Case and Constraint-Based Apartment Construction Project Planning System: FASTrak-APT

Kyoung Jun Lee† *, Hyun Woo Kim, Jae Kyu Lee*, Tae Hwan Kim***, Chang Gon Kim**, Myoung Kyun Yoon***, Eui Jun Hwang***, and Hyun Jeong Park*****

*Korea Advanced Institute of Science and Technology, **Hyundai Information Technology, Co., Ltd., ***Hyundai Engineering and Construction, Co., Ltd.
*Dept. of MIS, 207-43, Cheongryang, Seoul 130-012, Korea. ** Technology Support Division, 140-2, Kye-dong, Chongro-ku, Seoul, Korea. ***Construction Management Dept., 140-2, Kye-dong, Chongro-ku, Seoul, Korea.
*{leekj, jklee}@msd.kaist.ac.kr, **kimhw@msd.kaist.ac.kr, ***mission3@hyundai.hdec.co.kr

Abstract

To effectively generate a fast and consistent apartment construction project network, Hyundai and KAIST have developed a case and constraint based project planning expert system for apartment domain, FASTrak-APT, inspired by the fact that human expert project planner uses previous cases for planning a new project and modifies them using her/his knowledge on domain constraints. This large-scale, case based and mixed-initiative planning system integrated with intensive constraint-based adaptation utilizes semantic level meta-constraints and human decisions for compensating incomplete cases imbedding specific planning knowledge. The case and constraint based architecture inherently supports cross-checking cases with constraints during the system development and maintenance. This system has drastically reduced the time and effort required for initial project planning, improved the quality and completeness of the generated plans, and is expected to give the company the competitive advantage in contract bids for new contracts.

Project Planning Problem in Construction

Generation, verification and modification of construction project schedule networks in PERT/CPM (Project Evaluation and Review Technique/Critical Path Method) chart, as shown in Figure 1 is an essential task for successful project planning and management in the construction industry. However, generating a project network of a specific construction project is a time consuming and difficult task. For example, even for a senior engineer with 10-years of experience in construction planning, it takes a couple of days to make a project network of an apartment building construction. In order to compete with other companies for a contract, it is critical for a construction company to quickly generate a good and consistent project plan.

Hyundai Engineering and Construction (HDEC) -- a Korea based leading construction company world-wide -- has dedicated to the development of a series of expert systems for the automatic generation, verification, and modification of construction project networks. This five-year project has been performed in cooperation with the Intelligent Information Systems Laboratory in Korea Advanced Institute of Science and Technology (KAIST) and Hyundai Information Technology (HIT) Co.

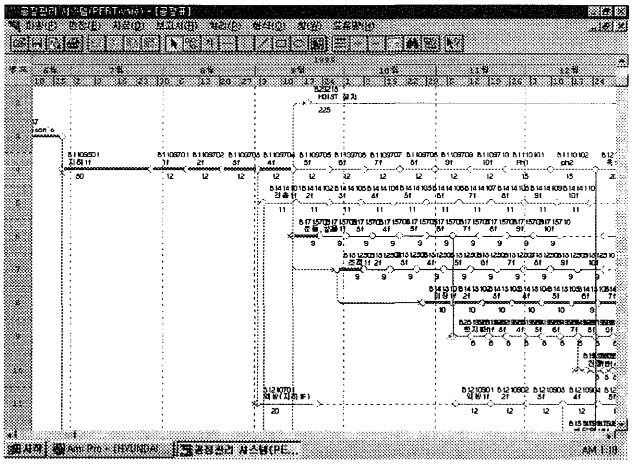

Figure 1. An Illustrative Construction Project Network in PERT Form

Rationale for Using Case and Constraint-based Approach

For a human project planner, it is very hard job to newly generate a project network without any previous cases because the amount of required knowledge is too large and complex. Therefore, human experts usually look for a similar previous project network for reference and get

†Visiting Scientist, until August 1997, at the Robotics Institute of Carnegie Mellon University, Pittsburgh, PA 15213-3891, USA. leekj@cs.cmu.edu

previous human expert's knowledge by referencing previous similar network. After finding the most similar network, they modify it to be fitted for the current project by using their own domain knowledge.

To simulate the expert's behavior of using a past similar case, the *case based reasoning approach* is adopted as a fundamental AI technique in this project. So far, most studies in CBR are developed for toy problems except for some systems (Aamodt and Plaza 1994), but this project is a full scale study applicable for the real world situation and deals with very large cases, each of which usually consists of hundreds of activities and hundreds of precedence relationships. In addition, as Shrobe pointed out in the excellent review of past IAAI Conferences (Shrobe 1996), there has been a significant gap between practical uses of case-based reasoning, which are limited almost totally to case retrieval, and the full paradigm as it emerged in the research community, which involves not just retrieval but also case adaptation, debugging, and so on. However, our system deals with case adaptation and maintenance intensively as well as case retrieval in large-scale real world context.

A new case usually has some discrepancies with the past cases. So a retrieved case has to be modified to fill the gap between them. In construction domain, the modification implies the addition and deletion of some activities and their associated precedence relationships. For instance, suppose the past case is a 20-floor building while the new one is for 18-floor building. Then we have to delete the activities for the 19th and 20th floors. After deletion, the succeeding activities should be pulled forward, unless any constraint is violated. In order to effectively modify the previous network to be suitable for the new construction project, we adopt a *constraint-based case adaptation approach.*

For our first attempt, in 1995, the construction of apartment complex was selected as the first application field because the planning task for apartments is relatively simple and can effectively employ the Case Based Reasoning (CBR) with the constraint based verification and modification. We named the apartment construction project planning system as FASTrak-APT.

Integration of the case-based approach and constraint-based approach has the following two advantages. First, construction domain constraints acquired from field experts compensates incomplete cases imbedding specific planning knowledge and improves the system's performance consequently. Secondly, during the system development, cross-checking of cases with constraints has improved the quality of both of them. Through the cross-checking process, the system developers could refine the previous cases to the high-quality referential cases, and simultaneously validate and verify the domain constraints.

Previous Approaches

In the area of project management, there has been a lot of research and development on project scheduling methods and management techniques assuming that a project network is given to the project manager (Bent and Thumann 1994). However, since the earliest research prototype CONSTRUCTION PLANEX (Hendrickson et al. 1987), there has only been a limited amount of research to automate or support the generation of progress networks using Artificial Intelligence or knowledge-based techniques, such as GHOST (Navinchandra, Sriram, and Logcher 1988), SIPE-2 (Kartam, Levitt, and Wilkins 1991), and HISCHED (Ory and Abraham 1995). To run these systems, users have some burden of inputting much information about activities, because these systems are not designed to utilize past cases. OARPLAN (Winstaniey, Chacon, and Levitt 1993), a model based planning system, utilizes past cases, but its user has to input the precedence relationships between activities. In contrast to these systems, the system that we have developed in this project doesn't require the users to input any precedence constraints because the system utilize past cases containing precedence constraints.

Zhang and Maher used a case-based reasoning method for the structural design of buildings (Zhang and Maher 1995). They claim that CBR as a design model is appealing intuitively because much of design knowledge comes through the experience of multiple, individual design situations. This also holds true in construction planning situation. For many domains where construction planning knowledge is difficult to acquire and may not be objectively applicable, the case-based paradigm can provide a model for the acquisition and reuse of specific planning knowledge because previous cases contain much valuable knowledge in themselves.

It is the first time for Hyundai people to adopt an AI technique to project planning. Previously human planners have created the project plan without any automated aid. They have not had structured case base accumulating past project cases either. Some other companies have tried database oriented approach, but it is not effective because it requires much knowledge and effort to modify a project network. For these reasons, Hyundai and KAIST considered using AI techniques: case-based technique for reusing and accumulating good cases and constraint-based technique for adapting a past case according to the construction knowledge.

Application Description

Figure. 2 shows the architecture of FASTrak-APT. The system supports a *mixed-initiative planning* procedure (Veloso 1996). A user can interact with the system by inputting the design specification, accumulating good cases into the case base, selecting constraints among the relaxation candidates, and informing the system of his/her intentions or management strategies contingent to project situation.

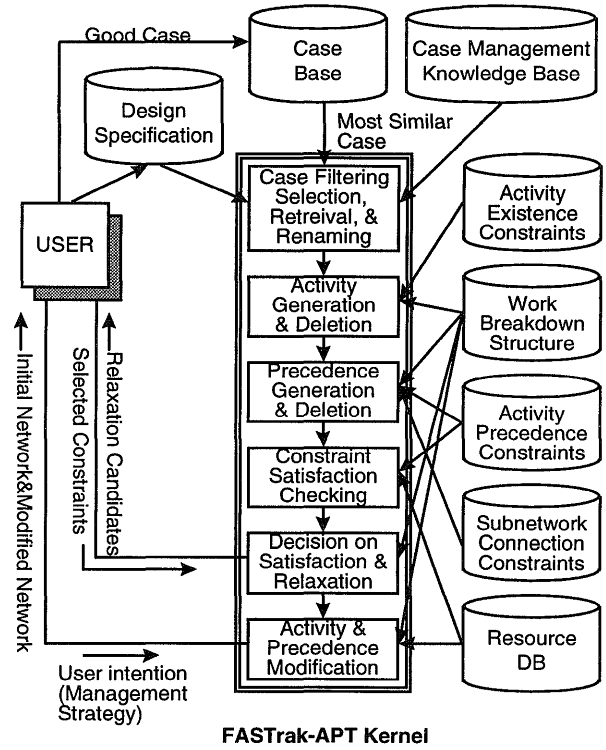

Figure 2. Architecture of FASTrak-APT and the Main Procedure of its Kernel System

Knowledge Representation

We use frame-based representation scheme for representing design specification, cases, and constraints. A design specification of an apartment is illustrated as follows.

{{Hukseok_Hyundai_Apt_Project
is-a : Project
name : Hukseok Hyundai Apt
address : Hukseok, Seoul
start-date : 1995/05/01
due-date : 1996/06/21
ground-type : Flatland
topography : (clay 40 %) (fragile-rock 60 %) (soft-rock 0 %) (solid-rock 0 %)
number-of-building : 1
area : 5707
construction-area : 571
household : (36 68)
}}
{{ Hukseok_Hyundai_Apt_1
is-a : APT
part-of : Hukseok_Hyundai_Apt_Project
apt-type : Corridor
base-last : 1
ground-last : 10
pent-last : 1
phases : 2
start-date : 1995/05/01
finish-date : 1996/06/21
household : (36 34)
construction-method : PC-framing, ...
}}

The kernel system uses the project case base, case management knowledge base, database for WBS (Work Breakdown Structure) and resources, and constraint base containing knowledge about activity existence, precedence relationship between activities, and subnetwork connection. We explain the main procedures of the kernel system one by one.

Case Filtering, Selection, Retrieval, and Renaming

When the user inputs the design specification of the current project, FASTrak-APT selects the most similar case using a two-step procedure. The first one is *case filtering*. Since every case in case base cannot always be adapted for the current project, we should filter out cases which, if adopted, could lead to adaptation failure. We have the case filtering knowledge in the case management knowledge base. The filtering knowledge has been accumulated through the experience of adaptation failure during system development and testing. For example, the variable *phases* is the one of the variables used for the filtering process. The filtering module filters out the cases which have different *phases* value with that of the current case, because the variable determines the grouping of concurrent activities and therefore significantly affects the topological shape of project networks. After the filtering process, we select the most similar case among the remaining candidates using the similarity measure. For the effective retrieval of suitable cases, we extracted six important properties from domain experts, as follows, and use the weighted sum of the distance functions of the variables.

$$Similarity = \frac{\sum_i W_i}{\sum_i W_i X_i}$$

W_i = Weight of each variable,
X_1 = Difference in number of floors,
X_2 = Difference in employed construction method,
X_3 = Difference in number household,
X_4 = Difference in building space,
X_5= Difference in ground type, and
X_6 = Difference in topography of buildings.

After the selection and retrieval, the system renames the activities and their precedence relations of the retrieved case using a naming rule which gives the unique codes to entities.

Activity Generation and Deletion

The retrieved case usually needs some activities to be added or has some activities to be deleted, so the next step is activity generation and deletion. There are two types of activity generation and/or deletion in our system. The first is caused by the difference of the number of floors (of ground or basement) between the selected case and a new project. For instance, if the selected case is a 20-floor apartment while the new project is an 18-floor building, then the activities for the 19th and 20th floor should be deleted.

The other is caused by the difference between employed construction methods. For example, if the selected case used reinforced concrete framing method, while the new project will use the pre-enforced concrete framing method, RC framing (b11097) activities should be deleted and PC framing (b11099) activities should be generated. The following object represents this constraint.

```
{{Construction-Method-Constraint-1
  is-a          : construction-method-constraint
  method-type   : framing
  value         : PC-framing
  use           : b11099
  delete        : b11097
}}
```

Precedence Relationship Generation and Deletion

When the activities are generated or deleted, the precedence relations associated with them should be created or removed. We integrate the two kinds of methods for generating and deleting the precedence relations: *constraint-based approach* and *principle-based approach. Constraint-based method* uses precedence constraints acquired from domain experts. If the newly generated activity has an associated precedence constraint, it can be converted to a new precedence relationship satisfying itself. In addition, when planning multi-apartment building construction, we need to connect the network of each building. To do this, we use subnetwork connection constraints which define the inter-building activity precedence relationship.

The *principle-based method* uses general network principles for maintaining project networks. Bell's work (Bell 1989) can be classified into this. We maintain the soundness of a project network by keeping the following basic principles.

1. There should be no cycle in the network.
2. There should be no isolated activity.
3. There should be only one start node and only one end node.

For example, in the case of 18-floor building case, if an 18th floor activity succeeded by an activity is deleted, the 17th floor activity of the same kind as the 18th should be succeeded by the activity. As such, this approach is not based on knowledge but based on logical rationale.

Constraint Satisfaction Checking

Constraints are used for verifying the current project network as well as for generating relevant activities and relationships. The first step in constraint satisfaction checking is calculating the earliest start time, earliest finish time, latest start time, latest finish time of all activities using CPM (Critical Path Method). With these values, we check the satisfaction of the constraints. We have acquired approximately 430 domain constraints about activity existence, activity precedence relationship, and subnetwork connection For the convenient representation and maintenance of the large number of constraints, we should represent them in semantic level. On the other hand, for the reasoning efficiency of the system, we should tightly couple the constraints into the activities. To satisfy the two criteria, i.e. user convenience and reasoning efficiency, we developed a *meta-constraint representation method* where a semantic level constraint for users is converted to a couple of instantiated constraints.

In addition, most constraints should be able to have different parameter values contingent to the situation. For example, when we construct a 10-floor apartment, the first floor's plastering activity starts after starting the 3rd floor's framing activity. However, in the case of 15-floor apartment, the first floor's plastering activity should start after starting the 4th floor's framing activity. To support the contingency, we connect forward-chaining rules with the meta-constraint. The forward chaining inference is supported by the tool UNIK-FWD (Lee et al. 1994). Figure 3, 4, and 5 show an illustrative constraint and its associated rules for the above example, instantiation process using forward chaining inference, and the instantiated precedence constraint respectively.

```
{{ Precedence-Constraint-3-b
  is-a                   : precedence-constraint
  relationship-type      : FS  ; Finish-to-Start
  relationship-operator  : >=
  relationship-value     : 0
  value-type             : day
  predecessor-wbs        : framing
  predecessor-floor      : <contingent-value-1>
  rule-groups : (<contingent-value-1>
                 Precedence-Constraint-3-b-rule-group)
  successor-wbs          : plastering
  successor-floor        : <base-first>
  importance             : 0.7
}}

(fwd-RULE Precedence-Constraint-3-b-1
[RULE-GROUP Precedence-Constraint-3-b-rule-group]
(precedence-constraint-control   ^current-apt <apt>)
(apt ^frame-name (= <> <apt>) ^ground-last (<= <> 12 ))
 --> (new-value 'precedence-constraint-control
                'contingent-value-1 3))
(fwd-RULE Precedence-Constraint-3-b-2
[RULE-GROUP Precedence-Constraint-3-b-rule-group]
(precedence-constraint-control   ^current-apt <apt>)
(apt ^frame-name (= <> <apt>) ^ground-last (>= <> 13 ))
 --> (new-value 'precedence-constraint-control
                'contingent-value-1 4))
```

Figure 3. An Example of Precedence Meta-constraint and its Forward Chaining Rules

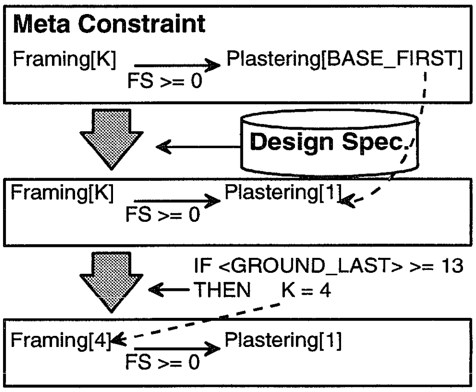

Figure 4. An Illustrative Instantiation of Meta Constraint

```
{{PC3-B-P2B1-0
    is-a                   : precedence-constraint-instance
    value-type             : day
    relationship-value     : 0
    relationship-operator  : >=
    relationship-type      : FS
    predecessor            : Hukseok_Apt_1-Framing-04
    successor              : Hukseok_Apt_1-Plastering-01
    importance             : 0.7
}}
```

Figure 5. An Instantiated Precedence Constraint

Decision on Satisfaction and Relaxation

We check the project network with the instantiated constraints. If a violated constraint is found, we should decide whether it will be satisfied or relaxed. There are two ways for the decision: automatic and manual. For automatic selection of the constraints to be relaxed, each constraint has an *importance* value. Constraints with the higher importance value should be satisfied preferentially. The other way is user selection. If the system shows the violated constraints, then user can select some of them to be relaxed. For example, the button 'Add All' of figure 6 can be used when a user wants to satisfy all of the violated constraints. If the user wants to relax some constraints, she or he click the 'Delete' button after selecting them in the lower box.

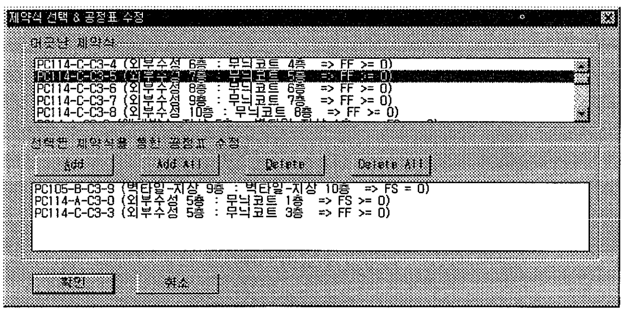

Figure 6. The Window for Selecting Violated Constraints

Activity and Precedence Modification

The modification occurs from the effort satisfying a violated constraint or satisfying user's intention. To satisfy a violated precedence constraint, we convert the constraint to a precedence relationship. Since adding a precedence relationship into the network can increase the project makespan, it is important to keep the makespan stabilized and simultaneously satisfy important precedence constraints. We use three methods for reducing the makespan. First, we can delete relatively unimportant precedence relationships. The precedence relationships which are not defined in the precedence constraint base are good candidates to be deleted. Of course, deletion of a precedence relationship should be confirmed by the user. Secondly, we can reduce the value of the precedence relationship as far as satisfying its associated precedence constraint. Finally, we can reduce the duration of activities in the critical path. To keep from unreasonable duration reduction, the system uses the WBS database and considers the status of the resource utilization.

Evaluation and Enhancement of Solutions

The plans generated by FASTrak-APT have been proved by human experts to be technically sound, and even satisfied more constraints than the cases prepared by domain experts. Therefore, the results were used for enhancing case base, and the refined case base helped improve the quality of generated plans. The cross-checking and mutual enhancement is one of important benefits from the integration of case-based approach and constraint-based approach. One of remained issues we are working on for user satisfaction is flexible visualization of plans to support the demand on various levels of views from the different levels of users in organization.

Knowledge Acquisition, Implementation and Maintenance

Before entering the knowledge acquisition phase, we standardized and constructed three-leveled WBS (Work Breakdown Structure) database, which contains a lot of information such as average duration, hierarchical structure, location, and seasonal factor of each activity etc. Ten domain experts participated in the task. After that, we made a cross-table between all WBS activities and let a 9-year-experienced expert mark in the cell if the two activities have any technical precedence relationships. The result were checked and refined by other five experts. The collected technical precedence constraints could not be complete because there can be some managerial and conventional precedence relationships. However, we could observe that the lack of information is compensated by case base.

FASTrak-APT has been implemented on Windows 95 using Visual C++ and UNIK (Lee et al. 1994), an expert system tool developed by KAIST and IntelliNet Co. Since KAIST has the right to use and enhance the source codes of UNIK under the cooperation with IntelliNet Co., it is a good choice to use it for developing flexible and expandable systems. Currently, FASTrak-APT has about 430 meta-constraints that can create thousands of instantiated constraints (e.g. approximately 2000 ones for a 20-floor building). So far, we have accumulated 50 high-quality cases prepared and verified by human experts. The project team consists of one project manager, two research programmers, two application programmers, and three domain experts. The roles of each group are described in Table 1. Hyundai and KAIST have been extending this system for bridge construction and power transmission tower construction planning since September 1996.

Project Manager	1	Methodology Consulting Overall Manaeement & Control
Research Programmers (KAIST/HIT)	2	Knowledge Engineering Case Base Design Kernel System Development R&D on Bridge Case
Application Programmers (HIT/HDEC)	2	GUI Development Interface to Project Management S/W System Verification and Maintenance R&D on Power Transmission Tower
Domain Experts (HDEC)	3	Requirement and Specification Knowledge Transfer and Verification Case-base Construction System Validation Knowledge Maintenance

Table 1. Roles of Members in Each Group

Development Cost

The development costs of FASTrak-APT for the two years were calculated at approximately $42,000 for the hardware, $162,000 for the outsourcing, and $417,000 for the internal manpower. So, the total cost was about $621,000.

Application Use and Estimate of Payoff

FASTrak-APT has been used by the Construction Management Department since September 1996 and is now ready to be used for the construction sites. FASTrak-APT has been proved to reduce the effort required to generate an initial project plan from 7 man-days to 1 man-day. The cost of updating a plan, which occurs every three months per project, has been also reduced from 2 man-days to 0.5 man -day. The company expects to be able to reduce the effort to complete bid document from 10 man-day to 1 man-day if the system is enhanced to support aggregating resource utilization and the cost of activities calculation. The expected annual benefit using the above parameters is about $616,000. Using the FASTrak-APT relieved the company of the suffers from the deficiency of project management experts. In addition, the faster simulation and feasibility analysis will give the company the competitive advantage over other companies in contract bids for new contracts. The accumulated good cases and the digitalized and refined knowledge became an invaluable asset for the company. In addition, the company now uses the system in training employees for construction management.

Acknowledgments

This research is funded by Hyundai Engineering. & Construction Co., Ltd. and Hyundai Information Technology Co., Ltd. in Korea. We are particularly grateful to Executive Directors Jae Ho Yoon and Seok Boo Choi for their strong support and helpful advice.

References

Aamodt, A. and Plaza, E. 1994. Case-based Reasoning: Foundational Issues, Methodological Variations, and System Approaches, ***AZ Communications,*** 7(1):39-59.

Bell, C. 1989. Maintaining Project Networks in Automated Artificial Intelligence Planning, ***Management*** Science, 35(1): 1192-1214.

Bent, J. and Thumann, A. 1994. ***Project Management for Engineering and Construction (2nd edition),*** Fairmont Press.

Hendrickson, C., Zozaya-Gorostiza, C., Rehak, D., Baracco-Miller, E., and Lim, P. 1987. An Expert System for Construction Planning, ***Journal of Computing in Civil Engineering,*** 1(4):253-269.

Kartam, N., Levitt, R., and Wilkins, D. 1991. Extending Artificial Intelligence Techniques for Hierarchical Planning, ***Journal of Computing in Civil Engineering,*** 5(4):464-477.

Lee, J. K. et al. 1994. ***UNIK User's Manual*** (in Korean), Intelligent Information Systems Laboratory, Korea Advanced Institute of Science and Technology.

Navinchandra, D., Sriram, D., and Logcher, R. 1988. GHOST: Project Network Generator, ***Journal of Computing in Civil Engineering.,*** 2(3), ***239-254.***

Ory S. and Abraham W. 1995. Knowledge Based System for Construction Planning of High-rise Buildings, ***Journal of Construction Engineering and Management.,*** 121(2):172-182.

Shrobe, H. 1996. The Innovative Applications of Artificial Intelligence Conference: Past and Future, ***AZ Magazine,*** 17(3): ***15-20.***

Veloso, M. 1996. Towards Mixed-Initiative Rationale-Supported Planning. ***In A. Tate (Ed.), Advanced Planning Technology.*** Menlo Park, CA: AAAI Press.

Winstaniey, G., Chacon, M., and Levitt, R. 1993. Model-Based Planning: Scaled-up Construction Application, ***Journal of Computing in Civil Engineering.,*** 7(2):199-217.

Zhang, D. and Maher, M. 1995. Case-based Reasoning for the Structural Design of Buildings, ***Proceedings of IEA/AIE (Industrial and Engineering Applications of Artificial Intelligence and Expert Systems) 95, 14 1 - 150.***