INTEGRATING BUILDING CODE COMPLIANCE CHECKING INTO A 3D CAD SYSTEM

Tang-Hung Nguyen[1] and Eric Asa[2]

ABSTRACT

This paper outlines an integrated framework for a building design system capable of automatically building code compliance checking. Most previous studies have focused on the processing of design codes for conformance checking. In this work, the authors investigate 3D CAD (three dimensional Computer-Aided-Design) techniques to support automated code compliance checking during the design process. In particular, the paper addresses the following issues (1) Extraction of the knowledge from a building code document (e.g. International Building Code), and representation of the knowledge in a computer-interpretable format; (2) Data representation of building designs in a 3D CAD system. (3) Mapping building code information with building design information that enables the computer to assist designers with the automatic code compliance checking; and (4) Integration of the knowledge in building codes and the design knowledge into a computer framework. The proposed framework will be able to identify code requirements relevant to a particular building design and notify designers whether the design violates the building code.

KEYWORDS

Automated Building Design System, Building Code Compliance Checking, Building Data Modeling, Computer-Aided Design, Three-Dimensional Solid Modeling.

INTRODUCTION

Traditionally, design objectives concerning various performance aspects (e.g. building code compliance, energy efficiency, budget limit, constructability, etc.) of a building at a particular design phase are not evaluated until all design works for the building have been completed. Further, the evaluation task is usually conducted in a manual fashion through numerous meetings. This in turn makes the evaluation process time-consuming, potentially expensive, and prone to error, since the meeting participants or evaluators often become overwhelmed with a huge volume of project information and design criteria. One example of the traditional design evaluation tasks is checking for the compliance to building code of a

[1] Assistant Professor, Department of Construction Management and Engineering, North Dakota State University, 120H CME Building, Fargo, ND 58105. Telephone: 701-231-8646. Email address: hung.nguyen@ndsu.edu
[2] Assistant Professor, Department of Construction Management and Engineering, North Dakota State University, 120C CME Building, Fargo, ND 58105. Telephone: 701-231-7246. Email address: eric.asa@ndsu.edu

building component. In the case that the building design does not comply with the current building codes, the designer need to meet with other project participants, whose designs may have affects on the code violations, to determine how to appropriately modify the design so that the code violations can be avoided. Checking a building design for conformance with applicable building codes is a tedious, laborious, and complicated task. Misinterpreting or overlooking provisions of the building codes, which is often the case with inexperienced users, may lead to serious consequences. At present, with computer supports, the building code compliance checking can be handled with relative ease. For example, a knowledge-based computer system is capable of reasoning the logics contained in building codes to provide advice with diagnostic problems in code checking (Nguyen, 1996) and an 'expertext' system, a combination of the best features of expert systems and hypertext, facilitates the building code compliance checking using the semantically rich nodes of hypertext and the well-specified, computable links of expert systems (Casson and Stone, 1991). However, the existing computer software is unable to explicitly capture the building code rationale for selecting a building system or component that may affect the overall design performance, forcing designers to manually verify code violations. This shortcoming makes it difficult for the designer (e.g. architects and engineers) to determine how to adjust the design information in the case of design changes so that the building code regulations will be completely complied. The current research effort is aimed at developing a 3D CAD system capable of automatically identifying, as the design proceeds, code requirements relevant to a particular building design and notify designers whether the design complies with the building code. Specifically, the major research tasks include:

- Extraction of the knowledge from a building code document (e.g. International Building Code), and representation of the knowledge in a computer-interpretable format;
- Data representation of building designs in a 3D CAD system;
- Mapping building code information with building design information that enables the computer to assist designers with the automatic code compliance checking; and
- Integration of the knowledge contained in building codes and the design knowledge into a computer framework. The proposed framework will be able to identify code requirements relevant to a particular building design and notify designers whether the design violates the building codes.

RESEARCH METHODOLOGY

The research methodology includes the following steps:

1. Study previous work on computer-based systems for checking designs against building regulations. This study identifies the needs for the development of an automated building code checking system.
2. Examine a building code document, e.g. International Building Code (IBC), from which the knowledge of one or two Chapters will be extracted for the prototype implementation.

3. Extract the knowledge from the selected sections in the IBC and represent them in a computer-interpretable format.
4. Develop a framework linking building product information with relevant building code information, which enables the automated code compliance checking
5. Implement the framework in a CAD system (e.g. AutoCAD by Autodesk, Inc.).

PREVIOUS WORK

In the last decade, the applications of advanced information technology (IT) to the development of computer tools that assist in accessing, interpreting and applying regulatory information have been given particular attention. Examples of such IT applications include hypertext (Kumar et al., 1995), expert systems (Rosenman et. al., 1986; Dym et al., 1988; Sharpe, 1991; Moulin, 1992; Frye et al., 1992; Heikkila and Blewett, 1992; and Delis and Delis, 1995), expertext (Casson and Stone, 1991), and intelligent systems (Nguyen, 1996, Nguyen et. al., 1996).

While many computer-based prototype systems for checking designs against selected regulations have been developed during the last decade, these systems are usually stand-alone systems that require the user to input manually the pertinent information describing the building. Unlike the previous work, the proposed CAD system is designed to quickly retrieve basic building data (e.g. geometry and functionality) of a particular building design stored in the CAD system to deduce more complex building information (e.g. spatial relations, areas and widths of a particular building component) required for building code compliance checking.

THE BUILDING CODE DOCUMENT

The first step taken in this research is to select a building code document. The three building codes in widespread use previously in the United States included: the Uniform Building Code of the International Conference of Building Officials (ICBO), the Southern Building Code of the Southern Building Code Congress International, Inc. (SBCCI), and the National Building Code of Building Officials and Code Administrators International, Inc. (BOCA). Recently, these three model codes have been reformatted into a single national model code, named International Building Code, which was developed by the International Code Council, an agency made up of representatives from the three model-code organizations ICBO, SBCCI, and BOCA (IBC, 2000). Therefore, the International Building Code (IBC) was selected as the building code document for implementing the automated code checking prototype system. To develop an automated code checking system, the regulatory requirements in the building code are used as the main source of information to establish its knowledge base. However, code texts are written in a natural language format that is unable to be encoded directly to the computer-based system. Thus, the procedural logic of the code should be restructured and formalized into a suitable format for the implementation of the computer application. The regulatory source used to extract the knowledge for the code checking system includes code regulations found in two Chapters (e.g. 'Fire-Restrictive Construction' and 'Means of Egress') of the IBC document.

EXTRACTION AND REPRESENTATION OF THE BUILDING CODE KNOWLEDGE

The next step taken in the research is to extract the necessary knowledge contained in the IBC Chapters "Fire-Restrictive Construction" and Means of Egress". The knowledge is then represented in a computer-interpretable format to support the automated building code checking. The development of such a format requires suitable knowledge representation forms. Rule-based and frame-based approaches are two techniques that have been examined for selection. Most of code provisions consist of a number of conditional statements which, when satisfied, lead to a set of requirements or consequences.

Table 1. **IBC Knowledge and Representation**

IBC Knowledge	IBC Representation
Openings in Fire Walls *705.8:* Each opening or window through a firewall within a building that is not equipped with an automatic sprinkler system shall not exceed 120 square feet (11 m^2) in size and the aggregate width of the opening shall not exceed 25 percent of the length of the wall (IBC, 2000).	Component: *Fire Wall* Element to be Checked: *Openings* Code Reference: *IBC-Section 705.8* Automatic Sprinkler System: *Y/N* Req'd Opening Area: $\leq$ *120 s.f (11m^2)* Req'd Opening width: $\leq$ *0.25xLengthOfWall*
Fire-Resistance of Structural Members: *713.2.1* requires individual structural members be fully protected on all sides for their entire length if they are required to have a fire-resistance *713.2.2* requires the full height of a column to be protected, including its connection to beams and girders, even if it extends through a rated ceiling assembly (IBC, 2000)	Component: *Structural Member* Element to be Checked: *Fire Protection* Code Reference: *IBC-Section 713.* Required Fire Resistance: *Y/N* Connected-to: *Beam/Girder* Required Length of protection: *Length of the structural member and connection*
Egress Doors *1003.3.1:* requires that egress doors should never be blind doors, should be not less than 80" (2032 mm) high, must have a minimum clear width of 32" (813 mm), measured from the face of the door to the stop when the door is open 90^o. (IBC, 2000)	Component: *Egress Door* Element to be Checked: *Height/Width* Code Reference: *IBC-Section 1003.3.1* Is-A Blind Door: *Y/N* Required Height: $\geq$ *80" (2032 mm)* Required Width: $\geq$ *32" (813 mm)*

Hence, a rule-based technique seems the most natural for representing the knowledge contained in building codes, as the form seems to match that of the code provisions. In addition, a frame that is similar to a form of data-structure may be used for representing stereotypical (and hierarchical) information of a building. For example, a building object 'Interior Wall' may have the following attributes: Object ID, Function, Material, Location, and Number of Openings (Figure 1). Table 1 below presents examples of the knowledge extracted from the IBC document and its frame-based representation.

3D SOLID MODELING REPRESENTATION OF BUILDING COMPONENTS

Unlike traditional CAD systems, which are merely tools to graphically represent a design by means of primitives such as points, lines, and arcs, solid modeling provides various representation schemes using primitives such as vertices, edges, and faces. Such primitives are suitable for a complete and unambiguous description of design objects (Zeid, 1991). Furthermore, in solid modeling systems, the objects observed in buildings can be quickly created as blocks, thus making it simple to perform calculations of volume and mass properties of the building objects. Such building information is necessary for different design activities (e.g. building code compliance checking) throughout the design and construction process. There are various 3D geometric representation techniques including Boundary representation (Brep), Constructive Solid Geometry, Parametric Representation, and Feature-Based Representation. This research makes use of 3D boundary representation or Brep of solid modeling for representing building objects since Brep provides both geometric and topological data necessary for deducing complex information (e.g. spatial relationships between building components) to support code compliance checking. A detailed description of these solid modeling representation techniques is beyond the scope of this paper and the reader may refer to other references such as (Kalay, 1989) and (Zeid, 1991) for more information.

THE PROPOSED FRAMEWORK

Once the building code knowledge has been extracted and organized in the computer-interpretable format, a framework linking the information describing building components and building code knowledge can be developed. Specifically, the information of building codes associated with each of building components will be linked and stored in an external database. The framework functions as a bridge (a mapping tool) between building designs produced in a CAD system and the database enabling the automated code compliance checking. For example, a building object 'Interior Wall' that functions as a Fire Wall (Figure 1) can be linked to building code knowledge concerning the firewall stored in database. The element to be checked for code compliance could be an opening *contained in* the firewall. Geometric data (e.g. location or dimension) describing the building component available in CAD system can be extracted to deduce the actual opening area or width, which will be mapped with the relevant building code requirements (e.g. required opening area, width, etc.) to verify code violation. Additionally, the *contained-in* information that is referred to as a spatial relationship between the opening and the firewall can be automatically deduced in a 3D CAD system (Nguyen, 1999). The description of the algorithms to deduce different spatial relationships between building components such as *adjacency, connection, containment, separation,* and *intersection*, can be found in (Nguyen, et. al., 2004).

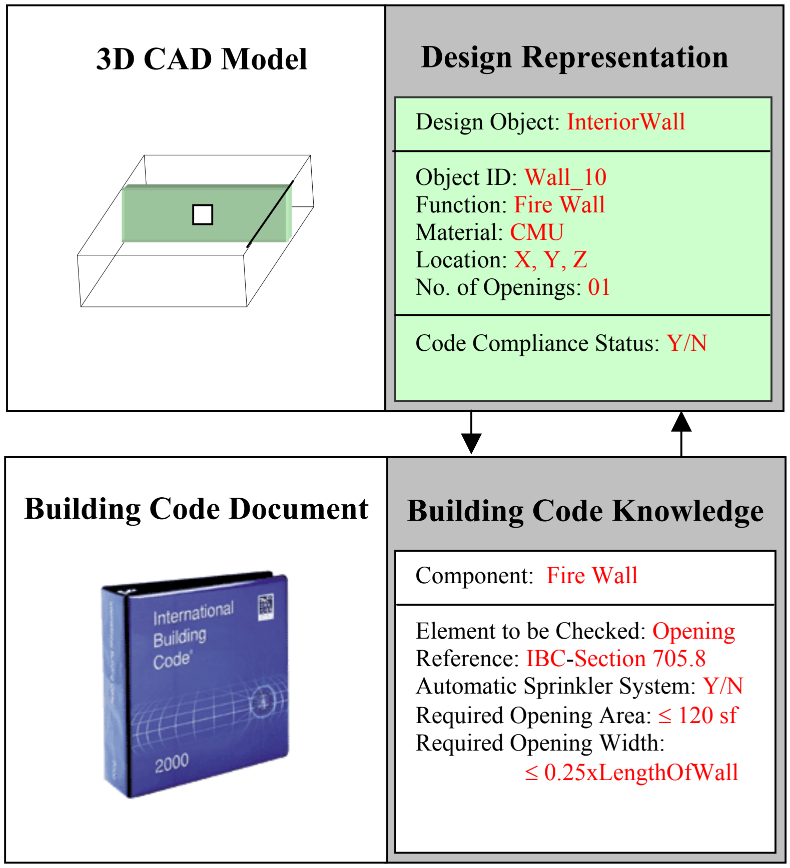

Figure 1. **The Framework Linking Building Design Information and Building Code Knowledge**

IMPLEMENTATION

The architecture of the automated system comprises of three major components: a 3D CAD system, an application development tool, and an external database (Figure 2).

The system architecture

The 3D CAD system (i.e. AutoCAD by Autodesk, Inc.), as a key component of the prototype system, undertakes various function including representations of 3D geometric data as well as functional attributes of building components. The application development tool selected for this prototype system is ObjectARX. This development tool is an AutoCAD® Runtime Extension programming environment that includes a number of C++ dynamic link libraries (DLLs) that enable developments of AutoCAD applications (ObjectARX, 2002). The main reason for selection of ObjectARX as a programming environment for the implementation of this work is that the set of DLLs (Dynamic Library Links) in ObjectARX can operate directly with core AutoCAD data structures and code, thus providing suitable mechanisms for accessing AutoCAD database to extract geometric data of building components that are necessary for deducing spatial relations and building code compliance checking. The development tool ObjectARX is used to create new AutoCAD commands (e.g. CodeComplianceChecker) to check building designs for code violations. As it is a compiler of ObjectARX applications, Visual C++ is used as a programming development environment for the user interface. The third component of the system architecture is the database, i.e. Microsoft Access, where the information contained in the selected building code is stored and extracted as needed for explaining code violations of the building design.

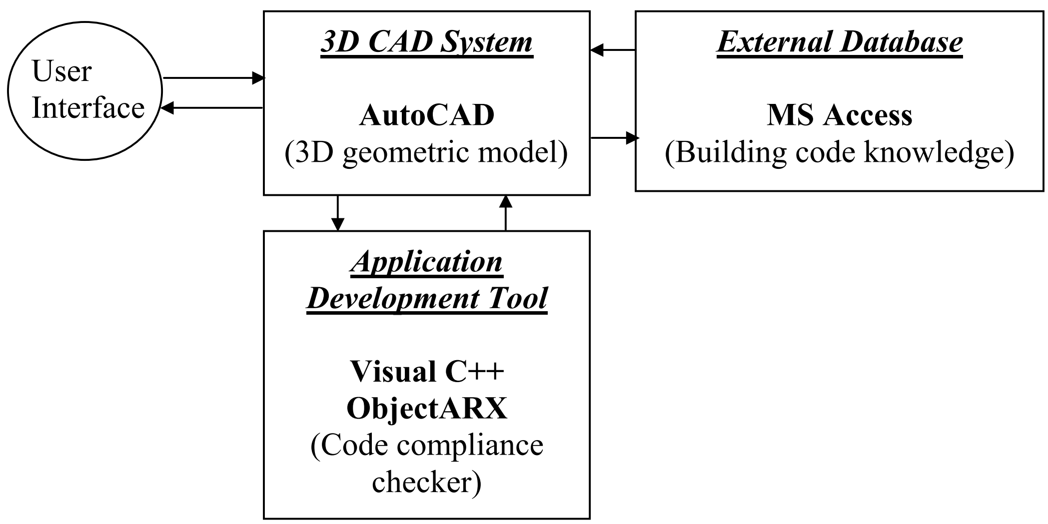

Figure 2. **Architecture of the Automated Code Compliance Checking System**

BUILDING CODE COMPLIANCE CHECKER

The current research effort is aimed at developing a 3D CAD system capable of automatically retrieving the basic building data (e.g. geometry and functionality) of a particular building design stored in the CAD system to deduce more complex building information (e.g. spatial relations, areas and widths of a particular building component) required for building code compliance checking. This can be done by means of two new AutoCAD commands, SpatialRelationGenerator (SRG) and CodeComplianceChecker (CCC), which were developed and inserted into the CAD system. The SRG command is used to extract basic building information necessary for deduction of spatial relations between building components, while the CCC is designed to enable designers, during the design process, to keep track on the effect of various designs on the overall performance of the project with respect to building code requirements. The development of these two AutoCAD commands leads to a need for the development of algorithms for code compliance checking and deducing spatial relationships among building components to be required for code compliance checking.

The algorithms for checking the compliance of the building code requirements were developed through two steps. First, the building code document is classified into specific categories to be considered during the building compliance checking. The IBC document was written with multi subdivisions such as Chapters, Sections, Subsections, Articles, etc.; the information under this hierarchical structure cannot be incorporated directly into the computer-based system. It must be reorganized in such a way that all compliance topics contained in building codes can be verified in a systematic fashion. For example, the compliance categories to be checked for Chapter 7 in the IBC include: Exterior walls, fire walls, fire barriers, shaft enclosure, so on. Each category may be further divided into several compliance subcategories, depending on its complexity and nature.

Next, based on the procedural logic for each specific compliance category, algorithms are developed in such a manner that the data required for checking is systematically collected whilst unnecessary queries are avoided. The interpretation of some code provisions could be a difficult task because of multiple cross-references and the qualitative nature of information. In addition, the existing information contained in the IBC is not always explicit; thus, it cannot be interpreted in a proper fashion without using appropriate analysis capability. Below is an example explaining the procedural logic for the subsection 705.8 Openings in a firewall which has been extracted from (IBC, 2000).

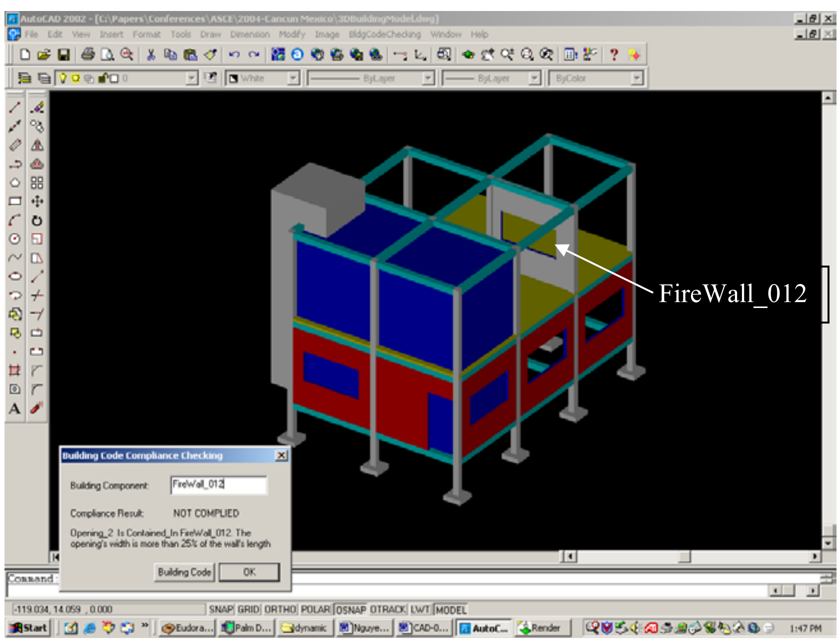

Figure 3. **Example Output from the Prototype System**

To check for the code compliance of an opening in the firewall, its geometric data available in CAD database first is retrieved to verify if the opening is *contained-in* a firewall (e.g. Opening_ 2 is *contained-in* FireWall_012). The *contained-in* information that is referred to as a spatial relationship between the opening and the fire wall can be automatically deduced in a 3D CAD system. The algorithms for deducing various spatial relationships between building components including *adjacency, connection, containment, separation,* and *intersection* have been developed and presented in prior work by (Nguyen, et. al., 2004). The next step in code compliance checking for the opening is to extract the geometric information about the wall's dimensions in the CAD database for computing the area and the length of the wall. Finally, the system checks if the wall area is less than 120 square feet and the opening width is not be more than 25 percent of the wall's length. Figure 3 shows an example output from the proposed system.

CONCLUSIONS

The design of a building component should be checked against the applicable building code requirements as the design progresses so that design changes or reworks due to building code violations, that are usually costly and time-consuming can be avoided. With the support of current advanced CAD technologies, the building code compliance checking can be performed with relative ease. This paper presents a computerized framework underlying an automated building code compliance checking system. The automated framework has been implemented in a 3D solid modeling system, where building components are represented by both geometric (e.g. dimensions, locations) and non-geometric data (e.g. fire resistant rate of a firewall). These basic data can be extracted to deduce more complex information (e.g. spatial relations) that can be used for building code compliance checking. The knowledge contained in Chapters "Fire-Restrictive Construction" and "Means of Egress" of the International Building Code (IBC) has been extracted and represented as a computer-interpretable format suitable for the implementation of the computer application. The proposed framework is designed to link the building code knowledge and information describing building components that enables the automated code compliance checking. In particular, the proposed system is able to interpret building designs as well as to deduce spatial relationships among individual building components to be needed for the code compliance checking. The knowledge base of the automated system should be expanded to completely address all building code provisions contained in the IBC document. Furthermore, a standard data exchange protocol, e.g. IFC (Industry Foundation Classes), needs to be incorporated into the proposed system to enable the interoperability among different AEC applications. The IFC is an object-oriented protocol for sharing data between dissimilar AEC (Architecture, Engineering, and Construction) software, which was developed by the International Alliance for Interoperability (IAI, 2003).

REFERENCES

Casson A. and Stone D. (1991). "An Expertext System for Building Standards". CIB W78 and CBS Joint International Workshops on Computer Integrated Construction and Computers and Building Standards, Montreal, Quebec.

Delis, E.A. and Delis, A. (1995). "Automated Fire-Code Checking Using Expert-System Technology". *Journal of Computing in Civil Engineering*, Vol. 9, No. 2, pp. 141-156.

Dym, C., Henchey, R., Delis, E.A., and Gonik, S. (1988). "A Knowledge-Based System for Automated Architectural Code Checking". *Journal of Computer-Aided Design*, Vol. 20, No. 3, pp. 137-145.

Frye M.J., Olynick D.M., and Pinkney R.B. (1992), "Development of an Expert System for the Fire Protection Requirements of the National Building Code of Canada", *Joint CIB Workshops on Computer and Information in Construction*, Montreal, Quebec, Canada, pp. 215-226.

Heikkila Eric J. and Blewett Edwin J. (1992). "Using Expert Systems to Check Compliance with Municipal Building Codes". *Journal of the American Planning Association*, Vol. 58, No.1, pp. 72-80.

IAI (2003). "International Alliance for Interoperability" access to http://www.iai.org. Industry Foundation Classes IFC Release 2.X, December 2003.

IBC (2000). "2000 International Building Code". International Code Council, Inc., Falls Church, Virginia

Kalay Y.E. (1989), "Modeling Objects and Environments. Wiley Interscience, New York.

Kumar, B., McLeod, I.A., and Neilson, A. (1995). "Representing Codes of Practice Information", *Proceedings of the Sixth International Conference on Computing in Civil and Building Engineering*, Berlin, Germany, Vol. 1, pp. 209-213.

Moulin, B. (1992). "Automated Generation of Rules from the National Building Code Text", *Proceedings of the Joint CIB Workshops on Computers and Information in Construction*, National Research Council, Institute for Research in Construction, Montreal, Canada, p. 343.

Nguyen T.H. (1996). " Building Inspection With Automated Code Compliance Checking", M.S Thesis in Building Engineering, Center for Building Studies, Concordia University, Montreal, Canada

Nguyen T.H., Bedard C., and Ha K.H. (1996). "Automated Code Checking for Building Inspection", *Proceedings of ASCE Conference on Computing in Civil Engineering*, Anaheim, California.

Nguyen T.H., Oloufa A.A., and Nassar K. (2004). "Algorithms for Automatically Deducing Topological Information", *Journal of Automation in Construction*, Vol. 14/1, pp. 59-70. (

ObjectARX (2002). "ObjectARX Software Development Kit for AutoCAD 2002", Autodesk Inc.

Rosenman M.A., Gero J.S., and Oxman R. (1986), "An Expert System for Design Codes and Design Rules", *Journal of AI in Engineering: Tools and Techniques*, pp.745-758.

Sharpe R. (1991). "BC Aider-Expert Assistant for the BCA". *Proceedings of the Fourth International Conference on Computing in Civil and Building Engineering*, Tokyo, Japan, July 29-30, pp. 247.

Zeid I. (1991). "CAD/CAM Theory and Practice", McGraw-Hill.