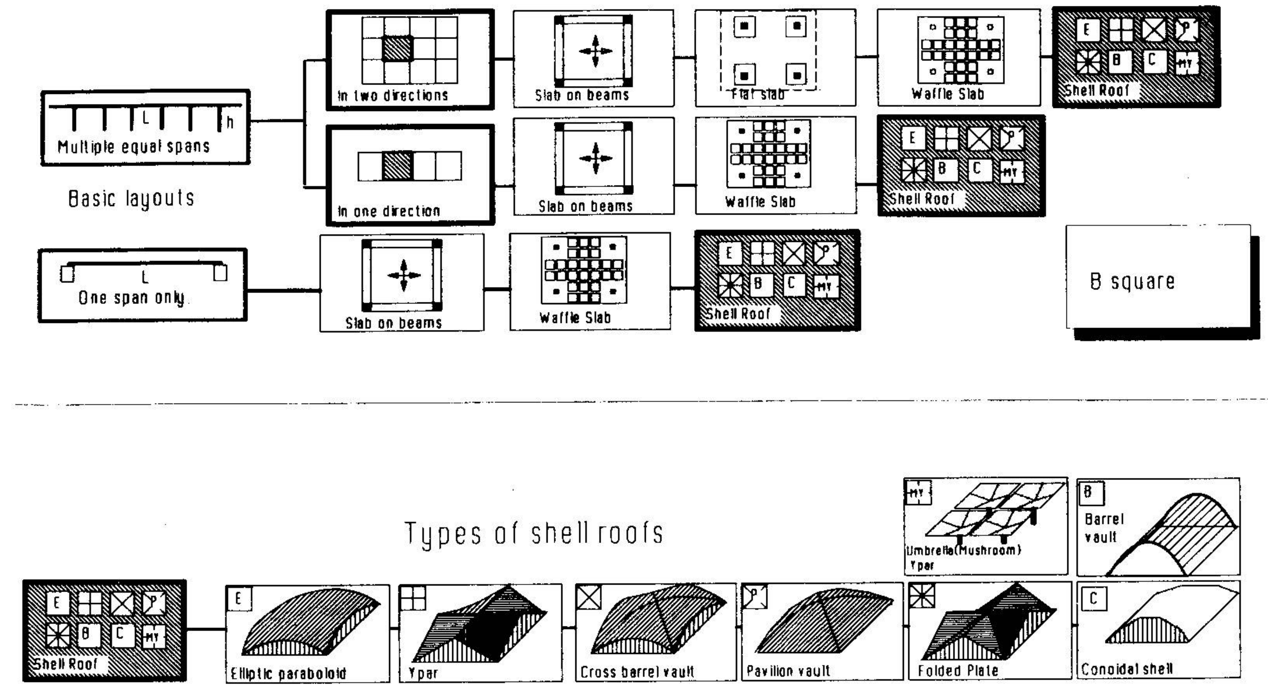


Fig.1-Reinforced/Prestressed concrete floors and roofs classification

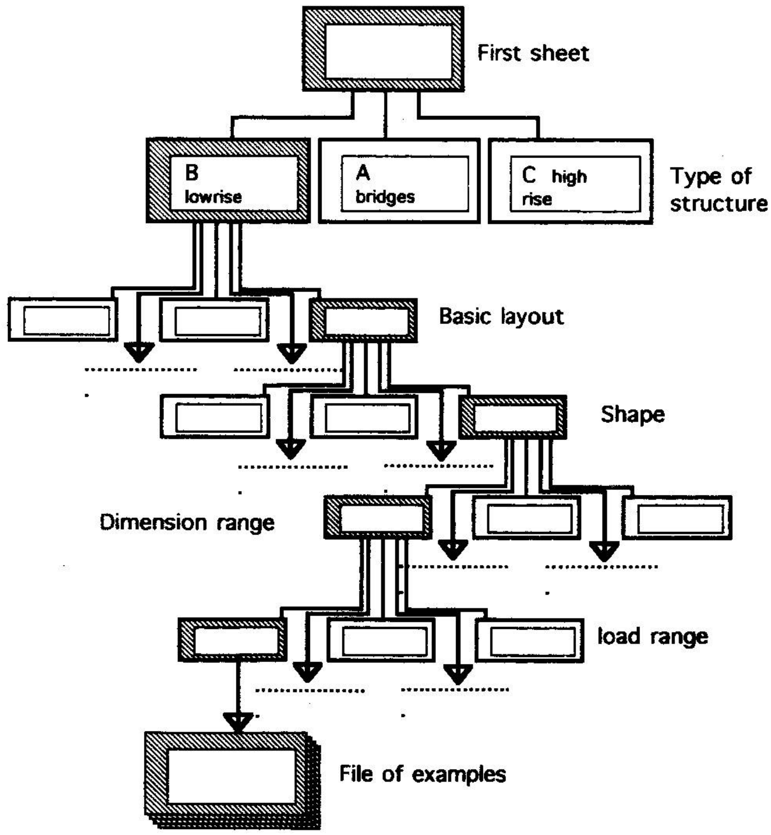


Fig.2-Structure of Hypermedia data base

To collect the needed data , the operation to perform is the organization of a graphical data base in which relevant data concerning realized succesful designs are organized in such a way that data interesting the building of the knowledge bases can be quickly extrapolated.
It is convenient to organize the data in an hypermedia environment[1][7],in which drawings as well as numerical data can be easily stored in the form of "sheets".
For easy access the data base can be organized according to a "treelike"structure,adopting a classification similar to the one used for the definition of the design sub-spaces.
The "leaves" of this "tree" represent short descriptions of realized designs organized according to structural shapes,loadings and spans.
In addition ,for every structural shape ,sheets containing diagrams devoted to the trial dimensioning of structural elements should be included(for example diagrams giving the floor depth in function of span and loads)

4. USES OF DATA BASE IN HYPERMEDIA FORM

In addition to its main purpose,that is the definition of models ,and functions giving element dimensions ,to be introduced in the knowledge bases, the hypermedia data base can also be used for the refinement of a preliminary design such as the one which is obtained with the procedure which is illustrated in [2] by comparison with existing designs.
In fact at this level,it should be extremely useful a rational overview of existing ,realized and tested designs whose specifications be not much different from the design which is being elaborated.
In other words at the level of detail definition "case based" design procedures should be used.

5. CONCLUSIONS

In the preparation of an expert system devoted to design two basic steps can be individuated:
-The preparation or adoption of a "Shell" containing an inference mechanism and graphical interfaces suitable for structural design .A similar shell,with a discussion concerning its use is described in ref.[2]
-The definition of knowledge bases for each design level which can incorporate as much experience and wisdom as possible.
To get this result interviews with single designers are not sufficient;the exam of data relative to a great number of existing designs in different countries and environments to extract rules and dimensioning criteria are necessary.
This data ,to be correctly used,must be organized in a suitable data base.
A data base organized in Hypermedia form according to criteria similar to those utilized in defining the stages of design seems the most useful to perform the task of "building up" the knowledge base.

6. REFERENCES

[1]Cauvin A,Stagnitto G.,Stagnitto D,Expert Systems in design of Structures.An application to bridges.From the Volume:Progress in Civil engineering,Edited by Grierson,Franchi,Riva.Kluwer Academic publishers,September 1991.
[2]Cauvin A,Stagnitto G.,A "Model Oriented" Expert system for the preliminary design of bridge structures,First workshop of EG-SEA-AI,EPFL,Lausanne 1994.
[3]Cauvin A,Stagnitto G.,Classification of Structures and organization of hypermedia data bases for Structural Design,First workshop of EG-SEA-AI,EPFL,Lausanne 1994.
[4]Tong C,Sriram,Artificial Intelligence in Engineering Design,Vol.1,chapt.1,introduction,Academic Press Inc.,1992
[5]Faltings B,Reasoning Strategies for Engineering Problems,Proceedings of IABSE Colloquium "Knowledge Based Systems in Civil Engineering",Bejing 1993.
[6]Steinberg L,Design as Top Down Refinement Plus Constraint Propagation,from Chapter 8 of book:Artificial Intelligence in Engineering Design,Vol.1,Academic Press Inc.,1992
[7]Hypercard 2,User's Manual,Apple Corp.1990
[8]Cauvin A.,Use of Neural Network for preliminary structural design,Research Report,Dept. of Structural Mechanics,University of Pavia,Pavia,Italy, October 1994

Development of Knowledge-Based System for Cofferdams

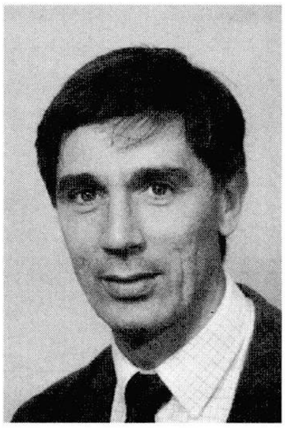
Développement d'un système à base de connaissance pour les barrages de palplanches

Entwicklung eines wissensbasierten Systems für Fangedämme

Charles J. VOS
Prof. of Constr. Technology
University of Technology
Delft, The Netherlands

Charles J. Vos was trained in civil eng. at the univ. of Delft, NL, and Leeds, UK. Since 1963 he has been active in the design of bridges, quay walls and concrete offshore structures. Since 1982 he has also been lecturing construction technology of concrete structures at TU Delft.

René BUVELOT
Manager, Information Technol.
Delta Marine Consultants
Gouda, The Netherlands



René Buvelot studied Civil Engineering at Delft University of Technology, NL. For seven years he worked as Design Engineer on civil engineering projects. Presently he is responsible for Information Technology at Delta Marine Consultants.

SUMMARY
The use of expert systems is currently limited to banking and insurance companies. Apart from experiments, no operational systems for the construction industry are described in the consulted literature. This paper discusses the development of a knowledge-based design system for construction pits, emphasising the system development process instead of describing the system itself. After the realisation of a prototype, it was concluded that knowledge-based systems are very promising tools for construction engineering.

RÉSUMÉ
L'utilisation de systèmes experts est actuellement limitée aux banques et compagnies d'assurance. Dans le génie civil, on en est encore au stade de l'expérimentation; il n'y a en tous cas aucun système opérationnel décrit dans la littérature. Le cas particulier d'un système à base de connaissance pour les barrages de palplanches permet de décrire le processus d'évolution du système, plus que le système lui-même. La réalisation d'un prototype permet d'envisager des applications prometteuses en génie civil.

ZUSAMMENFASSUNG
Expertensysteme sind zur Zeit auf Banken und Versicherungen beschränkt. Für das Bauwesen stehen sie noch im Experimentierstadium, zumindest ist in der Literatur kein operables System beschrieben. Am Beispiel eines wissensbasierten Entwurfssystems für Fangedämme wird weniger das System selbst als vielmehr der Entwicklungsprozess beschrieben. Nach der Entwicklung eines Prototyps lässt sich sagen, dass solche Systeme im Bauingenieurwesen mächtige Werkzeuge darstellen.

1. INTRODUCTION

Delta Marine Consultants (DMC) designs civil engineering structures. As a subsidiary of the Dutch general contractor HBG, one of the activities DMC is involved in is the design of temporary works such as construction pits, often used to enable the construction of basements and shallow foundations. Optimizing this design requires experience of the engineer as well as many tedious calculations, growing in complexity with the implementation of new design codes, while the available time to create designs, especially in tender phases, is reducing. Moreover, the industry faces a loss of experience due to early retirements and increased holiday periods.

DMC decided to investigate the possibility to create a Knowledge Based Design System to take over all engineering and estimating activities that can be defined in straightforward set of knowledge rules.
In this paper the approach and the results of the project are reported.

2. DEVELOPMENT TOOLS

The key development tool used is Design^{++}, a Unix based product of Design Power Inc. (Finland). This is an object oriented knowledge based reasoning shell, based on product modelling that can contain 3-D geometric data of objects. To represent the geometric data, Design^{++} is linked to a common CAD-system (AutoCAD). A second link is made to Oracle, merely to retrieve product information.

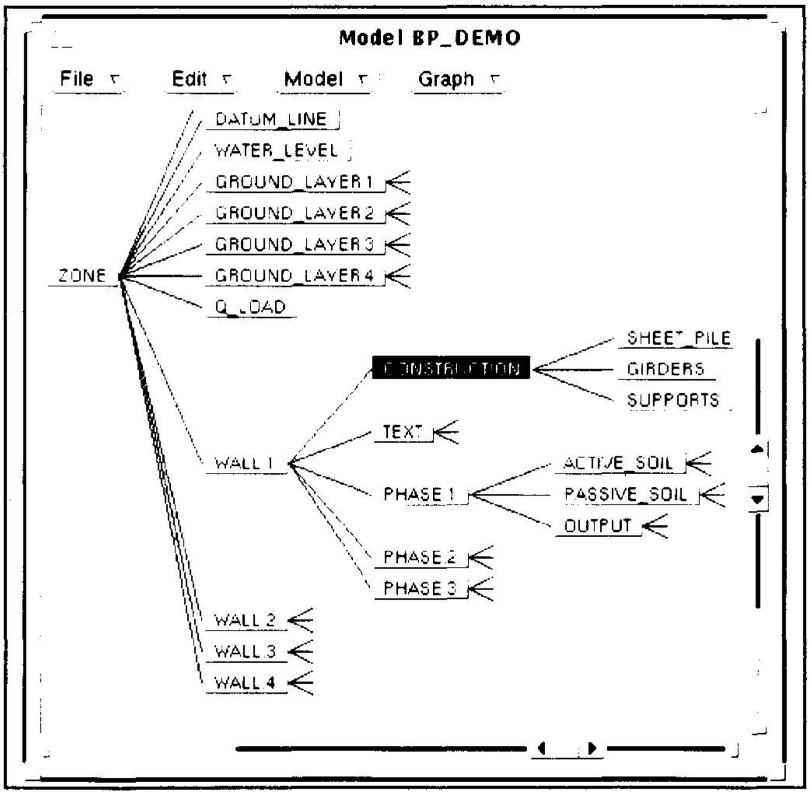


Fig. 1 Partial product model of construction pit

Design^{++} is suited to build configuration systems which link components from the database in order to create installations, structures, etc. according to the knowledge rules. Key element of each Design^{++} application is a product model that consists of all objects which may occur in the structures covered by this product model and knowledge rules describing the relations between these objects. The Design^{++} (forward) reasoning system is determined by these knowledge rules in Lisp, containing information to calculate and select values for attributes of objects such as selecting products from the database, determining which objects are relevant for the present design task and creating input files for external computer programs.

If the reasoning system lacks information, it will ask the user for input to proceed. This enables interactive operation of the system. Figure 1. shows an example of a part of the product model. The reasoning system needs the constraints given by the user as input to generate the structure within the limits of the Design^{++} model. It is the task of software developers to create the product model and to describe knowledge rules. This paper discusses the development of the product model for construction pits, named Pit Design^{++}.

3. APPLICATION AREA

3.1 Objectives of Pit Design^{++}

The application area is subject to the goals of the system, being:
- *reducing design time.* Use of the system will lead to a considerable time saving of 50 % or more, used not only to reduce costs but also to take more time to look into design alternatives or sensitivity analyses of different parameters. This will lead to improved and more optimized design solutions.
- *improving quality.* Use of a knowledge based system for a design task will not only standardize the creative design process but also ensure a proper use of design codes and company rules. The latter may be one's own interpretation of codes and rules dealing with the constructability of the design solution, in view of economy, safety and environmental impact.

3.2 Process covered

In order to develop an efficient system, an IDEF-0 analysis has been carried out (as illustrated in figure 2), concerning the process of:
1. choosing typical solutions;
2. elaborating the design;
3. estimating detailed design;
4. detailing design;
5. constructing sheet piled excavations.

Phases 1, 2 and 3 are required for both tender design and detailed designs. They will consequently be used for all design projects. Phase 1 was excluded from the program since it was found impossible to provide reliable knowledge rules. This phase will only occur for projects in progress and needs a high level of variable detail. It was therefore decided that Pit Design^{++} should cover phase 2 and 3 of the process only. This means elaboration of the chosen typical solution to the level of tender-design, including a calculation report for the client, drawings and a bill of materials priced with standard prices.

3.3 Products covered

Besides the process, the product range needed to be limited. It was decided that the area, which should be covered by the system, would cover 80 % of all possible construction pits. This means that apart from some exceptions all design tasks had to be dealt with by the knowledge system. Since designs have to be conform the design codes, these have been used as a guideline for the development of the system. Since DMC has an international working area both the German EAU\EAB and the Dutch CUR 166 Codes had to be implemented.

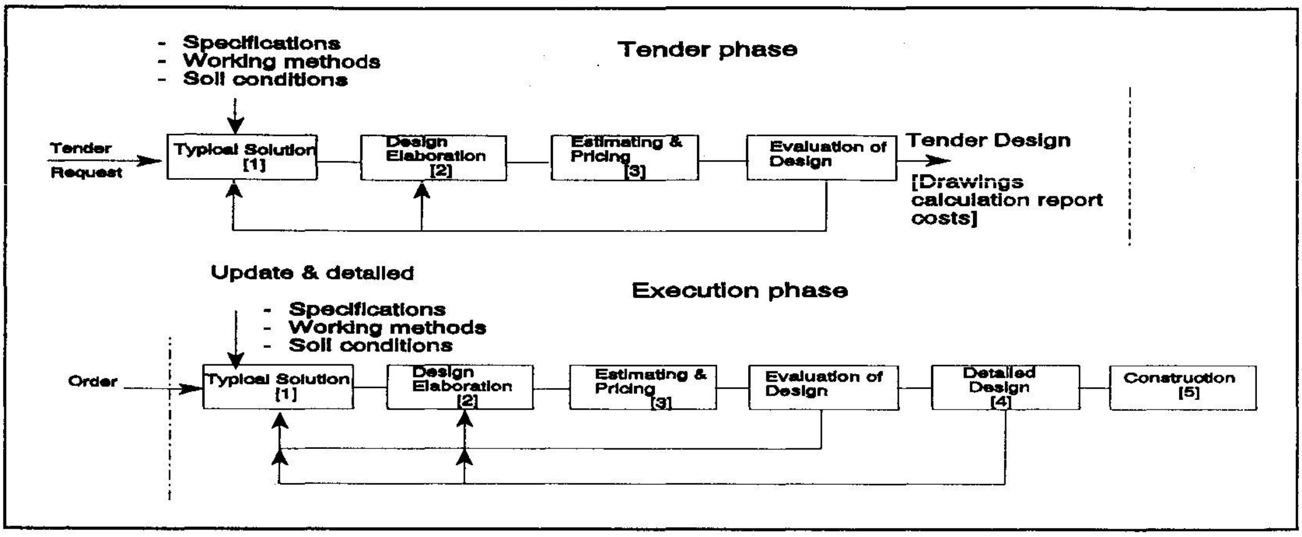


Fig. 2 Process scheme of the design of a cofferdam for a construction pit

4. APPROACH

4.1 Development method

Pit Design⁺⁺ is basically developed in discrete project steps. The following project phases are used: (1) proto-typing (2) defining system specifications (3) technical design, (4) programming and (5) implementation.

Proto-typing consists of elaborating a simple Design⁺⁺ model for a construction pit, containing all typical problems that may occur in the final version of the program. The development of the prototype learned that no serious problems were to be expected. Besides the realization it provided a large amount of information on the required effort to realize the final system and made it possible to quantify the running time reduction for construction pit designs. The prototype turned out to be a very useful tool to convince the organisation of the benefits of the project.

4.2 Specifications

Section 3.2 stated that 80% of the design tasks had to be covered by the system. This requirement led to the following main specifications:

- no limitations to construction pit lay-out
- six typical solutions to cross-section of the construction pit walls to be elaborated (see figure 3)
- computerized design of a sheetpile wall, anchors, anchor screen and supporting frame including computerized optimization
- different types of walls (combi-wall, sheetpiles, Peiner-wall, diaphragm wall, etc.)
- schematic presentation of standard details
- all other design tasks that can reasonably and practically be computerized will be programmed.

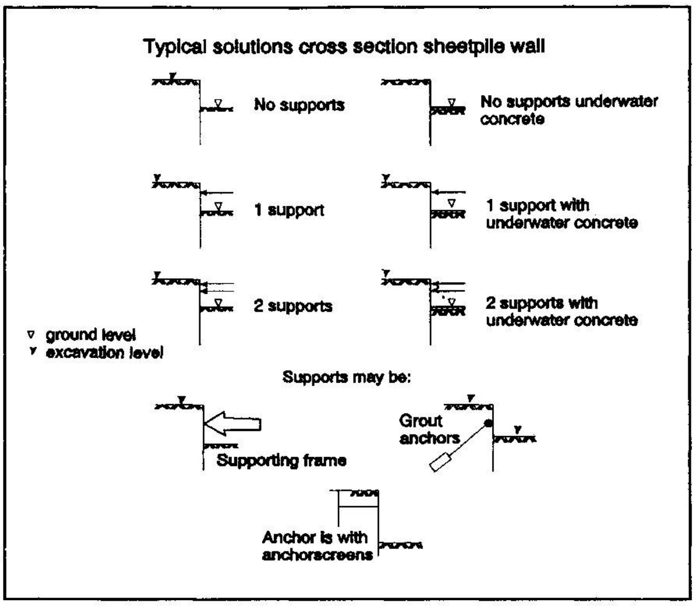


Fig. 3 Typical sheetpile walls as selected

4.3 Knowledge acquisition

Designers and estimators needed to be involved to supply the information to establish knowledge rules. In eight sessions with six specialists consisting of both designers and estimators it was decided which design checks had to be included in the system and the sequence of the design steps. It appeared that on various subjects the engineers had different opinions where the design process was concerned. The consensus reached on a standard design process, reported in a 300 page report on knowledge rules, was already an important result of the project.

It appeared that none of the specialists involved was reluctant to give his knowledge. They considered that the system would take over the boring design tasks so that they would have more time to do creative tasks.

5. LESSONS LEARNED

5.1 Design system

In Section 2 it was stated that Design^{++} can be used as a configuration system. During the project it appeared, however, that the design of a sheetpile wall for a construction pit is very complex since all the objects of the structure such as sheetpile wall, girders, support beams, anchors have many interdependent relations. This means that the reasoning system must execute a complex iterative solution process, making it difficult for the developer to guarantee the integrity of the data of the product model in each stage of the design process.

5.2 Development costs

The development of Pit Design^{++}, which cost several man years, confirms previous experience that high costs are involved in the developement of multi purpose engineering software within DMC.

5.3 Use of knowledge based design systems

It seems that in the construction industry only very few knowledge based systems are presently in use and that only few are under development.
Recent research within the industry has not led to similar cases of applied knowledge based technology in production environments. The experience with Pit Design^{++} shows, however, the possibility to build powerful systems leading to a considerable increase in productivity and quality if the application area is well chosen and a scenario for implementation is clearly defined.

6. LITERATURE

CLIVE L. DYM & RAYMOND E. LEVITT; Knowledge based systems in engineering; McGraw-Hill, Inc.; 1991

Integrating Knowledge-Based and Drawing Systems for Steel Construction

Intégration d'une base de connaissance et du projet assisté par ordinateur en construction métallique

Integration von Expertensystemen und CAD im Stahlbau

W. M. Kim RODDIS
Assoc. Prof.
University of Kansas
Lawrence, KS, USA

Dr W. M. Kim Roddis is Assoc. Prof. of civil eng. at the Univ. of Kansas. She is involved in research in expert systems and application of artificial intelligence to engineering and steel structures' behaviour . Member of expert systems committees and author of papers in the area.

Gregory P. PASLEY
Research Assistant
University of Kansas
Lawrence, KS, USA

Greg Pasley is a graduate research assistant in the Dept of Civil Eng. at the Univ. of Kansas. His research interests include collaborative engineering, shear behaviour of reinforced concrete, and artificial intelligence applications in structural eng. and engineering education.

SUMMARY
A computer-based tool, SteelTeam, is under development that integrates knowledge-based systems with a computer-aided design and drafting environment. SteelTeam is a communications tool that provides a medium for the electronic transfer of data in the steel industry. Through the use of knowledge-based systems, SteelTeam acts as an intelligent assistant to aid in the decision-making process, helping engineers understand the effects that early decisions have on the final product in order to avoid downstream conflicts.

RÉSUMÉ
Un outil informatique est en phase de développement: il combine un système basé sur la connaissance avec un environnement de projet et de dessin assistés par ordinateur. Cet outil de communication fournit un véhicule pour le transfert électronique de l'information dans l'industrie de l'acier. Il assiste intelligemment dans le processus de la prise de décision. Cela permet à l'ingénieur de comprendre les effets que les premières décisions ont sur le produit final et aide à éviter les conflits potentiels.

ZUSAMMENFASSUNG
Ein auf Computerbasis arbeitendes Hilfsmittel ist in Entwicklung, das intelligente Systeme mit rechnergestütztem Entwerfen verbindet. Es ist auch ein Kommunikationsmittel, das eine elektronische Datenübermittlung in der Stahlindustrie unterstützt. Durch die Anwendung intelligenter Systeme funktioniert es wie ein Assistent, der bei dem Entscheidungsfindungsprozess hilft. Die Anwendung gibt dem Ingenieur zu verstehen, welche Auswirkungen Entscheidungen in der Entwurfsphase auf das Endprodukt haben.

1. INTRODUCTION

In the design and construction industry, insufficient communication and coordination can lead to project failures and other problems. Because of the high degree of interdependency among the activities involved in collaborative design, the ability to communicate and coordinate the various members of the design team is crucial to the production of the best product and to the success of the project. This is especially true in the domain of steel building design, where each phase in the design, detailing, fabrication, and erection of steel buildings is normally carried out with little interaction between the participating parties. Much attention has been paid to this phenomenon occurring in the manufacturing sector, and has been informally characterized as engineering tossing completed designs "over the wall" to manufacturing.

A computer-based tool, SteelTeam, is under development that integrates knowledge-based expert systems (KBES) with a computer-aided design and drafting (CADD) environment. SteelTeam helps bridge communication gaps in the steel building industry by providing a communication tool for use by the various members of the collaborative design team. SteelTeam is used to electronically transfer data between the design engineer, detailer, fabricator, and erector. This improves the accuracy, efficiency, and completeness of the information used by each member of the design team. The inclusion of KBES makes expert advice available for informed decision making among all parties in the steel building industry. Thus the quality of the engineered product improves by making design, detailing, fabrication, and erection knowledge available to each member of the design team.

Several computer-based tools exist that integrate KBES and CADD. An early example of such a system is the LSC Advisor [1] that uses a KBES to assist an architect in making certain that floorplans are consistent with major fire safety code regulations. The KBES in the LSC Advisor operates on top of a geometrical database contained in a CADD system. Several systems have been developed that combine graphics and expert systems [2]. Both BERT and Evaluator use rule-based shells and CAD packages [2]. Many systems in existence have characteristics similar to the SteelTeam system. The PMAPM [3] system uses an object-oriented information model to facilitate information sharing in the design, construction, and management of a facility. The intelCAD system [4] integrates an object-oriented inference engine with AutoCAD [5]. Finally, the Interdisciplinary Communication Medium, ICM, [6] uses KBES and graphic modelling techniques to facilitate communication in collaborative conceptual design. SteelTeam combines these characteristics to create a computer-based tool to facilitate communication in the steel building project.

2. PROGRAM DESCRIPTION

Collaborative engineering requires that the knowledge of various team members be brought into the design process at the necessary times. In the steel industry this communication is not always possible because of the separation between the design, detailing, fabrication, and erection phases. Often the design is completed without input from the fabricator or the erector. This input, though desirable, is not always available because many times the fabrication and erection firms are not hired until the design is completed. The most exact and innovative design cannot overcome shortcomings that make the structure difficult to fabricate or erect. Therefore, close dialogue between the design-detail side and the fabrication-erection side of the steel building project is necessary to obtain an economical building design where all costs of producing the finished structure are considered. The quality of the communication and cooperation among the design team members directly affects the quality and value of the finished product. A problem exists when different parties in the design process are not aware of what suits other parties as the best solution to the problem being considered [7, 8]. Therefore, it is important to communicate design intent along with the design artifact.

SteelTeam runs under AutoCAD [5] on a PC-compatible platform. AutoCAD is chosen because of its widespread use in the engineering industry and because of the availability of AutoLISP as a built-in programming language. SteelTeam incorporates knowledge bases that represent the expertise of the various parties involved in the steel building design, detailing, fabrication, and erection. The KBES advise the user regarding fabrication issues, constructibility issues, design issues, section availability, and completeness of the design document. The presence of this knowledge allows SteelTeam to act as a decision support tool. SteelTeam helps the engineer understand the effects of early decisions, and through this understanding, aids the engineer in avoiding downstream conflicts. SteelTeam provides the knowledge needed to investigate design alternatives by presenting varying views where there are often conflicting goals.

The design information that is passed by SteelTeam is in the form of an object-based project model. The representation of the building's elements follows the object-oriented convention of encapsulating data structures along with the procedures needed to manipulate that data within objects. For example, the object representing a simple shear connection contains data slots for values of the reactions due to unfactored dead, live, and other loadings as well as a data slot for the value of the reaction due to the governing factored load combination, which may be generated automatically by the attached procedure. The SteelTeam project model must carry all the information used by the design team members to perform the tasks of design, detailing, fabrication, and erection of the steel building. The information includes all data that traditionally is passed between the design team members in the form of hardcopy drawings and specifications. This data is represented in SteelTeam by the electronic AutoCAD drawings with the augmented object-based model. In addition, information on issues of fabrication, constructibility, availability, drawing completeness, and other design areas are contained in the knowledge bases. For example, the fabrication knowledge base contains data on material and labor costs along with procedures to determine relative cost of such tradeoff options as using column stiffeners versus increasing column web thickness. These knowledge bases allow tailoring of their contents for regional and fabricator specific assumptions and preferences.

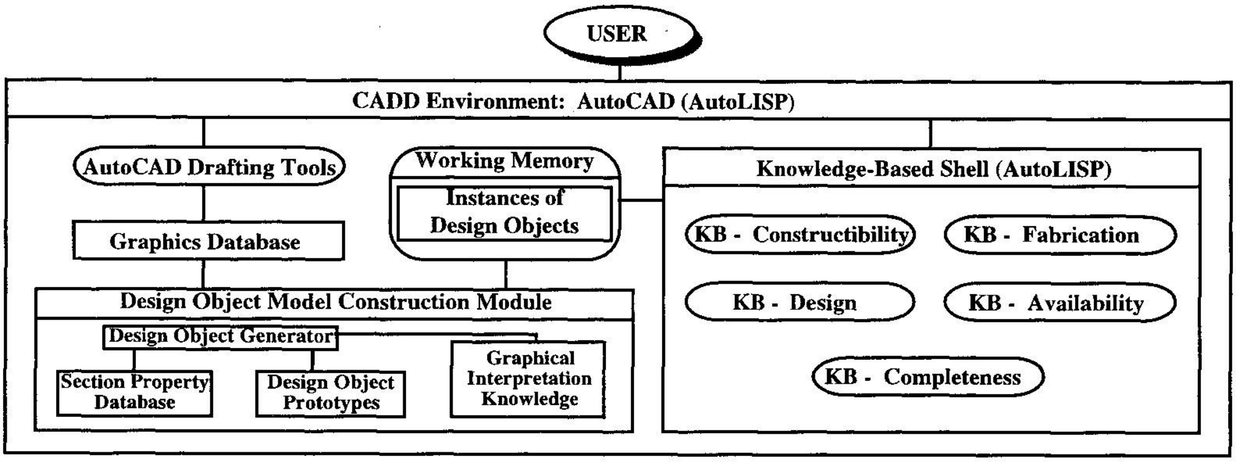


Fig. 1 SteelTeam system architecture

Figure 1 shows the architecture of the SteelTeam system. From within the AutoCAD drafting environment, the user may choose to either enter the design object model construction module or query the knowledge-based system shell. When first generating the design documents, the user enters the design object model construction module where the drawing entities and their associated attribute data are defined. This module contains the design object generator that assigns instances to the attributes that describe an object. The objects and their instances that make up the building model are stored in the working memory of the system. The building model consists of the collection of design instances that describe the beams, columns, connections, and other information needed to describe the structural configuration of the building. The second module that the user accesses is the knowledge-based system shell. This shell is a typical rule-based expert system shell implemented in AutoLISP. Only the component knowledge bases are illustrated in Figure 1. The knowledge bases cover constructibility issues (e.g. placement of column splices to avoid excessive erection costs), fabrication issues (e.g. notifying the user when a heavier section may be more economical than attaching stiffeners to a lighter section), design issues (e.g. bay layout of beams and girders for maximum economy), section availability (e.g. flagging sizes only available in large tonnage lots), and completeness (e.g. if highly skewed intersection on plan, then highly skewed connection detail should be addressed). The knowledge-based system shell accesses the working memory of the system to obtain information about the design being considered. The KBES is being enhanced to operate in a cooperative distributed problem solving (CDPS) architecture, allowing each knowledge base to function independently [9].

Current status of the knowledge-based shell only allows single or sequential operations of each knowledge base. In this single knowledge-base mode, the integration of CADD and the KBES has been explored for purposes of drawing checking, red-flagging possible problem areas. Details of the operation on a sample building may be found in [9]. The drawing of the building is checked for fabrication issues such as required coping and simple connection clearance issues. Currently, the fabrication KBES is being expanded and the other KBES are under construction. In the near future, the interactions between the different KBES will be explored using the CDPS architecture.

3. PRELIMINARY RESULTS

The benefits from the SteelTeam system include:

- The data exchange mechanism that allows transmittal of a more complete set of information. This increased access to better information allows more informed decisions to be made by all of the parties involved in the steel building design.
- The SteelTeam system performs a valuable completeness check on the documents that are being transmitted between parties.
- The expert knowledge that is available to the user in the form of the KBES provides the engineer with the ability to look ahead in the design process and avoid downstream problems and conflicts.
- SteelTeam allows the members of the design team to check the performance of alternative designs under the scrutiny of other experts.
- SteelTeam provides a two-way communication network for the members of the steel design team, allowing early and continued exchange of expert knowledge between the designer, detailer, fabricator, and erector.

ACKNOWLEDGEMENTS

This research is supported by the National Science Foundation under NSF Grant MSS-9221977 and the Center for Excellence in Computer-Aided Systems Engineering at the University of Kansas.

REFERENCES

1. DYM, C. L., HENCHEY, R. P., DELIS, E. A., and GONICK, S.,"A Knowledge-Based System for Automated Architectural Code Checking," *Computer-Aided Design*, April, 1988, pp. 137-145.
2. JAIN, D. and MAHER, M. L., "Combining Expert Systems and Cad Techniques," *Microcomputers in Civil Engineering*, December, 1988, pp. 321-331.
3. ITO, K., LAW, K. H., and LEVITT, R. E., "PMAPM: An Object Oriented Project Model for A/E/C Process with Multiple Views," *Second CIB W78 & W74 Joint Seminar on Computer Integrated Construction*, 1990, pp. 165-172.
4. CALVERT, T., DICKINSON, J., DILL, J., HAVENS, W., JONES, J., and BARTRAM, L., "An Intelligent Basis for Design," *IEEE Pacific Rim Conference on Communication, Computers and Signal Processing*, 1991, pp. 371-375.
5. AUTODESK, INC., "AutoCAD Release 12," Autodesk, Inc., 1992.
6. FRUCHTER, R., CLAYTON, M. J., KRAWINKLER, H., KUNZ, J., and TEICHOLZ, P., "Interdisciplinary Communication of Design Critique in the Conceptual Design Stage," *Computing in Civil and Building Engineering*, 1993, pp. 377-384.
7. RICKER, D. T., "Value engineering and steel economy," *Modern Steel Construction*, February 1992, pp. 22-26.
8. THORNTON W. A., "Designing for cost efficient fabrication," *Modern Steel Construction*, February 1992, pp. 12-20.
9. PASLEY, G. P. and RODDIS, W. M. K., "Using artificial intelligence to bridge communication gaps in the steel building industry," *Mid-America Conference on Intelligent Systems*, 1994, pp. 49-55.

Knowledge Support for Functional Design of Buildings

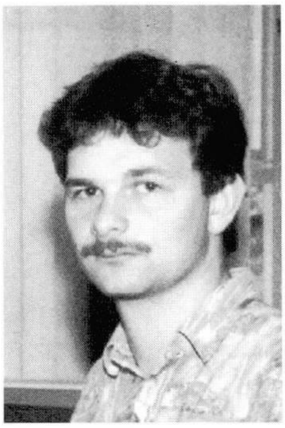
Système à base de connaissance pour le projet fonctionnel de bâtiments

Wissensbasierte Hilfsmittel für den funktionalen Gebäudeentwurf

R. HUEBLER
Prof.
Hochschule für Arch. & Bauwesen
Weimar, Germany

F. STEINMANN
Civil Eng.
Hochschule für Arch. & Bauwesen
Weimar, Germany

T. HAUSCHILD
Computer Scientist
Hochschule für Arch. & Bauwesen
Weimar, Germany



SUMMARY
Processes in the early stages of architectural design can hardly be mastered by common means of information science. A starting point for knowledge-based support of early stages of structural design is provided by linguistic models. Knowledge on the conceptual level, such as classification schemes and functional relations, is used as a pattern for construction elements, which has to be refined later. The system is able to apprehend the fuzziness of design objects and to evaluate it by means of a simple constraint system. The sub-system which provides function plans is developed as a prototype.

RÉSUMÉ
Le processus de projet architectural, dans sa phase initiale, ne peut pas être maîtrisé avec les techniques habituelles de l'informatique. Un point d'accrochage, à ce stade, pour une aide au projet structural est fourni par des modèles linguistiques. La connaissance de schémas de classification et de relations fonctionnelles permet d'esquisser des éléments de construction, qui seront mieux définis par la suite. Le système permet d'appréhender des éléments flous du projet et de les définir au moyen d'un simple système contraignant. Un sous-système fournit des plans fonctionnels types.

ZUSAMMENFASSUNG
Prozesse in frühen Phasen des architektonischen Entwurfs können mit gebräuchlichen Techniken der Informatik nicht beherrscht werden. Einen Ansatzpunkt zur Unterstützung derartiger Prozesse liefern linguistische Modelle. Dabei wird konzeptuelles Wissen wie Klassifikationsschemata und funktionelle Relationen als Muster für Entwurfsobjekte verwendet. Diese anfangs abstrakten Objekte können später verfeinert werden. Entwurfsobjekte können unscharf beschrieben sein. Die Handhabung der Unschärfe erfolgt in einem einfachen Constraint-System. Prototypisch realisiert ist ein Teilsystem zum Erstellen architektonischer Funktionspläne.

Introduction

The motivation for the support of early stages of architectural design arises from their crucial influence on the quality and economy of a building. These design process stages are dominated by synthesis tasks, which are not supportable by traditional means of computer science. The reason is the great variety and fuzziness of decision influences and the apparent chaotic character of the process and lack of algorithms. For that reason the present contribution aims at the realization of a plausibility-saving support system, which accomodates the individuality of design work.

Characteristic of design

Current research in the field of design support is focused on the routine design. Such configuration models make the mastery of the ambiguous transformation problem 're- quirement - solution' possible. It works by the application of multiple selections from a storage of solution elements with backtracking. For the field of structural planning the mentioned techniques are able to provide design support for restricted or extensively prepared application domains. However, these narrow restrictions don't apply to ar- chitectural design in general.

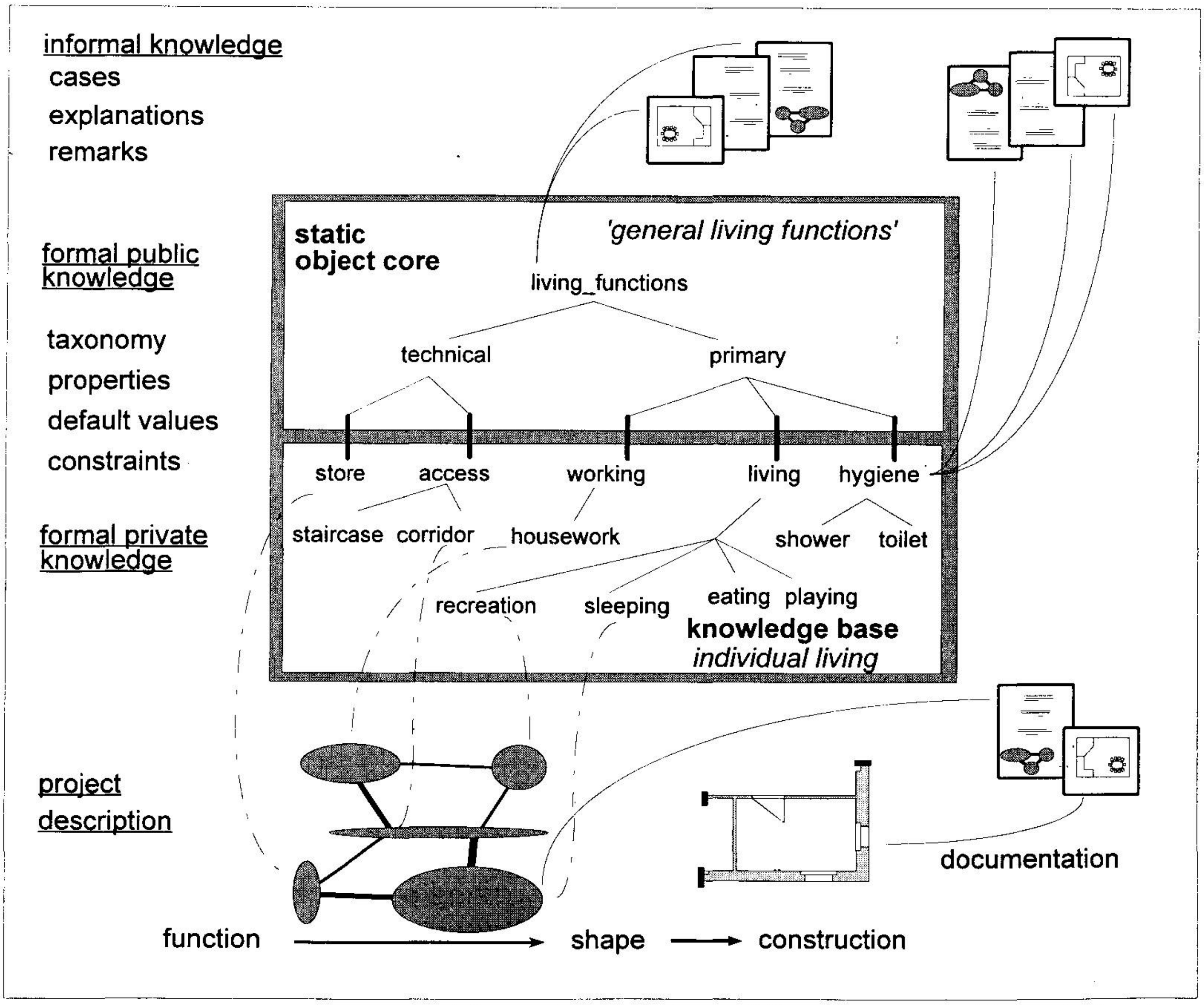


Figure 1 Structure of the taxonomical knowledge base

Most buildings are unique objects with a high measure of latitude for the designer. Comparatively, there are only a few global restrictions. Thereby, structural planning in this generalized sense is **innovative design** (/CHA/). A starting point for design supports in this field is offered by linguistic models (/BRO/). They proceed from the thesis that initially a design decision concerning the realization of a function designates a construction object to be created solely as type. For example: for a designated function 'making two floors accessible' an *functional element* is created, and it will be satisfied by a *realization element* of any kind of a staircase (an abstract object).
This approach is analogous to the human mode of expression. During the design process the development of more and more precise objects *(realization-elements)* occurs by a description of the construction element like for example an exact appearance of the staircase (concret object).

This concept is similar to object-oriented modeling in the field of software design. Thereby this form of a-priori knowledge bounded to a type allows on the conceptional level a representation in classes, from which instances may be derived which reflect the singular situation of a special design. In the iconic approach of design this a-priori knowledge is used by the designer as a pattern (concept) for construction elements which have to be refined later. In regard of the early stages of design this conceptual knowledge is restricted to the classification structure of the particular design domain and to basic functional relations between the embodied design objects.
The domain-specific design knowledge is provided by the system in two ways :

- *formalized knowledge*
  patterns for classification, structural correlations, default values, restrictions
- *informal knowledge*
  texts, pictures, examples, hypermedial information

Project *PREPLAN*

Design processes in practice are structured vertically in a sequence of design stages due to accountability and data-exchangeability (see /HOAI/). PREPLAN is a system of tools, which adds a horizontal organization of the design steps to the vertical stages according to some basic design actions. If the above-mentioned knowledge is used the focal point of design-support results in the following actions:

- the continuous application-specific concept refinement *(specification)*
- the instanciation of design pattern *(generation)*
- their consistency verification *(evaluation)*

They are supported by *informing* in the sense of a homogeneous integrated retrieval of informal knowledge (see Fig.1 and 2).
The following is valid for design objects in the supported design stages :

- design objects are fuzzy and are represented symbolically if neccessary
- design objects may be generalized **or** specialized during the design process
- default assumptions for design objects may be adopted
- there are restrictions for objects and relations between them
- aggregate objects were constructed according to changing strategies (top-down, bottom-up)

Design decision and the control of the design process remains with the designer.
By making this knowledge contextsensitively available through the support-system not only a decision base for the designer but also a facility for decision auditing on the basisof the specification is offered. In doing so, the basic item is the supply of formal knowledge

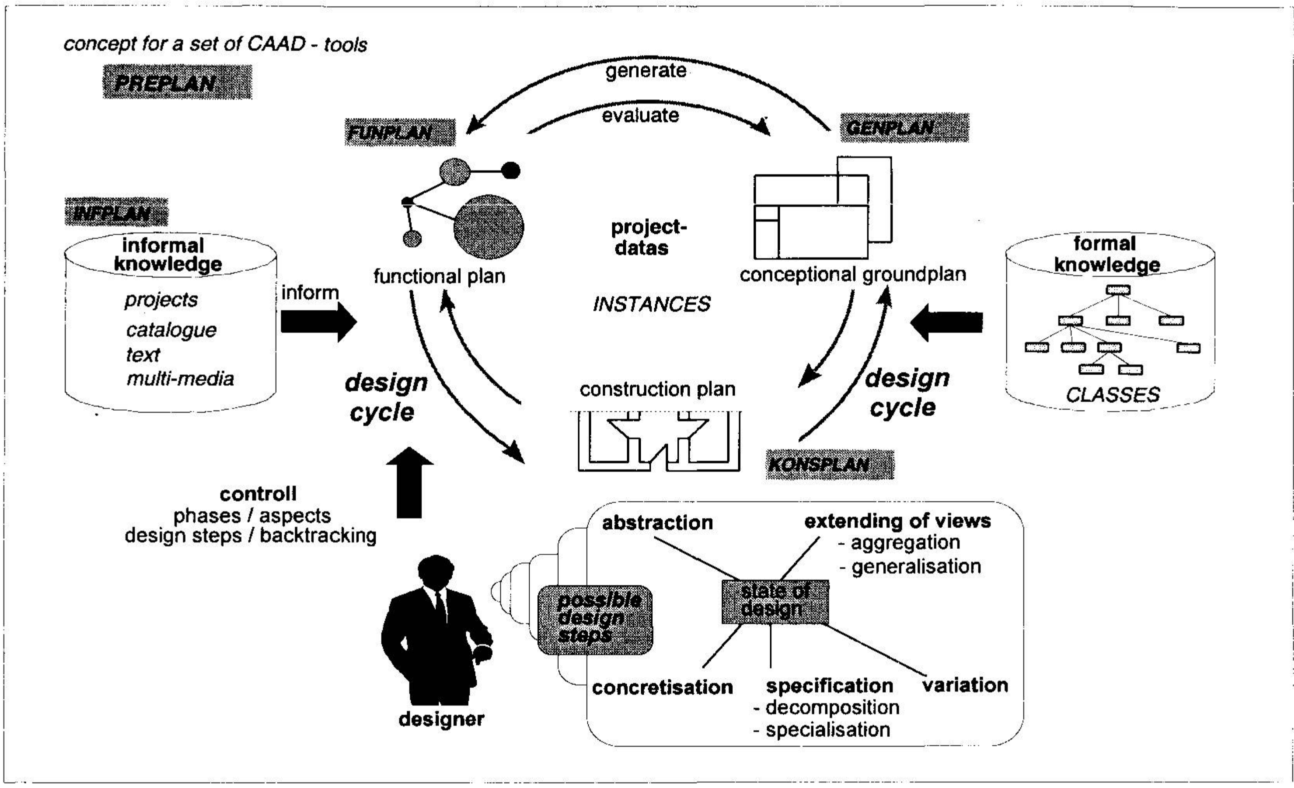


Figure 2 Passive design-support and design cycle

in the sense of a taxonomy of the design-scope in form of a class hierarchy. This is both the base for the instanciation of special design-objects and also the base for the contextsensitive informal knowledge storage.

Thereby it is assumed that general accepted concepts and notions may be preinstalled as a taxonomical base sphere. Refinements in the sense of a continuous application-specific specialization respectively individualization could be done by the architect (see Fig. 1).

In contrary to computations, whose specific algorithm has to be defined at compilation time, it is possible to realize the auditing of design decisions on the base of formalized restrictions at runtime. The restrictions themselfes have a purely descriptive nature, i.e. value-propagation is abandoned along such constraints. These constraints are administered in facets of attributes of objects, which participate in such relations. So far the attribute types *numerical, symbolical* and *relational* are distiguished in the constraint-system. To take into account the small commitment of the domain knowledge stored as those contraints, the constraints have to be relaxable (see Fig. 3).

In early stages of design fuzziness is an important feature of objects . For that reason, specialization and abstraction processes must be supported on the taxonomy level, as well as stepwise refinement at project level. Fuzziness must be administerable and processable by the tools for adequate design-support. The treatment of fuzziness occurs through application of methods of the object-core. The possibility of considering fuzziness in attributes with facets and to evaluate them by means of the constraint system seems to be a suitable modelling principle. The transmission of the fuzzy object characteristic into the project level takes place by inheritance of the class-related data-structure to the instances according to the object-oriented approach. In this way the the usage of equal methods for consistency-saving in the knowledge base and project storage is permitted.

Beyond that, procedures for analysis and visualization may work not only with the probable attribute values but also with crisp values.

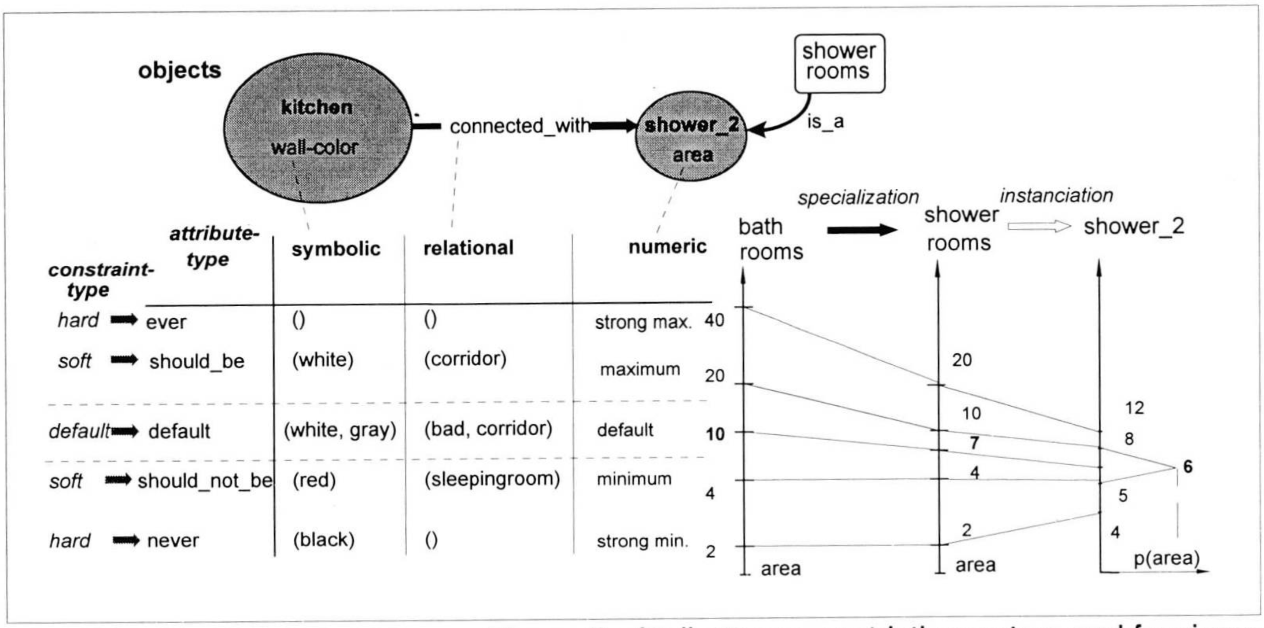


Figure 3 : Attributtypes, restrictionsystem and fuzziness

State of development

At present a prototype implementation of the subsystem FUNPLAN exists, which realizes the above mentioned functionality at the level of function plans. It was realized in a first version by utilization of the object-oriented development environment KappaPC(c). A AutoCAD-linkage for support of graphic communication facilities is available. These implementation was supported by a self-developed object system for AutoCAD (/ALOS/). The reference to the informal knowledge level is currently carried out only in a prototypical way by some hypertext systems. The integration of a database for administering and archiving large projects is under investigation. First tests with FUNPLAN were performed for some application domains. /DON/.

References

/BRO/ Broadbent, G.:"Design in architecture"
Wiley&Sons, London 1973

/CHA/ Chandrasekaran,B.; Brown,D.C.: "Design problem solving"
Pitman publishing, London 1989

/DON/ Donath,D.; Steimann,F.; Ehlers,H.
'Funktionales Entwerfen auf der Basis taxonomischen Wissens', unpublished,
Hochschule fuer Architektur und Bauwesen Weimar 1993

/ALOS/ Steimann, F. :
'Experimentelles AutoLisp-Objektsystem ALOS - V 0.4 - Referenzhandbuch"
Hochschule fuer Architektur und Bauwesen Weimar ,1993

/HOAI/ German Regulations on fees for engineers and architects

Leere Seite
Blank page
Page vide

An Approach to the Integration of the Design Process

Une méthode pour l'intégration du processus du projet

Ein Ansatz zur Integration des Entwurfprozesses

Olga RIO
Dr Eng.
Instituto Eduardo Torroja
Madrid, Spain

Marina ALVAREZ
Dr Arch.
UPM
Madrid, Spain

Roman JIMENEZ
Computer Scientist
Instituto Eduardo Torroja
Madrid, Spain

Olga Rio, born 1955, obtained her doctorate in civil engineering at the University of Madrid. She has been working in structural analysis at the Torroja Institute for the past 10 years.

Marina Alvarez is professor at the Computer Science Faculty of the University of Madrid. She has been working for the past 5 years in CAD systems in construction.

Roman Jimenez, born 1966, He has finished the studies at the Computer Science Faculty of the University of Madrid. He has been working 8 years at the Torroja Institute software system.

SUMMARY
A new approach to the integration of the design process is presented. The model considers not only tools that could be solved by means of the traditional computation science, but manifests the necessity to use other techniques useful in representing the knowledge of the experts.

RÉSUMÉ
L'article présente une nouvelle méthode pour l'intégration du processus du projet. Le modèle emploie des outils qui peuvent être traités par l'informatique traditionnelle. Il envisage aussi l'usage d'autres techniques qui sont très utiles pour la représentation de la connaissance.

ZUSAMMENFASSUNG
Es wird eine neue Möglichkeit vorgestellt, die Arbeit aller am Entwurfsprozess Beteiligten zu integrieren. Dieses Modell nützt nicht nur die Möglichkeiten der traditionellen Computerwissenschaft, sondern zeigt auch die Notwendigkeit auf, neuere Techniken zu verwenden, die sich besonders zur Verarbeitung von Expertenwissen eignen.

1. INTRODUCTION

Nowadays design process is developed in a very fragmented succession of activities and in spite of a lot of computer applications have been carried out to improve the work, due to the advances in hardware and software, these applications imitate this fragmented scheme. In other words, the philosophy has not changed: each area has better tools than before but there is no connection among them. Since the 80's a new philosophy has existed: the idea is to integrate the building process during design and later throughout the building's life cycle, giving the computer a more active role in the entire process of designing Gero[3], Eastman[2], Luiten et al.[4]. All these works have a common pourpose that is to obtain the integration of the design process, but the principal difference among them lies in the way in which this integration is reached.

From the analysis of these works the following points can be drawn out if the integration of the design process would be obtained:

- It is neccessary to define a *conceptual model* for structuring all data about a specific building, to be used in design, production and maintenance.
- It is neccessary to store the information contained in the conceptual model in one or more *databases*. Nevertheless, these databases are only able to store objects and their characteristics (shapes, sizes, physical properties, materials used, etc).
- It is neccessary to store the experience of the different partners that intervene in the process, by means of *knowledge bases*, in order to make the system able to take decisions and to adquire knowledge.

This paper describes the study of the different problems which arise as a result of the definition of a conceptual model for integrating the building design process.

2. DESIGN PROCESS MODELLING.

The construction process must be understood as a set of stages that must be executed to obtain the final result. For this reason the process and its stages must be perfectly defined. It will be necessary to take into account each stage of the process separately. In this way a structured organization of the data handled by each stage could be obtained. Once these sets of data exist they must be integrated in objects to the greatest possible extent.

Each object would have different views of the process. Thus the integration of all the data handled by the process will be possible in such a way that a change in one view will imply the change of the instance of the object as a whole.

The acquisition of knowledge precedes to the conceptualization stage. The general vision that we have of the problem depends in part on this stage. Once this stage is done we can pass to the following stage: the modelling, whose result will be the conceptual model, where the structured and integrated set of data handled by the process is represented. This conceptual model, all our knowledge of the problem will present.

The object oriented data bases would be a solution for implementing this conceptual model. In this way both the data structures and the functions of these data as soon as the inheritance properties could be integrated.

2.1. Analysis of the constructive process model

The design method that the architect uses could be emulated by means of an efficient structuring of the architectural process. An algorithmic process can be developed if this process is split into modules that define different stages. This algorithmic process could be solved by means of mathematical tools as the combining topology is. But on the other hand the process needs to have other information due to the set of objective as well as subjective requirements that will be stored in DKB that the process needs to use or modify depending on the stage it considers.

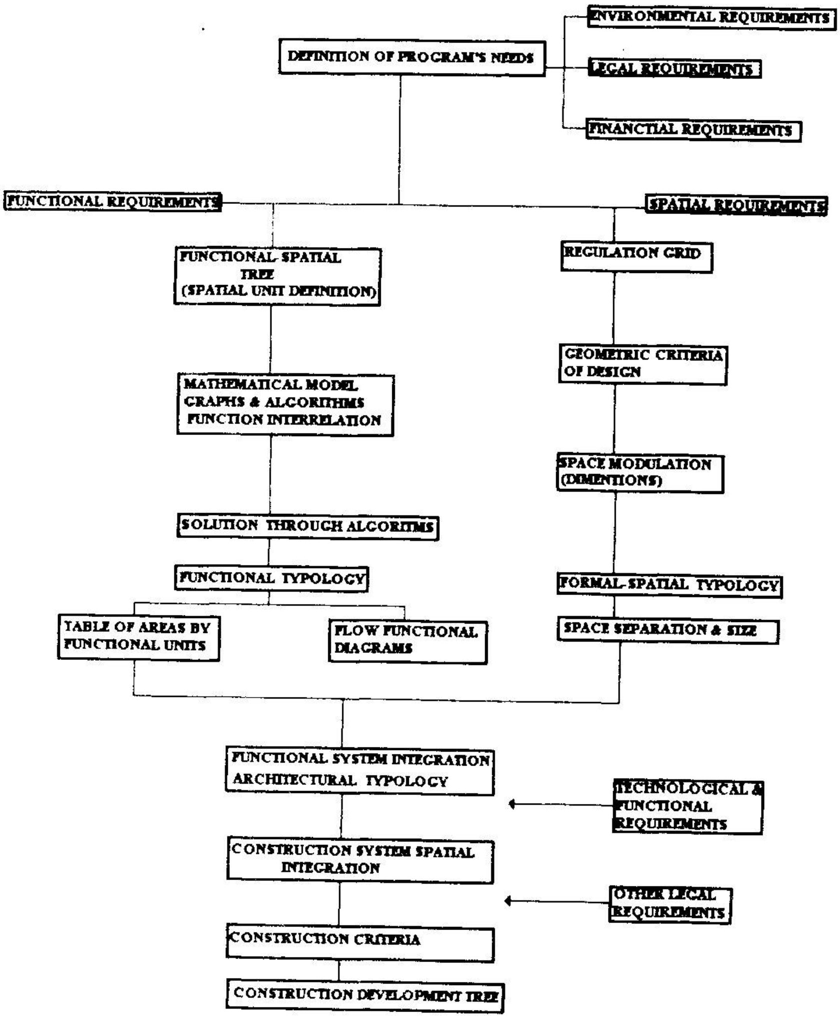


Fig.1 Structure of the Proposed Architectural process as well as the different modules.

In figure 1, the following modules can be distinguished: The module of definition of general program requirements. It is a function of the objective to reach (efficiency, functionality, comfort, etc) and the needs to settle. At this stage a kind of objective and quantitative requirements as can be the legal and environmental or physical ones, must be defined. These requirements could be stored in data bases. On the other hand, other kind of requirements, that have qualitative character must be considered. These last requirements cannot stored in a data base.

Once, the above mentioned stage has been developed, the second module can be developed. This second module, called the architectural design module can be split in two sub-stages Alvarez et al.[1]. In the first of them, the architectural spaces are generated taking functional requirements as a base. In this way the structure of the building can be defined and, as a consequence, the functional typology can be obtained. At the other sub-stage the space is modulated and geometrically structured, taking into account the spatial requirements. In this way the dimentions can be given, and the spatial typology is defined.

With the integration of both sub-stages the architectural typologies are defined, in which the functions are spatially represented and the morphological image is generically outlined. With both of them an architectural typology file will be created that will be stored in a data base.

The following stage, called the spatial integration of the constructive process, will be developed taking as a base a constructive development tree previously defined. Taking into consideration the technological and functional requirements that has been previously defined and which will be stored in a data base, the constructive system can be choosen. Then the technical calculations will be made with adequate tools and following the legal requirements of the standards of constructive and technical aspects. As a conclusion at this stage, a new data base is generated in which all the elements and components are stored. Lastly, both the functional-spatial stage and the construction development stage must be treated in conjunction to obtain the formal image of the building.

3.CONCLUSIONS

It is very important to bear in mind that the process cannot be implemented as a succession of stages executed in a sequential order only. It seems to be clear that in the integration of the construction process the concepts of subjectivity, personalization and individuality of each design must be dealt with.

It is neccessary to have a system capable of holding the knowledge that the experts integrate within the construction process and which allows going through the graph until reaching the final integrated design.

The interactions between different stages in certain cases may produce feedback between these stages that will originate changes in the products or objects handled during the whole process allowing the control and check of the eventual errors of design.

REFERENCES

1. ALVAREZ M., RECUERO A., Propuesta de Modelo Conceptual de una Vivienda Unifamiliar, ARECDAO 93, 1993, Barcelona pgs. 151-162
2. EASTMAN CH., Modelling of buildings: evolution and concepts, Automation in Construction 1, 1992, Pgs. 99-109
3. GERO JS., AI and New Computational Models of Design, IABSE Colloquium, Beijing, 1993, pgs. 25-34.
4. LUITEN G., FROESE T., BJÖRK B.C., COOPER G., JUNGE R., KARSTILA K., OXMAN R., An Information Reference Model For Architecture, Engineering and Construction, Management of Information Technology for Construction, 1993, Singapore. pgs. 391-406.