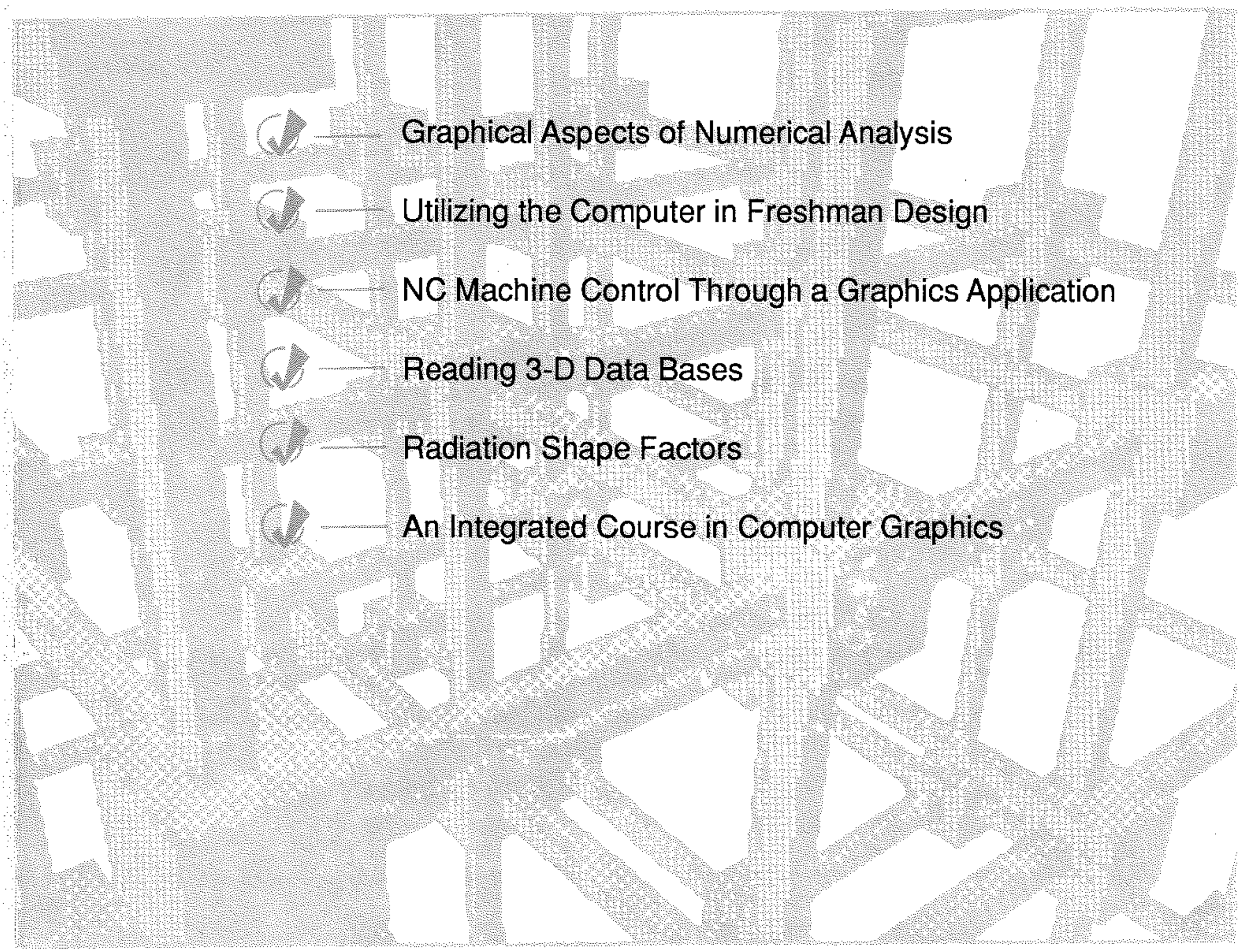
# THE ENGINEERING DESIGN GRAPHICS Journal

Winter 1988 Volume 52 Number 1

ENGINEERING DESIGN GRAPHICS DIVISION

AMERICAN SOCIETY FOR ENGINEERING EDUCATION



- Graphical Aspects of Numerical Analysis
- Utilizing the Computer in Freshman Design
- NC Machine Control Through a Graphics Application
- Reading 3-D Data Bases
- Radiation Shape Factors
- An Integrated Course in Computer Graphics

ISSN 0046-2012

## PUBLICATION BOARD



## CALENDAR OF EVENTS

ASEE ANNUAL CONFERENCES

1988 PORTLAND hosted by Portland State University

EDGD MIDYEAR MEETINGS

1988 EVANSVILLE hosted by The University of Southern Indiana

1989 TUSCALOOSA hosted by The University of Alabama

---

The Engineering Design Graphics Journal is published one Volume per year, three Numbers per Volume in Winter, Spring, and Fall by the Engineering Design Graphics Division of the American Society for Engineering Education, for teachers and practitioners of Engineering, Computer, Design, and Technical Graphics. The views and opinions expressed by the individual authors do not necessarily reflect the editorial policy of The Engineering Design Graphics Division or of the American Society for Engineering Education. The editors make a reasonable effort to verify the technical content of the material published; however, the final responsibility for opinions and technical accuracy rests entirely upon the author.

## MICROFILM

Article copies and 16, 35, and 105mm microfische are available from:

UNIVERSITY MICROFILM, INC.
3000 ZEEB ROAD
ANN ARBOR, MI 48106

## SUBSCRIPTION RATES

| YEARLY RATES | |
|---|---|
| ASEE Member | $3.00 |
| Non-Member | $6.00 |
| Canada, Mexico | $10.00 |
| Foreign Subscribers | $20.00 |

| SINGLE COPY RATES | |
|---|---|
| ASEE MEMBER | $1.50 |
| Non-Member | $2.50 |
| Canada, Mexico | $3.50 |
| Foreign Subscribers | $7.00 |

All fees are payable to the Engineering Design Graphics Journal at: The Engineering Design Graphics Journal, Department of Engineering Graphics, The Ohio State University, 2070 Neil Avenue, Columbus, OH 43210. Back issues are available at single copy rates (prepaid) from the circulation manager and are limited in general to numbers published within the past six years. Subscription expiration date appears in the upper right corner of the mailing label as follows: ASEE/EDGD member is the same month/year as ASEE dues e.g. 6/86; all others last issue paid e.g. W86 for Winter 1986. Claims for missing issues must be submitted to the Circulation Manager within a six-month period following the month of publication: February for Winter, May for Spring, and November for Autumn.

## OBJECTIVES OF THE JOURNAL

The objectives of The Journal are:

1. To publish articles of interest to teachers and practitioners of Engineering Graphics, Computer Graphics, and subjects allied to the fundamentals of engineering graphics education and graphic technology.

2. To stimulate the preparation of articles and papers on topics of interest to its membership.

3. To encourage teachers of graphics to experiment with and test appropriate teaching techniques and topics to further improve the quality and modernization of instruction and courses.

4. To encourage research, development, and refinement of theory and application of engineering graphics for understanding and practice.

## DEADLINES

The following are deadlines for submission of articles, announcements, and advertising: FALL-September 15; WINTER-December 1; SPRING-February 1.

## STYLE GUIDE FOR JOURNAL AUTHORS

1. All copy is to be typed, double spaced, on one side only using white paper with black ribbon in standard English. Dot matrix copy is acceptable if high quality.

2. All pages of the manuscript are to be numbered consecutively.

3. THREE copies of each manuscript are required.

4. Refer to all graphics, diagrams, photographs, or illustrations in your text as Figure 1, Table 1, etc. Be sure to identify all material. Illustrations cannot be redrawn. Accordingly, be sure that all linework is black and sharply drawn and that text is large enough to be legible when reduced to 4.25" in width. Good quality photocopies of sharply drawn illustrations are acceptable.

5. Submit a recent, glossy black and white photograph (head to chest). Make sure that your name and address are on the back. Photographs, illustrations or other submitted materials cannot be returned without postage prepaid.

6. The editorial staff will edit manuscripts for publication after return from the board of review. Galley proofs cannot be returned for author approval. Authors are encouraged to seek editorial comment from their colleagues before submission.

7. Enclose all material, unfolded, in a large envelope. Use heavy cardboard to prevent bending.

Continued inside back cover.

THE ENGINEERING DESIGN GRAPHICS

# Journal

Volume 52 Number 1

ENGINEERING DESIGN GRAPHICS DIVISION

AMERICAN SOCIETY FOR ENGINEERING EDUCATION



## DEPARTMENTS



## FEATURES

Jon M. Duff, Editor

## FROM THE DESK OF THE EDITOR

Many campuses are abuzz with discussions of what to do during the impending shortage of college faculty. That there is a greying of the professoriate is well known. By many estimates nearly half of those teaching in colleges today will be retired by the year 2000. You need look no further than the latest issue of *Engineering Education* to see concern for the professor shortage. Add to this a bottoming out in the decline in the college-age population and you can see that all of us in higher education are in for some difficult but interesting times.

In graphics, we are and have been experiencing this problem. Where to find qualified and interested faculty, how to groom faculty in-house, how to retain faculty, and how to promote those who do stay—all are issues that continue to be addressed. I would like to attend to one of those issues in this column...possibly each of you might consider what you have done either to contribute to this problem, or to its solution.

The problem I'm speaking about is the aura we seem to surround ourselves with, the aura of great knowledge. When I was an undergraduate I never dreamed of becoming a Professor because professors, you know...they professed. They professed about a lot of things, about almost everything. They acted professorial. They seemed so intelligent and knowledgeable about their discipline that they appeared bewildered at an undergraduate's simple questions. Their understanding was so deep that simple explanations became obtuse. They rushed here or there to do something important, and we as students simply tried to stay out of their backwash lest we be bowled over. How could I, in my short lifetime, ever hope to acquire the knowledge of...A Professor? They quoted chapter and verse out of the books they had written; they knew the entire ANSI standard; they had memorized all trig functions in multiples of 5 degrees; they knew the serial numbers of every Timken tapered roller bearing manufactured in the past forty years; they could cite the torque values of every fastener on a GE jet engine. And this was just a start! They could ask questions on a test that made you look around the room to see if you hadn't by mistake dropped in on a class on nuclear brain surgery. Little did I know back then that none of this was the business of being a professor and that being brilliant took a distant back seat to having a genuine interest in acquiring knowledge, being inquisitive, and developing a devotion to your discipline.

***"How could I, in my short lifetime, ever hope to aquire the knowledge of ...A Professor?"***

Are we discouraging future professors by continuing an aura of academic elitism? Are we attracting only those who

Continued on Page 6

# chairman

## A MESSAGE FROM THE DIVISION CHAIRMAN

Another successful EDGD Midyear Meeting was held this past November in Louisville. The meeting was well-attended (I counted 85-plus heads at the Business Luncheon), and the twenty-five technical presentations were both informative and stimulating, as we all brace for the era of 3-D Engineering Graphics. For the first time in recent history, a bound proceedings of all papers presented at the meeting was produced, which greatly enhanced the professional stature of the meeting. I trust that this precedent will establish a quality standard that all future midyear meetings will aim for.

Successful gatherings do not happen automatically. Special people are always behind the planning and organizing of successful events, and I would like to acknowledge those special persons who contributed to the Louisville Midyear meeting:

• Bob Matthews was overall meeting chairman and facilities planner, and was responsible for the excellent arrangements at the Galt House.

• Frank Croft was program chairman, and was editor of the inaugural midyear Proceedings.

• Dick Latimer was an able assistant to Bob Matthews and helped with registration.

• Dean Gerhard of the Speed Scientific School at the University of Louisville, which graciously hosted the event.

• General Electric Corporation, who provided an eye-opening, computer graphics-filled tour of their plant.

• All moderators and paper presenters, for providing the innovative, technical information which we all came to the meeting to hear.

Beside the excellent technical, business, and social events that transpired at the meeting, two important ad hoc committees were formed in Louisville.

1.**The Ad Hoc Committee for Review of the Engineering Design Graphics Journal** was charged with the responsibility to thoroughly review all operating facets of the Journal. The committee will study the recently established paper review process, the layout style of the Journal, sources for advertising, and publication timetables. The committee will also investigate whether a permanent desktop publishing system should be acquired by theDivision for the original layout of the Journal. Merwin Weed, Garland Hilliard, and Bob LaRue were asked to co-chair this important committee, and I am sure they would welcome your input.

2.**The Ad Hoc ABET Study Committee** was formed to study clauses in ABET Accreditation guidelines and to formulate a proposal for changes which cast a better light on the importance of engineering graphics in engineering education. If you read the guidelines, you would note that the only mention of graphics in the General Criteria is a negative statement: "Coursework devoted to developing drafting skills may not be used to satisfy the engineering design requirement."

**"Coursework devoted to developing drafting skills may not be used to satisfy the engineering design requirement."** *ABET Guidelines*

Continued on Next Page

## Chairman's Message from page 5

There is no positive support anywhere for engineering graphics, engineering drawing, or computer graphics. While there are strong statements for both written and oral communication, there is no mention of graphical communication skills. The committee ultimately would like to change this existing condition, and we need your support! If you have any thoughts on the matter, and in particularly if you have any documentation that supports our belief in engineering graphics, please forward them to me. This support can take the form of:

- Papers
- Surveys
- Industrial Sponsorship
- Letters and Citations
- Government Reports

I will be chairing this committee, and plan to place a high priority on its success during the 1988-89 time frame. Please contact me at the address or telephone numbers below. We need your help!

**Ronald E. Barr**
Mechanical Engineering Department
University of Texas at Austin
Austin, Texas 78712
(512) 471-3008

## Editor
Continued from Page 4

view the professoriate as a haven for the elite?

If students only knew that being a professor, like any other job, is 10% inspiration and 90% perspiration! Being a professor means acknowledging that something new is learned about your discipline—its teaching and learning— every day; moreover, as a professor you actually *enjoy* learning something new every day. It's what drives you along, keeps you going. You're not afraid of a student who knows more about a given topic than you do because the acquisition of knowledge is your business. No one is keeping score on students vs. professors.

What is the problem in Engineering Education? Undergraduates, especially engineers, find out rather quickly that the world of work can be boring, without challenge. Why is that? Possibly it is because engineering as it is taught in colleges and universities, and engineering as it is practiced in industry, are two very different animals. Our engineering students, in reality, aren't being prepared to be engineers, rather they are being prepared to be future engineering faculty. The question is, why don't they realize that? The characteristics of new problems, new knowledge, changing schedules, and self-directed time management that are part of the undergraduate engineering curriculum are often lacking on an engineering job. It would seem that collectively we need to spread the word that a career as a professor may just be the best job there is.

## Editor Suggests Rethinking of Computer Choices

Paul Schrier, editor of *Personal Engineering & Instrumentation News*, reported in a recent issue that "IBM has lost its lead. True most engineering applications still run on the IBM, but the excitement has shifted to other platforms. Further, new hardware and software products for engineering tasks are changing the way you'll look at technical computing."

He goes on to say "users must rethink their choices of host computers. IBM by no means continues to have a lock on such applications and we don't see any developments that indicate that IBM is terribly interested in technical users. Other host vendors, in contrast, are making their units ever more attractive. Prices are dropping to the point where you can have more than one kind of computer at your workplace and enjoy the best of all worlds."

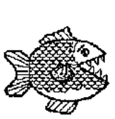
# News and Notes

Who is looking for faculty this next year? The EDGJ runs job notices free of charge to member schools. Send notices of retirements, promotions, etc. to the Division Editor.

You should have completed the ABET Survey Sheet that may have been mailed to you. A positive comment by ABET on graphics would be a real boost.

If you are going to Vienna for the International Conference, let the Editor know. I would like to publish a list.

Read with interest the proposal for EDGD history on page 9. A fine idea. Another project might be a list of publications of current and past EDGD members.



ASEE Annual Meeting
Portland Oregon
see page 32

# EDGD Notes the Passing of a Long-Time Graphics Teacher, ASEE Leader

## RALPH SEAL PAFFENBARGER

Ralph S. Paffenbarger, Professor Emeritus of Engineering Graphics at The Ohio State University, passed away in Columbus, Ohio, on January 21, 1988, at the age of 93. Following in the footsteps of Thomas E. French, Professor Paffenbarger chaired the OSU Department of Engineering Graphics during his last 20 years of service on the faculty prior to his retirement in 1964.

Professor Paffenbarger was a long-time member of ASEE and a very active and enthusiastic member of the Engineering Design Graphics Division. He served as Chairman of the Division in 1950-51, and received its Distinguished Service Award in 1956.

Professor Paffenbarger was also a very active participant in the work of the American Standards Association, now the American National Standards Institute. He was a member of both the Y14 and Y15 Committees. He served as Secretary of the Y14 Committee during much of the period of development of the ANSI Y14 Standards for Engineering Drawing and Related Documentation Practices.

"Paffy" was a dyed-in-the-wool BUCKEYE. He designed the first seating plan for the OSU Football Stadium prior to its dedication in 1922. He officiated at Ohio State track meets from the time of Jesse Owens until well after his retirement, acting as Head Judge for most of those years. He was a member and Secretary of the OSU Athletic Board for several years. He prided himself on having not missed a single day of teaching due to illness in 44 years at OSU, and on being in attendance at all Ohio State home football games for over 50 years. The Ohio State University recognized his dedicated career by awarding him its Distinguished Service Award in 1977.

Professor Paffenbarger was one of the last of a vanishing group of distinguished pioneers in engineering graphics and engineering education. We sadly note his passing.

*Clyde Kearns*

# review

A REVIEW OF PRODUCTS AND SERVICES

## PRENTICE-HALL INTRODUCES TWO SOFTWARE-KEYED GRAPHICS TEXTS.

**Learning CAD With AutoCAD** *by Mihir K. Das, California State University, Long Beach. 1988.*

This spiral-bound text represents a new breed of text-workbooks. It does not make any pretensions to be a graphics textbook. Rather it, as does the Reichard book also reviewed here, assumes that graphics knowledge has been acquired elsewhere and the task at hand is to apply a particular software tool to the production of engineering drawings. More precisely, in the case of Das, not to the production of drawings but to the mastery of geometric and textual operations that at a later date might be combined into a set of skills necessary for production drawing.

The book is uniformly formatted sothat information is found in the same location on each page...a desirable feature if the book is to be used as a reference. The depth of concept explanation however makes this book more suitable for training than for education. All traditional CAD topics are covered in a format not unlike traditional FORTRAN texts.

There is no reference to using AutoCAD as a design tool though AutoLISP routines are briefly covered. The lack of production or design examples, however, does not do justice to the tool's capabilities. *-ed*

**Exploring CADKEY** *by David C. Reichard, Charles County Community College. 1988.*

The other major IBM PC CAD software package is covered in this student-oriented text. A spiral format allows for laying the book flat on a table. This is less of a reference book of possible commands, and more of a student tutorial guide. The writing is more conversational, and thereby, more educational. Each chapter begins with specific objectives, followed by instruction, examples, tutorials, and student assignments. The author goes into detail about the functionality of a particular CAD command, making this book suitable if a broad understanding of CAD concepts is an important objective. Both books unfortunately treat modeling as an application of computer drawing rather than drawing as an outcome of the modeling process.

*-ed*

# comment

Larry Goss
University of Southern Indiana

"There is No History of Computer Graphics before, say, 1968"

I heard that statement made at the mid-year meeting in Louisville as you and others in attendance did also. Those of us who have been working with computer graphics and computer-aided design since before that time were aware of its error, but said nothing about it at the time. I was asked by one of the vendors in attendance at the mid-year meeting why the statement was not challenged. My response was that we were just too polite to engage in public argument.

But now you have given the statement credence by restating it in print (EDG Journal, Volume 51, Number 3, Autumn 1987, page 4). Let me assure you that computer graphics does have a history before 1968—way before.

In 1968 I was working with a computer graphics hardware/software system from IBM that was so old and out-of-date that IBM no longer supported it and was giving it away. In 1963, when I was teaching at Purdue, I was assembling literature from vendors in an attempt to find turnkey computer graphics systems that were affordable for use in the engineering graphics department. In the early 60's

Continued on Page 16

The Executive Committee of

# The Engineering Design Graphics Division - ASEE

Requests Proposals on the

# History of the EDG Division

## Background

The EDG Division of ASEE was founded in 1928 and is 60 years old this year. In 1993, ASEE will celebrate its centennial anniversary at the Annual Summer Conference to be held at the University of Illinois. The EDG Division has a very rich tradition of people, places, and events that have greatly contributed to engineering education. Some of this tradition perhaps has been documented through the years in various issues of the EDG Journal. It would be a fitting tribute, in keeping with the centennial ce;ebration of ASEE, that a book be professionally produced on the "History of the EDG Division."

## Proposal

The proposal should be brief (four type-written pages) and should include:

1. The author (or authors) qualifications to write the book should be clearly stated.
2. The plan of approach and sources of EDG history information should be identified.
3. The types of archival information to be included in the book, as well as the style of prose, should be indicated.
4. The project timetable should be outlined, bearing in mind that it is desirable to have the finished product before 1993.

## Budget

It is expected that the time spent to author the book will be donated as a service to the EDG Division, hence no salaries can be accepted in the proposal. However, in order to entice serious professionals, a 15% author's royalty, above production costs, is permissible. Budget items for secretarial assistance, copying costs, telephone, mail, and justifiable travel costs can be included.

## Deadline

Please submit your proposal by **June 8, 1988** to:

Dr. Ronald E. Barr, Chairman
EDG Exectutive Committee
Mechanical Engineering Department
University of Texas at Austin
Austin, Texas 78712

**Inquire:** (512) 471-3008

# papers

The review process of the Engineering Design Graphics Journal can be long and time-consuming. Papers submitted to The Journal are first reviewed for timeliness and subject matter applicability by the Editor. If a paper is appropriate, it is forwarded to the Technical Editor who distributes copies of the paper to members of the Board of Review whose technical areas match the paper's content. When returned, the paper may be accepted for publication, accepted conditionally for publication pending revision by the author, or rejected for publication. Of submitted papers, The Journal currently accepts 30%, conditionally accepts 18%, and rejects 52%.

## Graphical Aspects of Numerical Analysis

M.M. Khonsari
University of Pittsburgh

D.E. Brewe
AVSCOM
NASA Lewis Research Center

### INTRODUCTION

The widespread availability of digital computers and substantial increase in computing power in recent years has resulted in computers becoming a major research tool in all branches of engineering and science. Today's engineers rely heavily on computers to study a new design before building an actual prototype of the model.

Most physical systems can be mathematically represented with a set(s) of differential equations having appropriate boundary (and initial) conditions. The governing equations of a system may be linear or non-linear and are categorized as Ordinary Differential Equations (O.D.E.) or Partial Differential Equations (P.D.E.).

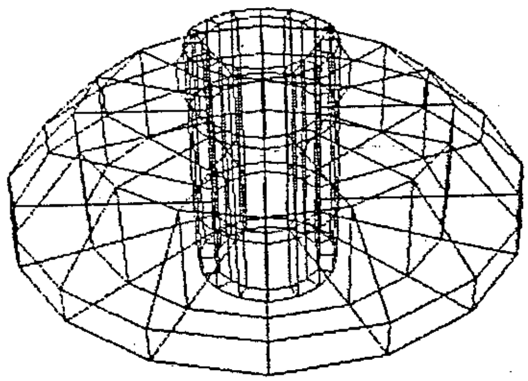
Inherently, information is required from numerical analysis that is often overwhelming. Further, if the information is received as numerical output, it cannot be easily perceived in an immediate and integral fashion. In other words, many of the important details of the results such as the location and magnitude of peaks and valleys (maximum and minimum) are lost mainly because of the quality of output. In this regard, graphical representation of data becomes far more superior to numerical data. Hence, computer graphics has become an essential part of numerical analysis. In recent years, tremendous progress has been made in the area of computer graphics. The designer is now equipped with advanced CAD systems capable of performing large scale simulations as well as interactive computer graphics.

This paper discusses various forms of graphical data presentations utilized in numerical analysis. Graphical aspects of two widely used numerical methods known as the *finite element method* and *finite difference method* are discusses. In addition, a case study of animation of numerically generated data is presented. The study pertains to the effect of cavitation on the performance of a dynamically loaded journal bearing.

### FINITE ELEMENT METHOD (FEM)

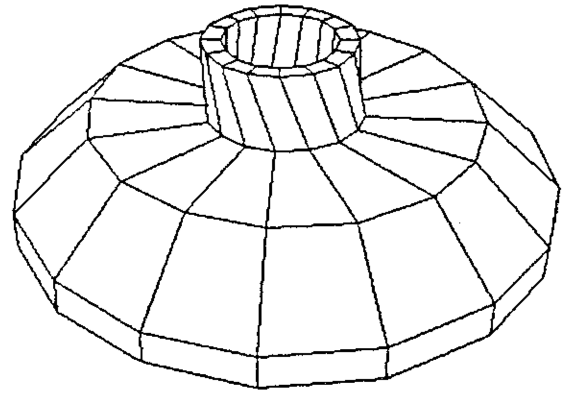
Finite element method was developed primarily for calculating stress/strain and deformation in structural analysis. Although FEM is extensively used in the field of solid mechanics, it has been successfully applied to problems in other areas such as thermal science and fluid mechanics.

Fundamentally with the FEM method, the continuum defined as the body of geometry is divided



Mesh of Three-Dimensional Elements Formed Using Gear Solid Model Geometry.

Figure 1



Displacements from Structural Analysis Displayed Using both Surface and Wire Frame Graphics.

Figure 2

into a series of *mesh elements* which can be of various shapes such as triangular, rectangular, etc. In this fashion, FEM casts the problem in an integral sense so that the solution to the governing differential equation is a *piece-wise* approximation to the problem.

The greatest strength of the finite element method is in the handling of arbitrary geometries such as curved boundaries whereas the finite difference method is best suited for rectangular geometries. Because of the flexibility of FEM to complicated shapes, the method has created a great market for development of commercial software packages.

One of the essential features of such FEM packages is an automatic mesh generation routine with mesh refinement capabilities in critical regions of the geometry where greater resolution and accuracy are needed. An example of a three-dimensional mesh element is shown in Figure 1 which depicts the geometry of a solid gear. Figure 2 presents an exaggerated plot of the gear when subjected to externally applied torque. All boundary conditions for this example were entered interactively. With the aid of interactive graphics one can observe results of changing parameters both as shaded surfaces or wire-frame models. This provides immediate feedback to the analyst.

## FINITE DIFFERENCE METHOD (FDM)

Finite difference method was developed long before the method of finite elements. Fundamentally it is based on approximate representation for the derivatives appearing in a partial differential equation. Thus as opposed to FEM which provides a piece-wise approximation to the equation, FDM is a point-wise approximate solution. Finite difference method tends to exhibit more similarity to the underlying governing equation derived based on physical laws. It is therefore easier to understand However, it is best suited to geometries of simple shapes, although mesh generation routines for a finite difference solution in complicated geometries are available as well. Furthermore, through some suitable mathematical transformation, it is possible to map the physical domain into a rectangular computational domain so that a finite difference method can be easily and effectively applied. For example, the convergent-divergent domain shown in Figure 3a can be transformed to a

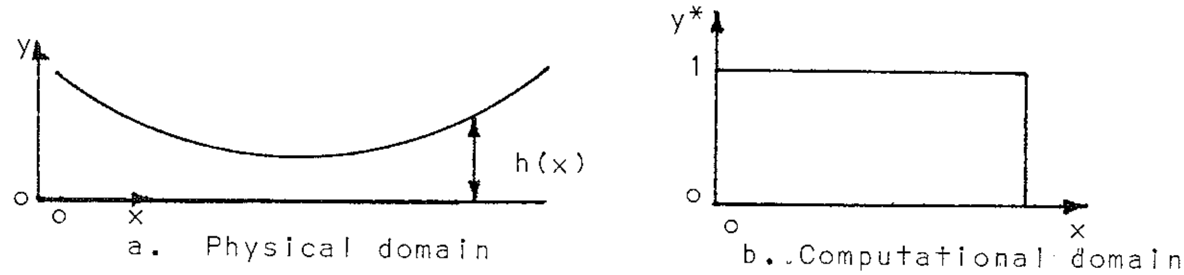



Figure 3

rectangular domain as depicted in Figure 3b by making the transformation of $y^* = y/h(x)$.

Finite difference programs are much easier to develop than finite element programs. However, they are mostly unique to the geometry being modeled and the program is not easily transferable to other geometries.

## GRAPHICAL PRESENTATION OF NUMERICAL SOLUTIONS

Large amounts of numerical data can be interpreted much easier if presented in graphical form. Various forms of plots are available for presentation purposes. For most applications the results are displayed on a plotter medium. However, for presentation of long documents particularly those containing a time-dependent element, it may be necessary to store results on videotape.

The earliest (most primitive) type of diagrams which became available on interactive computer graphic systems were column diagrams and histograms used for displaying functions of one variable. The next generation of graphical packages added the capability of plotting ordinate versus abscissa type diagrams with an option for smoothing the data which can be done by linear interpolation of the data. Several curves can be plotted on the same diagram using colored pens and/or with different line styles such as dashed line, center line, etc. Representation of numerical data that are functions of two and three dimensions require more sophisticated software. Among the methods available are the so-called marker clouds and contour plotting. Marker clouds is a popular method of presenting material movement in the field of fluid mechanics. Basically, markers of different sizes can be displayed based on the material density: the larger the density, the larger the size of the marker. The disadvantage of the method is that, depending on the plotting medium, it may be very time consuming to plot all markers.

A contour diagram is probably the most popular method for representing a scalar function of two variables in a single plot. Contour plots provide an immediate feedback on the numerical solutions so the analyst can quickly identify the regions of, for example, high stress concentrations in a structure or hot spots in a bearing. Most commercial packages only display contour plots in a rectangular shaped geometry. See, for example, Figure 4 which shows the contour diagrams of isotherms in a bushing surface on a journal bearing. By a suitable coordinate transformation, one can easily display a more realistic plot of temperature as shown in Figure 5. These results were obtained by finite difference method.

The next degree of sophistication is the display of three-dimensional plots using surface algorithms. Such algorithms are generally very time-consuming since a hidden line removal algorithm must be used. Figure 6 presents a plot of the pressure distribution generated in a dynamically loaded bearing. This plot represents a typical frame of motion-picture graphics discussed in the following section.

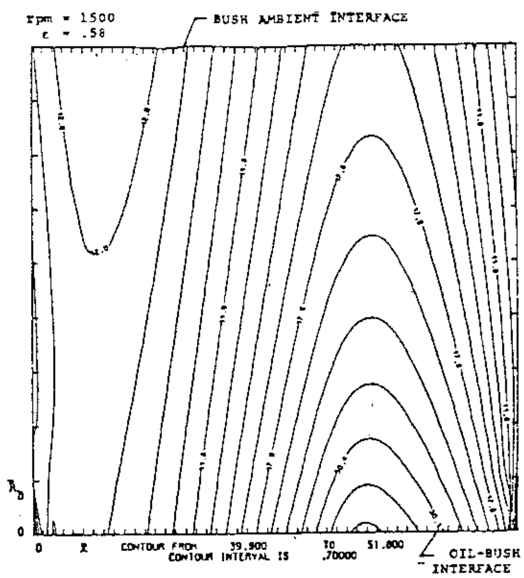



Figure 4

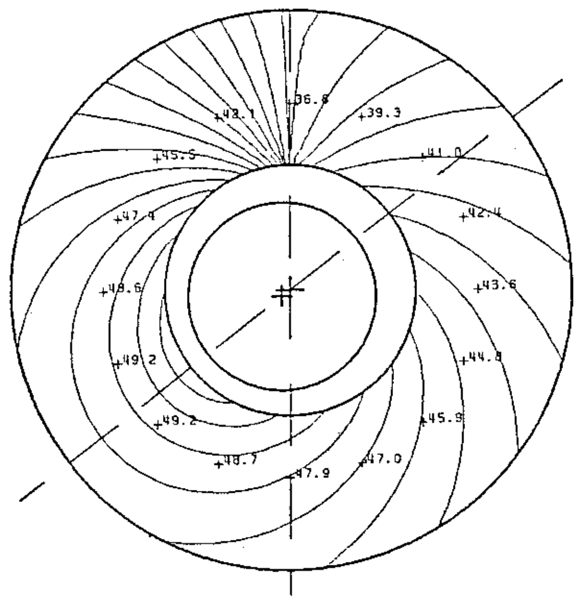



Figure 5

## ANIMATION

One of the most fascinating aspects of computer graphics is animation. The ability to create motion pictures by computer has become a reality in recent years, particularly for computer generated arts; computer animation in science is growing in popularity as well. The results of a numerical study, particularly time-dependent problems, can be animated. Such a method of data presentation can be used for the purpose of testing a new computational algorithm, a new design, or for studying the physics of complicated phenomena.

A study was made to analyze a dynamically loaded journal bearing using a computer generated movie. The rationale behind this study was:

*a.* to get a clear picture of the fluid dynamics of the lubricant as it travels inside the bearing;

*b.* to predict the results of some experiments conducted at NASA Lewis Research Center which required high-speed photography;

*c.* to understand the details of the formation of vapor cavitation and its effect on the stability of dynamically loaded journal bearings.

It is well-known that journal bearings are an important component of almost all rotating machinery. The surface of the rotating element, namely the shaft, and the housing is separated by a layer of fluid film. Bearings are subjected to a condition known as cavitation during which the dissolved gases in the lubricant come out of the solution. Two forms of cavitations observed in bearings are known as gaseous and vapor cavitation. Theoretical study of cavitation in has been the subject of various rsearch papers, a summary of which is given by Dowson and Taylor [8]. The importance of including a realistic model in the design stages of a bearing is of considerable importance since it influences the bearing performance. Further, cavitation is known to play an important role in the stability of hydrodynamic bearings. There are various forms of instabilities that may set in rotor-bearing systems. One of the most serious forms is known as the oil-film whirl. This undesirable phenomenen gives rise to the transmission of vibration throughout the equipment which

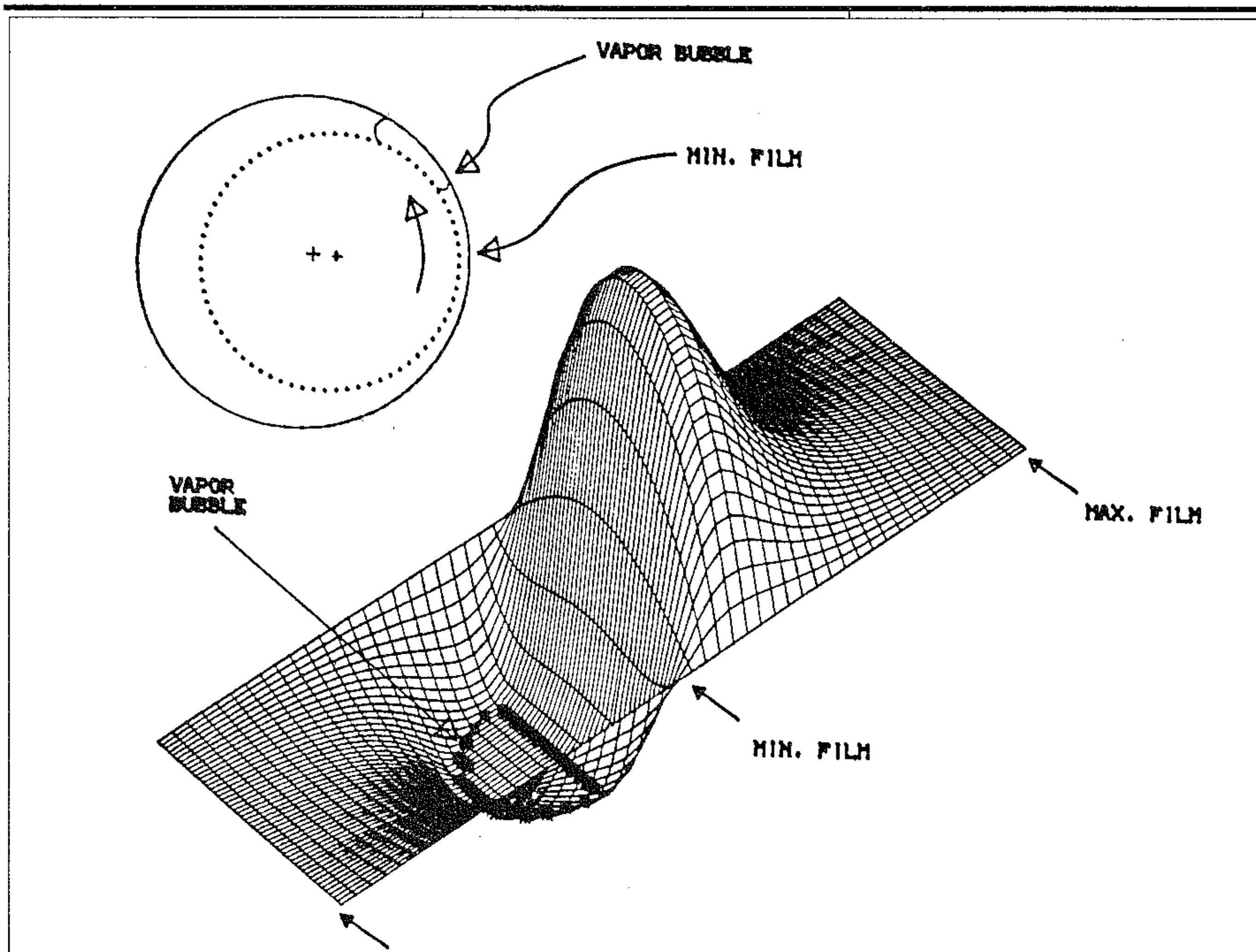


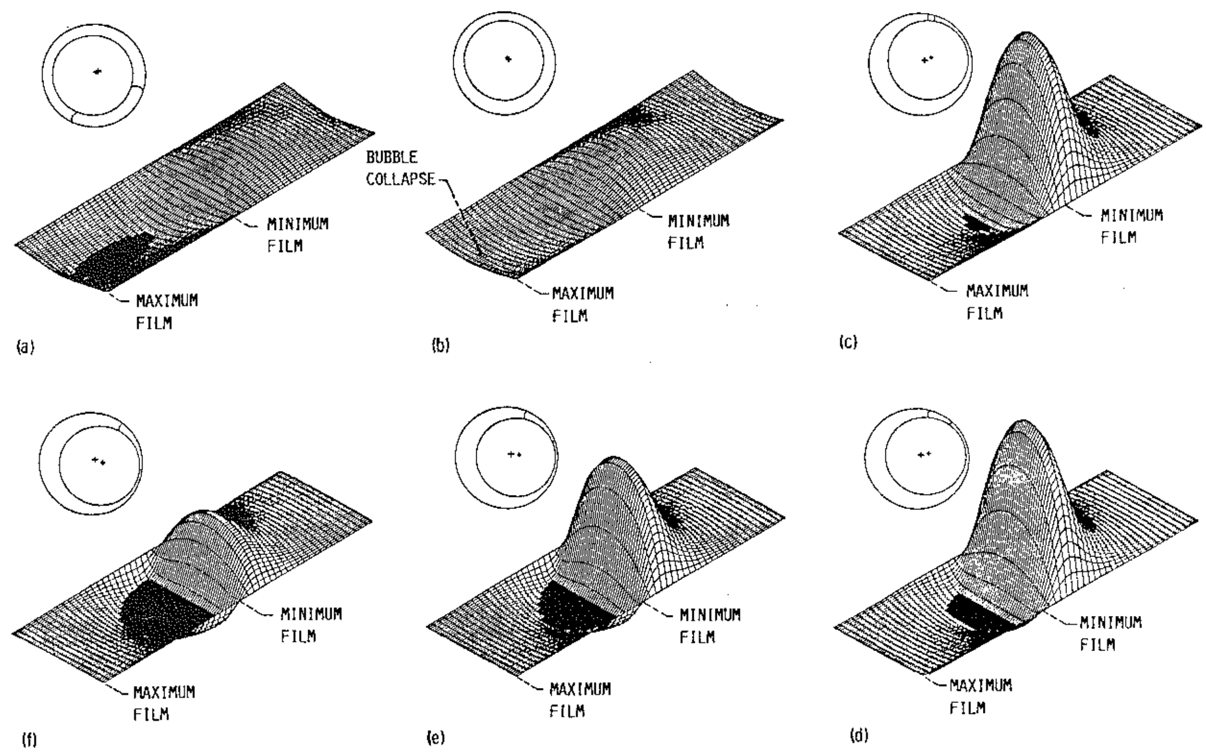

Figure 6

may cause fatigue on the bearing components and leads to an ultimate failure.

It is now known that whirl instability is a form of a self-excited, self-induced vibration, meaning that the motion is controlled by the motion itself. The self-induced vibration continues to build up with the speed as the self-excitation continually transfers energy to the system. Therefore, if the speed of rotation is increased above the whirl threshold, the large orbiting amplitudes results in bearing failure [4].

A cavitation algorithm was developed for dynamically loaded bearings. [5]. The results are obtained by finite difference solution of the time-dependent Reynolds equation. The computational model predicts the formation of the cavitation bubble, its downstream movement, and final collapse. The entire sequence of events are captured in computer generated movies using a three-dimensional graphics package. The results show the pressure generation capacity during normal operation as well as the loss of the entire load capacity at the instant when the oil-whip condition takes place. As an example, a full cycle of cavitation history is shown in Figure 7 which shows the oil-whip condition at T= 33.91 ms at an eccentricity ratio of e = .08.

The pressure plots using the GRAPH3D plotting package are available at NASA Lewis Research Center. The package can perform standard two and three dimensional plots with hidden line removal capability, contour plots and surface and solid modeling. The package runs on an IBM 370/3033 System. An ENVISION 230 color raster




PRESSURE DISTRIBUTION AND BEARING CONFIGURATION FOR FULL PERIOD OF SHAFT WHIRL. ($\omega_S$ = 0 RAD/S. FIGURES (a) TO (f) ARE VIEWED CLOCKWISE IN TIME.)

Figure 7

terminal was used at the development stages of this project. The slides were produced on a MATRIX 3000 Film Recorder.

## CONCLUDING REMARKS

While closed form solutions to mathematical models are always desirable, they are restricted to simple geometrical and physical constraints. Before the advent of computers, researchers were forced to make many simplifying assumptions which would lend the problem to analytical treatments. With high speed digital computers, researchers are now in a position to accurately solve complicated problems by numerical methods.

An inherent disadvantage of numerical methods is a huge output of data generated making the assimilation of the results difficult. For this reason, computer graphics has become an integral part of numerical analysis. Interactive computer graphics is extensively used in commercial numerical analysis packages both during the modeling state, for example, to display mesh elements in finite element formulation and in the presentation of the final solutions.

Various forms of graphical presentation of numerical solutions are discussed. Among the most elegant types are colored 3-D surfaces and solid models generated interactively. To aid visualizing the behavior of complicated models, or to verify the correctness of a computational algorithm one may resort to computer animation. This is particularly important for cases where a fourth dimension, namely time, plays an important role.

A case study of the effect of cavitation on the performance of dynamically loaded journal bearings was presented. A motion picture was created with authenticated some experimental work conducted at NASA Lewis Research Center. The movie was found to be extremely helpful in understanding the mechanism of the flow of a complicated phenomena.

### References

1. Huebner, K., The Finite Element Method for Engineers, John Wiley Publication, 1975.

2. Encarnacao, J., Schlechtendahl, E., Computer Aided Design, Springer-Verlag Publication, 1983.

3. Duane, J., Khonsari, M., "Graphics for Finite Element Analysis," Engineering Design Graphics Journal, pp. 33-40, 1985.

4. Khonsari, M., "Computational Topics in Hydrodynamic Journal Bearings," Proceedings of the ASEE Indiana-Illinois Meeting, pp. 217-223, 1986.

5. Brewe, D., "Theoretical Modeling of the Vapor Cavitation in Dynamically Loaded Journal Bearings," ASME Journal of Tribology, pp. 628-638, 1986.

6. Brewe, D., Khonsari, M., "Effect of Shaft Frequency on Cavitation in a Journal Bearing for Non-centered Circular Whirl," submitted for publication in the ASLE Transactions.

7. Brewe, D., Sosoka, D., "The Study of Journal Bearing Dynamics Using Motion Picture 3-D Graphics," Engineering Design Graphics Journal, pp. 23-25, 1986.

8. Dowson, D., Taylor, C., "Cavitation in Bearings," Annual Review of Fluid Mechanics, V. II, Annual Reviews, pp. 35-66., 1979.

# Comment

from page 8

articles on computer graphics were appearing in the EDG Journal and monographs on computer graphics were being published under the sponsorship of ASEE. (Talk to Steve Slaby, he was coordinating editor of the monographs. Earl Black was writing articles about computer graphics at General Motors, and Bob Christianson was Journal Editor.) When I started working for the General Electric Company in 1955, they were in the process of installing computer graphics capability to "keep up" with the competition in the electric motor industry.

It should be obvious from these comments that there is a known history of computer graphics, that it pre-dates 1968, and it is recognized by vendors and individuals who got their start long before the micro computer revolution.

Among the other items that were also "learned" at that same technical session is that IGES is an "international" standard. Against the possibility that this statement should also find its way into the EDG Journal, let me clarify that the acronym, IGES, stands for Initial Graphics Exchange Standard. It is a long way from being international in any context. Once again the vendors who were present in Louisville were aghast that this statement was made with such aplomb and elicited so little reaction.

It is little wonder that the division has difficulty obtaining vendor support for our activities when we continue to insult those who do support us by our apparent ignorance.

I know you are not soliciting my advice, and I'm not sure what advice I would give if you were to ask for it, but somehow you need to become aware of those in the division who speak from a myopic and unfounded point of view and whose statements are not backed by either experience or good research.

*Larry Goss raises several points in his comment. Our relationship with vendors can always stand improving. I just wish someone would stay long enough with one company so that a relationship could be developed. And certainly there is much misinformation that gets passed around. I did not intend* the editor *to be a contributor to the confusion. The 1968 date was not used to set an exact point in time before which there was no computer anywhere making graphics. Were that the case, non-electronic computers—mechanical contrivances used for plotting perspectives and used in the 14th and 15th centuries—would also be considered as computer graphics equipment as would the X-Y but manual Etc-a-Sketch of the 1930's. The date was used only to illustrate that we are dealing with relatively recent technology. If someone states that they were doing computer graphics in 1955, then they probably were. I told myself that I was doing computer graphics when I submitted card decks to make character graphics on a line printer. In reality, there probably isn't a date that can be fixed as the conception of computer graphics since it represents an evolutionary process, and it was probably wrong to even try.*

*-ed*

# UTILIZING THE COMPUTER IN FRESHMAN DESIGN

Steven K. Mickelson
Department of Freshman Engineering
Iowa State University

The process of design is of great importance in engineering. As computers have increasingly become a vital tool in industrial design, they have also become a vital tool in engineering education of design. At Iowa State we are taking a closer look at these changes caused by the computer and are altering our freshman design component accordingly. The design component at the freshman year provides an important first step in the students' engineering education.

The Accreditation Board for Engineering and Technology (ABET) requires one-half year of engineering design, which translates to approximately 16 semester credits of design. Of this one-half-year ABET asks that the majority be taught in the upper-division level courses, including a senior design course which would utilize the student's background in mathematics, basic and engineering sciences and the humanities and social sciences.

At Iowa State one credit of design required by ABET is taught in our freshman engineering graphics and introductory design course. This course is a 3 credit semester course which integrates fundamental graphics, computer graphics, and engineering design. One of our goals in this course is to expose the freshman engineering student to a design experience. Although students at this stage of their studies do not have the technical background to deal with dynamic equations or optimization techniques, students can gain valuable experience with regard to the overall design process. Another goal in this course is to give the students an appreciation of the capabilities of computer graphics in the design process. Early introduction to the design process including computer graphics is desirable and can be a useful stepping stone to upper-level design.

## FRESHMAN DESIGN AT IOWA STATE

To develop the design experience at the freshman level at Iowa State, the students are assigned open-ended design problems. By following a structured design process the students develop a solution to a problem and communicate their work with a written report and an oral presentation. The following design steps are used:

1. Identify Need
2. Define Problem
3. Search
4. Criteria, Constraints
5. Alternatives
6. Analysis
7. Decision
8. Specification
9. Communication

Conceptual design is accomplished by breaking up a typical class of 25 students into teams of 3-5 students.

*"Early introduction to the design process including computer graphics is desirable..."*

These teams are then assigned the same open-ended design project. Each team then develops a problem statement, conducts a limited search (due to time limit), decides on criteria and constraints, and develops several possible alternative solutions. After going through a decision process to determine the "best" solution, conceptual sketches of the solution are developed for communication purposes. The team then communicates its results in written and oral form.

Presently at Iowa State the analysis and the specification steps are omitted with separate analysis and design drawing problems substituted into the design process after the conceptual design. The analysis step in the design process is introduced to the students by using problems which involve repeated analysis and computer-assisted problem solving. Typical problems are discussed in references 1, 2, and 3. The advantage of using the

DATE: March 26, 1986
TO: Fr. E. 170H DESIGN TEAMS
FROM: MANAGEMENT
RE: Drainage Facility for State III Subdivision

Our consulting firm has been asked for a preliminary study and design of a drainage facility. This facility is necessary to carry surface water runoff from an upstream watershed through the new Stage III subdivision in Ames, Iowa. Such a facility must be designed to carry a flow of approximately 80 to 100 cubic feet per second. The total length of the proposed facility is 1000 feet. The first three-fourths of the distance is in a residential, single family area, and the remainder is in farm land. The land appraisals, land values are $2500/acre for agricultural land and $5000/acre for the residential property. Consideration should be given to maintenance, aesthetics, safety, costs, and other appropriate items when determining necessary land acquisition for the project.

Based on our previous experience, we will limit our alternatives for the project to corrugated metal pipe, reinforced concrete pipe, reinforced concrete box, and an open channel. The preliminary design and recommendations must be completed by Wednesday, April 16. Written documentation and an oral presentation, following company guidelines, will be given to our clients on that date.

A sketch of the area is in a file titled 'Stage'.

Figure 1

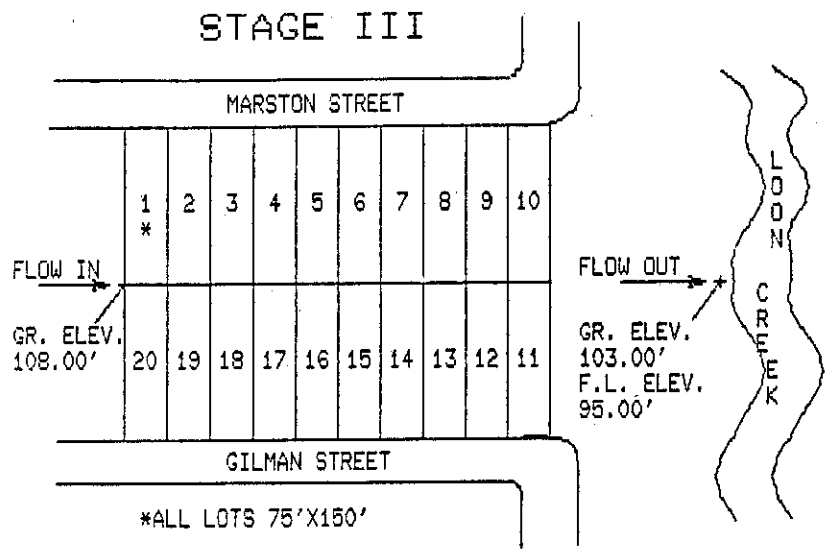



Figure 2

computer in the analysis step is easily realized by the students. Since the students in the design class are mainly freshmen, with little or no programming experience, it is not possible to have them program their own analyzers. Therefore, it is necessary for the instructor to have the appropriate software analyzers available when the students need them.

Specific design drawings, as determined by the instructor, are used for the specification phase of the design process. Problems have ranged from design drawings for a three point hitch to design drawings for a child's wagon. These problems have been separate from the conceptual design.

The major drawback of this present approach is that the introduction of design is somewhat fragmented with the separate conceptual, design-by-analysis, and design drawing problems. Ideally an "overall" design which included all the design steps would be the best. In a new honors course, this overall design was attempted

## NEW HONORS COURSE

In the spring of 1986, Freshman Engineering at Iowa State introduced an honors section of our engineering graphics and design course. Prerequisites for the honors section were trigonometry, credit or enrollment in the first semester calculus course, and credit or proficiency in FORTRAN programming language.

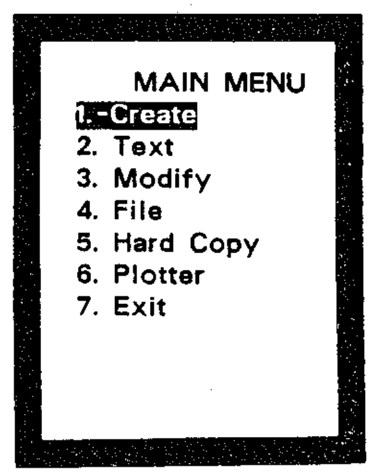



Figure 3

Due to the strong math and computer prerequisites, utilization of these skills in our introductory design project was possible.

The semester before the honors course began, a team of three faculty undertook the task of developing an open-ended design problem that would effectively incorporate the computer. The design project was to be assigned

*"One disadvantage was the preparatory time necessary for the instructors..."*

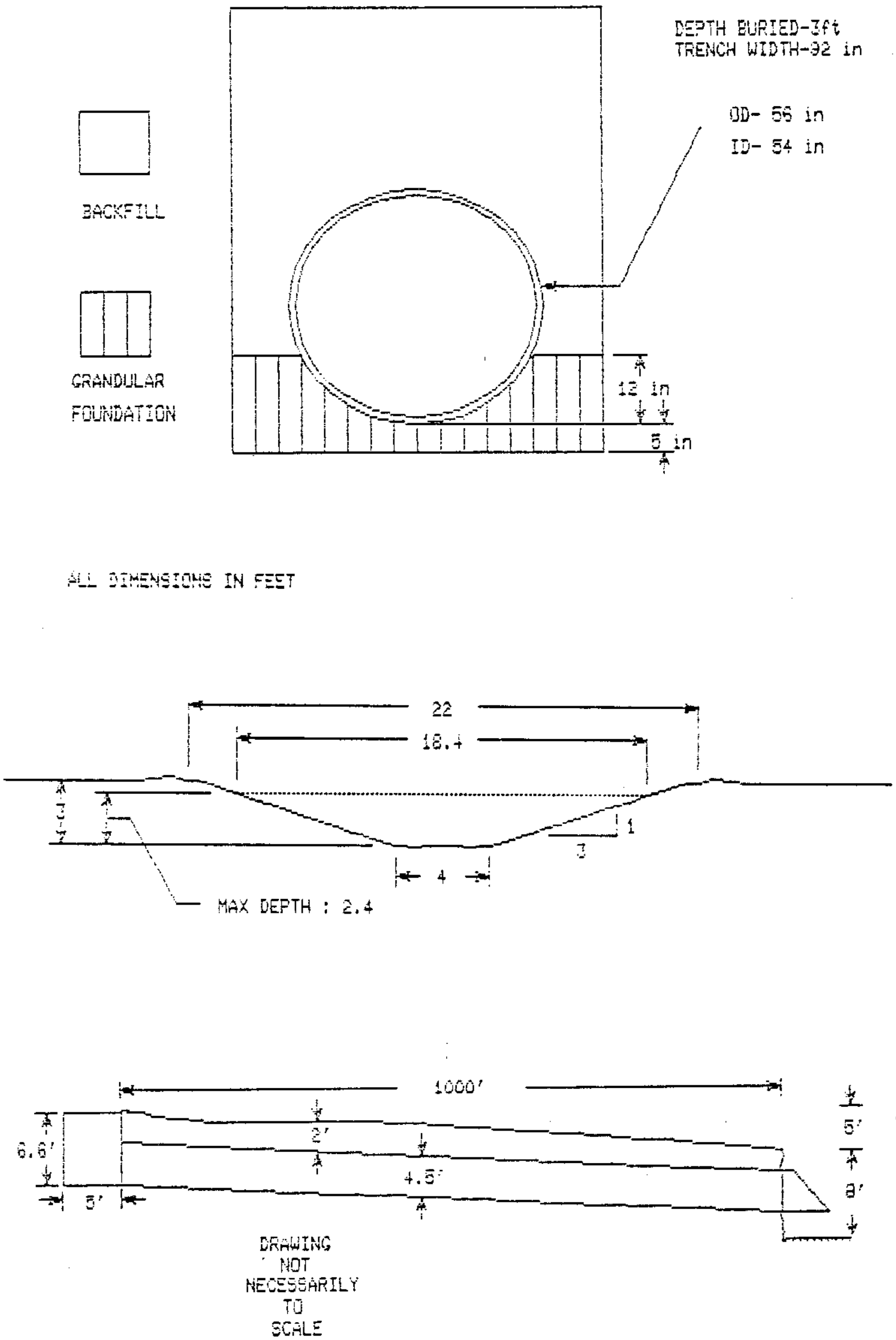

DEPTH BURIED-3ft
TRENCH WIDTH-92 in
OD- 56 in
ID- 54 in
BACKFILL
GRANDULAR
FOUNDATION
12 in
5 in
ALL DIMENSIONS IN FEET
22
18.4
3
1
3
4
MAX DEPTH : 2.4
1000'
6.6'
2'
4.5'
5'
5'
8'
DRAWING
NOT
NECESSARILY
TO
SCALE

Figure 4

to 4 or 5 student teams, with each team solving the same project. It was our goal that this introductory design project would consolidate conceptual design, design-by-analysis, and design drawings into an "overall" design project, plus incorporate the computer into the conceptual design and design-by-analysis sections.

## OPEN-ENDED DESIGN PROJECT

The project developed by the faculty team was a drainage facility design and study for a new housing project. Each team of students was given known criteria and constraints as presented in a letter from management of a fictitious company (shown in figure 1). Also available to the teams was a computer sketch, in their computer directory, of the specified drainage area (figure 2). Limiting the alternative solutions for the project to four types of drainage facilities, the teams were asked to determine the "best" alternative and to present their preliminary design and recommendation in a written report and in an oral presentation.

## UTILIZING A 2-D GRAPHICS PACKAGE

The teams were asked to use "SKETCH," our 2-D graphics software package, for all their conceptual drawings for their reports. "SKETCH" is an interactive 2-D drawing package that was developed by faculty specifically for our graphics course as an introduction to some capabilities of computer-aided drafting.

SKETCH is menu-driven, and is a quick graphics tool which students can easily learn and utilize. The main menu for SKETCH is shown in figure 3. SKETCH allows the user to create graphic elements such as an arc, box, curve, line, polygon, or marker. Various sizes and angles of text can also be created or added to a drawing. SKETCH creates a database that can be easily saved, redrawn, erased, or modified. A hardcopy of the created graphics can easily be obtained using a dot matrix printer or a pen plotter. SKETCH was introduced early in the semester to the honors students, Examples of their computer sketches are shown in figure 4.

## UTILIZING FORTRAN IN ANALYSIS

After completing the preliminary design, the individual team members were then asked to limit their study to an open channel with a trapezoidal cross section. They were to assume that this was the channel type picked by management as the best conceptual design. They were then asked to analyze this channel

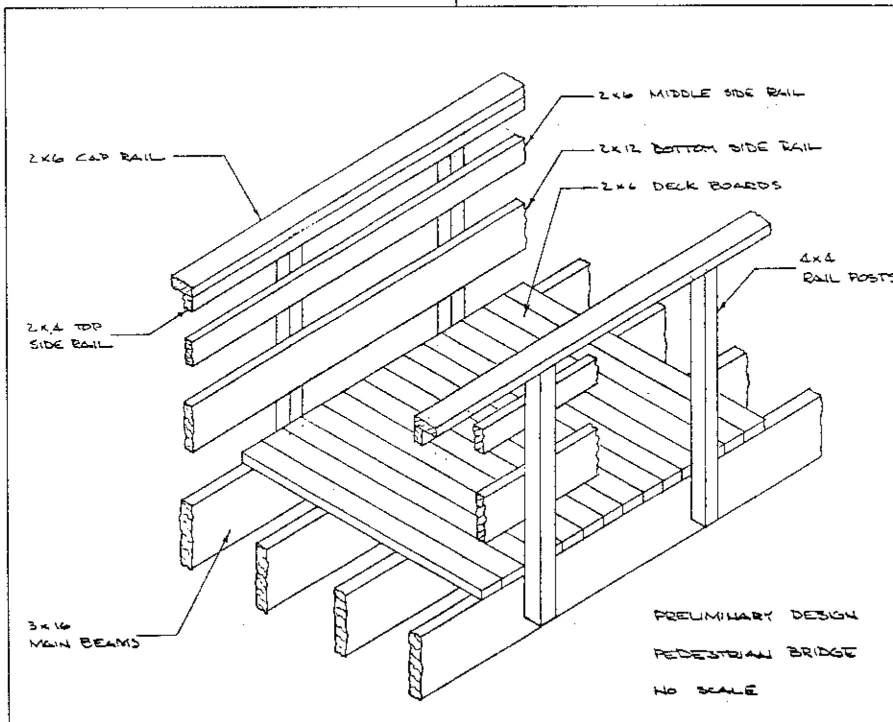



Figure 5

type using their FORTRAN programming experience along with the Manning Equation. The Manning Equation relates the velocity of fluid in an open channel to the roughness coefficient, the hydraulic radius, and the slope of the channel. The Manning Equation is as follows:

$$V = \frac{1.486}{n} R^{2/3} S^{1/2}$$

where,

V = Velocity, (ft/sec)
n = Roughness Coefficient
R = Hydraulic Radius = $\frac{\text{Cross Section Area, ft}^2}{\text{Wetted Perimeter, ft}}$
S = Slope of the channel along the length

After writing a FORTRAN program relating these values, the students were asked to discuss the effects of changing the roughness of the channel, the slope of the channel, and the side slopes of the trapezoidal cross section. Finally the students were asked to recommend a cross section for the best design using the criteria and constraints developed in the conceptual design process along with the analysis findings. This approach greatly enhanced the analysis step in the design cycle. The major advantage of this analysis method was that the students created their own analyzer software, and quickly determined the effects of changing design variables.

To incorporate the design drawings into this "overall" design project, the team members were given a preliminary design for a pedestrian bridge for crossing the design facility. (Figure 5) The students were required to present selected design and detail drawings of various parts of the bridge, plus compile a bill of materials for the total bridge design.

## CONCLUSION

The incorporation of an "overall" design project, including design, design-by-analysis and design drawings, was well received by the students, and was successful in introducing freshman students to the design process. The potential of the computer in the design process was realized in the report drawings and in the analysis step of the design cycle. "SKETCH" software was used to develop computer drawings of design concepts, while FORTRAN was used to assist in the experience of the iterative process of the analysis step of design.

One disadvantage was the preparatory time necessary for the instructors to develop this type of design. Yet, time was saved due to the fact that the students wrote the analyzer software instead of faculty.

The honors section of our graphics and design course will try this overall design approach again this spring.

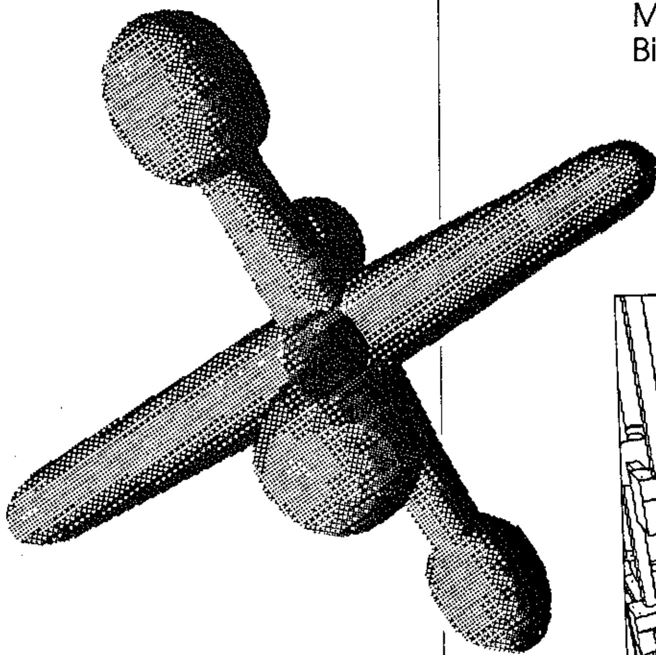
## REFERENCES

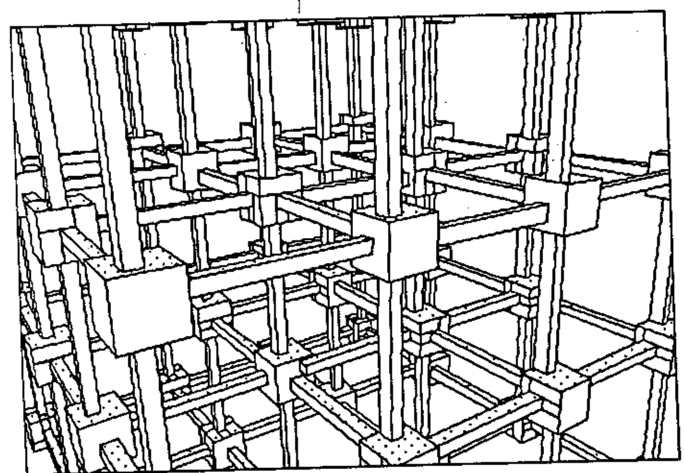
1. Eide, A.R., R.D. Jenison, "Integration of Computers into First Year Design," Paper presented for ASEE, University of Cincinnati, June 1986.

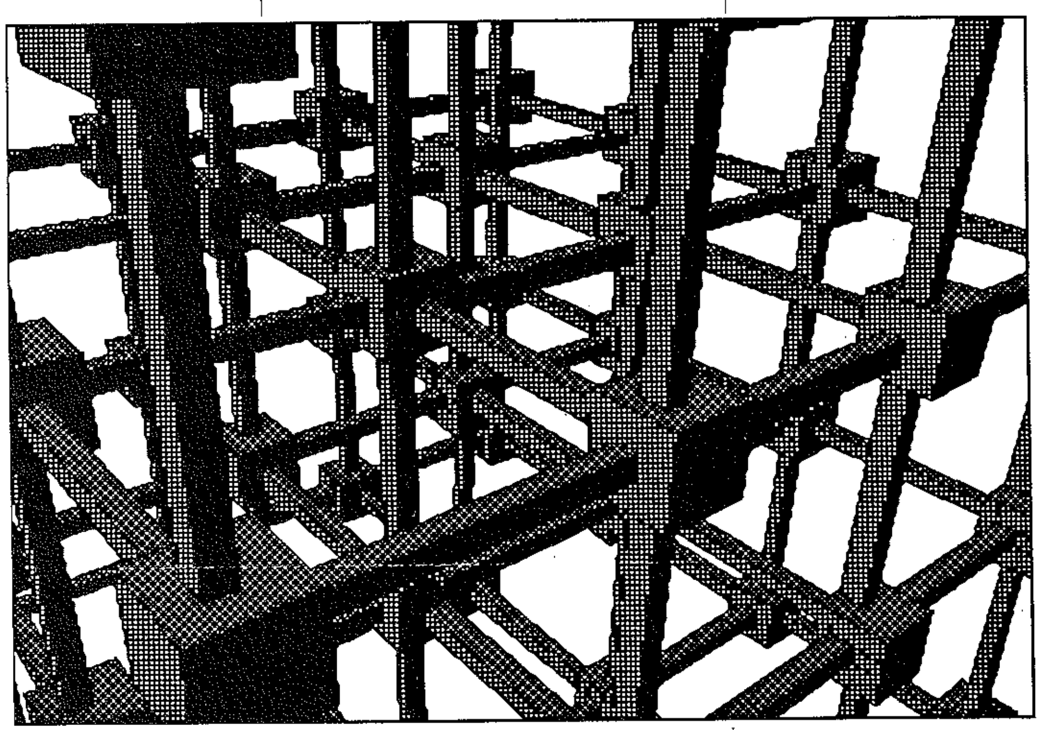
2. Eide, A.R., R.D. Jenison, L.H. Mashaw, L.L. Northup, and C.G. Sanders, "Engineering Graphics Fundamentals," McGraw-Hill, New York, 1985.

3. Jenison, R.D., "A New Look at Introductory Design Instruction," Paper presented at the 93rd annual convention of ASEE, Georgia Tech, June 1985.

Microcomputer Models by
Bill Ross/North Carolina State University







# Development and Implementation of a CAD Application for a Numerical Control Machine Operation for use in a Laboratory Course

J. Patten, T.A. Allison,
H. Razavi
Departments of Engineering Technology and Computer Science
University of North Carolina at Charlotte

## INTRODUCTION

This paper provides a description of the development of a CAD based system for the design, simulation, numerical control (NC) program generation and execution utilizing CAD/CAM principles and technology.

The system configuration is based upon commercial equipment, hardware and software, generally available at colleges and universities. The system design information is available, through this paper and supplemental system documentation, to interested persons for development of their own instructional programs. This paper is intended to provide a stimulus and useful information for others to develop systems and programs for hand-on instruction and teaching of CAD/CAM applications. The program goal at UNCC was to develop a system to provide CAD/CAM instruction utilizing available equipment. The emphasis of the program was to develop a teaching tool for student use in learning CAD/CAM principles and applications in a laboratory and classroom environment.

The project involved the adaptation of an numerical controlled (NC) milling machine to "direct computer numerical control" (DNC/CNC) using modern computer integrated manufacturing (CIM) techniques such as computer aided design (CAD) and computer aided manufacturing (CAM). The project involved four distinct but interrelated steps. These are: 1) Laboratory Development: Project origination and management, 2) Emulation of the NC paper tape function through software, 3) Design and construction of an electronic interface board, and 4) Development of software to interface a commercial CAD software program (AUTOCAD) through the interface board to the NC machine. This project has resulted in two distinct new uses of the NC machine. First, programming can be accomplished off-line using standard hardware and software eliminating altogether the use of the paper tape, teletype machine and NC reader. Secondly, a design developed on a CAD package can be simulated and downloaded to the NC machine as machining instructions, again eliminating the use of the punched paper tape.

**"..results of this project are used to provide educational training to students studying CAD/CAM with hands-on applications in a laboratory environment."**

The project was initiated as a method to upgrade and modernize existing equipment by replacing the paper tape, teletype, and reader of the NC machine with a personal computer (PC). The personal computer would serve as programmer, editor and controller for the NC machine functions. This required a communications link between the PC and the NC machine. The project was expanded to include linking the design and manufacture (CAD/CAM) of a milled part. This involved developing software to accept a CAD program file, graphically simulating the milling operation and creating a milling instruction file which could be downloaded to the milling machine via an interface board.

As the project was developed to serve as a laboratory experience, the IBM-PC was selected as the computer for the project. The UNCC College of Engineering has available a sufficient number of suitable units for this purpose. The general arrangement envisioned was that of students creating the NC programs, via programming on a PC or through a CAD drawing and NC simulation routine, in one of the PC or graphics laboratories. The students then transfer the files to a PC located at the milling machine for execution of the machining operations. This arrangement provides for maximum utilization of resources and exposes the greatest number of students to the available equipment.

The conduct of the project involved the active participation of many individuals forming a project team. Four faculty members, two from computer science, and one each from the electrical and mechanical disciplines; six students, four from computer science and two from mechanical disciplines; and four technicians, one each from the electrical, computer, mechanical and machine shop trades were directly involved in the project. The project took approximately one and a half years to complete. Partial funding for the project was obtained from a small grant awarded from the College of Engineering. The students involved in the project accounted for two senior projects and three graduate student projects. The total system was developed using existing in-house equipment, consisting of the NC machine, PC's and commercial software packages.

The paper is divided into four logical sections detailing the development of the items listed above. Each will be covered separately in the following sections. These sections are:

1. Laboratory Development:
2. Paper Type Emulation
3. PC-NC Interface Device
4. CAD-NC Software

```
IMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM;
:  DIRECT   NUMERICAL   CONTROL   :
HMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM<

1 = Create/Edit Workfile        5 = Erase Workfile

2 = Save File                   6 = Transmit Workfile

3 = Retrieve File               7 = Display Workfile

4 = Remove File                 8 = Help

             9 = Exit Program

      Enter selection and press RETURN:
```

Figure 1

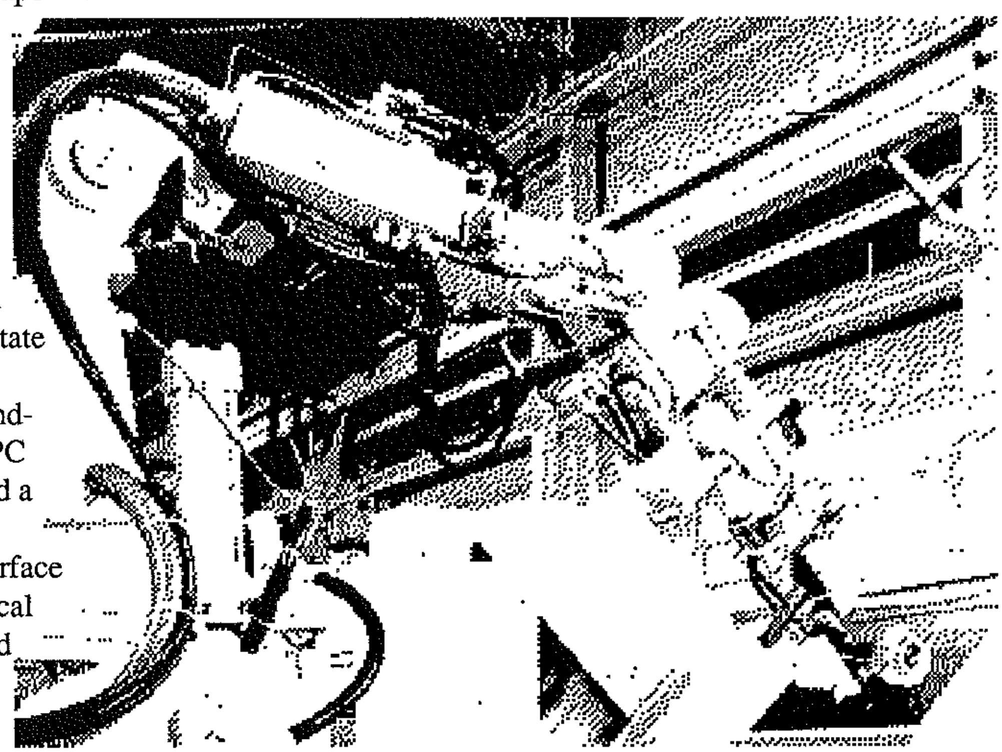
This paper will describe the various stages of the project in detail.

## LABORATORY DEVELOPMENT

A lack of sufficient laboratory equipment, a problem common to many universities and colleges, was the seed for this project. The College of Engineering has an old Bridgeport Mill equipped with a Slo-Syn NC controller. This system had been used for demonstrations, and was a remnant of an old technology. Two degree programs, mechanical engineering and engineering technology, were teaching NC, CAD/CAM/CIM courses without a laboratory component. As a partial solution for the need to expose students to this technology, first the NC mill was put board was necessary since neither the electrical characteristics of the timing/handshaking characteristics between the PC and the NC mill match.

The printer port of the PC operates under standard Centronic's port specifications when driven with most PC software. The interface board, however, makes use of only the Strobe and Busy control lines thus assuming the use of custom port driver software. The NC mill port is located in the SLO-SYN controller and exists as a cable and connector to the existing paper tape reader. The tape reader port uses two handshaking signals. Only one of these signals, the Forward signal, is required by the interface.

Matching the handshaking signals between the PC and the mill required a simple state machine design. The state machine coordinates the handshaking signals so that the PC considers the interface board a printer and the NC mill controller considers the interface board a tape reader. Electrical characteristics were matched by using LS 5 volt TTL logic on the PC slide of the interface and 15 bolt discrete transistor logic on the NC mill side of the interface. Connectors were provided so that the OX printer cable and the NC mill tape reader cable could be plugged directly into the interface board.



Power for the board was derived from a separate 120 volt supply.

When connected, the interface board and PC emulate the tape reader. All controls on the SLO-SYN controller function normally in that pauses, speed changes and the like can be made from the controller. The only function not available is the tape rewind. The rewind function must be handled by software at the PC.

### CAD-NC SOFTWARE

The original project envisioned utilizing CAD drawings directly, as input into the NC machine, to eliminate altogether the need for user developed NC programs. The paper tape emulation software, discussed previously, still required the user to have an understanding and detailed working knowledge of NC programming using G and M codes. What was desired was a system or program which would make the NC program transparent to the user. A student in this case would be able to design a part on a CAD system, suitable for production on the NC mill, and transmit this design to the NC machine in a form directly readable or executable by the mill. The emulation program and interface devices were already developed, therefore these were used as the basis for the CAD-NC software program. The alternative which was not selected, was to eliminate the existing NC SLO-SNY controller altogether and control the stepper drive motors directly from the software through the necessary interconnecting hardware. The approach chosen was to use the PC-NC interface device developed for the emulation program, and develop a software package that would format, simulate and transmit a CAD file of a drawing directly to the existing NC controller through the interface device. The interface protocol developed for the emulation software was again used to communicate to the NC machine. However now, the NC program is transparent to the user. Thus, the user need not be familiar with NC programming to execute this operation. At the same time, it was realized that the user may want to modify or document the program. The program code is made available to the user in a format identical to the paper tape and emulation program, complete with TAB, EOB, etc., statements included.

The above was accomplished using the AutoCAD software package coupled with a simulation package and post-processor to drive the NC machine. These two programs, the simulation and post-processor, were again developed as student projects. AutoCAD was selected as it was running in a PC environment and was available on 6 IBM-PC/XT computers in the graphics laboratory. Another software package, DOGS by PAFEC, was considered for this use but was rejected as it was not operating on the IBM-PC and, therefore, it was not easily portable or transferable to the manufacturing laboratory where the NC machine is located. This option will be investigated in the future when the College installs a local area network (LAN) connecting the computer laboratories to the engineering and technology laboratories.

Again, the goal of the PC-NC software was to make the NC programming transparent to the user. The user in this case, being a student, would be functioning as a draftperson or designer. They would develop a design drawing, using AutoCAD, and then utilizing CAD/CAM and CIM technology transmit the design to the NC mill, thus eliminating the programming step entirely. No new technology had to be developed for this project. The application of existing technology was integrated into an educational environment. This technology exists as turnkey systems from vendors and industry makes use of these systems. However, the cost of such a system, which can amount to tens of thousands to hundreds of thousands of dollars, was prohibitively expensive to the College and University.

The other item addressed was the question of whether or not the students should be aware of or understand NC programming, G and M codes or APT. The software package was developed including the capability to provide a complete listing of the NC code for documentation purposes. The student can type in the code, using the paper tape system or the emulation software, and obtain the same end results produced by the CAD-NC software. The instructions to the milling machine will be identical, for a given part program, regardless of the method used. As with any computer program, the students should be capable of verifying the results. The students can determine the NC code required to produce their design and compare it to the computer code generated from this software package. With the aid of the simulation software, the students can evaluate the effect of various cutter sizes, in the range 1/64 inch through 2 inches. The final form of slots, radii, etc., can be evaluated and an optimum cutter size determined.

The visual feedback provided by the simulation was obtained with the use of the Graphics Tool Kit software package from IBM. This provided the necessary software tools to produce an animation of the milling operation. However, the simulation portion of the software package is the limiting factor to more sophisticated part designs. At present, not all NC milling instructions are executable due to limitations in the simulation program. As the post processor, that generates the NC code, relies on the simulation program for input, any limitation in the simulation restricts the actual operations executable by the NC mill. This is not a major problem as the NC mill is fairly limited by nature of its design. The stepper motors, on the x and y table movements, are open loop control. The distance, and not speed, is controlled. Therefore, the program is restricted to straight line motion and simple part configurations. The limitations imposed by the simulation software do not adversely affect the learning process of the students, relative to the project's original goals. While the system would be inadequate for industries' needs, in terms of a complete functional CAD/CAM system, for student use as an educational tool the system is excellent. The limitations imposed are documented and actually help to simplify the system, making it easier for students to learn and work with.

## RESULTS

The students now have two additional methods available to them to accomplish NC programming. The paper tape punch method is also still available. The first method, paper tape emulation using the IBM-PC, still requires the students to know NC programming (G & M codes, etc.). The system does automatically take care of some of the overhead functions such as RWS, TAB, EOB, etc. The system does allow the students to edit, save, and transfer programs which previously were not easily accomplished. The system also allows many more students access to the NC system as many PC's are available for programming whereas only one single purpose dedicated paper tape punch was formerly available. The second method of transferring CAD files directly to the NC machine, now made it possible for a student to study CAD and NC operations, in a CAD/CAM environment, while eliminating the requirement of a prior knowledge of an NC programming language. At the same time graphical simulation of the NC milling operation, required to produce the part, is provided for visualization to promote design optimization.

An integral part of the project was the requirement that the student must provide documentation of their work (2-5). In addition to the project report, the students developed user's guides. These are used to train students on the operation and use of software programs. This provided the vital link between the project and the end user, the student.

## CONCLUSIONS

CAD/CAM and CIM was made possible in an undergraduate student laboratory environment with a modest investment ($1,200) through utilization of existing equipment and resources, primarily the NC mill and PC's. The $1,200 investment was pleasantly spent to compensate a faculty member for time devoted to development of this project.

Six student projects, 1 mechanical engineering technology, 1 mechanical engineering and engineering science, 1 undergraduate and 3 graduate computer science projects were completed in conjunction with the project. This project represents an excellent example of using student talent to assist with the development of instructional programs, educational goals and requirements. The use of students to assist with such projects is unquestionably a benefit to all concerned. With the number of people involved in the project, 6 students, 4 faculty and 4 technical staff members, the project obviously required careful attention to details such as scheduling and communication within the group and among the various project members.

The end use of the project results is the real key and the long-term reward. Many students now have access to NC/DNC/CNC technology which would not have been possible previously. This was also accomplished using a CAD system applying modern CAD/CAM techniques to a university instructional laboratory environment. Enough options were designed into the program to provide broad student exposure and educational experience in the area of CAD/CAM and NC technology. Additionally, faculty and technical staff further advanced their capabilities and experiences. This will undoubtedly lead to new and interesting and useful endeavors.

The project was truly interdisciplinary, as indicated throughout this paper, as are most manufacturing tasks. It involved the departments and staffs from computer science, mechanical engineering and engineering science, engineering technology, electrical engineering, and technical assistance and cooperation at all stages of the project. While the computer science students provided the software development, mechanical engineering and technology students provided the initial impetus of the project. It will be primarily the manufacturing technology and mechanical engineering students who will utilize this resource. The project does provide the computer science and electrical engineering and technology students an example of a successful design project.

Overall, looking back, the project was quite successful. The only major hitch was finding faculty time to develop the PC-NC interface board.

As with most projects of this nature, a wish list was developed for future project work. These items will be the starting point for additional undergraduate student projects, primarily senior design projects, and possibly graduate student projects. More importantly, however, the results of this project are used to provide educational training to students studying CAD/CAM with hands- on applications in a laboratory environment.

Another added benefit of this project is the possibility of a cooperative venture with local industries. At least one company has expressed an interest in the project, from the standpoint of implementing the system for use in a machine shop environment. Such an arrangement will provide additional rewards which were not originally considered in the project.

## REFERENCES

1. "Interfacing Microcomputer with Numerically Controlled Machine," Z.J. Czajkiewicz and Y. Parikh, Department of Industrial Engineering, Wichita State University

2. Computerized Numerical Control, Nicholas M. Kontos, CSC 450/451, Volumes I, II & III, Project 85-21, May 1986, University of North Carolina at Charlotte, Department of Computer Science.

3. "Direct Numerical Control using Program DNC; User's Manual/Guide," Nicholas M. Kontos, UNCC-Computer Science Department, May 1986.

4. "PC Control of a Numerically Controlled Milling Machine," Arun Chopra, Marcia Hostetler, David Seniw, UNCC-College of Engineering, May 1986.

5. "Computerized Numerical Control", Users Manual, Software Version 1.0, Arun Chopra, Marcia Hostetler, David Seniw, May 6, 1986.

# ASEE
## 1988 Annual Conference
## Portland, Oregon
### Hosted by the
### University of Portland
### June 19, 23

## Engineering Design Graphics Division Hosted Sessions

**Monday, June 20, 1988**

1138
Creative Engineering Design Competition and Display Judges Breakfast

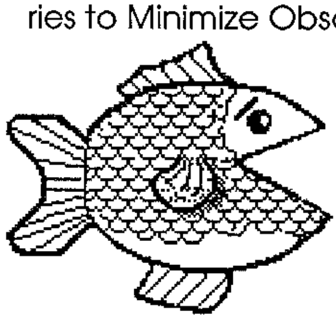
1238
Student Evaluation of CADD Instruction I

1239
Creative Engineering Design Competition and Display

**Tuesday, June 21, 1988**
2238
Student Evaluation of CADD Instruction II

**Wednesday, June 22, 1988**
3238
Designing Engineering Graphics Laboratories to Minimize Obsolescence

A full schedule of evens is planned for the conference to include spouse and family activities, professional tours, and workshops. A tour of Oregon's north coast, a Columbia River Gorge and sternwheerler cruise, tours of the city, the Pendleton wollen mills (with shopping), wine country tours—all are plabbed attractions. For the children, consider the Oregon Museum of Science and Industry, Washington Park Zoo, Flying M Ranch, or nature hikes. Tours of Intel, Precision Castparts, Hewlett-Packard, General Electric, and the University of Portland are scheduled, and if you are interested in professional workshops, pre- and post-conference workshops may be attended.



**Salmon on the Barbie!**
$8 for salmon, drinks, and salad
Doorprizes and Entertainment
Sunday, June 18 5-7pm at the Quadrangle

# Learning to Read 3-D Data Bases Through Solving Descriptive Geometry Problems

Gerald W. Hannah
Associate Professor
of Engineering
Mount Allison University
N.B., Canada

## INTRODUCTION

The type of three-dimensional database that students in Engineering Graphics at Mount Allison first encounter is that of a wire-frame model. Such a database consists of the coordinate data of the figure together with a column giving a code specifying if the particular movement is made with, or without, drawing a line. Upon first encountering such a database students view it with apprehension; they appreciate what it represents, but they cannot "read" it.

This paper describes a method of solving descriptive geometry problems in which students gain expertise in working with the problem's 3-D database. In this method the figure is first translated to the origin then is subjected to a series of rotations about the appropriate axes (and possibly further translations) so that the distance or angle of interest shows its true value in the X—Y plane. This approach is an alternative to the use of a "viewpoint" described by Helmlinger and Northup (1) and although the sequence of transformations needed here lacks the elegance, in comparison, each step is confirmed by simple inspection of the database.

The method utilizes a commercial graphics program "GRAPHPAK" (IBM Corporation) to display isometric diagrams of the figures as well as the database. A calculator and a sketch pad are the only other tools used. A salient feature of the method is that the computer graphics program and display are not essential components but are simply aids, to be discarded once the method has been learned. The computer requirements are then lowered to a text display unit running a matrix-handling program.

The method described here was used at Mount Allison University for the first time in the winter term of 1985-86 in a sophomore course in Engineering Graphics and Design.

*"Prior to learning this method students have completed the manual graphics techniques..."*

## PREPARATION

There are three parts to the preparation:

- background in graphic techniques
- using the graphics program
- geometrical transformations

### Background in Graphite Techniques

Prior to learning this method students have completed the manual graphical techniques of solving problems involving lines and planes in space - true length of a line, distance from a point to a line, mutual perpendicular to skew lines, true shape of a plane, perpendicular from a point to a plane, and the location of the point at which a line pierces a plane. Thus they have developed a reasonable ability to visualize relative orientations in space.

### Using the Graphics Program

Since the computer graphics program being used, GRAPHPAK (2), is an APL workspace, some knowledge of APL becomes an essential part of the preparation. The amount required is actually quite small, consisting of the method of creating two-dimensional arrays, displaying arrays, performing matrix addition and multiplication, plus some housekeeping commands.

The database of a figure, termed the "object matrix" in GRAPHPAK, is to be entered in the following format:

Col. 1 - if line is to be drawn to this point from the previous point; 0 if no line is to be drawn
Col 2. - X Coordinate
Col 3. - Y Coordinate
Col 4. - Z Coordinate

For example, the database for the plane of Figure 1 would be, as entered into the matrix OBJ:

OBJ

| | | | | |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | (pt.1) |
| 1 | 80 | 60 | 0 | (pt.2) |
| 1 | 50 | 0 | 90 | (pt.3) |
| 1 | 0 | 0 | 0 | (pt.1) |

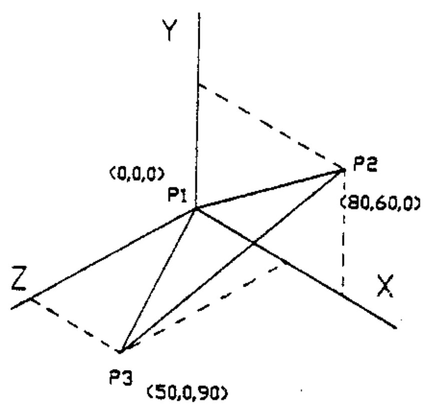



Figure 1

GRAPHPAK provides simple commands to produce drawings of the database on the screen:

THREE VIEWS OBJ
- gives the three principal orthographic views

SKETCH ISOMETRIC OBJ
- gives an isometric drawing

The latter is found to be the most useful in beginning to work with this method.

### Geometrical Transformations

Movement of a figure in space is accomplished by changing the coordinates of the points in the database in an appropriate, systematic manner. Such operations are termed geometrical transformations, two of the most common of which are those which perform translation and rotation. These matrix operations are probably best understood by considering the two-dimensional case, first, then the three-dimensional.

Two-dimensional translation could be accomplished by adding the displacement values to all the elements in the X- and Y coordinate columns using a matrix summing operation. The same result can be accomplished with matrix multiplication, using the concept of homogeneous coordinates (3), this being the generally preferred method since it permits concatenation of transformations. Translation of a point at coordinates $x$ and $y$ by amounts $x_t$ and $y_t$ is performed by Figure 2.

The transformed coordinated being $x'$ and $y'$ , as shown in Figure 2.

Two-dimensional rotation about the origin is relatively easy to perform (Figure 3).

Using homogeneous coordinates increases the order of the operational matrix to three, thus making it conformable for matrix multiplication with a translation matrix.

$$[\ x' \ \ y' \ \ 1\ ] = [\ x \ \ y \ \ 1\ ] \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ x_t & y_t & 1 \end{bmatrix}$$

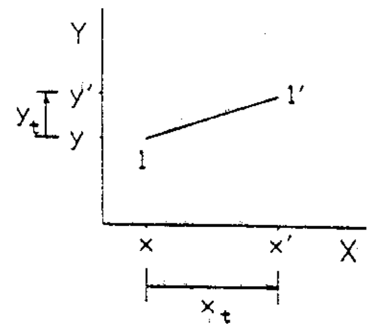



Figure 2

$$[\ x' \ \ y' \ \ 1\ ] = [\ x \ \ y \ \ 1\ ] \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

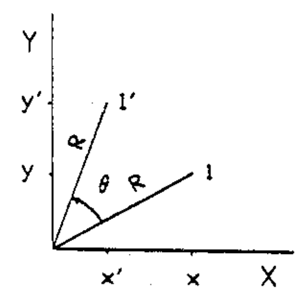
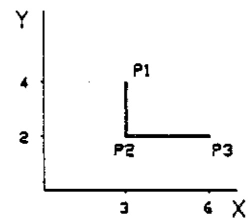



Figure 3

Rotation about some point other than the origin is accomplished by first translating the figure so the axis point is at the origin, then rotating the desired amount about the origin, and finally, translating the figure so the axis point is moved from the origin back to its original position. This is illustrated by the example shown in Figure 4.

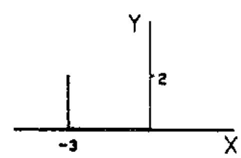
**Example**: The coordinate matrix of the object shown in Figure 4(a) is given in homogeneous form in matrix C. It is desired to rotate the object about point 3 through an angle of 90 degrees, clockwise.

$$C = \begin{bmatrix} 3 & 4 & 1 \\ 3 & 2 & 1 \\ 6 & 2 & 1 \end{bmatrix}$$



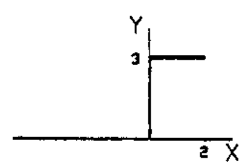

To translate the object so as to place point 3 at the origin, multiply C by $T_1$ , where

$$T_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -6 & -2 & 1 \end{bmatrix}$$




Then to rotate the object -90 deg. about the origin, multiply the resulting matrix by R, where

$$R = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$




Finally translate the resulting matrix back so that point 3 occupies its original position; multiply it by $T_2$ , where

$$T_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 6 & 2 & 1 \end{bmatrix}$$

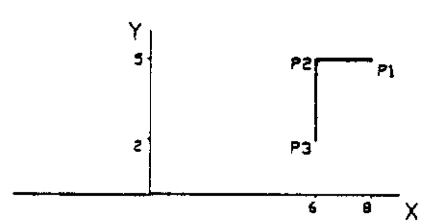



Figure 4

Three-dimensional translation and rotation transformations must accommodate an additional equation, for z, as in the following translation operation:

$$[\,x'\ \ y'\ \ z'\ \ 1\,] = [\,x\ \ y\ \ z\ \ 1\,]\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ x_t & y_t & z_t & 1 \end{bmatrix}$$

Three-dimensional rotation about the Z-axis is equivalent to the 2-D case of rotation about the origin:

$$[\,x'\ \ y'\ \ z'\ \ 1\,] = [\,x\ \ y\ \ z\ \ 1\,]\begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

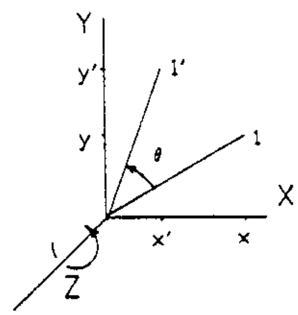



Figure 5

The positive direction of rotation is taken as the direction a right-hand screw must be turned to advance along the axis, as shown in Figure 6.

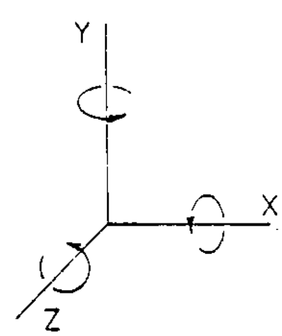



Figure 6

Applying positive rotation about the X-axis, as shown in Figure 7, yields a set of equations which can be acquired from the following transformation:

$$[\,x' \; y' \; z' \; 1\,] = [\,x \; y \; z \; 1\,] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & \sin\theta & 0 \\ 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

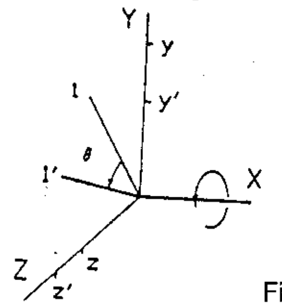



Figure 7

Finally, for positive rotation about the Y-axis, as shown in Figure 8:

$$[\,x' \; y' \; z' \; 1\,] = [\,x \; y \; z \; 1\,] \begin{bmatrix} \cos\theta & 0 & -\sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

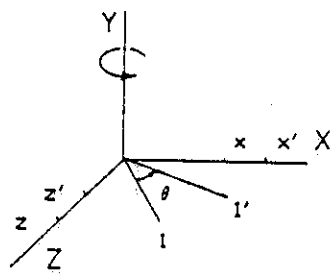



Figure 8

Within GRAPHPAK the two types of transformations we require are accomplished by using the TRANSLATE and the ROTATE functions, as follows:

NEWFIG ← XT YT ZT TRANSLATE OLDFIG

where the desired displacements in the X-, Y- and Z-directions are XT, YT and ZT, respectively;

NEWFIG ← ANGX ANGY ANGZ ROTATE OLDFIG

where the desired angles of rotation (in degrees) about the X, Y- and Z-axis are ANGX, ANGY and ANGZ, respectively.

## METHOD

The general approach to solving descriptive geometry problems using this method is as follows:

1. With the computer running GRAPHPAK, input the lines, planes and the axes to create the database.

2. Decide which of the two transformations, translation or rotation, you wish to perform to bring you one step closer to solution.

3. Perform the transformation, inspect the database to verify that the operation was done properly. If desired, display an isometric view to assist in visualizing the situation.

4. Repeat steps 2 and 3 until you have reached the solution. "Reaching a solution" means having the objects oriented in space so that the distance or angle of interest shows its true value in the X-Y plane (or one of the other principal planes of projections).

Two aids external to the computer are used: a sketch pad, and a calculator to compute desired angles of rotation.

The general strategy for solving a problem is to first translate the objects so that one end of a particular object line lies at the origin, then determine the sequence of transformations (usually just rotations about the axes) that will reach a solution. Choosing the particular object line to anchor at the origin and choosing the sequence of transformations are the key activities and it is here that good spatial visualization ability provides for ease in solution.

Following are examples showing the application of this method to three problems of increasing complexity: determining the true length of a line; determining the true shape of a plane (i.e. true lengths and angles); determining the angle at which a line pierces a plane.

### True Length of a Line

Given a line in space, first decide the sequence of steps which will result in the line being placed in the X-Y plane. (Using a pencil to represent the line and the corner of a book to represent the origin may be of assistance.)

Step 1: translate to place P1 at the origin (9a).

Step 2: compute angle alpha and rotate about the Y-axis by that amount (9b).

Step 3: line now lies in X-Y plane; compute true length from coordinates (9c).

Now follow through with the execution, beginning with the inputting of the axes and the line. (It is assumed you have already set the viewport and window bounds.)

Display the matrices to verify proper entry:

```
       LN
 0   40   30   20
 1   20   80   50

       AX
 0   99    0    0
 1    0    0    0
 0    0   99    0
 1    0    0    0
 0    0    0   99
 1    0    0    0
```

To reinforce your visualization you can create an isometric drawing by switching the display to graphics mode then inputting:

```
SKETCH ISOMETRIC AX
SKETCH ISOMETRIC LN
```

```
AX←6 4ρ 0 99 0 0 1 ... etc.
LN←2 4ρ 0 40 30 20 1 20 80 50
```

When inputting a database it helps guard against errors if you make the point number correspond to the row number of the matrix, placing the coordinates of P1 are placed in row one, P2 in row 2, etc.

Now re-display LN to determine the displacement values to be used in translation:

```
     LN
 0  40  30  20
 1  20  80  50
```

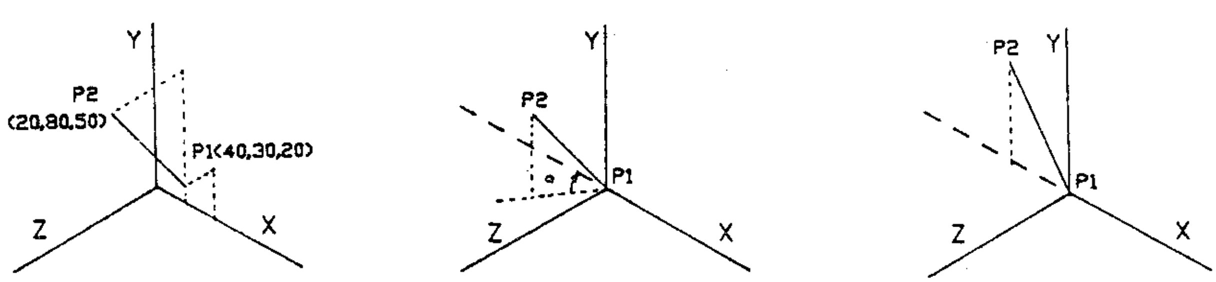



Figure 9(a)   Figure 9(b)   Figure 9(c)

Figure 9

Since it is easier to visualize objects above the X-Z plane, choose P1 as the point to translate to the origin. The X-Y-Z displacement values are obviously -40, -30, -20.

Perform the translation with:

LNT←¯40 ¯30 ¯20 TRANSLATE LN

and create an isometric drawing for visual confirmation:

SKETCH ISOMETRIC LNT

Switching the display to text mode, display the database as it currently stands:

LNT

| | | | |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 1 | ¯20 | 50 | 30 |

(Even at this early stage in using the method it becomes obvious that inspection of the data base provides the visual confirmation, and that displaying it is easier than creating isometric drawings on the graphics screen.)

A freehand sketch of the line in space is easily made by inspection of LNT, as shown in Figure 10, from which the desired rotation is determined.

The rotation about the Y-axis is accomplished with:

LNTR←¯56.310 0 ROTATE LNT

and inspection of the matrix LNTR shows that the line has been swung into the X-Y plane, the Z-coordinate of P2 having become zero.

LNTR

| | | | |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 1 | ¯36.056 | 50 | ¯0.00004 |

A freehand sketch of the line is made by inspection of LNTR as shown in Figure 11, from which the determination of the true length of the line is:

$$\text{True length} = (36.056^2 + 50^2)^{1/2} = 61.6$$

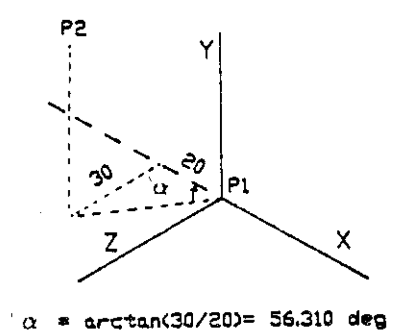



Figure 10

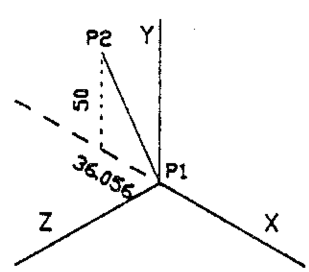



Figure 11

**True Shape of a Plane**

The general approach is illustrated in the diagrams of Figures 12(a) to 12(e).

First, translate so that P2 will lie at the origin (12a).

Then rotate about the Y-axis to place the line P1-P2 in the X-Y plane (12b).

Then rotate about the Z-axis so the line P1-P2 will be collinear with the X-axis (12c).

Finally, rotate about the X-axis in order to bring the vertex P3 into the X-Y plane (12d).

In its final position the Z-coordinates should all be zero. The length of any side and the value of any angle in the plane can be easily computed from the database (12e).

In applying the above procedure to the following example no mention will be made of the use of the SKETCH ISOMETRIC function, although the user can utilize it at any state. Confirmation will be sought in the database. From the displayed database freehand sketches are made as shown in Figures 13(a) to 13(f) to determine the next transformation.

First, the database for the plane is entered, and displayed.

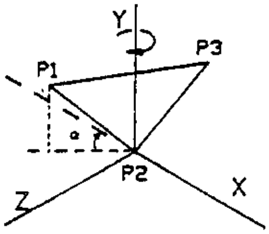



Figure 12(b)

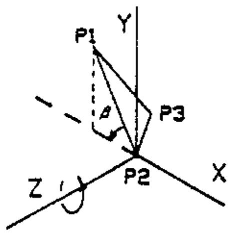



Figure 12(c)

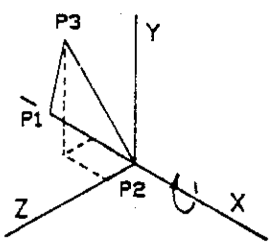



Figure 12(d)

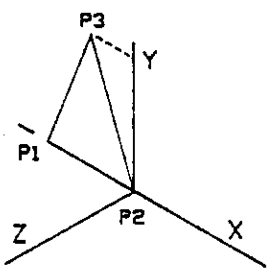



Figure 12(e)

Figure 12

PL

| | | | |
|---|---|---|---|
| 0 | 10 | 50 | 60 |
| 1 | 30 | 10 | 40 |
| 1 | 120 | 70 | 50 |
| 1 | 10 | 50 | 60 |

Deciding that P2 is to be translated to the origin (13a):

PLT ← ¯30 ¯10 ¯40 TRANSLATE PL

Then displaying PLT:

PLT

| | | | |
|---|---|---|---|
| 0 | 20 | 40 | 20 |
| 1 | 0 | 0 | 0 |
| 1 | 90 | 60 | 10 |
| 1 | ¯20 | 40 | 20 |

Rotate through the angle alpha with (13b):

PLTR ←0 ¯45 0 ROTATE PLT

and display with:

PLTR

| | | | |
|---|---|---|---|
| 0 | ¯28.284 | 40 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | 56.569 | 60 | 70.711 |
| 1 | ¯28.284 | 40 | 0 |

Rotate through the angle beta with (13c):

PLTRR←0 0 54.73 ROTATE PLTR

and display:

PLTRR

| | | | |
|---|---|---|---|
| 0 | -48.99 | 0 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | -16.330 | 80.829 | 70.711 |
| 1 | -48.990 | 0 | 0 |

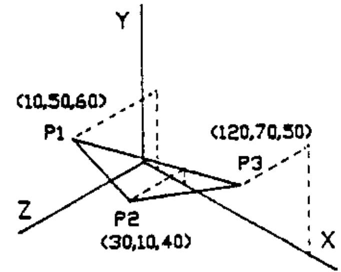
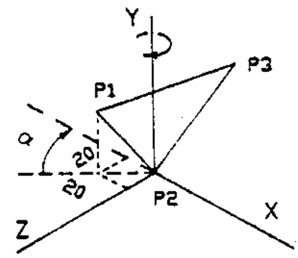
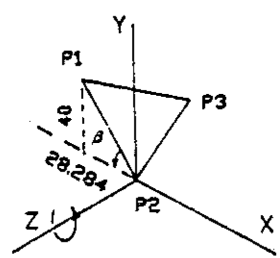



Figure 13(a)




α = arctan(20/20)= 45.000 deg

Figure 13(b)




β = arctan(40/28.284)
= 54.736 deg

Figure 13(c)

Figure 13

Rotate through angle gamma with (13d):

```
PLTRRR ← -41.180 0 0
ROTATE PLRRR
```

and display with (13e):

PLTRRR

| | | | |
|---|---|---|---|
| 0 | -48.990 | 0 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | -16.330 | 107.393 | 0 |
| 1 | -48.990 | 0 | 0 |

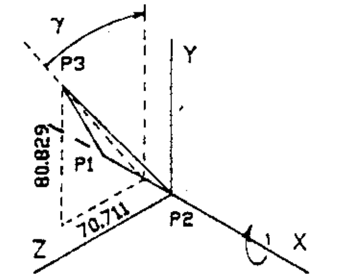
The coordinate matrix confirms that the plane is lying in the X-Y plane (13f). You can now easily compute true lengths, angles, etc.

### Line Intersecting a Plane

In this example only the approach will be described, with sketches. The working of a numerical example is left to the reader.

The given situation is a plane defined by P1, P2, P3, pierced by a line, P4-P5, as shown in Figure 14(a). We wish to orient the objects so the plane lies in the X-Z plane while the line lies in the X-Y plane; that orientation will allow easy determination of various items of interest, such as the location of the piercing point on the plane or the angle between the line and the plane.

The original position (14a).




γ = 90 - arctan(80.829/70.711)
= 41.180 deg

Figure 13(d)

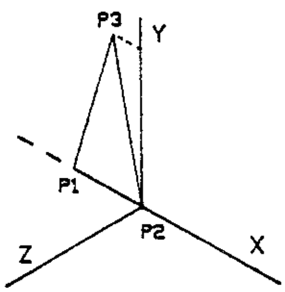



Figure 13(e)

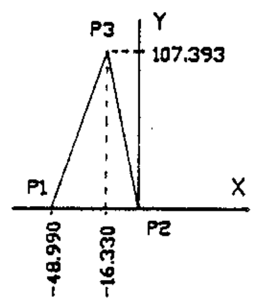



Figure 13(f)

Figure 13 (cont)

First, translate so one vertex, say P3, lies at the origin.

Then rotate about the Y-axis through an angle alpha to bring P1-P3 into the X-Y plane (13b).

Then rotate about the Z-axis through an angle beta to align P1-P3 with the X-axis (13c).

Then rotate about the X-axis through an angle gamma to bring P2 into the X-Z plane (13d).

The plane is now lying flat in the X-Z plane (13e).

Now translate the required amount in the X- and Z-directions so that P4 lies on the Y-axis (14f).

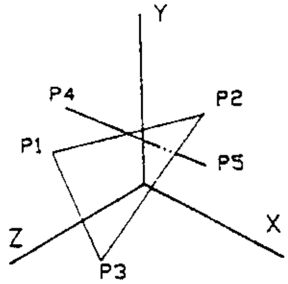


Figure 14(a)

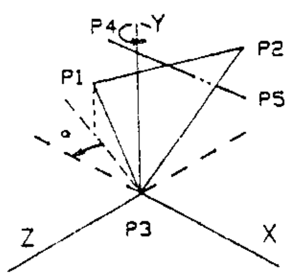


Figure 14(b)

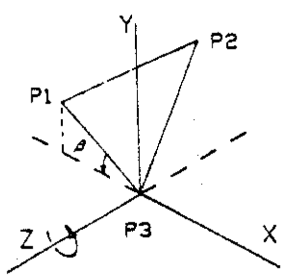


Figure 14(c)

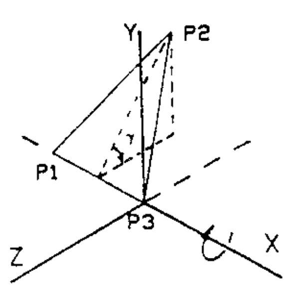


Figure 14(d)

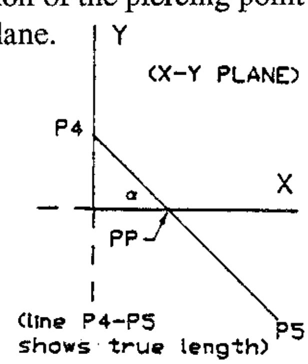
Figure 14

Then rotate about the Y-axis (the required angle can be determined from the coordinates of P5)

so that P5 lies in the X-Y plane (14g).

The figure is now in its final position, with the plane P1-P2-P3 lying flat in the X-Z plane while the line P4-P5 lies in the X-Y plane.

From the database of the plane and line in this final position and with the aid of a freehand sketch of the horizontal and frontal views as shown in Figure 15, quite simple computations will produce the angle of incidence, the length of line P4-P5 lying on either side of the plane, or the position of the piercing point on the plane.



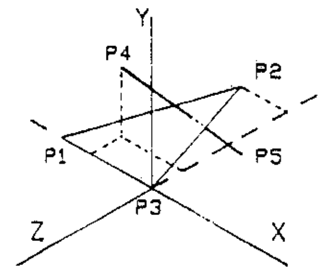



Figure 14(e)

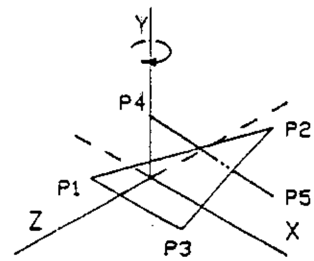


Figure 14(f)

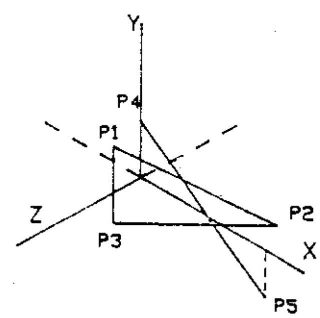


Figure 14(g)

Figure 14 (cont)

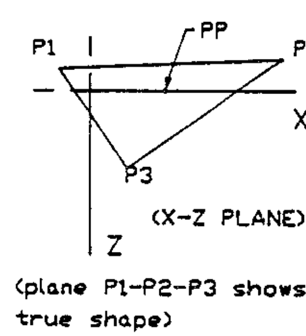



Figure 15

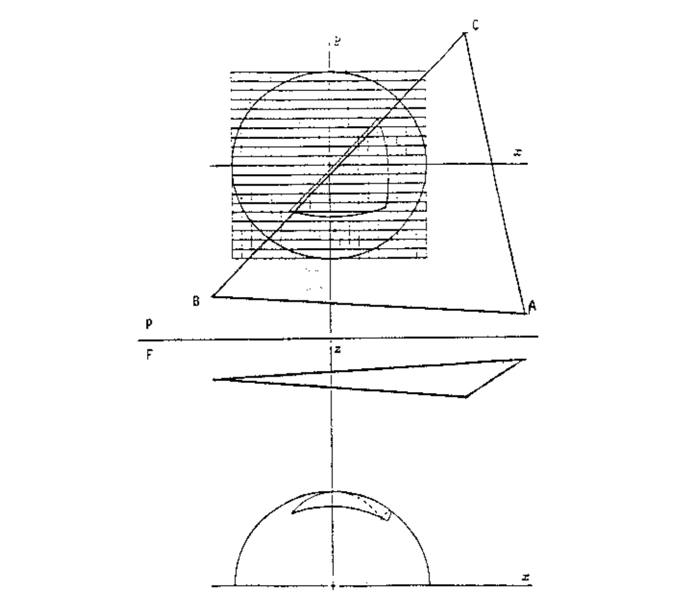



Figure 5

Figure 5 shows the solution and in the plan view, a grid has been imposed. The area $A_b$ can be determined either by counting squares or by some numerical technique. In this solution, we count squares to obtain an area for $A_b$ of 54.5 squares. The circle (in this same scale) has an area of $100\pi$. So the shape factor is:

$$F_{dA1-A2} = \frac{54.5}{100\pi} = 0.173$$

## MICROCOMPUTER SOLUTION

A computer program has been written to solve the problem. Again we begin with Figure 3 which illustrates the problem statement. The user must input the coordinates of the corners of the plane. The equation of the hemisphere is

$$x^2 + y^2 + z^2 = r^2$$

The user is permitted the option of selecting a radius $r$. The computer then finds the equation of the line extending from the origin ($dA_1$) to any point on the edge of the plane. The equation of the line and of the hemisphere are solved simultaneously to find the $x$, $y$, $z$ coordinates of the point of intersection. The computer then selects a new point on the edge (close to the preceding point) and repeats the procedure. The user has the option of specifying the number of points selected on each line segment (e.g., 100 points on $AB$).

Eventually the points on the edges of the plane $A_s$ in Figure 4 are known. Finally to obtain the projection of $A_s$ onto the $x - y$ plane (yielding plane $Ab_j$, the z coordinate of all intersecting points is set equal to zero as is Z in the hemisphere equation (Equation 5 above).

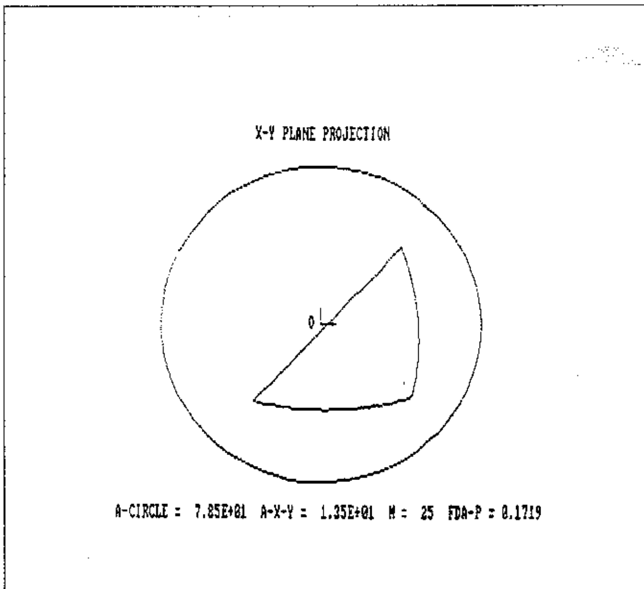



Figure 6

The Green's Theorem area line integral around the projected curve is then used to calculate the area of A$b$ and, of course, the area of the circle is $\pi R2\cdot$ The ratio of A$b$ to $\pi R2$ is the required shape factor. Figure 6 shows a computer drawing of the hemisphere and the plane both projected onto the x — y plane.

## RESULTS

For the problem of Figure 3 solved graphically, it has been determined that the shape factor is 0.173. When solved with the microcomputer, the problem of Figure 3 yielded a shape factor of 0.1719. The error is

$$\%\ error = \frac{0.173 - 0.1719}{0.173} \times 100\% = 0.63\%$$

## ACCURACY

The problem described above (which was solved graphically and with the microcomputer) was composed only for illustrative purposes. In order to get an idea of the accuracy of the microcomputer solution, however, it is prudent to compare results with an exact solution obtained by integration. Figure 7 shows one of the problems for which there is an exact solution: a differential area and a disk. The shape factor is [8]:

$$F_{dA1} - A_2 = \frac{H}{2} \left\{ \frac{H^2 + R^2 + 1}{[(H^2 = R^2 + 1^2)]^{1/2}} - 1 \right]$$

where $H = h/l$ and $R = r/l$. We now select specific values of $h$, $r$ and $l$ in order to perform a calculation; arbitrarily,

$h = 4$ units
$l = 6$ units
$r = 2$ units

Equation 6 gives $F_{dA1-A2} = 0.0356$. The microcomputer was set up to solve this problem in two ways: where the radius of the hemisphere is large enough to encompass the disk ($r = 50$ units), and where the hemisphere excludes the disk ($r = 2$ units). The results obtained in both formulations are identical. The shape factor (both methods) is $F_{dA1-A2} = 0.0331$. The

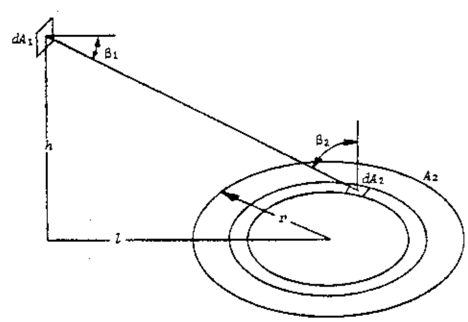



Figure 7

microcomputer solution is illustrated in Figure 8.

The error is

$$\% \ error = \frac{0.0356 - 0.0331}{0.0356} x100\% = 7.0\%$$

## CONCLUSIONS

The graphical method is a time-consuming technique for finding shape factors. The results obtained for a specific problem solved graphically were within 1% of that obtained with the microcomputer. It is believed that the microcomputer solution is closer to being exact (than the graphical method) because numerical methods were used by the computer to find areas, whereas counting squares was used in the graphical method. Results of both methods gave surprisingly good agreement.

Comparing the microcomputer-obtained solution to one obtained by direct integration shows that the microcomputer solution gives fairly precise results that are useful for engineering work.

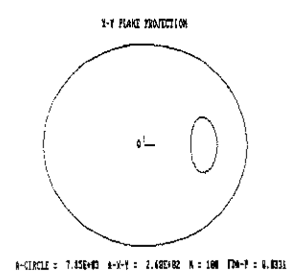


Figure 8

## BIBLIOGRAPHY


1. J.H. Earle, Drafting Technology, Addison-Wesley Pub. Co., 1982.
2. F. Woodworth, Graphical Simulation, International Text book Co., 1967.
3. R.T. Hinkle, Kinematics of Machines, 2ed, Prentice-Hall, Inc., 1960.
4. F. Kreith, Principles of Heat Transfer, 2ed, International Textbook Co., 1969.
5. W.S. Janna, Engineering Heat Transfer, PWS Engi neering, 1986.
6. T.G. Keith and W.S. Janna, "Graphical Solution of Unsteady Viscous Flow Problems", Int J Mech Engrg Edu 5, 2, 1977, ppg. 97-105.
7. W. Nusselt, "Graphische Bestimmung des Winkelverhaltnisses bei der Wärmestrahlung", VDI Z 72, 72, 1928, pg 673.
8. R. Siegel and J.R. Howell, Thermal Radiation Heat Trans fer, McGraw-Hill Book Co., 1972.


## ACKNOWLEDGEMENTS

The authors wish to express their gratitude to Mr. Edwin Edwards for producing the ink drawings of this paper. For further information about the program, interested readers are advised to contact Mr. David Alciatore at the University of Texas at Austin, Department of Mechanical Engineering.

# An Integrated Course in Computer Graphics: Merging Old and New

Manjula B. Waldron
John T. Demel
The Ohio State University

## INTRODUCTION

Engineering graphics includes learning of graphics to visualize, understand, represent and/or solve engineering problems and designs. As such, it formed an important part of the engineering curriculum [1]. However, the addition of computers and science courses to the engineering curricula resulted in the elimination or reduction of graphics courses [2]. The early nineteen-eighties saw a rebirth of graphics in the form of computer graphics. The ability to simulate engineering systems and to represent them graphically appealed to the designers. There was a resultant increase in CAD/CAM software which has provided more graphic media for representation [3] [4].

There are many approaches for incorporating computer graphics which have been proposed by different educators. Some educators believe that knowing how to write computer programs is not essential and, after an introduction to traditional graphics, the students can learn computer assisted drafting without worrying about programming [5-7]. The difference between this approach and ours is very fundamental. We believe that the problem-solving skills learned in both programming and graphics are very important for engineers. This is especially true of top down and modular approaches to problem solving that we have adopted. These skills, once acquired, assist students in looking at any problem, analyzing it in a set of logical steps or subproblems, and then solving each subproblem until the entire problem is solved. Hence, we felt that we could make the instruction of programming and graphics much more efficient by integrating these into the first course of a three-course sequence that contains graphics, computer graphics, and interactive computer programming. Interactive programming through graphics, apart from being instructional, is also fun for students, because they quickly see the accuracy of their results.

> ***"We believe that the problem solving skills learned in both programming and graphics are important for engineers"***

## COURSE DESIGN AND DEVELOPMENT

Engineering course design and development is a dynamic process and is dependent on industry needs, the student and faculty feedback, and the hardware/software availability. Our first course as presently taught has undergone many changes, but the underlying philosophy remains the same.

The original concept was to integrate the three courses with programming, problem solving, graphics, and computer graphics interlaced [8]. This innovative and exciting design was in some ways dictated by the interactive and time-sharing hardware availability of the late seventies and early eighties. But, after two years of evaluation, it was apparent that while the students benefited from the approach as measured by the knowledge they gained, the work load on the students, the faculty, and the system was much too large. Unfortunately, it also gave the courses an image of being suitable only for the brightest and harder working students. To some extent, the problem was the pilot test which was done with honors students.

When the course sequence was opened to the general student body, the assignments had to be changed in scope so as to reduce the student work load. The student group projects were dropped and the course content in the three-course sequence was separated so that a smaller number of topics was covered in each course. This change helped transfer students and reduced administrative difficulties [9].

In 1982-83 the personal computers became powerful enough to support FORTRAN and good graphics. These standalone computers could provide students with experience in dealing with the entire computer system rather than a terminal on a multiuser system. Evaluation of the available software and hardware followed by a pilot study showed that an IBM PC with Waterloo WATCOM interpreted FORTRAN software gave us a system suitable for the first course in the sequence. The particular strengths of the IBM PC/WATCOM configuration are: ease of use, full screen editing, graphics, FORTRAN 77 language (also, PASCAL), and potential networking capabilities. The latest version of the WATCOM editor is easy to learn and has sufficient editing power including cut and paste and file transfer. Experimentation showed that the interpreter executed the programs slowly and that the WATCOM serial network was very slow for frequent file transfer [10].

The design of the first course reflects the philosophy that the programming principles can be taught through interactive computer graphics. This can be done by teaching the students some basic graphics principles through the sketching of orthographic views and pictorial drawings. Then they can use a particular computer language (FORTRAN 77) and system (IBM PC) to create the same pictures on the terminal.

The second course covers drafting techniques and descriptive geometry. We are currently evaluating computer-aided drawing packages for the IBM PC that can be used effectively to replace some pencil and paper exercises.

The third course in the sequence introduces the students to a multiuser computer system similar to those used in the other engineering departments. This course covers more advanced interactive computer graphics and numerical programming algorithms. The computer system is a trio of networked VAXes that share user disks and run VMS. They support medium resolution (640 x 480) raster graphics terminals. FORTRAN 77 and the Precision Visuals DI 3000 graphics subroutine library are used by the students for all programming exercises. DI 3000 was chosen by the College of Engineering (and subsequently by the campus computer center) to be the graphics package available on most systems and replace PLOT10. In the first course, graphics primitives allow the programming emphasis to be modular and topdown. For example, the student uses a program that displays several vectors on the CRT screen and then studies the code to see how a subroutine provides simpler coding to achieve the same effect. In this case one programming statement CALL VECTOR (X1, Y1, X2, Y2) replaces CALL JMOVE (X1, Y1) AND CALL JDRAW (X2 Y2). The student then is given a drawing composed of a series of 2-D graphic primitives and is required to first write and then check subroutines individually to create the primitives.

These subroutines provide a variety of learning experiences. They require the student to learn about intrinsic functions (trigonometric, math, etc.) and various features of the programming language (loops, conditional statements, etc.), and computer graphics concepts (move, draw, translation, scaling, rotation). Then he/she is required to create a data file of 2-D information for creating the pictures. Once the data file is created, the student writes a subroutine to read from the file, choose the correct subroutine for the set of data elements, and call the subroutine to do the drawing. The subroutines are individually checked by test programs and then combined to make one program which creates the given picture.

The fourth step in this progression of computer graphics programs is to create a pictorial of a 3-D object through creating a data base, and writing subroutines to read from the data base (file). The resultant display is 2-D orthographic projections which the student had sketched by pencil earlier in the course. Through this exercise, the student learns the rotation of an object to display different views, and in the process learns the use of one dimensional array manipulation in the programming language. In our approach the computer graphics concepts of translation, scaling and rotation are taught with algebra and trigonometry rather than matrices.

In teaching, the instructors are made aware of good programming proverbs to be emphasized with the students during the delivery of the course material. The examinations reflect the programming (computer graphics) and the traditional graphics (paper and pencil) aspects of the course with quizzes or hour examinations once a week and a required final examination. The final examination originally was made optional to encourage the students to develop good daily work habits rather than face the end of the quarter rush. This practice was dropped because it appeared that the comprehensive final examination provide an opportunity for the student to consolidate the course material.

## HARDWARE

In 1986-87, the air conditioned laboratory houses 80 IBM PCs with 640 KB memory, RGB monitors, and two disk drives. Nine dot matrix printers are shared by the students. Priority for the printers is based on the students need for a hard copy output. Each computer has PC-DOS 2.1 and the WATCOM FORTRAN 77 interpreter. The faculty members teaching the course also have this basic configuration in their offices so that they can develop course materials and answer student problems. The laboratory is staffed by a full-time manager, assistant manager, and part-time student consultants and site monitors.

The faculty member and at least one student teaching assistant are present during the scheduled laboratory sessions to grade and monitor student progress. At other times the laboratory is open and students are encouraged to work on their own and do extra exercises from text books and practice with supplemental tutorial material. The philosophy behind this approach is to encourage the good students to learn as much as possible rather than restrict them to specified times. It also provides all students with a chance to work on the system when expert help is available.

## COURSE DELIVERY

The first hour (as stated above) is devoted to introduction to new material and has a lecture/discussion format. The second hour is spent in the laboratory going over materials covered in the lecture. The students do preassigned work under the instructor supervision. Each aspect of the course can then be individually covered and the instructor can assign extra tutorial work to students who need it. This allows for each concept in the course to be emphasized with every student. The top down and the modular approach in the students' thinking can be reinforced before many bad problem-solving and programming habits are ingrained.

Most textbooks in FORTRAN do not seem to emphasize modularity and interactive programming. Hence, a supplemental manual detailing the hardware/software use and introduction to subroutines was developed by our staff for student use. The manual also includes the tips for good programming practices necessary for top down and modular programming, such as the use of mnemonic names, good documentation, headers, problem-solving techniques, hand checking the program, having a good logic diagram so that the program can be understood by those who read it, etc. Based on student feedback as well as software updates, the manual has been modified to make it more effective.

The course grade is based on the student obtaining 60% grade or better in both the programming and the drawing part of the course which is weighted roughly 2:1 in favor of computer graphics work. This distribution reflects the time spent by the student in integrating programming and graphics rather than learning graphics. Typical computer problems that the students encounter in the laboratory are; Familiarity with the WATCOM editor and IBM PC; Development of subroutines and driver programs to test subroutines; Manipulation of data from unformatted input data files; Manipulation of 3D data to draw and rotate objects through the use of arrays.

Figure 1 shows a typical example of the documentation students learn to use in their programs. Figure 2 shows how they combine some of these primitives through data specification to create a composite picture. Figure 3 shows the 3-D wireframe object representation done through 3-D data manipulation and rotation. Rotations are done through simple trigonometric functions as presented in Demel & Miller [11] rather than advanced matrix concepts as presented in many texts. After completing this exercise the students have a better understanding of orthographic projection and isometric views. They will usually create different objects on their own and experience the power of modular programming and creation of a user-oriented software.

## SUPPLEMENTAL TUTORIALS AND COMPUTER AIDED INSTRUCTION

Several tutorial software packages have been developed to assist those students who have trouble in visualizing objects in space from either the orthographic or the isometric drawings. The tutorial allows students to look at surfaces of selected objects in the isometric and the three orthographic projection views [12], [13]. Tutorials which will present the student with an isometric drawing and allow the students to create three orthographic views and vice versa with immediate feedback on errors is available on the IBM PC [14], [15].

These tutorial systems are user-friendly and students can start the program and look at different objects, surfaces, lines or create drawings on their own without assistance. Usually the instructor will demonstrate the tutorial use to the needy student and let the student come and practice on their own time. The use of the tutorial is voluntary on the part of both the student and the instructor.

## EVALUATION

This course format on the IBM PC has been followed for 10 quarters now. The original course design required formal exercises on problem-solving skills. It was found that both the formality of presentation for the students and its grading by the instructors proved to be very time consuming. Hence the problem-solving part which is still taught and stressed in their course has become part of the presentation of graphics and programming rather than being a separate topic. What impact this will have on student learning has yet to be assessed. But the general feeling is that the benefit in eliminating the frustration from overloading will far outweigh the possible decrease in student retention of problem- solving skills. In the last student survey more than

```
*
*****************************************************************************
      subroutine RORECT(X1,Y1,W,H.BETA)
*****************************************************************************
*                                                                           *
*     PURPOSE: TO PLOT A RECTANGLE WHEN GIVEN ONE CORNER, THE HEIGHT,       *
*              THE WIDTH, AND THE ANGLE OF ROTATION                         *
*                                                                           *
*     I/O      : THE INPUT FOR THE SUBROUTINE IS FROM ARGUMENTS             *
*                                                                           *
*     CALLED BY: MAIN PROGRAM AND SUBROUTINES                               *
*                                                                           *
*     CALLS  TO: IGPP-PC SUBROUTINE IN THE IGPP-PC.FOR PACKAGE              *
*                                                                           *
*     VARIABLES:  X1, Y1     - COORDINATES OF FIRST CORNER                  *
*                 X2, Y2     - COORDINATES OF SECOND CORNER                 *
*                 X3, Y3     - COORDINATES OF THIRD CORNER                  *
*                 X4, Y4     - CORRDINATES OF FOURTH CORNER                 *
*                 H          - RECTANGLE HEIGHT, SCREEN UNITS               *
*                 W          - RECTANGLE WIDTH, -SCREEN UNITS               *
*                 BETA       - ANGLE OF ROTATION IN DEGREES                 *
*                                                                           *
*****************************************************************************
*
*          CONVERT DEGREES TO RADIANS
*
      ANGLE = BETA*6.2832/360.
*
*          CALCULATE CORNERS
*
      CA = COS(ANGLE)
      SA = SIN(ANGLE)
       X2 =  X1 + W * CA
       Y2 =  Y1 + W * SA
       X3 =  X2 - H * SA
       Y3 =  Y2 + H * CA
       X4 =  X1 - H * SA
       Y4 =  Y1 + H * CA
```

Figure 1

70% were satisfied with the time required and the course presentation. Considering the university has an open admissions policy and all entering freshmen in engineering take this course, we feel that this is a positive indicator.

At present we are experimenting with different networking systems, but the slow access times and relative unreliability of the original networking software/hardware system dictated stand alone PCs with software distribution on floppy disk. This aspect is still under evaluation.

The strength of the course we believe lies in the fact that the students integrate problem solving into graphics which will assist them in approaching CAD/CAM workstations in a more visually literate manner. Students enjoy producing graphics and the interactive nature of the environment. The weakness of this course lies in the fact that there are no current textbooks promoting the philosophy presented here. Hence, both students and faculty must rely on the faculty member's experience and locally prepared visual aids and handouts to assist in instruction.

## FUTURE DIRECTIONS

It is difficult to hold a crystal ball in these fast changing technological times and see which choices to make. Very recently we have seen the price of computer memory falling rapidly, paving way for an easy merger of video and digital architectures. On the software side the impact has been in the Artificial Intelligence (AI) type of languages and expert systems. For service courses such as freshman graphics and/or computer graphics, the impacts are often dictated by the college of engineering as a whole and hence are even harder to predict. This is why it is important to have a good underlying philosophy. This allows the details to change as dictated by hardware and software but the philosophy remains the same. Our philosophy is that the engineer should know how the system works and be able to modify it through software changes and be able to create, to understand, and to use graphics for effective communication.

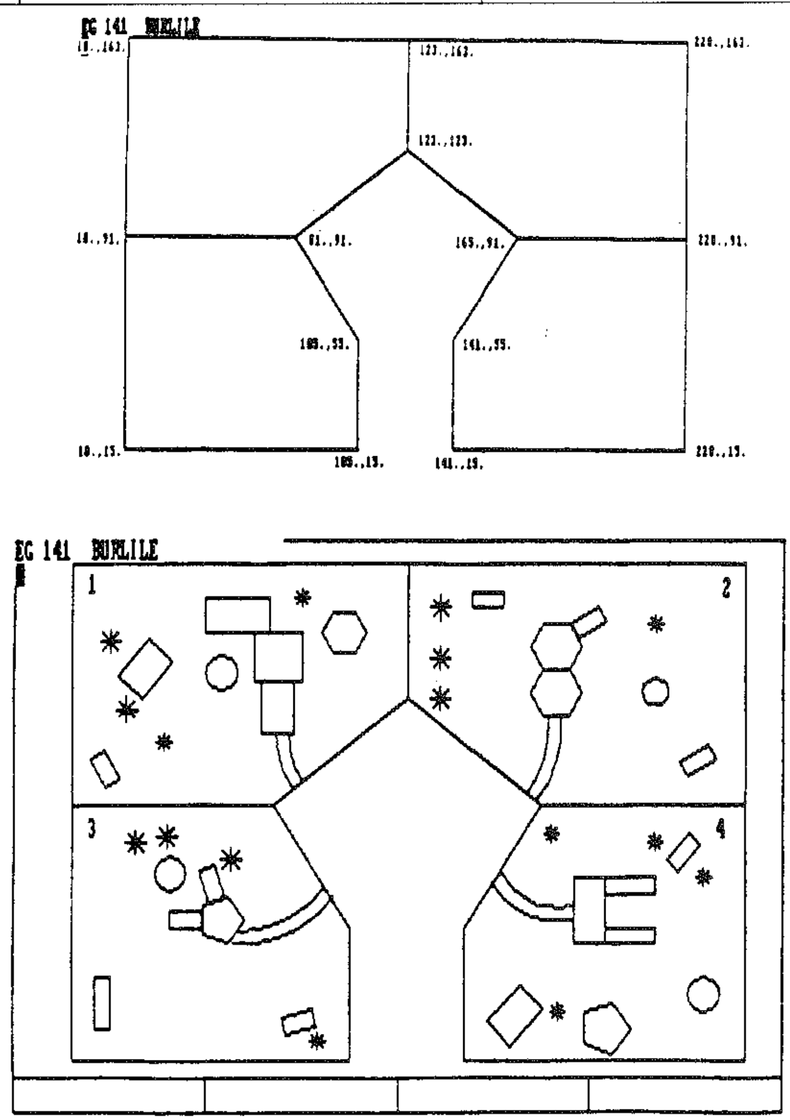


Figure 2

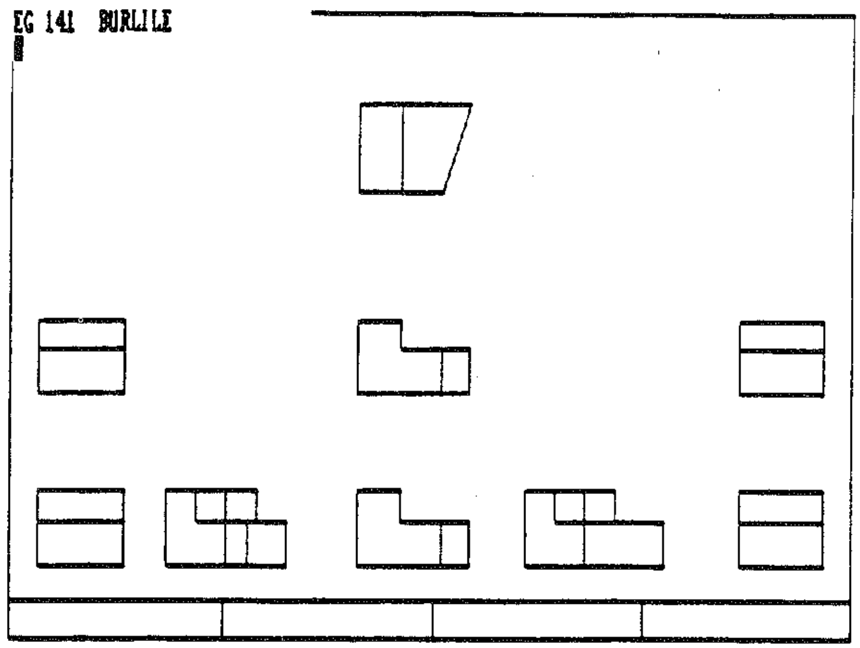



Figure 3

In the computer graphics area, one thing is certain, more engineers will need to be visually literate. Engineers will need visual skills to assimilate and utilize the vast number of interactively controlled visual devices which are becoming increasingly available. In the future, this literacy will be provided through computer graphics rather in addition to paper and pencil, because computer graphics can provide for a more dynamic medium of instruction.

The foregoing leads us to state that there will be a need for interactive, intelligent tutorials on graphics and computer graphics related topics to assist students in developing visual literacy, especially in the area of rotation and object manipulation. Since engineers by profession will continue to be problem solvers, they will need to continue to learn good programming and problem solving skills with modular and top down approaches. CAD/CAM software and workstations will be the tools engineers use in industry; hence, an intelligent introduction and use of these in the engineering curriculum will be necessary. Some of these introductions of necessity will have to be made at the freshman level.

In summary, in the next few years the graphics content will not change while the delivery systems may change. Descriptive geometry will take on additional forms, including solid modeling and a rebirth of rotation methods done by the computer, thus saving time while increasing visualization. The language of instruction in most engineering schools may change from FORTRAN to some other language, but engineering problems will have to be solved using graphics and the computer through good problem-solving techniques. We feel that our teaching approach for freshman graphics/computer graphics has the most potential for future curriculum adaptation.

## REFERENCES


1. Earl, J.H., Engineering Design Graphics, Reading, MA; Addison Wesley, 1983.

2. Nee, J.G., "The PSI Approach to Teaching Engineers Graphics," Engineering Design Graphics Journal, Vol. 43, 3, Fall 1979, pp. 55-58.

3. Waldron, M.B., & Hang, R.I., "Computer Graphics and Teaching Descriptive Geometry," Proceeding of World Congress on Computers in Education/ 85, Norfolk, VA, August 1, 1985, pp 657-661.

4. Demel, J.T., Wilke, R.A., Coppinger, J.T. & Barr, R.E., "Computer Graphics," College Station, TX, Creative Publishing, 1979.

5. Pearson, W.B., "Practice Classroom Computer Graphics," Engineering Design Graphics Journal, Vol. 44, 3, Fall 1980, pp. 60-71.

6. Mosillo, F.A., "Computer Graphics - Perplexing and Troublesome Questions," Engineering Design Graphics Journal, Vol. 45, 1, Winter 1981, pp. 3.

7. Brillhart, L.V. & Bell, E.A., "Freshman Graphics at Triton College," Engineering Design Graphics Journal, Vol. 47, 3, Autumn 1983, pp. 14-21.

8. Miller, M.J., Wolf, J.J. & Demel, J.T., "The OSU Freshman Graphics and Programming Sequence," Engineering Design Graphics Journal, Vol. 46, 3, Autumn 1982, pp. 16-21.

9. Demel, J.T., Waldron, M.B., Wolf, J.J., "Materials for Integrated Freshman Engineering Course," final report Learning Resource Center teaching grant, The Ohio State University, August, 1982.

10. Waldron, M.B., Demel, J.T., and Barber F.E., "Teaching Interactive Computer Graphics on the Microcomputer," 1984 Frontiers in Education Conference Proceedings, pp. 614-617. .

11. Demel, J.T., & Miller, M.J., "Introduction to Computer Graphics," Monterey, CA, Brook/Cole Engineering Div., 1984.

12. Waldron, M.B., "A Computer Graphics Tutorial to Teach Orthographic Projection," Engineering Education, January, 1985, pp. 237-239.

13. Waldron, M.B., Lalitha, P.S., "Multisensory Learning and Orthographic Projection: A Comparative Study," Engineering Design Graphics Journal, Vol. 49, 1 Winter 1985, pp. 30-34.

14. Waldron, M.B., Hopkins, M.R., "Orthographic Projections and Isometric Drawing Tutorials Using Computer Graphics," 1985 ADCIS (Association for the Development of Computer-based Instructional System) Proceedings, March, 1985, pp. 46-49.

15. Waldron, M.B., Dybiec, S., "Semi-intelligent Tutorials to Teach Graphics Concepts," to appear in Proceedings of Frontiers in Education.

16. Waldron, M.B., Hang, R.I., & Smith, L.F., "Need for Descriptive Geometry: A Foundation for CAD/ CAM," to appear in Engineering Education.


## ACKNOWLEDGMENTS

This research was partially supported by a grant from International Business Machine (IBM) and the Learning Resources Center at The Ohio State University. Authors are grateful to Franco Barber and Greg Burlile for evaluating the software, typing the manuals and some of the class exercises used in the courses.

8. All articles shall be written using Metric-SI units. Common measurements will be accepted at the descretion of the editorial staff.

9. Send all material in one mailing to:

Jon M. Duff
Editor, EDGJ
355 Knoy Hall of Technology
Purdue University
West Lafayette, IN 47907

## REVIEW OF ARTICLES

All technical articles are submitted to the EDGJ Board of Review for blind review. This board represents national authority on graphic technology, engineering education, and technical specialties within engineering and technology. Authors will be notified as to the results of the review. Manuscripts may be accepted as submitted, accepted with provisions, or not accepted for publication.

The Journal reserves the right to refuse letters to the editor due to space constraints, because the material is potentially libelous, because the issues discussed are no longer of a timely nature, or because of a large number of similar letters on the same topic. Letters may be edited for spelling, punctuation, and style, and may be shortened or abstracted although meaning and tone will not be changed.



A repeating decimal is a number that stutters and mumbles numerical rage. It trips a digit on a decimal point and stumbles across the page.

Aiber

Circulation Manager
The Engineering Design Graphics Journal
The Ohio State University
2070 Neil Avenue  Columbus, OH  43210

NON PROFIT
ORGANIZATION
U.S. POSTAGE PAID
BULK RATE PERMIT
NO.477
DUBUQUE, IOWA