Carnegie Mellon University
Research Showcase @ CMU

Department of Civil and Environmental Engineering

Carnegie Institute of Technology

1986

Graphics based preliminary structural design

Mary Lou. Maher
Carnegie Mellon University

Weiguang Zhang

Carnegie Mellon University.Engineering Design Research Center.

Follow this and additional works at: http://repository.cmu.edu/cee

This Technical Report is brought to you for free and open access by the Carnegie Institute of Technology at Research Showcase @ CMU. It has been accepted for inclusion in Department of Civil and Environmental Engineering by an authorized administrator of Research Showcase @ CMU. For more information, please contact research-showcase@andrew.cmu.edu.

NOTICE WARNING CONCERNING COPYRIGHT RESTRICTIONS:
The copyright law of the United States (title 17, U.S. Code) governs the making of photocopies or other reproductions of copyrighted material. Any copying of this document without permission of its author may be prohibited by law.

Graphics Based Preliminary Structural Design

by

M. L. Maher, W. Zhang

EDRC-12-04-86 ^

September 1986

Graphics Based Preliminary Structural Design

Mary Lou Maher and Weiguang Zhang

ABSTRACT

The development of graphic based preliminary design programs is appealing because preliminary design involves the consideration of alternatives as concepts or sketches. The use of such a program would allow engineering students and new graduates to gain design experience directly, through their own trial and error. This paper describes a graphic based preliminary design program that was developed to facilitate teaching architect and engineering students about structural design.

Graphics Based Preliminary Structural Design

Mary Lou Maher, A.M.ASCE[1] and Weiguang Zhang[2]

1. Structural Design Of Buildings

The structural design process can be divided into three phases: preliminary design, analysis, and detailed design. During preliminary design, alternative structural configurations are explored and one or more alternatives are selected for further consideration. During analysis, a structural configuration is analyzed and modified until its response to expected loads is satisfactory. During detailed design, the structural configuration is completely specified such that all relevant design constraints are satisfied. The use of graphical interfaces is becoming popular during the analysis and detailed design phases, where graphics provides a friendly interface to conventional analysis and component sizing programs. The development of graphic based preliminary design programs is appealing because preliminary design involves the consideration of alternatives as concepts cr sketches.

The formal education of a structural engineer typically emphasizes behavior and analysis cf structural members and systems, that is. the mechanics of evaluating the response of a specified system to its intended environment. The student has very little exposure to th,3 synthesis of structural systems. The practicing structural engineer slowly acquires enough experience, through analysis and detailing, to be prepared to undertake a design problem. Most of the experience of the new graduate is acquired indirectly. The use of a graphics based preliminary structural design program would allow engineering students and new graduates to gain design experience directly, through their own trial and error.

This paper describes a graphics based preliminary design program that was developed to facilitate teaching architect and engineering students about structural design. Current work >.n this area involves the development of an 'intelligent' graphical environment in which tf ·? program contains knowledge about both the graphical representation of the design and ir j structural implications of the design.

2. SDU: A Structural Design User Interface

SDU[11][6] is an environment in which the engineer can graphically 'build[1] a structure using a mouse to move icons representing structural components. The structure can be bu t from components such as beams and columns or from user-defined subsystems such as rig J frames and braced frames. After the structural system is graphically defined, SDU provides 3 set of design aids to help the engineer evaluate the design. These aids include an on-I/* ? calculator, a database of steel sections, a stability checking routine, and a finite elemert

---

[1] Assistant Professor of Civil Engineering, Carnegie Mellon University, Pittsburgh, PA 15213

[2] Rese2ᵣch Assistant, Department of Civil Engineering, Carnegie-Ve'ion University, Pittsburgh, PA 15213

University Libraries
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

program. The interaction among these aids is handled by SDU and is transparent to the user.

2.1. Design Considerations of SDU

Computer software in structural engineering is usually a forwarding only process: the sequence of processes is input-calculate-output. Generally, the user cannot modify the input data after execution has begun. The user-machine interface, data input/output, is a bottle neck of most engineering application software. This issue is especially critical to the development of software to aid preliminary structural planning and design.

Structural planning is the study of structural system alternatives within the context of an unique set of program criteria. The goal of structural planning is to select the most appropriate, efficient and economical structural system which satisfies the design criteria while integrating the mechanical requirements and enhancing the intended aesthetic. The structural planning process is most effective if initiated during preliminary design, when major decisions regarding form, function, and aesthetics are being firmly established [10].

SDU was developed to aid the preliminary structural design phase of building design. The design of a building involves coordinating two major disciplines: architecture and structural engineering [4]. In most cases, the structural engineer is expected to meet the requirements of the architectural design. The preliminary structural design is critical to the performance cf the building since many structural systems are feasible. To choose an effective structural configuration, the structural engineer needs to compare and modify alternative feasible configurations. SDU supports this process by facilitating the development and analysis of structural configurations.

2.2. Structural Design Using SDU

SDU supports three major activities: synthesis, analysis and feedback. These activities are represented by the following three modules.

I. The synthesis module provides a 3D structural system editor for defining the structural system. This module also performs a completeness check and a stability check to insure that a valid input file can be prepared for the analysis program. The displays of 2D structural planes and the 3D structural system give the user visual feedback on the structural system he synthesized.

2. The analysis module provides an automatic interface to a finite element program. The interface translates the structural system description data into the format that can be accepted by the analysis program and then translates the analysis results into SDU's data representation after the analysis finished.

3. The feedback module provides some simple warning signals by checking the maximum stress and maximum deflection of the structural system under the specified load conditions. The magnitude of the internal force of specific components can be displayed by pointing to the component on the screen.

SDU focuses on the synthesis and feedback activities and these are discussed in more detail below.

2.2.1. Structural System Synthesis

A well defined structural system is a combination of structural components according to a specified topology and geometry. The synthesis of a structural system involves manipulating grids, components and substructures. These concepts are described below and the definitions and the relations among them are illustrated in Figure 1.

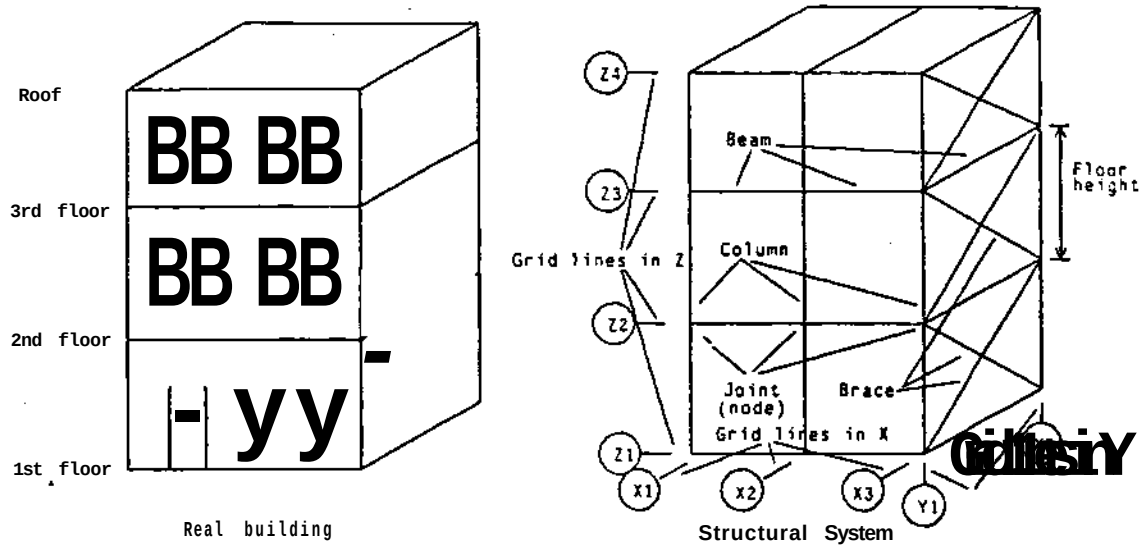


Figure 1: Real Building and Structural System

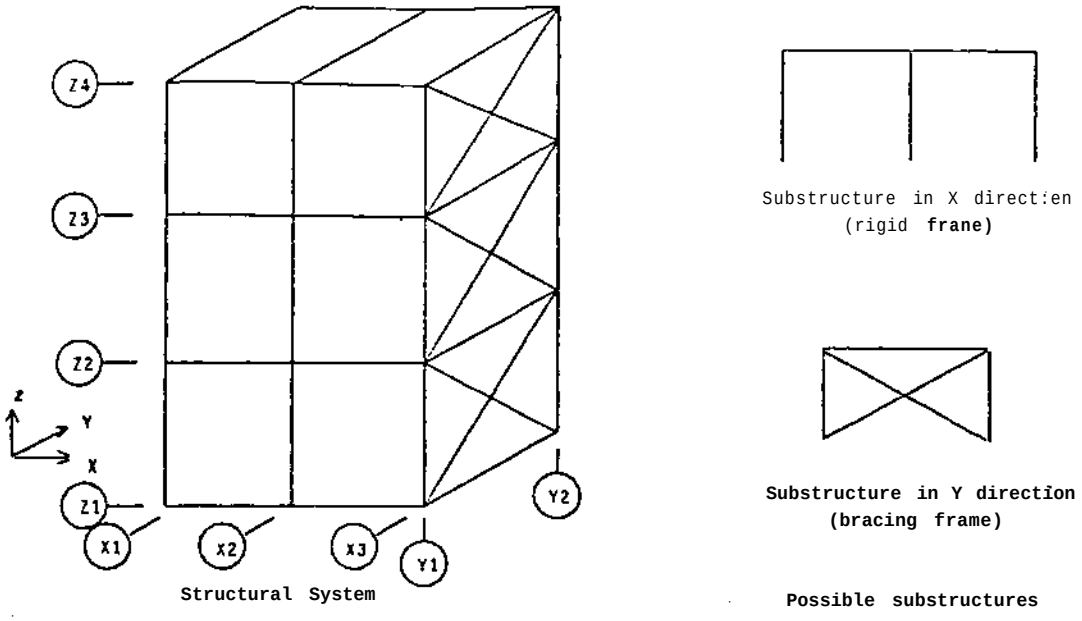
A grid is a two or three dimensional pattern of lines. In the structural design of buildings, a grid is typically used to represent the regular spacing between components. In SDU, a grid is specified by the user and the lines on the grid represent allowable component locations. Once the grid is defined, the user is free from specifying locations as points in three dimensional space; the user need only specify a grid line or node.

Components are the basic units to form a structural system. The components used in SDU are beams, columns, and bracing. A beam is a horizontal component used to support a roof or floor. The gravity load is uniformly distributed on the beams. A column is a vertical component used to support gravity and lateral loads; these loads are only applied at the ends of the column. Bracing components are inclined and are used to support lateral loads. The position and topology of a component are determined by its end nodes. Currently, the connection between a component and a node can be either simple or rigid.

It is quite common in a structural system that a set of components comprise a distinct functional group. A structural system usually has several similar groups which have the same material and mechanical properties. In SDU, these component groups are called substructures. For example, the rigid frame substructure is used to resist the lateral load in the X direction in the structural system shown in Figure 2 and the braced frame substructure is used in the Y direction. Each user defined substructure is represented by an icon to be used in a manner similar to the basic component icons.

2.2.2. Design Feedback

SDU provides feedback at two different stages in the design process: immediately after



Figu re 2: Structural System and Substructures

synthesis and after analysis. The feedback after synthesis includes visual feedback, and stability and completeness checks. The feedback after analysis includes checking for overdesigned and underdesigned components.

Visual feedback is provided by updating the displays after the user edits each structural plane. The structural system is displayed in 3D with perspective in one window and a specified plane with the component end connection information in another window. The 3D display gives the user a stereoscopic image of the whole structural system. The plane display provides end connection information and can be used as a reference when synthesizing another structural plane. The spatial reasoning capacity of human beings is wonderful. The user can find most mistakes in the topology of the structural system.

The stability and completeness checks insure that the analysis program can be run successfully. The stability check insures that there are no mechanisms in the synthesized structure. The completeness checks insure that all components are part of one structural system and that every component has associated section properties.

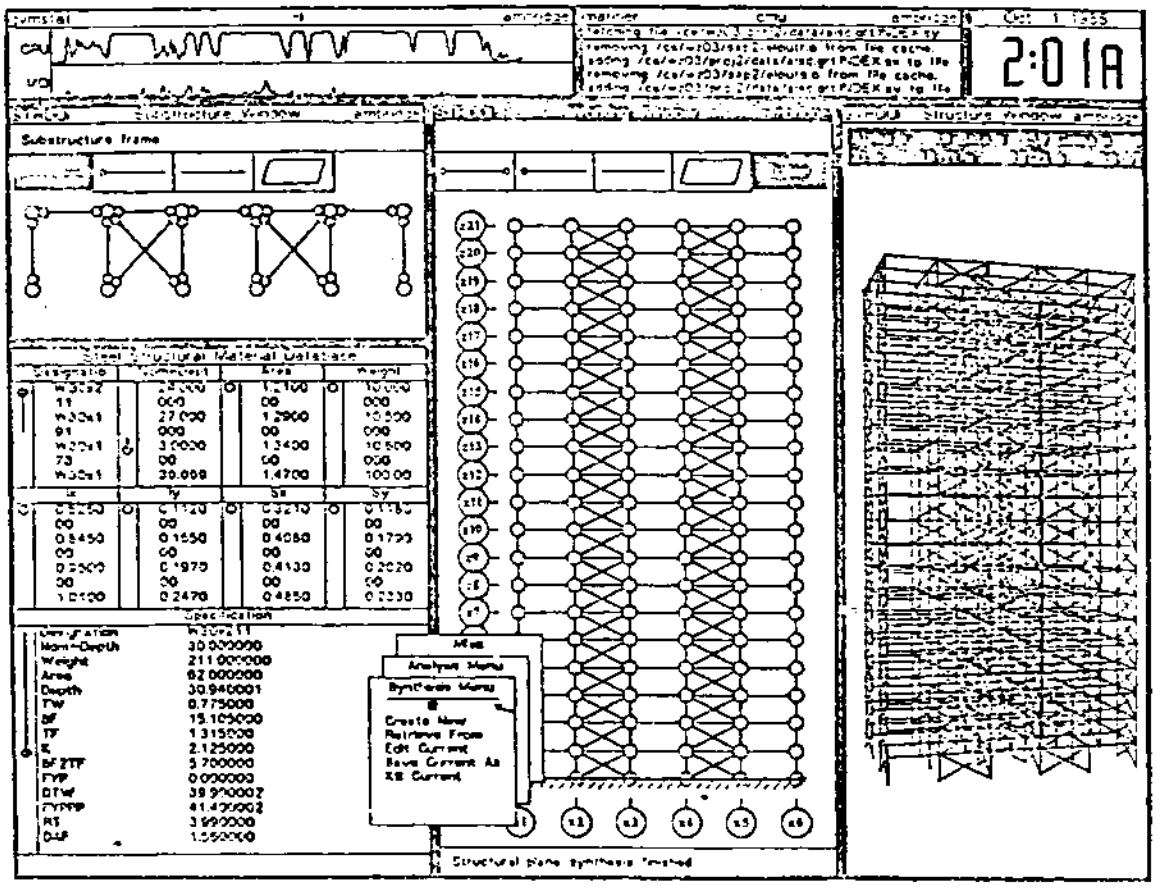
SDU schecks for overdesigned or underdesigned components by comparing allowable and actual stresses and deflections. An underdesigned component is one in which the actual stress or deflection exceeds the allowable. An overdesigned component is one in which the maximum stress or deflection is less than 10% of the allowable.

The user is notified of a violation of one of the above checks graphically. The nodes and components which cause a problem blink one by one in the 3D structure display window and an error message is displayed in text.

2.2.3. WiTidow Organization

SDU is a multiple window display program. Each window has its own input, pop menu

options and screen display. Most of the input is done by mouse pointing instead of keyboard stroke. When the program is executed, the following four windows are opened : working window, substructure window, plane window and structure window. The sizes and locations of the windows can be changed by the user. A possible organization of screen display is ^o-.ŋ in Figure 3. These windows provide different functions for the graphical interface as described below.



Figu re 3: A Typical Window Organization **of SDU**

Working Window :

This window serves as the area for synthesizing a structure, including defining load conditions, defining member section parameters, and saving **and** retrieving a synthesized structural system. Most of the user-machine interactions are performed in this window. The working window is shown on the right of the screen in Figure **3.**

Substructure Window :

This window is a work space in which the user can synthesize, save, retrieve, modify and delete substructures. A synthesized substructure v. ;i appear in the working window as a basic component type in the iccn option. The substructure window is shewn on the top left of the screen in Figure 3.

Plane Window : A 2D structural plane of the current structure indicating components with end support conditions is displayed in the plane window. The structural plane can be chosen from the pop up menu in this window. The plane window is shown on the bottom left of the screen in Figure 3.

Structure Window :

The structural system defined in the working window is displayed in the structure window in 3D perspective. The user can scale, rotate, move the structure and change the view point. During the structural system checking, such as stability checking, stress checking, etc., the component or node with a problem will blink in this window. The structure window is shown on the middle of the screen in Figure 3.

2.3. Implementation of SDU

SDU is developed on a SUN workstation[3] based on the Andrew software environment. The Sun workstation has a 32-bit architecture CPU and 2 megabytes of main memory. The operating system supports 16 megabytes virtual address space per process by interchanging data with the local hard disk. The 10 million bits/second Ethernet local area network interface joins each workstation with the main server of the Andrew environment. The monitor is a 19" high-resolution (1152 x 900) bit-mapped monochrome display. Input is via the keyboard or an optical mouse. There is no co-processor for floating point calculation on the current workstation, resulting in a slow rèsponse for numerically intensive applications.

The software environment used for this project is called Andrew; it is developed by the Information Technology Center (ITC) of Carnegie-Mellon University (CMU) for educational purposes. Andrew is a joint project of IBM and C-MU. Andrew is designed as a machine independent software system. The Sun workstation is the prototype workstation chosen for the development of Andrew. Each workstation is based on Unix 4.2 BSD operating system. Almost all of the system software en the Sun is written in the C language. SDU uses three software packages that were developed independent of this project: the window manager program that runs in Andrew [7], a relational database package, GRITS [8], and a finite element program, SAP IV [1].

3. Intelligent Graphics

Our current effort in the area of graphics based preliminary structural design includes the study of the potential for 'intelligent' graphics. Typically, computer software is developed using techniques from specific domains, i.e. drafting tools only know how to draw and analysis packages only know how to analyze. The concept of intelligent graphics involves the restructuring of the information inside the computer program so that the capabilities of the software are more comprehensive. The following features would provide a more intelligent design environment.

3.1. Display with Detail Level Control

One aspect of intelligent graphics is the combination of structural and graphical

[3] S?JN is the trade mark of SUN microsystems Inc.

information. In the current version of SDU, the structure is displayed as a line drawing, where eah component is represented by a line. There is currently no facility for displaying detailed information about the components, although this information is stored in the data structures of SDU. The display of all information about the structure at the 3D level would result in a confusing and busy display. A more comprehensive program would include both graphical and structural information at all levels and provide a facility for controlling the level of detail displayed at any time.

3.2. Spatial Reasoning

Human beings have a very strong capability for graphical understanding and spatial reasoning. People can quickly extract much more information from a picture than from text or digital table. This is the most important reason for using computer graphics as the man-machine interaction media. Currently, CAD programs store and display the picture the user has defined and know little about the objects that the picture represents.

To make graphical structural design tools more intelligent, geometric modeling techniques are necessary. The program should store information about the spatial relationship among the objects as well as information about the isolated objects. In addition, the software should have the ability to infer information about the geometry of the structure from stored information; in other words, all geometric information that is ever needed should not be stored declaratively.

3.3. Approximate Analysis Techniques

The finite element method is a universal way to solve for the internal forces and nodal displacements of a structural system. It can theoretically deal with any type of structural system. But for complex structures it is an expensive and time consuming method. Various approximate analysis methods have been developed for different types of structural systems based on experiments and design experience. These are the techniques that are used by engineers during preliminary design. Unfortunately, these techniques are seldom used in computer programs, primarily because the program does not understand the structural system in the manner necessary to apply the appropriate approximate method. By combining geometric and structural knowledge, it may be possible to make intelligent, automatic use of approximate analysis techniques.

3.4. Expert Structural Design Consultant

To evaluate a structural system by interpreting detailed analysis results is not suitable at the preliminary design phase. In order to truly aid the structural designer, the program should understand the building configuration enough to provide intelligent criticism of the user's design. The use of expert system techniques make this kind of interaction possible. It is conceivable that a knowledge base of design heuristics could be used to monitor the design and intervene when appropriate.

In addition to design evaluation, a structural design consultant is able to supply the client with alternative feasible design solutions. At the preliminary design phase such a consultant would allow the designer to easily pursue more alternatives and would give him the option of developing a design solution from both his own experience and the experience recorded in the expert system. Prototype expert systems of this type have been developed at Carnegie-

Mellon University [5, 9].

4. Conclusions

SDU, as a graphics based preliminary design environment, is successful in its current form; it has been successfully used in a structural design course for architecture students. SDU does not, however, provide much more than a friendly graphical interface to a finite element program. The preliminary design process is not adequately supported by SDU primarily because there is very limited knowledge of structural engineering and spatial reasoning in the program. One way to improve SDU would be to continue to add more knowledge to the program; this would essentially be an ad hoc approach to intelligent graphics.

An alternative way to provide an intelligent graphical environment is to reconsider the way in which knowledge is organized in the computer. The environment provided by an object oriented programming language, such as Smalltalk [2] or KEE [3], is particularly appealing because it allows the programmer to create chunks (or objects) of related information. Each object can contain declarative, procedural, and heuristic information, and can inherit information from similar objects. A potential organization for an intelligent graphical environment would be to represent structural entities as objects and associate graphical, structural, and experiential information with each entity. A reorganization of the knowledge stored in a graphics based preliminary design program will result in an efficient and comprehensive design environment.

References

[I] Bathe, Klaus-Jurgen.
SAP IV A Structural Analysis Program for Static and Synamic Response of Linear System
College of Engineering, University of California, Berkeley, California, 1973.

[2] Goldberg, A., and Robson, D.
Smalltalk-80. The Language and its Implementation.
Addison-Wesley, Reading, MA, 1983.

[3] undef.
KEE: Knowledge Engineering Environment
IntelliGentics, Inc., 1984.

[4] T. Y. Lin.
Structural Concepts and Systems for Architects and Engineers.
John Wiley & Sons, Inc., 1981.

[5] Maher, M.L , Sriram, D., Fenves, SJ.
Tools and Techniques for Knowledge Based Expert Systems for Engineering Design.
Advances in Engineering Software ,1984.

[6] M.L. Maher and W. Zhang.
Graphics Based Preliminary Structural Design.
Proceedings of the Fourth Conference on Computing in Civil Engineering , October, 1986.

[7] Neuwirth, Chris.
Programmer's Guide to the Window Manager. An Introduction for the Uninitiated
The Communications Design Center (CDC) Carnegie-Mellon University, 1985.
for Release 1 of the Information Technology Center (ITC) Prototype Workstation.

[8] TC Peters.
GRITS Programmer's Guide
Information Technology Center, Carnegie-Mellon University, 1985.

[9] Sriram, D.
Knowledge-Based Approaches For Structural Design.
PhD thesis, Carnegie-Mellon University, February, 1986.

[10] Wichersheimer, David J.
Computer Graphics in Structural Planning.
1985.
Unpublished Report.

[II] Zhang, Weiguang.
Interactive Graphic User Interface for 3D Structural Design.
Master's thesis, Carnegie-Mellon University, March, 1986.