Zeitschrift: IABSE reports = Rapports AIPC = IVBH Berichte

Band: 68 (1993)

Rubrik: Session 0: Keynote

SESSION 0

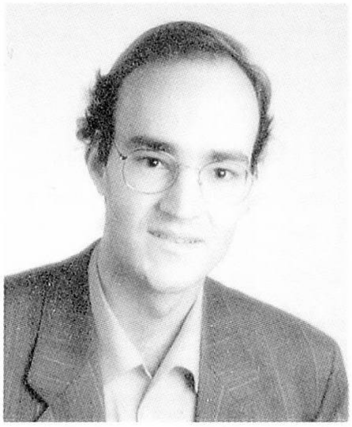
KEYNOTE

Reasoning Strategies for Engineering Problems

Stratégies de raisonnement dans les problèmes de génie civil

Strategien des Schliessens bei Ingenieurproblemen

Boi FALTINGS
Dir.AI Lab.
Swiss Fed. Inst. of Technol.
Lausanne, Switzerland



Prof. Boi Faltings received a diploma from the ETH Zurich and a PhD from the Univ. of Illinois, both in Electrical Eng. He founded the Artificial Intelligence Laboratory of EPFL in 1987. His research interests include intelligent design systems.

SUMMARY

Classical expert systems are based on deductive inference. However, most engineering problems require abductive reasoning. This paper discusses the problems caused by simulating abductive reasoning using deductive rules, and how the framework of model-based reasoning allows explicit implementation of abductive inference and thus avoids these problems. The model-based framework also makes it possible to use dependencies for efficient solutions to the problem of constraint relaxation. Model-based reasoning is thus not only useful as an efficient way of formulating knowledge, but also allows more powerful inference strategies.

RÉSUMÉ

Les systèmes experts classiques se basent sur le raisonnement déductif. La plupart des problèmes de génie civil exigent toutefois une manière de procéder plus abstraite, habituellement simulée par de simples règles déductives. Il faut aborder les problèmes qui en découlent par des réflexions se rapportant à des modèles, et qui permettent une mise en application explicite de la transmission par abstraction. Le cadre basé sur un modèle permet d'utiliser des relations pour résoudre le problème de la relaxation de conditions secondaires. Le raisonnement rapporté à un modèle spécifique n'est pas seulement une manière efficace de formuler les connaissances, mais de permettre également des stratégies de transmission plus performantes.

ZUSAMMENFASSUNG

Klassische Expertensysteme basieren auf dem deduktiven Schliessen. Die meisten Ingenieurprobleme erfordern jedoch ein abstrahierendes Vorgehen, das üblicherweise lediglich mit deduktiven Regeln simuliert wird. Es wird gezeigt, wie daraus entstehende Probleme durch auf Modelle bezogene Überlegungen umgangen werden, die eine explizierte Implementierung der Übertragung durch Abstraktion gestattet. Der modellbasierte Rahmen macht es auch möglich, Abhängigkeiten für eine effiziente Lösung des Problems der Lockerung von Nebenbedingungen zu nutzen. Modellbezogenes Schliessen ist daher nicht nur eine wirkungsvolle Art, Wissen zu formulieren, sondern es erlaubt auch leistungsstärkere Strategien der Übertragung.

1 Reasoning Strategies

A very general tool for modeling knowledge and reasoning on computers are *inference rules* taking the form:

```
conditions ⇒ conclusion
```

meaning that whenever `conditions` are given, `conclusion` is also true. The formulation of knowledge as inference rules originated in research on human psychology and was proposed as a formalism for computer programs by Newell and Simon ([5]).

The most natural way to apply inference rules is by *deduction*. A deductive inference engine is a computer program which starts with a set of premises - presumed to be true - and iteratively applies inference rules to add new conclusions to this set of known facts. Rules engines for expert systems often distinguish between *forward* and *backward* chaining, where backward chaining means that inferences are guided to lead to particular *goals*.

Deductive inference has been proven to be *Turing-equivalent* ([4]), meaning that any computation which can be carried out on a digital computer can also be achieved using deductive inference. This may become intuitively clear by seeing that a FORTRAN statement of the form:

```
C = A*A + B*B
```

can be translated into a deductive rule:

```
(∀ x) (∀ y) (A = x) ∧ (B = y) → (C = x*x+y*y)
```

which can be applied as soon as the values of A and B are known.

However, deduction is not the only form of logical inference. Consider the following propositions and rule:

```
a) bird(Tweety)
b) flies(Tweety)
c) (∀ x) bird(x) ⇒ flies(x)
```

Three types of inference are possible between these elements, depending on which of them is desired as a conclusion:

- **deduction**: a), c) $\rightarrow$ b)
  the conditions and the rule justify the conclusion.
- **induction**: a), b) $\rightarrow$ c)
  the rule is inferred from observing the example of a bird that flies.
- **abduction**: b), c) $\rightarrow$ a)
  the condition of the rule is inferred to explain the conclusion.

Now consider the typical engineering activities:

- *analysis* =
  find the performance of a given structure: **deduction**
- *diagnosis* =
  find causes that explain given symptoms: **abduction**
- *design* =
  find a structure that satisfies given specifications: **abduction**
- *learning* =
  find a rule that summarizes given observations: **induction**

The surprising conclusion is that many of the activities in which engineers hope to use knowledge-based systems in fact require not deductive, but abductive and inductive reasoning! It is therefore worthwhile to examine the properties of these other kinds of inference.

2 Abductive and Inductive Inference

Abduction and induction are distinguished from deduction by the fact that they usually produce *ambiguous* answers. For example, given the rules:

a) `poor-drainage` ⇒ `excessive-staining`
b) `low-quality-concrete` ⇒ `excessive-staining`
c) `insufficient-covering-of-reinforcement` ⇒ `excessive-staining`

abduction gives three different explanations for the premise `excessive-staining`, corresponding to the rules `a)`, `b)` and `c)`. Different explanations are distinguished only through *corroboration* with additional information, possibly also obtained by abduction. For example, another abductive inference:

d) `poor-drainage` ⇒ `wet-pavement`
e) `humid-climate` ⇒ `wet-pavement`
+ assertion: `wet-pavement`

results in two solutions of which one, `poor-drainage`, is in agreement with one of the choices for the first abductive inference, and gives reason to select it over the other candidates.

Similarly, ambiguities arise in induction because there are usually many rules which fit a given set of observations. Thus, the examples:

`Bridge-27:` {`poor-drainage,freeway,excessive-staining`}
`Bridge-34:` {`poor-drainage,multiple-simple-spans,excessive-staining`}
`Bridge-53:` {`poor-drainage,multiple-simple-spans,freeway,excessive-staining`}

could justify any combination of the following rules:

a) `poor-drainage` ⇒ `excessive-staining`
b) `freeway` ⇒ `excessive-staining`
c) `multiple-simple-spans` ⇒ `excessive-staining`

The ambiguities must be resolved by *refutation*: observing counterexamples to the hypothesized rule.

In fact, the occurrence of ambiguities is the main motivation for using *symbolic* or *qualitative* models for abductive or inductive inference: numerical models would often result in infinite sets of choices which cannot be dealt with in a computer algorithm. It is thus not surprising that knowledge-based systems are an attractive technology for activities which require inductive or abductive inference: learning, diagnosis and design.

3 Implementing Abductive Inference

Although abduction is one of the main motivations for applying knowledge-based systems in engineering, classical expert systems are based on *deductive* inference only, since deduction is most straightforward to implement in an algorithm. Using a deductive system for abductive tasks such as diagnosis means that abduction must be *simulated* using deductive rules. This is carried out most easily by inverting rules defining the knowledge:

`poor-drainage` ⇒ `excessive-staining` is transformed into:
`excessive-staining` ⇒ `poor-drainage`

However, this conversion cannot express the ambiguity which arises when several rules could explain the same observation. To distinguish different possibilities, many expert systems use *certainty factors* or similar measures which estimate the likelihoods of candidates.

Such certainty factors could be computed on the basis of the absolute probabilities that candidates are in fact present. More precisely, given a set of rules:

a ⇒ x
b ⇒ x
c ⇒ x

a set of a priori probabilities p(a), p(b) and p(c) that a, b or c are the correct candidates, and the assumptions that:

- the propositions a, b and c are mutually exclusive.
- there are no other possible explanations for x (closed-world assumption).

one can follow the principle of Bayes and construct a *probabilistic* set of inference rules where the conclusions are asserted to be true with certain probabilities:

$$x \Rightarrow a,\ p = \frac{p(a)}{p(a)+p(b)+p(c)}$$
$$x \Rightarrow b,\ p = \frac{p(b)}{p(a)+p(b)+p(c)}$$
$$x \Rightarrow c,\ p = \frac{p(c)}{p(a)+p(b)+p(c)}$$

Even though many expert systems do not explicitly follow such a construction, the heuristic certainty factors present in systems such as MYCIN ([1]) are an attempt to approximate such inference and thus they are subject to the same limitations[1]. When an assertion is corroborated - asserted through a different inference - its certainty factor is increased correspondingly to reflect this added degree of confidence.

Thus, assuming the probabilities:

```
P(poor-drainage) = 0.1
P(low-quality-concrete) = 0.15
P(insufficient-covering-of-reinforcement) = 0.25
```

the knowledge about excessive-staining can be transformed into the following deductive rules:

```
excessive-staining ⇒ poor-drainage (CF = 0.1/0.5 = 0.2)
excessive-staining ⇒ low-quality-concrete (CF = 0.15/0.5 = 0.3)
excessive-staining ⇒ insufficient-covering-of-reinforcement (CF=0.25/0.5=0.5)
```

The assumptions underlying the simulation of abduction through deduction, however, lead to significant difficulties. First, there is no correct method for combining certainty factors which can take into account interdependence between inference rules. Consequently, it is not possible to guarantee that the results of the inference are always correct. Second, the different possibilities are usually not mutually exclusive. For example, there may well be several causes for one and the same problem. The deductive framework provides no reliable way for dealing with multiple solutions.

Third, the closed-world assumption underlying the construction of the rules is put into question as soon as new knowledge is discovered and has to be added to the system. For example, imagine that it is newly discovered that overstressing causes excessive cracking which in turn causes excessive staining. This could be expressed as a rule:

```
overstressing ⇒ excessive-staining
```

But this means that *all* certainty factors involving excessive-staining have to be revised. Assuming that the probability of overstressing is P(overstressing) = 0.1, the revised rules would now read:

```
excessive-staining ⇒ poor-drainage (CF = 0.1/0.6 = 1/6)
excessive-staining ⇒ low-quality-concrete (CF = 0.15/0.6 = 0.25)
excessive-staining ⇒ insufficient-covering-of-reinforcement (CF=0.25/0.6=5/12)
excessive-staining ⇒ overstressing (CF = 0.1/0.6 = 1/6)
```

[1]The construction given here should not be confused with the technique of *Bayesian networks*, which perform abduction with probabilistic knowledge.

Especially when certainty factors have been obtained through heuristic estimates and tuned so that the system gives the correct answers, such a revision can be a very expensive, if not impossible, task. The limitations become completely inacceptable when rule sets are incomplete and have to be modified while the system is being used.

It is therefore desirable to look for other ways of implementing abductive inference that do not give rise to such problems. This is a primary motivation for *model-based reasoning.*

4 Abductive Inference in Model-based Reasoning

Knowledge about physical systems is generally available in the form of *models.* A model of a device or a part thereof is expressed by a simulation rule of the form:

`cause` $\Rightarrow$ `effect`

When knowledge is formulated as models, tasks such as diagnosis and design require abductive inference. In fact, during the previous discussion in this paper, it was tacitly assumed that knowledge was given in the form of models.

While classical knowledge-based systems *compiled* models into deductive rules, *model-based reasoning* allows using models directly without the need for such compilation. Model-based reasoning systems have many advantages over heuristic expert systems:

- knowledge is straightforward to formulate and maintain.
- the results can be guaranteed to be correct whenever the underlying models are correct.
- combinations of multiple solutions can be found.

Among users of the technology, the prime motivation for model-based reasoning has been the ease of formulating knowledge. However, as we shall now see, the explicit implementation of abduction in model-based reasoning systems also offers significant advantages from the computational point of view, namely guarantees of correctness and the ability to generate combinations of solutions. Model-based reasoning (MBR) has therefore become increasingly successful in recent years.

The general problem of abductive inference in the context of a MBR system can be stated as follows:

> Find all sets of combinations of causes $\{C_1, C_2, ..., C_k\}_j$ which logically entail *all* of the observed effects:
> $\{C_1, C_2, ..., C_k\}_j \vdash \{E_1, E_2, ..., E_n\}$.

This problem can be solved by inverting all rules which allow the inference of an effect E_i to generate the set of individual causes $\{C_j, C_l, ...\}$ which entail E_i. The set of potential solutions is then the set of all combinations of causes such that at least one possible cause for each E_i is contained in the combination. However, this set will contain enormously many solutions, making the problem intractable for practical problems. This is in fact one important reason for constructing heuristic expert systems.

As an example, consider the problem of diagnosing failures of an overhead projector using the models of the device shown in Figure 1. Given the problems

$\neg$`image-lit`: the image is not lit, and
$\neg$`hum`: there is no hum

abduction would first consider the 5 candidate combinations:

a: { $\neg$`proj-power` }
b: { $\neg$`proj-power` $\wedge$ `bulb-broken` }
c: { $\neg$`proj-power` $\wedge$ `fan-broken` }
d: { `bulb-broken` $\wedge$ `fan-broken` }
e: { $\neg$`proj-power` $\wedge$ `bulb-broken` $\wedge$ `fan-broken` }

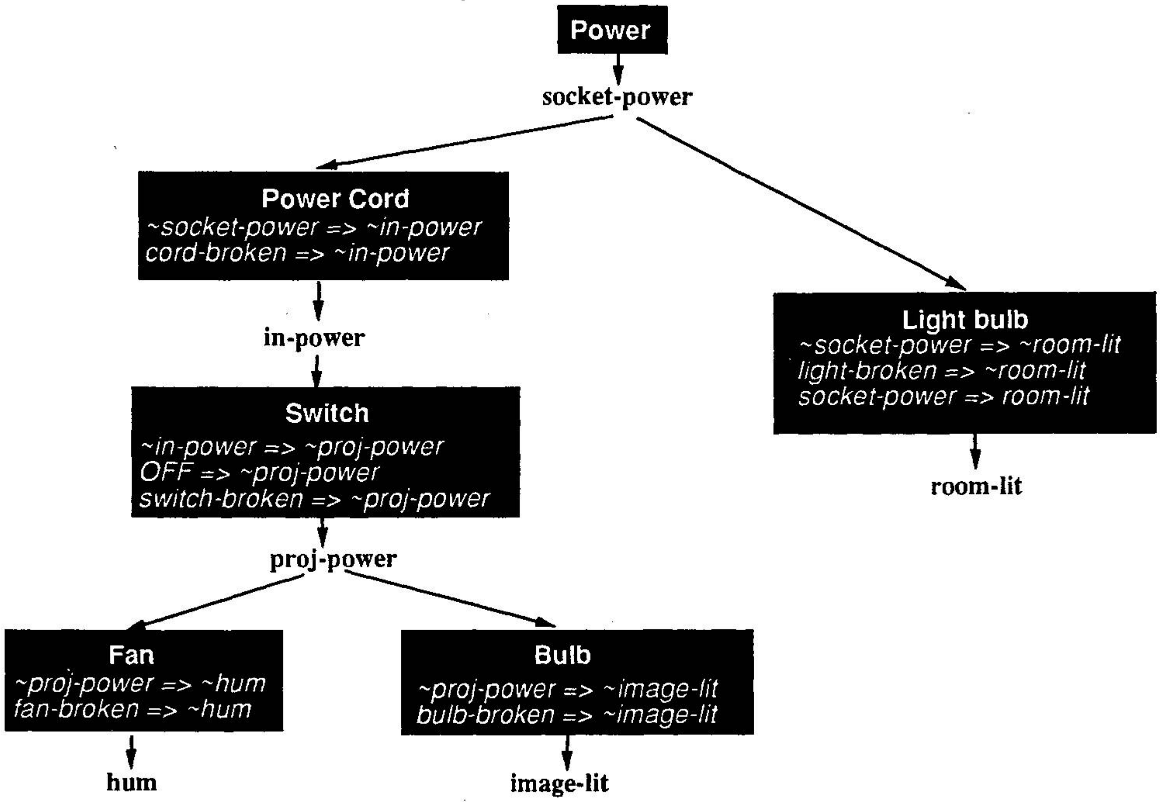


Figure 1: *Models used for diagnosing a projector.*

where candidate a) is of course the most likely one. By recursive abduction, ¬proj-power can in turn be explained by any of the $2^4 = 16$ combinations of

`switch-broken`, `OFF`, `cord-broken`, `¬socket-power`

This means that there are altogehter $5 \cdot 16 = 80$ candidate combinations of causes to be searched. In an example of practical size, there may be thousands of candidates, resulting in an unmanageable complexity.

DeKleer observed ([2]) that the set of solutions could be described by specifying only *minimal* combinations which are required to entail the given conclusions. Any solution to abductive inference is in fact a superset of one or several such minimal combinations. This observation and its realization in an associated reasoning engine, the assumption-based truth maintenance system (ATMS), have been the basis for the practical success of model-based reasoning.

For the example of diagnosis, the intuition behind DeKleer's observation can be explained as follows. Assume that causes for failure are modelled by giving the faulty component, and that the set $\{C_1, C_2, C_3\}$ of faulty components entails all observed failures and is thus a solution to the diagnostic problem. Imagine now a fourth component C_4 which is really faulty, but its fault is masked by the faults of C_1, C_2 and C_3. Obviously, $\{C_1, C_2, C_3, C_4\}$ is also a solution to the diagnostic problem. In fact, since any component could potentially play the role of C_4, *any* superset of $\{C_1, C_2, C_3\}$ is a diagnosis. The very large space of potential diagnoses can be represented by the minimal candidates only, often an extreme economy. In the example of the projector failure, the space of 80 candidate combinations obtained from the symptoms ¬image-lit and ¬hum can thus be represented by the minimal candidates:

{ bulb-broken ∧ fan-broken }, { switch-broken }, { OFF }, { cord-broken }, { ¬socket-power }

Contrary to systems based on deductive rules which map symptoms directly into faults, it is now straightforward to reason about combinations of multiple faults. Furthermore, it is possible to bring

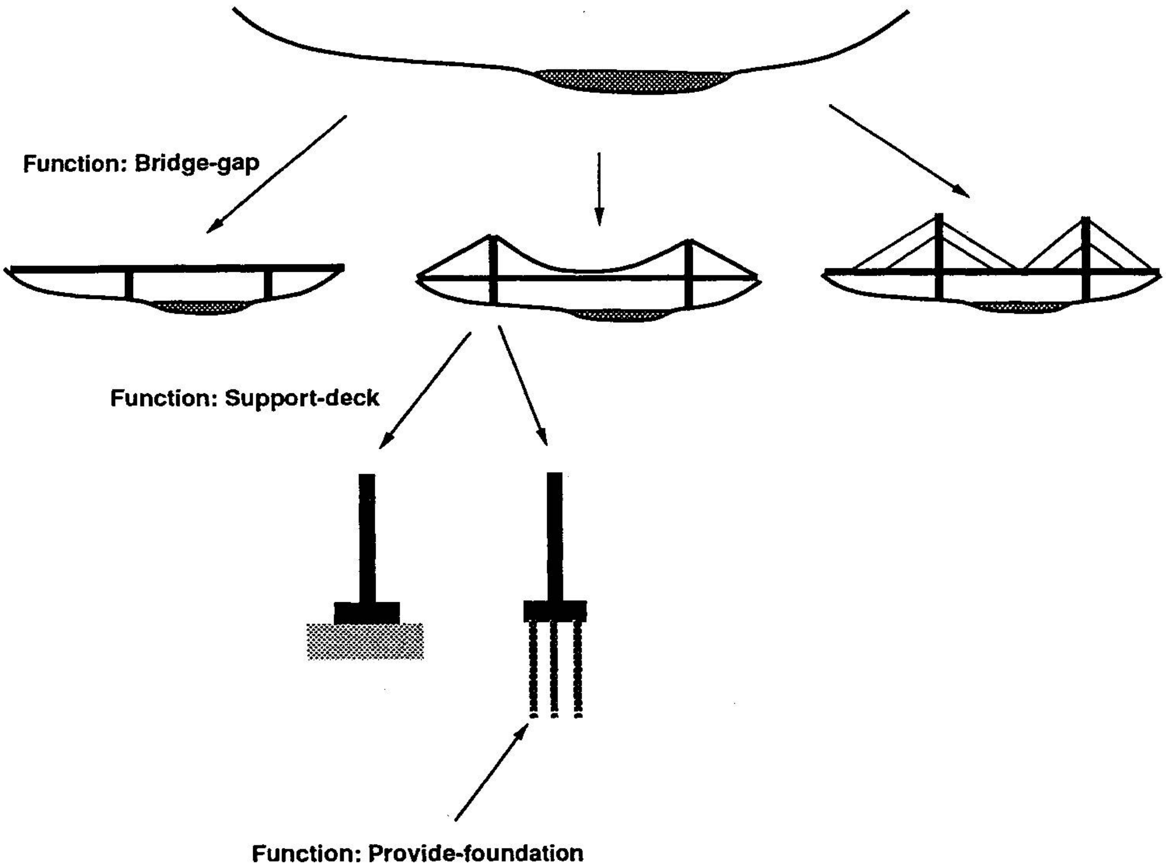


Figure 2: *Designing a bridge by functional decomposition. Shown here are two functions: stable support and providing the deck.*

new elements into consideration without profound changes to the knowledge base. For example, the fact that the room lighting is usually fed by the same electric circuit can be added to the knowledge base as the model of the light bulb, as shown on the right in Figure 1. Once this model has been added, the observation that ¬`room-lit` can be abduced to ¬`socket-power` and give this candidate a much higher probability. Conversely, the observation `room-lit` allows the abduction `socket-power` which rules out the conflicting candidate ¬`socket-power`.

The use of minimal candidates has been proposed as the key idea of a program called the General Diagnostic Engine (GDE, [3]) which uses an assumption-based truth maintenance system (ATMS) as a tool which generates the set of all minimal candidates in parallel. Since then, it has been shown that many abductive inference problems in diagnosis and design can also be solved efficiently using a sequential search, but nevertheless maintaining the advantage of computing with minimal combinations only. In general, when abduction can be applied recursively to arbitrary depths, as is common in design, the space of potential solutions is infinite and cannot be obtained using an ATMS, requiring instead sequential search.

Explicit abduction based on models has also been used in design, but with a less systematic approach due to the fact that design not as well-defined as diagnosis. Systems that perform design by functional decomposition, such as VEXED ([6]), perform abduction on rules of the form:

structure $\Rightarrow$ function

An example of how a bridge design might be obtained by such an abductive system is shown in Figure 2. The first goal to be abduced is that of providing a deck, which can be achieved by one of three bridge types. Depending on the solution chosen for the bridge, the second stage of abduction selects possible types of pier along with its foundations. In parallel, the goal of providing stable foundations

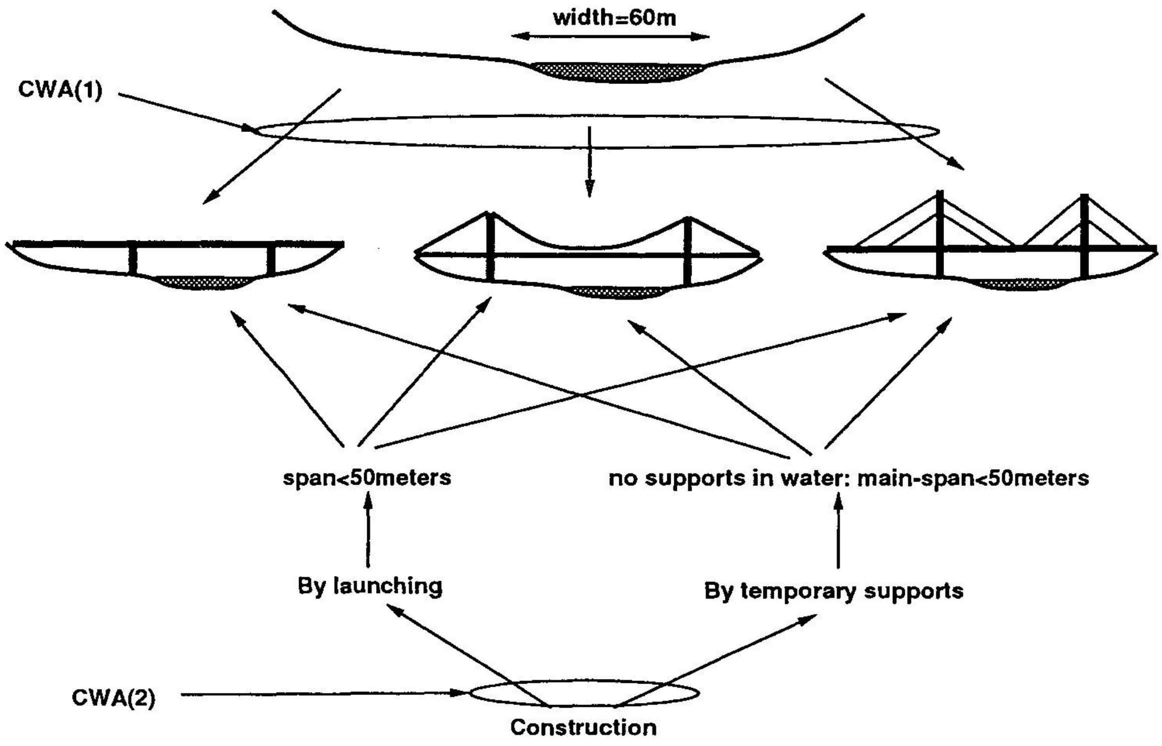


Figure 3: *Example of a conflict between conflicting requirements in a bridge design. The source of the conflict can be traced to one of two closed-world assumptions.*

leads to abduction of another set of choices on pier foundations. Since only one of them can be chosen, possible pier foundations are given by the intersection of the two sets.

Applying abduction in design is subject to complexity problems which are much worse than those observed in diagnosis. This is because the space of possible structures is usually very large, if not infinite, making the space of potential solutions impossible to search completely. Furthermore, because of compatibility constraints between components, the space is not *monotonic*: while a combination of structures $\{S_1, S_2\}$ can have function F, the superset $\{S_1, S_2, S_3\}$ may not have it due to interference of component S_3. The idea of minimal combinations is therefore far less useful in design than it is in diagnosis.

5 Using Explicit Closed-World Assumptions

The validity of an abductive inference depends crucially on a *closed-world assumption* (CWA) that there exists no other rule, unknown to the system, by which the observation could be obtained. The fact that a closed-world assumption is violated becomes obvious when the system does not find a solution, or when the solution proposed is wrong. In classical expert systems, it would be an extremely difficult problem to determine *why* the system did not find a better solution. In model-based reasoning, however, this can be solved more easily by explicitly representing the closed-world assumptions underlying the reasoning.

As an example, consider the design of a bridge across a river which is 60 meters wide (Figure 3). Assume that abduction to provide the main function - a deck spanning the river - results in three different bridge solutions shown in the figure, and a closed-world assumption CWA1 denotes the assumption that there are no other bridge types. On the other hand, designing the construction methods may leave only construction by launching and by using intermediate supports. The closed-world assumption CWA2 denotes the assumption that there are no other construction methods which apply to this problem. When no intermediate supports may be placed in the water, there is no construction method which is compatible with any of the proposed bridge types. This could mean that there is in fact no solution to the problem. However, it is more likely that the designer should look for other bridge designs or construction methods which do not have this conflict. This amounts to questioning one of the two closed-world assumptions, CWA1 and CWA2, which underly the contradiction.

A model-based reasoning system can use explicit closed-world assumptions in order to pinpoint the sources of conflict in reasoning. For design, this effect could be obtained by adding to every abductive inference an additional possibility `CWAi-violated` which is never in conflict with any other part of the solution. Solutions which contain such possibilities indicate solutions that would exist given additional possibilities. The designer can thus *choose* whether and where to look for innovative solutions in order to improve the design.

In model-based diagnosis, the use of explicit closed-world assumptions has been introduced by the work on GDE+ ([7]). The system starts its diagnosis by considering only a limited set of the most common faults. When no solution can be found at this level, potential violations of closed-world assumptions guide the system to extend the set of faults under consideration to new candidates which could result in extending the set of diagnoses. In the example of the projector, one might first start by only considering faults of the bulb the fan and the switch. However, if all of them have been inspected and found to be working, the system might extend its search to also suspect the power cord and the electricity supply. In this way, a very large space of potential diagnoses can be considered while still maintaining the efficiency of the system.

6 Conclusions

Most knowledge of physical systems is formulated in the form of *models*, mapping characteristics of devices and structures into behaviors. Consequently, many engineering tasks require abductive or inductive inference. In contrast to deduction, which always provides sound solutions, abduction and induction often produce ambiguous results. These ambiguities are one of the main motivations for the use of knowledge-based systems.

However, classical deductive expert systems provide poor support for such inference strategies. Simulating abduction in a deductive framework requires the use of certainty factors or other probabilistic mechanisms in order to discriminate between potential solutions. These require several unrealistic assumptions, and make it difficult, if not impossible, to extend an expert system with new knowledge.

The framework of model-based reasoning is based on explicit abduction on models. It allows formulating knowledge in a modular way as models, and performing abductive inference in a sound and potentially efficient way. Furthermore, explicit formulation of closed-world assumptions makes it possible to detect missing knowledge which precludes reasoning from providing useful results. Such advantages mean that model-based reasoning should be considered for every knowledge-based system in civil engineering.

Acknowledgements

I would like to thank Dr. Ian Smith, LIA-EPFL, for discussions about knowledge-based systems in civil engineering.

References

[1] **Randall Davis, Bruce Buchanan, Edward Shortliffe**: "Production Rules as a Representation for a Knowledge-based Consultation Program," *Artificial Intelligence* **8**, 1977

[2] **Johan deKleer**: "An Assumption-based Truth Maintenance System," *Artificial Intelligence* **28**, 1986

[3] **Johan deKleer, Brian Williams**: "Diagnosing Multiple Faults," *Artificial Intelligence* **32**, 1987

[4] **R.A. Kowalski**: "Logic for Problem Solving," Elsevier, 1979

[5] **Allan Newell, Herbert Simon**: "GPS - A Program that Simulates Human Thought," in E. Feigenbaum, J. Feldman (eds.): *Computers and Thought*, McGraw-Hill, New York 1963

[6] **Louis Steinberg**: "Design as Top-Down Refinement plus Constraint Propagation," in: C. Tong and D. Sriram (eds): *Artificial Intelligence in Design* **1**, 1992

[7] **Peter Struss, Oskar Dressler**: "Physical Negation: Integrating Fault Models into the General Diagnostic Engine," *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, Morgan-Kaufmann, 1989

Development of Knowledge-Based Systems in Civil Engineering in China

Développement des systèmes experts pour le génie civil en Chine

Entwicklung von Expertensystemen für das Bauwesen in China

Xila LIU
Prof. of Civil Eng.
Tsinghua University,
Beijing, China

Xila Liu, born 1940, received his PhD at Purdue Univ. USA. He is a member of Board of Directors of China Civil Engineering Society, and Head of the Department of Civil Engineering at Tsinghua Univ. Beijing.

SUMMARY

The necessity to develop knowledge-based systems in civil engineering in general and the urgent need for China in particular are explained. A well organized project, which was supported by the National Natural Science Foundation of China (NSFC), is introduced. According to both the development of artificial intelligence technologies and the reals life applications in civil engineering, the possibility and some suggestions for further development are given.

RÉSUMÉ

Si le développement des systèmes experts pour le génie civil en général semble s'avérer nécessaire, cela est encore plus urgent pour la Chine. L'article expose un projet qui, sous l'impulsion de la Fondation nationale des sciences de Chine, fut bien organisé en vue de développer les technologies de l'intelligence artificielle et de les appliquer dans la pratique de la construction. L'auteur souligne les possibilités qui leur sont offertes et fournit quelques suggestions pour leur développement futur.

ZUSAMMENFASSUNG

Erscheint allgemein die Entwicklung von Expertensystemen für das Bauwesen geboten, so gilt das mit besonderer Dringlichkeit für China. Die Nationale Stiftung für Naturwissenschaften des Landes unterstützte ein wohlorganisiertes Projekt zur Entwicklung von Technologien der Künstlichen Intelligenz und ihre Anwendung in der Baupraxis. Der Beitrag behandelt ihre Möglichkeiten und einige Vorschläge für die weitere Entwicklung.

1. INTRODUCTION

Computer has been developed for over 40 years without stopping. There is no any industrial product which can compete with it. Since the great influence of computer development, the knowledge structures for many different disciplines have been changed. In any case, however, computer is still a calculation tool, even it is very powerful, the definition of "Calculation" has not been changed yet. Therefore, any over-estimating on computer's function may not be realistic. In the present paper, the necessity of development of knowledge-based systems in civil engineering in general and the urgent need for China in particular are discussed. A well organized project, which was supported by the National Natural Science Foundation of China (NSFC), and its contribution in China are briefly introduced. For further development of knowledge-based engineering in civil engineering, the possibility and some suggestion are also given.

2. CIVIL ENGINEERING AND KNOWLEDGE ENGINEERING

2.1 The distinguishing Features of Civil Engineering

In studying engineering systems, there are two categories of systems to be considered. The former is so-called continuous system and the latter is so-called discrete system. For the former, one usually starts with a continuous mathematical model which is obtained by considering some natural laws. For the latter, however, the behaviour of systems is described by some dynamic models with discrete mathematical structures. From the product point of view, the number of finished products can not be counted in the former system, and it is countable in the latter system. But the construction project, i.e. the product of civil engineering, is very individual. For example, the well-known project, Three Gorge Dam on the Yangtse River, has been planned for more than three decades. A great number of research works have been done just for this individual project. From the continuity point of view, the civil engineering is on the counter part as shown in Fig.1 .

			Continuity →
Individual System	**Discrete System**	**Continuous System**	
Civil Engineering	Auto Engineering	Chemical Engineering	

Fig.1 Different Categories of Systems

It should also be mentioned that the decision making problems of civil engineering are synthetical. For example, the evaluation of the structural reliability does not only depend on the probabilities calculation, but also on psychology and social system. During planning and design of a high-rise building, besides architectural consideration and mechanical calculation, many environment and transportation problems must be considered. A new built highway may become a serious pollution source. Neglecting the durability of a highway bridge may cause heavy economic losses.

Unlike other engineering domains the individuality and multi-disciplines in civil engineering are very protrusive. In practice, during planning, design, construction and maintenance of a construction project, a great number of factors have to be involved. Usually, there are very strong interactions existing among these factors. Most of factors are uncertain and a great amount of incomplete information have to be treated. In addition, in many cases, it is difficult to do statistical survey. And it also seems that to build a comprehensive mathematical model is not so feasible. In fact, it is very often to solve a civil engineering problems only by experience or intuition of engineers.

2.2 The Knowledge Structure of Civil Engineers

The traditional knowledge structure for civil engineers is a two-apex structure, i.e. the practical experience and the theoretical basis. The practical experience should include the experimental skill, observation, and expertise. Due to the computer development the modern knowledge structure becomes a three-apex structure[1], i.e. the practical experience, the theoretical basis, and the computational training (Fig.2).

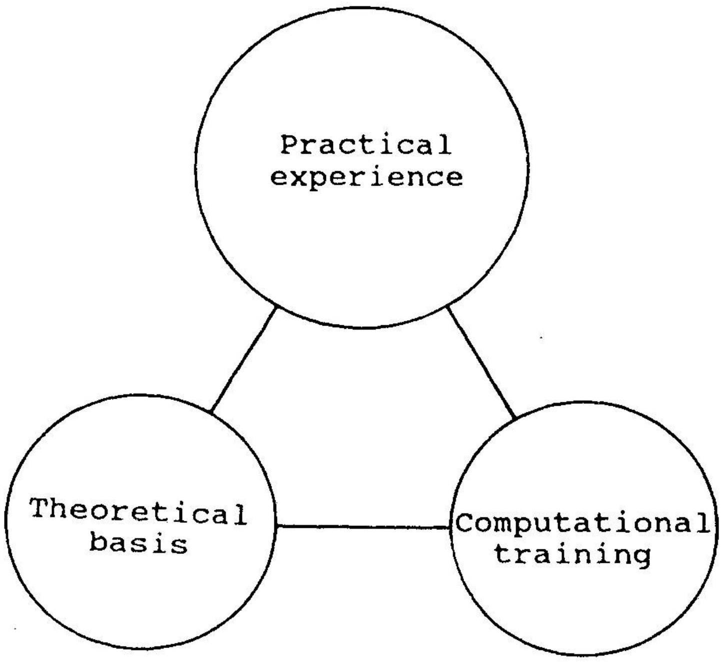


Fig.2 Modern Knowledge Structure of Civil Engineers

There is a more strong link between practice and theory. The computer makes the tests to be a real experimental science. Such as shaking tables and pseudo-dynamic tests are used to simulate the real earthquake. It is incredible to do these tests without computer. On the other side, however, the computer makes theory softer, which means that many uncertainties and incomplete information can be considered. Also, from the new development of knowledge engineering, it should be possible to code a great deal of experience in computer, which means the computer should make the experience harder[2]. It can be predicted that a civil engineer in the next century, who can handle the triangle knowledge structure, will be very active. Otherwise, he will be very passive.

2.3 The Education of Civil Engineers in The Next Century

Rethinking on the present education in civil engineering, there are two weak points existing. One is on systems engineering and another one is on practice.
It is very clear that the viewpoint of systems engineering has pushed the thinking mode of civil engineers forward. For example, the development tendency of structural engineering is to improve the pure analysis of individual elements to synthesis and control of the whole structure with its coupling systems, also to change the pure consideration of the structural service life to that of the whole structural life-cycle (construction, service life and maintenance). People have found that they made mistake such as on a Chinese steelyard shown in Fig.3, people used to be interested in refining the scale on the arm but forgot to check the sliding weight. But the problem is that, when they are dealing with a large scale system, a lot of uncertain factors and incomplete information have to be treated. People know quantitative calculation and analysis very well, but they are not so familiar with qualitative reasoning and syntheses.

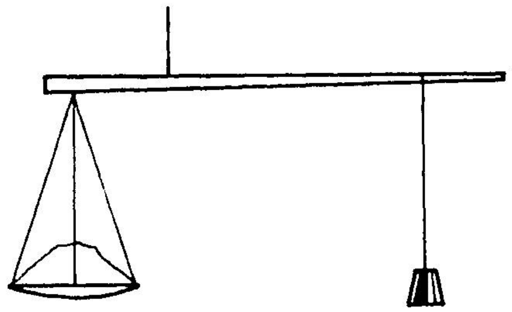


Fig.3 a Chinese Steelyard

The practical training probably is the most important part in civil engineering education, which is not only for future engineers but also for younger researchers. A man's knowledge consists only of two parts, that which comes from direct experience and that which comes from indirect experience. But it should be emphasised that direct experience is basic. Considered as a whole, knowledge of any kind is inseparable from direct experience. In order to have direct engineering feeling, for example, it is necessary to extend the practice training of students on construction site. But it is not the only way we can do. As one of the emerging technologies, knowledge engineering can help us to acquire, represent, and reuse indirect knowledge more efficiently. It is also possible to help senior engineers to code their own direct experience.

We are now in a great new era of technical revolution. There are so many advanced technologies which need to be handled. In reality, however, we only can handle a part of them, which may be very significant for developing our particular discipline. For some of them, the ebb tide may come very early or may be very low(Fig.4). How to make the choice? Here, the prediction on the ebb tide is very important. As discussed previously, it can be seen that the development tendency of knowledge engineering is going up and getting more and more important in general(Fig.4). Consequently, in the next century, knowledge engineering may become a fundamental technology for civil engineers.

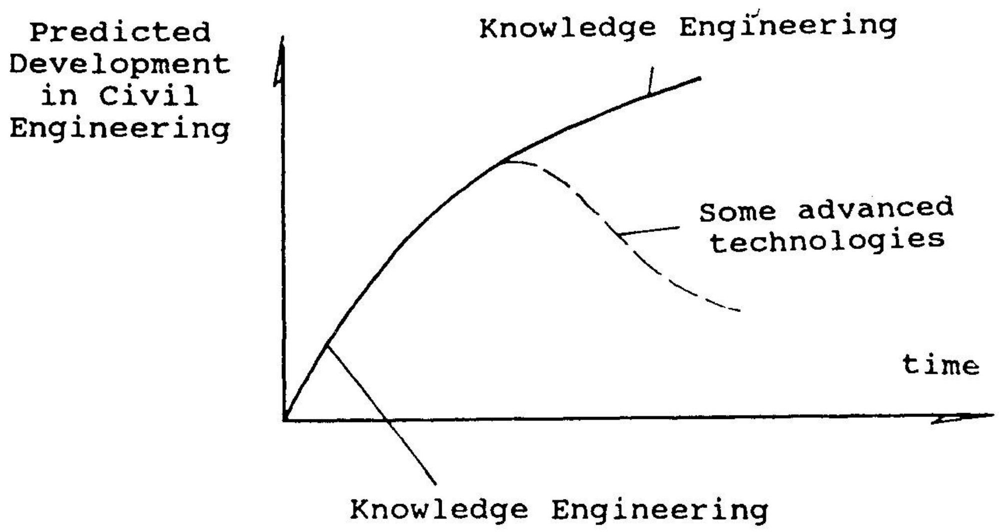


Fig.4 Prediction on Knowledge Engineering in Civil Engineering

3. KNOWLEDGE BASED SYSTEMS RESEARCH IN CHINA

3.1 Background

China is a developing country. The citification is an irrecusable tendency. At present, the ratio of city population is around 26%. By the end of this century, the ratio may increase to 33%. At that time, the economics of our country will develop by leaps and bounds. By prediction, in 2020, we will have 1100 cities and the ratio of city population may increase to 50%. It is true that our country is just shortly before the economic development at full speed.

In China, there are more that 20 million people working in construction companies. During the past 40 years, almost 40 million big construction projects and 1.64 billion meter square apartments have been completed. We rebuilt or expended 200 cities and more than 1000 towns. The living condition and environment of 400 million people have been improved. As predicted by experts, during the next 20 to 30 years, the hot regions of economic development in the world will be moved to the west coast of the Pacific Ocean. Probably, China will be the hot point in these regions.

Although the construction background in China is really beneficial to the academical development of civil engineering, some short-comings should still be noted. Such as the safety criteria of design codes are lower than the average level in the world, the quality control of construction is not so satisfactory, the structural maintenance and durability have been neglected for long time. In this case, by the end of this century, almost 50% of existing buildings in our country (i.e. 2.34 billion meter square) need to be repaired. The most serious problem is that after the cultural revolution, almost one generation of educated engineers has been lost. China is really short of senior civil engineers. Many experienced engineers are going to be retired. In order to save their expertise, it is worth to explore the new field - knowledge engineering.

3.2 Organization on Research Projects

Although the research funds and the total number of intellectuals are very limited in our country, China has enough foresight and sagacity to pay great attention to fundamental research, even on many engineering fields. We have already been adjusting the whole economic system in our country, but as far as the National Natural Science Foundation of China concerned the annual research funds are still increasing. For instance, in 1992, it has increased by 28% over 1991. After abundant proving and consulting by a large circle of experts, a priority joint project on intelligent decision support systems in civil engineering was organized by NSFC and seven ministries (such as construction, transportation, environment, and education et al.). Almost 25 universities and research institutes, 220 researchers including about 90 professors or senior engineers, some of them are very famous in our country, were involved. There were 10 aspects and 30 subprojects including: urban planning, seismic risk prediction, railway construction, highway and water transportation design, evaluation of ecological-environment qualities, preliminary design of structures, construction management and cost prediction, damage assessment of existing buildings, intelligent CAD and simulation, and treatment of uncertain information in civil engineering.

The predicted objectives of the mentioned project were: to build a group of relatively complete knowledge bases, to provide a great number of advanced papers or books, to complete a number of applicable softwares, and to train and bring up large numbers of qualified younger researchers for the next century. The administration system was very strict. Each subproject had to be re-evaluated by an advisory group once a year. The annual research fund of each subproject in the next year entirely depended on the contribution in the present year.

3.3 Recent Development

The mentioned priority joint project started in 1987 and has been checked and accepted item by item by the National Natural Science Foundation of China (NSFC) in December of 1992. In the five years, 45 research achievements have been appraised by NSFC, ministries, or different units, respectively. Among them, 9 projects have won research awards or prizes. 49 softwares have been provide to different users and 48 users reports on their qualification have been received. More than 480 papers have been published in various technical journals, conference proceedings, symposium volumes and technical report series. Four proceedings (914 papers in total)[4] and 16 professional books have been or will be published. During the past five years, we have trained a great number of graduate students by the joint project and 17 doctoral degrees and 120 master degrees were conferred. According to excellent personal contributions, 16 young researchers have been promoted to associate professors or full professors. As shown in Fig.5, the earliest knowledge-based systems in civil engineering in China were developed from 1985 to 1986, which were later than those developed in other domains for several years. But now, researches on knowledge based systems in civil engineering have become a very active field in China. It is very significant not only for the discipline development but also for the education of the next century. In the following five years, the National Natural Science Foundation of China will continuously support the same project (1993-1998) especially on the integration techniques and fundamental researches of structural design and construction control.

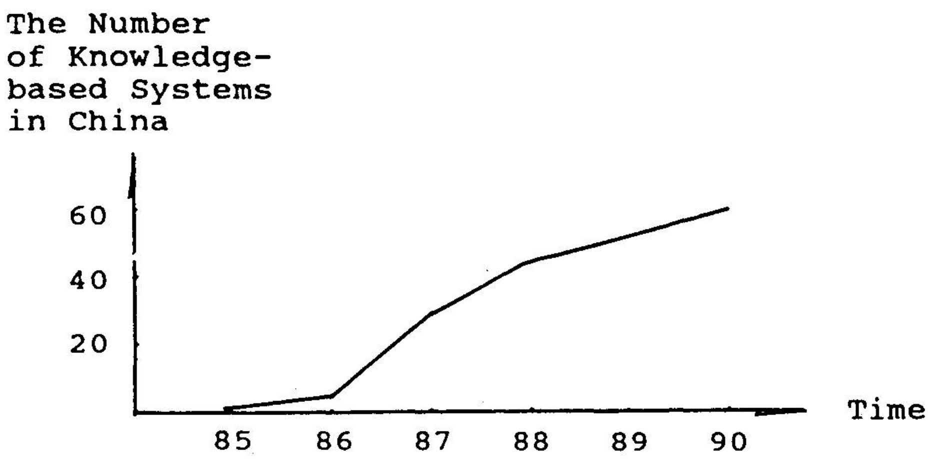


Fig.5 Development of Knowledge-based Systems in Civil Engineering

3.4 Some Remarkable Systems

PDSMSMB-1 and **PDKSCB-1** : Knowledge-based Systems for Prediction of Earthquake Damage to Urban Buildings (Institute of Engineering Mechanics of the State Seismological Bureau)

Based on a vast amount of collected data and facts, such as those from Xing Tai earthquake in 1966 and Tang Shan earthquake in 1976, both systems have been developed for damage prediction of masonry buildings. **PDSMSMB-1** is particularly for individual multistory masonry apartments and **PDKSCB-1** is for clusters of masonry buildings in a particular district, which includes predictions on damage degrees, casualties, economic loss, and some countermeasures. Both systems have been used in many cities, such as Tai Yuan (population of 1.5 million), Xia Men (0.3 million), Zhan Jiang (0.3 million) et al. Since systems have very solid knowledge-bases, the Ministry of Construction of China has decided to spread them in every seismic region of China in 1993.

RAISE-4 : Expert System for Reliability Assessment and Countermeasures of Reinforced Concrete Mills (Tsinghua University of Beijing)

In the present system, the moduli on the reliability assessment, the diagnosis of structural damages and the strengthening methods are integrated. Some algorithms on structural remained safety, reliability assessment specifications, and the strengthening code are also included. A great number of factors that affect the structural reliability are synthesized by a new fuzzy measurement method. Also, an advanced association model is proposed to obtain the most possible damage causes among all damage causes. By prediction, there is a very good market for the present system in China. The newest investigation shows that the annual income only on damage assessment and strengthening design of existing industrial buildings is around 172 million Yuan. Many research centres on structural diagnosis and strengthening in different provinces hope to buy the software RISE-4.

CARB : Expert System on Condition Assessment for Railway Bridges (Southwest Jiaotong University)

The special contribution of the present system is the technique for integrating an expert system with a dynamic database. The following measurements are taken, such as keeping the consistency of items of damage data with premises of assessment rules, auto-collecting of damage data, and auto-generating of damage data. Besides, two kinds of heuristic knowledge, i.e. the synthetical criterion of assessment and the current specification method, are used. Also, unauthenticity reasoning, unsuitability reasoning, and default reasoning are performed. This system is very useful for monitoring railway bridges. The similar system has also been used in highway bridge monitoring system in Guang Dong province.

UENC : Intelligent Decision Support System on Urban Environmental Noise Control (Tongji University)

The present system is a typical integration system including knowledge base, data base, models base, and algorithms. Its functions involve noise prediction, evaluation, and noise precaution. The traffic noise, industry noise, construction noise and area noise all can be controlled by the present system. Many practical noise data and calculation modules are provided for users inquiry, and the graphical interface is very friendly. A number of users reports with satisfactorily comments have been received. Also, after examining by a group of experts in December of 1992, it is recommended that this advanced integration system can be widely spread not only for noise precaution and control but also for management of noise information.

4. SOME COMMENTS ON DEVELOPMENT OF KNOWLEDGE-BASED SYSTEMS

4.1 Cognition Process is Primary and Its Conceptualization is Secondary

Since 1980s, although some fundamental researches on knowledge representation, inference of common knowledge, machine learning and the distributed knowledge artificial intelligence et al. have been developed gratifyingly, it seems that there is no considerable breakthrough in the artificial intelligence field. It is worth to rethink some basic questions[5] as: How much have we already known on cognition process? Is there really a single architecture underlying virtually all cognition? Actually, in the endless flow of absolute truth, each particular contribution in this field is heartening but only a relative truth, there may be a long way to go to reach the absolute truth. In this case, any over-optimistical or over-pessimistical viewpoint may not be appropriate.

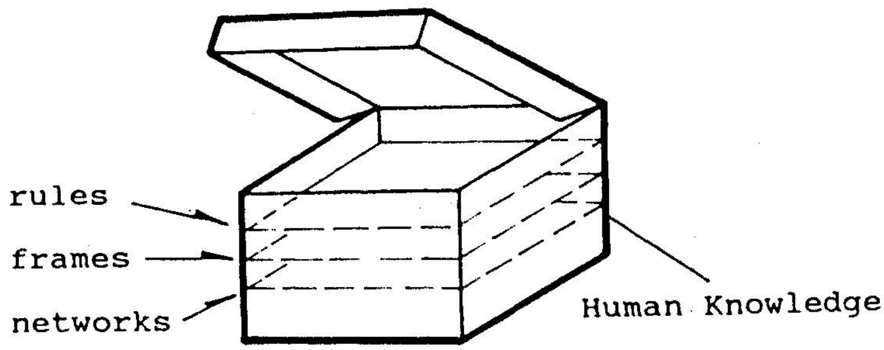


Fig.6 The Box of Human Knowledge

Until now, how much do we really know about the human brain? As shown in Fig.6, the three most popular ways of representing knowledge are rules, frames and semantic nets. Rule-based representation is a shallow representation on the first layer of the box of human knowledge, whereas schemes using frames and semantic nets are deeper representations on the second or the third layer. What about the next layers? It should be understood that the ways of representing human knowledge are very limited. We are not so familiar with human association (between different domains), human inspiration, knowledge distillation, the leap of cognition process et al. Even it is very hard to handle the human ability on searching and simplifying.

When we are talking about artificial intelligence in design, we have to clear the definition on design first. Actually, in most cases, "design" means a kind of innovation based on existing cases and knowledge, which is similar to the interpolation in numerical method. No matter what kind of cognition model is used, mostly, people have to make a closed domain (may use users interface) to obtain the final scheme. Strictly speaking, for the time being, basic knowledge coded in a knowledge base comes from human being, it is not automatically generated from another knowledge base. In this case, the computer creation is very limited.

On the other side, in general, domain knowledge is incomplete, intractable, incorrect and inconsistent. It is very hard to be generalized. The tunnel between widely generalized knowledge representation and unified inference process is still troublesome. From this point, a new generation of development environment of knowledge-based systems is not a easy task.

But there is no reason to be pessimistical. At present, we could not develop a knowledge-based system instead of domain experts, but it is possible to build a system to provide best advice when we have no domain expert. It is very difficult to develop an intelligent computer aided design system to create some newest schemes, but it is possible to develop a knowledge-based design system to ensure the generated scheme is not worse than the average design level, which is also very useful in application.

From Nilsson's strict logicism to Hewitt's open information systems semantics, from Lenat-Feigenbaum's thresholds of knowledge and Newell's SOAR in chunk to Brooks' intelligence without representation many contributions have been done from a particular aspect, but none of them is the absolute truth. In the endless flow it may be hardly to reach the end, but anyway, each contribution makes us to be closer to the absolute truth.

4.2 Practice is Primary and Knowledge is Secondary

Where does man's knowledge come from? It comes from his activity in material production, through which he comes gradually to understand the phenomena, the properties, the laws of nature, and the relations between himself and nature[3]. None of this knowledge can be acquired apart from practice. When human activity develops step by step from a lower to a higher level, consequently, man's knowledge also develops step by step from the shallower to the deeper level. It is true that, for artificial intelligence development the considerable breakthrough on theoretical research is needed, but successful application is more important.

The truth of any knowledge or theory is determined not by subjective feeling, but by objective results in social practice. Only social practice can be the criterion of truth. For a successful knowledge-based system, it is not enough to show some experts how to finish some particular examples, the real application during a certain period is necessary.

In China, the research group for each related subproject has to be "three in one", which means that three kinds of experts, such as the civil engineer, the knowledge engineer, and the software engineer, have to be involved. The head of each research group should be a real civil engineer. Furthermore, the civil engineer in the group is encouraged to learn some general concepts on knowledge engineering, meanwhile the knowledge engineer is suggested to know some basic ideas on civil engineering. In practice, the former seems more efficient.

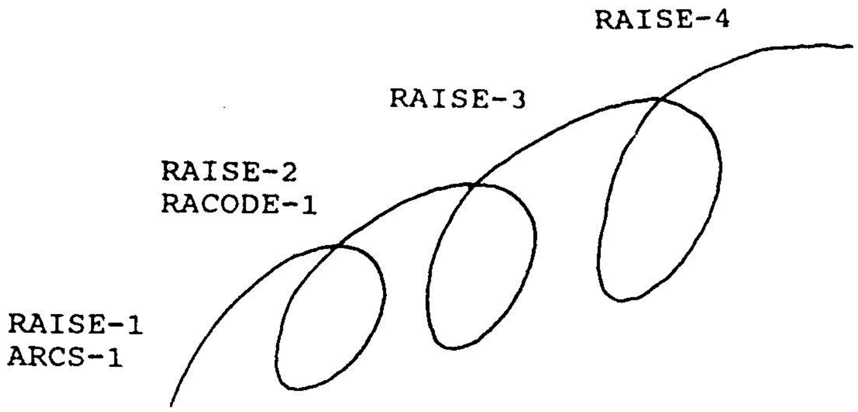


Fig.7 Development of RAISE Series

Usually, the life cycle of a software includes: planning, requirement analysis, software, design, programming, testing, running, and maintenance. It is so-called the falls model. But most of knowledge-based systems in China are following the fountain model shown in Fig.7. Following users requirement, we keep improving the previous prototype in each cycle. With each cycle the function of the software rises to a higher level. For example, during developing the RAISE series, we started from two original prototypes, i.e. RAISE-1, an expert system for damage assessment of single-storied reinforced concrete frame, and ARCS-1, also an expert system for damage assessment of reinforced concrete elements in industrial workshops. In the second cycle, we expanded the function of RAISE-1 to cover various single-storied workshops and also built another prototype RACODE-1 to store a new specification on structural reliability assessment. In the third cycle, we added a calculation program of safety factors on RAISE-2 and performed the new version RAISE-3 on personal computers. The Chinese version of RAISE-3 is called RAISE-4, which can be directly used by Chinese technicians. It is very clear that the engine to push the RAISE series forward is practice, is the application. Practice always produces some new requirement. After improving the previous version the prototype and knowledge has risen to a higher level. It is true that practice, knowledge, again practice, and again knowledge. This form repeats itself in endless cycles, and with each cycle the content of practice and knowledge rises to a higher level[3]. We believe that discover the truth of cognition process through practice, and again though practice verify and develop the truth.

4.3 Individual Character is Primary and General Character is Secondary

Cognition process always moves from the particular to the general, and then from the general to the particular; each cycle makes it more and more profound[3]. For example, the well known expert system MYCIN was built first and then the generalized EMYCIN was developed, afterwards EMYCIN has been used to develop several systems, such as PUFF and SACON. On a lower level, the inductive method can be used to extract some general rules from particular examples. But the scientific abstraction on a higher level may be based on some intuition which is very hard to be explained by existing knowledge. At present, we may not be able to find the architecture underlying virtually all cognition in a short time, but more particular systems or examples may give us more solid foundation for the creative inspiration. We belive that the individual character is the most abundant. The general character is contained in every individual character; it should not be converse (Fig.8). The considerable breakthrough in artificial intelligence seems not so easy. When we are wandering about the next knowledge distillation, why we do not pay more attention to individual systems? Without individual character, there can be no general character[3].

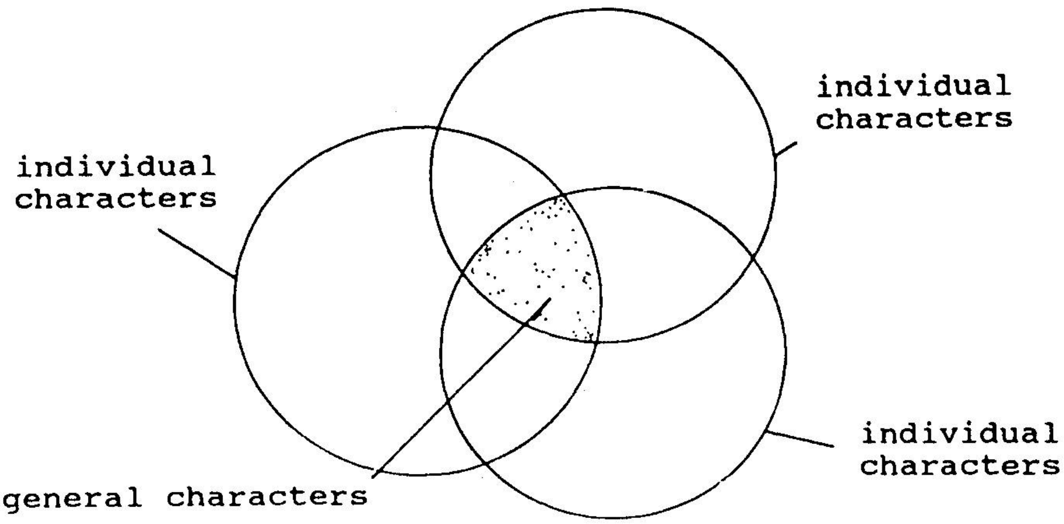


Fig.8 General Character vs. Individual Character

5. CONCLUSIONS

The necessity to develop knowledge-based systems in civil engineering not only comes from the individuality and multi-disciplines in civil engineering but also comes from the education for younger civil engineers in the next century.

Considering the financial and education background of China to organize a prior joint project to involve a large circle of related experts is a efficient way, especially in some advanced fields.

We are looking for some considerable breakthrough in artificial intelligence. But it is worth to emphasize that, at any time, the objective world and practice are primary.

REFERENCES

1. LIU XILA, Basic Consideration of the Development Strategy of Structural engineering in China. International Seminar on Evaluation and Granting Policies for development of Basic Disciplines, NSFC, Beijing, China, Nov., 26-28, 1991.
2. LIU XILA, Decision Making System in Civil Engineering in China. Proceedings of Expert Systems in Civil Engineering, IABSE Colloquium, Bergamo, 1989, pp.71-81.
3. AI SIQI, Dialectical Materialism & Historical Materialism. People Publish House, Beijing, 1961 (in Chinese).
4. AI Aided Decision Making Systems in Civil Engineering, edited by H.X. Liu, Xila Liu, et al., China Construction Publish House, Beijing, 1992 (in Chinese).
5. Artificial Intelligence, Vol.47, Special Volume: Foundation of AI, edited by David Kirsh, Elsevier, Amsterdam, 1991.

AI and New Computational Models of Design
Intelligence artificielle et nouveaux modèles de calcul
KI und neue Berechnungsmodelle im Entwerfen

John S. GERO
Prof. of Design Science
Univ. of Sydney
Sydney, Australia

John Gero holds degrees in eng., building science and architecture. He has been visiting professor at universities in the USA, UK and France. His current research is in artificial intelligence in design.

SUMMARY
This paper very briefly reviews the state-of-the-art in the application of artificial intelligence in design before describing ten research and development areas which support new computational models of design. Two new computational models are introduced. The first case-based design is founded on the concept that it is possible to reason from specific precedents (called cases) rather than computing new results using compiled knowledge. The second creative design, introduces the notion of expanding the space of possible designs through various computationsl constructs.

RÉSUMÉ
Après un résumé succinct de l'état actuel de la technique d'application de l'intelligence artificielle dans le domaine des projets, l'auteur décrit dix secteurs de la recherche et du développement assistés par de nouveaux modèles de calcul, dont deux font l'objet d'explications détaillées: projets rapportés au cas spécifique et projets créatifs. Le premier modèle se base sur le concept qu'il est possible d'opérer des déductions à partir de cas spécifiques précédents, sans devoir calculer de nouvelles solutions à partir de la banque de données. Le deuxième modèle repose sur l'idée que le domaine des projets possibles est extensible par diverses phases de calcul d'élaboration.

ZUSAMMENFASSUNG
Nach einem kurzen Ueberblick über den Stand der Technik in der Anwendung künstlicher Intelligenz im Entwerfen werden zehn Gebiete beschrieben, die in Forschung oder Entwicklung neue Berechnungsmodelle unterstützen. Auf zwei davon wird näher eingegangen: fallbezogenes und kreatives Entwerfen. Ersteres basiert auf dem Konzept, dass von spezifischen vorangegangenen Fällen geschlossen werden kann, ohne neue Lösung aufgrund der Datenbank zu berechnen. Das zweite Modell, kreatives Entwerfen, beruht auf der These, dass der Bereich möglicher Entwürfe durch unterschiedliche rechnerische Aufbauschritte erweiterbar ist.

1. INTRODUCTION

Among a nation's goals are competitive leadership in the international marketplace and excellence in industrial productivity. Superior design, a fundamental prerequisite for superior products and systems, is one of the important keys to achieving these goals. Computer-aided design has the potential to provide access to this key.

The early work on computer-aided design fell into two distinct and disparate groupings. The first was concerned with analysis methods embodied in computer programs. This has resulted in today's finite element method techniques and programs. The finite element method is, nowadays, a mature technology. The second was concerned with graphics, commencing with Ivan Sutherland's SKETCHPAD. For a long time this form of graphics led researchers to concentrate on data structures to support graphical image making. Later, the emphasis shifted away from graphical image making to the representation of models of objects leading researchers to concentrate on models and data structures for geometric modelling.

This work was and continues to be based on particular paradigms of the roles of the computer in the design process. In the work characterised by the finite element method the paradigm assumes that a sufficient representation can be encoded to allow analysis to be automated. In the work characterised by geometric modelling the paradigm assumes that the representation is the problem since the analysis is left to the designer. Neither of these is based on a paradigm which gives the computer a more active role in the entire process of designing.

From the beginning of the 1980s there has been a burgeoning interest in understanding and using approaches drawn from artificial intelligence. These approaches are often couched under such labels as information technology, knowledge-based systems, expert systems and so on. What they all have in common is the move from using the computer with algebraic models and numerical values for the variables in those models to symbolic models and symbolic values for the variables in those models. Along with the move to the use of knowledge-based systems has come an increasing interest in expanding the role of computers and redefining computer-aided design in the service of design.

2. STATE-OF-THE-ART

During the 1980s computational approaches to the provision of design assistance were researched and developed using the knowledge-based view of design with its concomitant computational machinery derived from artificial intelligence. Although a wide variety of techniques and methods have been used most of them addressed one of the following categories:

(i) representation of designed objects
(ii) analysis of designed objects
(iii) diagnosis of faults in designed objects
(iv) synthesis of designs

(i) *Representation of designed objects*—the object-oriented paradigm in which both data and methods are encapsulated has provided new ways of conceiving how to represent designed objects. Artificial intelligence concepts have allowed the possibility of representing and then reasoning about non-numeric features of a designed object.

(ii) *Analysis of designed objects*—analysis plays a pivotal role in design. New analysis processed using artificial intelligence approaches have been developed; for example the checking of a design against government design codes and codes of practice, where these contain requirements couched in logical rather than numerical terms.

(iii) *Diagnosis of faults in designed objects*—new model-based systems are now being used to diagnose design faults. Many of these are an outgrowth of diagnostic expert systems developed for use in medicine.

(iv) *Synthesis of designs*—formation processes based on such strategies as decomposition, design grammars and symbolic optimization are available for use in the synthesis of designs. In some domains such as VLSI design considerable effort has been expended in automating or semi-automating these synthesis processes.

All of this work has one over-arching view in common. Namely, that the systems operate within the context of routine design.

Routine design can be defined as that class of design activity where everything about the design process is known a priori. The variables as well as the processes, i.e. the knowledge, needed to find values for those variables are known a priori. This concept of routine design is analogous to but is not meant to model an experienced human designer tackling a well-known task. Routine design is often equated with parametric design but it has a larger ambit. There is a large body of research and development with a smaller number of applications in this area. *Concurrent design*, which can be considered within the ambit of routine design, is attracting increasing attention. It aims to incorporate knowledge about processes downstream of design into the decision-making in design. The primary focus here is to include buildability and manufacturability into the design process.

The knowledge-based tools being developed often end up automating some design task. However, there is remarkably little work which addresses the difficult problems associated with such areas as conceptual or non-routine design and collaborative design. It might be said that knowledge-based approaches to design have so far concentrated on areas which are relatively well understood in computational terms. In order to obtain a quantum increase in design quality and performance increasing research effort will, in the future, have to be put into other areas.

3. RESEARCH AND NEW COMPUTATIONAL MODELS OF DESIGN

New computational models of design based on the artificial intelligence paradigm make use of the fundamental concepts of:

— symbolic variables
— separation of knowledge from control
— symbolic reasoning

Ten of the most significant research areas supporting these new design models will be briefly described before elaborating two models in more detail.

(i) Representation in Design

A fundamental problem for artificial intelligence and design remains the one of representation. What is it that a designer knows and how does it get represented in a computer? There are two disparate kinds of knowledge of interest here: that concerned with design processes and that concerned with the artefact as it is being designed. Even if there is no concern with what a human designer knows there is still the question of what knowledge a computational model of design needs and how to represent it.

Knowledge-based design has moved from being treated as a knowledge-lean problem to being treated as a knowledge-rich problem. Thus, increasing amounts of knowledge need to be formalised, structured and represented. Three kinds of knowledge need to be represented:

— case knowledge (episodes or precedents)
— generalised or compiled knowledge (derived from cases)
— first principles knowledge

(ii) Design Semantics

Two issues are mentioned here. The first issue is the coding-decoding problem. How does a system decode a representation that has been altered after it has been coded or if the representation is being decoded in a different context. One important aspect of design is the shifting context it creates for its own activities. Such changes in context offer the opportunity for *emergence*—where an interpretation of the semantics of a representation is made which is different to that explicitly made in the representation. The second issue concerns how do you represent in an explicit and manipulable form the intentions, purposes or functions of the intended artefact in such a manner that they can be used. This has important implications for data exchange between designers and for data exchange standards.

(iii) Reasoning in Design

Much of the reasoning machinery brought across from artificial intelligence has been concerned with monotonic logics, with consistency maintenance and with resolving conflicting constraints. These reasoning processes have been developed for a static world. The design world by its very nature is not static and the appropriate reasoning mode is abductive (i.e. what could be) rather than deductive (i.e. what must be). It is common in design to maintain inconsistent beliefs for a time and to resolve conflicting constraints by designing them away.

(iv) Combinatorial Explosion in Design

Abductive reasoning brings with it the very real likelihood of combinatorial explosion of potential inferences. As soon as a system deals with what could be rather than what must be it could go on indefinitely. Constraint propagation, planning and heuristics are common ways of addressing combinatorial explosion. However, alternate approaches based on evaluating the satisfaction of solutions or solution directions are likely to be more useful in design.

(v) Indexing in Design

Design occurs in a knowledge-rich and knowledge-intensive environment. however, the more knowledge that is coded into the system the harder it is to find what is useful. Much design knowledge can be placed into one of the three categories of: cases (episodes or precedences), generalised knowledge based on cases and first principles knowledge. When and how to index these still remains a difficult question to answer.

(vi) Dynamic Modification—Learning in Design

In design synthesis, unlike in fields which rely exclusively on deductive processes, obtaining the same solution each time for the same problem is considered a failure of design. Designers learn from doing design and learn from their own and other's designs. This learning results in a dynamic modification of both the knowledge and knowledge structures used to represent the knowledge. Understanding this dynamic modification is still a question yet to be adequately answered.

(vii) Situation Recognition in Design

An important research area for artificial intelligence in design is how to produce systems capable of recognising situations at a semantic (strategic) level rather than simply at the syntactical (tactical) level. Much of the interest in non-routine design lies in the emergence of newly recognised situations, situations which were not produced intentionally but by extension.

(viii) Collaborative Design

Designers rarely work alone, design has become so complex an activity that many specialist designers are involved. How to provide real-time computational support to improve collaboration between individuals in a design team has become a critical issue. Ideas from distributed artificial intelligence provide useful starting points but fundamental issues remain.

(ix) Non-Routine or Creative Design

Design and creativity are often treated synonymously by many people. Clear definitional distinctions have been drawn between routine and non-routine design with the acceptance that not all design is creative. Basic questions remain: are there principles of creativity; are there creative processes; what kind of computational support can be provided in a non-routine design context?

(x) Evaluation in Design
The evaluation processes in design include not only the evaluation of the a priori defined technical performance of the designed artefact but an assessment of emerging performance as well as the assessment of its socio-ethical value. This latter aspect currently eludes any formal description. However, these issues need to be addressed.

4. NEW COMPUTATIONAL MODELS OF DESIGN

A number of new computational models of design using artificial intelligence concepts are under development. Two of these will be described here.

4.1 Case-Based Design

Case-based reasoning is a well-defined paradigm in artificial intelligence. It is based on the premise that humans reason from specific experiences rather than by following a set of general guidelines. For example, reasoning from precedents is one of the basic methodologies in law. Case-based reasoning relates a current situation to the closest most specific experience in memory and uses that experience to solve the problem at hand. It is thus a memory-based approach rather than a computation approach, whereby solutions to problems already solved need only be retrieved rather than computed again. The key factors in case-based reasoning are the storage of cases as complete patterns of experiences including the reasoning process, the ability to be reminded of the most appropriate case and the application of that case to the current situation. Application of the case may either be a direct application if the current situation and that of the case match exactly, or there may be a need for some modification of the case. This modification may be of various degrees of severity. Case-based reasoning uses the strategies of *modification* and *repair* to effect such modifications. New cases are thus produced either as variations on the previous case or, in extreme situations, as new cases if considerable modification took place. Case-based reasoning thus incorporates a learning capacity in the form of new cases being incorporated into a dynamic case base.

Searching for a case is based on indexing cases with regards to various factors, e.g. goals and attributes. The more efficient the indexing, the more efficient the search. Retrieval is a matter of pattern matching, i.e. matching a required pattern of requirements to an existing set. This match may be exact or partial. In the case of partial matches, some criteria are required to determine the 'best' partial match. Matches may be made to parts of several cases and a new case results from combining elements from these cases, if consistency is satisfied.

The processes involved in case-based design are search, match, retrieve, select, modify, repair and store.

Search. Given a problem description of requirements including functions to be achieved, required behaviour performances, the design environment and even constraints on values of structure variables, the case base must be searched to find an appropriate design case. The utility of case-based designing is strongly dependent on the efficiency of the search procedure. Searching could be sequential, parallel or direct using an indexing mechanism. Indexing must be done on the function, behaviour, structure and context features.

Match. An appropriate case for consideration is found with regards to the matching of above mentioned features. Perfect matching, i.e. where the required features are found exactly in a case, is unusual. Partial matching occurs when some of the features are matched or the features are matched to some degree.

Retrieve. A case which matches to some defined degree needs to be retrieved for consideration. This may or may not involve display of these cases to the users for perusal and consideration.

Select. A selection of a single case as the basis for determining the design solution has to be made. Alternatively, if only part of a design case is required, then several design cases may require to be selected, and the necessary parts of each extracted. In either situation, the 'best' matching design case should be selected. Selection of the 'best' design case can be on the basis of the most similar or the most useful match. Selection can be carried out by the system or by users after consideration of an appropriate set of candidates retrieved by the system. Selection by the system based on partial matching entails such factors as the importance of the features matched as well as how close they are matched.

Modify. Where a design case is selected which does not match the design requirements sufficiently, some modifications will be necessary. This may involve the replacement of variables with other variables or simply the alteration of some values of variables.

Repair. In many situations, a modification to an existing design case based on substitution of variables or modification of values will cause some performance failure in some other behaviour or function. For example, decreasing the cross-sectional area of a column to satisfy some new spatial requirement may cause buckling. Other modifications may be considered but none may be satisfactory. One of two directions now needs be taken. Either an alternative design case is selected based on the new information known regarding the necessity for modifications and the effects of modifications or the current selected design case is modified in such a way as to make it acceptable. This latter process is known as the process of *repair* in case-based reasoning.

Store. After a design case has been modified or repaired, a new design case has been generated. If this new design case is considered to be sufficiently important as a design experience different to existing design cases, then it must be stored in the case base with appropriate indexing. Where the failure of solutions is seen as an important piece of information to the anticipation of future problems, this must be noted in the design case.

4.2 Non-Routine or Creative Design

Non-routine design or *creative design* can be defined as that class of design activity when all the variables which define the structure and behaviour are not known a priori nor necessarily are all the processes needed to produce them. The implication of this conceptualisation of non-routine design is that the focus is on processes for the introduction of new variables into the design and their integration into the existing variable structure. It is suggested that this is one basis for the production of potentially creative designs.

For a given set of variables and processes operating within a bounded context any model will construct a bounded state space. Creative design can be represented in such a state space by a change in the state space. Routine design does not change the state space, it simply searches within it. There are two classes of change to the state space possible: addition and substitution. The *additive class of state space change* is represented in Figure 1 where the new state space, S_n, totally contains the original state space, S_o. The implication of the additive class of state space change is that new variables are added to the existing stock of variables.

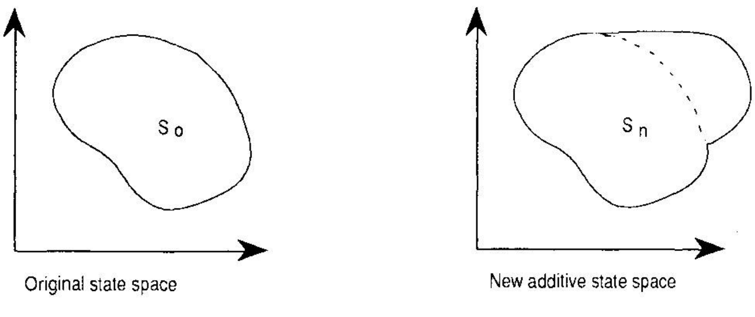


Figure 1. The change in state space due to the addition of new variables.

The *substitutive class of state space change* is represented in Figure 2 where the new state space, S_n, does not cover the original state space, S_o. The implication of the substitutive class of state space change is that some (or in the extreme case all) of the existing variables are deleted and new ones are added to the remaining stock of variables.

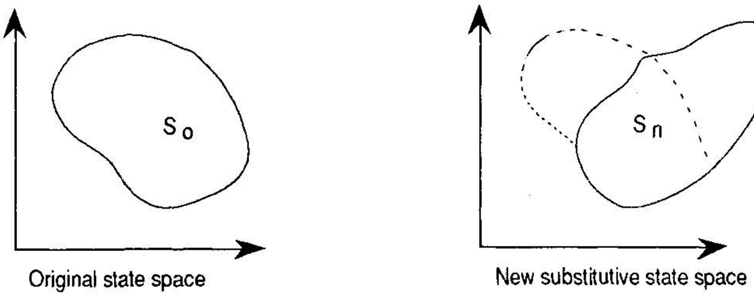


Figure 2. The change in the state space due to the substitution of new variables.

Whilst the additive and substitutive classes of state space change have been presented as if the variables being used are structure or behaviour variables only, this need not be the case. Modifications to the knowledge structures and to the contents of knowledge structures fall into these classes also and have the potential to be part of the creative process. For example, take a rule-based system for the production of a design. A design is produced by a defined sequence of executions of the rules, i.e. the plan or control, for a given set of rules. Concern with determining which is the best plan for the given rules places that endeavour in the realm of routine design. However, if there is a process for modifying the rules themselves within the planning process then it is possible to produce designs with behaviours or structures outside the original state spaces. Computational systems which exploit this concept are now being developed.

Such computational systems make use of a variety of processes, chief amongst them are the following:

(i) combination
(ii) mutation
(iii) analogy
(iv) first principles
(v) emergence

(i) *Combination*—as a creative design process combination involves the addition of components from two separate designs. This combination is expressed in terms of the addition of variables. One common computational model for carrying out this combination is based on modelling the design process as a genetic algorithm. Here the genetic process of cross-over is the analog of combination. Novel designs can be produced this way.

(ii) *Mutation*—as a creative design process mutation involves a modification to an existing design variable to produce a new design variable. Typical mutation operators include the algebraic and set theoretic operators. Thus, division, for example, divides a single variable into two like variables. Such an operation can affect the resultant topology of the artefact. Mutation is also a process in genetic algorithms.

(iii) *Analogy*—analogy is defined as the product of a process in which specific coherent aspects of the conceptual structure of one design are matched with and transferred to another design. Based on the nature of the knowledge transferred to the new design, analogical reasoning processes can be placed into one of the two classes of transformational analogy or derivational analogy. Transformational analogy adapts the structure of a previous design to be useful in the present design. Derivational analogy applies the design process used in a previous design to the production of the current design. The effect of transformational analogy is the introduction of new variables into the current design.

(iv) *First principles*—first principles relies on causal, qualitative or computational knowledge used abductively to relate intentions (functions) to behaviour and behaviour to structure without the use of compiled knowledge. Design using first principles is the least developed of the processes described so far.

(v) *Emergence*—emergence is the process whereby extensional properties of a design are recognised beyond its intentional ones, i.e. properties which were not intentionally explicit are recognised and made explicit. Computational models of emergence are only now being developed concentrating on shape emergence.

These two models of design—case-based design and creative design—are developments founded on concepts from artificial intelligence which have allowed an expansion of the possible roles of computers in the design process. This is the beginning of a redefinition of computer-aided design.

BIBLIOGRAPHY

Brown, D. and Chandrasekaran, B. (1989). *Design Problem Solving: Knowledge Structures and Control Strategies*, Morgan Kaufmann, San Mateo, CA.

Coyne, R. D., Rosenman, M. A., Radford, A. D., Balachandran, M. and Gero, J. S. (1990). *Knowledge-Based Design Systems*, Addison-Wesley, Reading.

Gero, J. S. (ed.) (1991). *Artificial Intelligence in Design '91*, Butterworth-Heinemann, Oxford.

Gero, J. S. (ed.) (1992). *Artificial Intelligence in Design '92*, Kluwer, Dordrecht.

Gero, J. S. and Maher, M. L. (eds) (1992). *Modeling Creativity and Knowledge-Based Creative Design*, Lawrence Erlbaum, Hillsdale, NJ.

Green, M. (ed.) (1992). *Knowledge Aided Design*, Academic Press, London.

Tong, C. and Sriram, D. (eds) (1992). *Artificial Intelligence in Engineering Design, Vols I and II*, Academic Press, Boston.

Rychener, M. (ed.) (1988). *Expert Systems for Engineering Design*, Academic Press, San Diego.