THE ENGINEERING DESIGN GRAPHICS

Journal

Winter 1986 Volume 50 Number 1

ENGINEERING DESIGN GRAPHICS DIVISION

AMERICAN SOCIETY FOR ENGINEERING EDUCATION

ISSN 0046-2012

PUBLICATION BOARD

CALENDAR OF EVENTS

ASEE ANNUAL CONFERENCES
- 1986 CINCINNATI hosted by The University of Cincinnati
- 1987 RENO hosted by The University of Nevada/Reno
- 1988 PORTLAND hosted by Portland State University

EDGD MIDYEAR MEETINGS
- 1986 AUSTIN hosted by The University of Texas at Austin
- 1987 LOUISVILLE hosted buy The University of Louisville
- 1988 EVANSVILLE hosted by The University of Southern Indiana
- 1989 TUSCALOOSA hosted by The Universityof 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.

SUBSCRIPTION RATES

YEARLYRATES

US, Canada, Mexico	
ASEE/EDGD Member	$1.50
ASEE Member	$3.00
Non-Member	$5.00
Foreign Subscribers	$15.00

SINGLE COPY RATES

US, Canada, Mexico	
ASEE/EDGD Member	$1.00
ASEE MEMBER	$1.50
Non-Member	$2.00
Foreign Subscribers	$5.00

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-September15; WINTER-December1; 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. Two 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 is 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 collegues before submission.
7. Enclose all material, unfolded, in a large envelope. Use heavy cardboard to prevent bending.

Continued inside back cover.

VOLUME 50 NUMBER 1

THE ENGINEERING DESIGN GRAPHICS Journal

ENGINEERING DESIGN GRAPHICS DIVISION
AMERICAN SOCIETY FOR ENGINEERING EDUCATION

Departments

Features

FROM THE DESK OF THE EDITOR

A MILESTONE ISSUE

This issue of the **EDGJ** marks an important milestone for our Division and our profession. Fifty years of bringing news of research, teaching and service in graphics is something we can all be proud of. It has taken much sacrifice by many of the figures that have become legends in our field. Almost every author, every teacher of national recognition, every administrator of a graphics program has found the time to serve his/her Society, Division, and profession. Each of you have my respect and gratitude for making our current jobs that much easier....and satisfying.

This issue is also a milestone in that for the first time The Journal has moved into the electronic age through the addition of interactive page composition software. There will be a few bugs to work out in the next few issues, but this change in way that the **EDGJ** is put together should make the job of the next Editor that much easier. If you have an interest in the publication of the **EDGD** I invite you to consider making such interest known.

Volume 50, Number 3 also begins a new era. Commencing with that issue, a strict and professional review procedure will govern the publication of technical papers. Glance at page 3 and you will see the first of our reviewers as well as our Technical Editor. A paper will be reviewed by three of the panel, subject to final review by the Technical Editor. This process should assure the kind of publication that best serves our Division and its members.

LETTERS

The desk has received several letters in the past months. Here are a few that might be of interest.

Irwin Wladaver:

Although I am pleased to read that my favorite adversary, C.Ernesto S. Lindgren, is peddling his scholarly (to me nonsensical) papers on 4-D and now plus all-D descriptive geometry, I was dismayed at the horrible way his name was spelled in the Autumn '85 JOURNAL.

Apology sent, thanks!...ed

And later from Vlad:

The first Journal came out first in 1936. Forty years later compiled an index- I guess you have seen it. Surely some ambitious drudge might volunteer to index 50 years of Journals. And to think I was editor thirty years ago: 1955-1958. I'm glad you thought of the significant anniversary.

any takers out there?......ed

From Larry Goss, newly appointed EDGJ Division Editor:

Now I describe the geometry on the computer, revolve it to the view I desire to have, and let the translation and rotation formulas do their wonder. Not only are my ellipse templates gathering dust, but so are my spring templates, schematic templates, lettering guides, kidney bean intersection templates, complete set of highway curves and ship's curves--in short, thousands of dollars worth of drafting equipment that I have accumulated over the past 30 years have suddenly become obsolete collector's items--just like my slide rule--useful only if the power goes off and my batteries go dead. There have been a few landmark situations which have developed within the Division which have guided its direction. One was the teaching of design. Another is happening as a direct result of readily available micro-computer based CAD systems. It is a warning to the more conservative among our membership that times are changing again. If we are not made aware of that and do nothing to keep our course offerings current with what the market wants from our students, then we may find ourselves out in the cold again.

see letters on page 12

chairman

Robert J. Foster
Chairman

A MESSAGE FROM THE CHAIRMAN

The membership welcomes the officers newly elected for three positions! **Ronald Barr**, University of Texas at Austin will serve as Vice Chairman, to become Chairman in 1987. **Josann Duane** of The Ohio State University begins a three-year position as Director of Programs. **James Leach** at Auburn Unversity will be director of Liaison, also a three year position. These persons have a background of involvement in the Division and coupled with their committment and ability, indicates successful tenures of service just ahead. Give them and the other officers your support by becoming involved yourself with the Division. The presentation of papers at conferences, submission of material for the **EDGJ**, and service on committees are all good and worthy ways to help the Division and your own professional growth.

My message in the previous issue indicated that engineering graphics is still a unique and important aspect of of both engineering education and industrial practice. It was suggested that the <u>ways</u> of expressing graphics evolve with time. For us in the teaching end of graphics, the question arrises as to the how best <u>teach</u> engineering graphics. I'd like to dwell a few moments on this question, and also on possible <u>trends</u> in our discipline.

We have seen at least four ways of teaching the subject emerge with varying emphases from school to school. The methods are, (1) the traditional manual use of text and drafting equipment equipment, (2) the use of television, (3) use of design problems, and (4) the use of computers, especially microcomputers. Methods (2) and (3) receive perhaps less emphasis. Some see a possible conflict between methods (1) and (4), both popular. Are the traditional and computer techniques really separate and in conflict, or do they supplement each other?

Persons both in education and industry seem to agree that it is critical for a student to learn to visualize object in three dimensions and then to be able to represent these objects in a two-dimensional format. The format may be that of paper, film, or a monitor used in CAD. Does a student lern to conceptualize better using traditional manual methods or by using computers? **Retha Groom** provided some insights into this in the Autumn 1983 issue of the **EDGJ**. It appears that learning is maximized when concepts are learned manually, if followed by using computer software covering the same concepts.

We in education need not posess the ultimate in CAD equipment. Industry would like the student to learn the basic concepts of computer expressed graphics. Students need not be experts on specific high-powered systems. Industry will gladly train the engineer on corporate-specific systems.

Continued on page 11

division

NEWS OF THE DESIGN GRAPHICS DIVISION

ASEE ANNUAL
Conference

June 22-26, 1986 marks the dates of the annual ASEE conference to be held in Cincinnati, Ohio. Hosted by the University of Cincinnati, the conference should prove to be of interest to all the EDGD membership and its central location means no doubt easy access to most of our readership. The University of Cincinnati is in a unique position in engineering and engineering technology education with their Institute of Advanced Manufacturing Systems. Between Cincinnati Milicron, General Electric , and Structural DynamicsResearch Corporation, the area is ripe with the latest applications of CADD, CAD/CAM, CAE, CIM, and most other cutting edge practices. See you there!

EDGD MIDYEAR
Meeting

Head to Austin, Texas next January for the 1987 ASEE/EDGD Midyear Meeting hosted by the University of Texas at Austin. If you like chili and Bar-B-Q bring your bibs and Ron Barr has agreed to personally lead a "chili breath-87" marathon through the haunts of the Lone Star State's star city.

Interested in presenting a paper? Contact:

Ronald C. Pare'
Mechanical Engineering Technology
University of Houston
4800 Calhoun
Houston, TX 77004
(713)749-4652

Send 500 word abstract by July 15, 1986

Topics:
engineering and computer graphics, CAD,CADD, micros, 3-D graphics, solid modeling, geometry, degree programs, graphics programming.

West Lafayette Midyear-1986
A Report

Pete Miller, facilities chairman for the 1986 Midyear meeting hosted by the Technical Graphics Department at Purdue University, forwarded a check to EDGD Treasurer Barry Crittenden, representing the net profit from the meeting. Contributing to the financial success was a list of 18 exhibitors whose attendence added greatly to both profitibility and interest.

Oppenheimer Award

The award for the outstanding paper presented at the 1986 Midyear meeting was shared by Larry Genalo of Iowa State University and John F. Freeman, Jr. of North Carolina State University. These were two of the finest papers presented in recent years, indeed the quality of all presentations spoke well of the teaching and technical expertise.

continued on page 47

1984 北京

Presented at the International Conference on Engineering and Computer Graphics

USING THE "ADDING" AND "SUBTRACTING" CONCEPT IN COMPUTER GRAPHICS

BY

Zeng Damin
South China Institute of Technology
People's Republic of China

Steve M. Slaby
Civil Engineering Department
Princeton University
Princeton, New Jersey
08540

INTRODUCTION

Most modern interactive computer graphics systems can only be used to draw complicated objects line by line -- straight lines or curves, and only one projection at a time -- orthogonal views, axonometric projection and perspective projection. So, a long process is required to draw a complicated object. However a complicated object can be decomposed into a group of primitive geometrical shapes, such as prisms, pyramids, cylinders, cones, spheres and torus', etc. Conversely, a complicated object can be "composed" by ADDING or SUBTRACTING primitive geometrical shapes one by one. Each kind of primitive geometrical shape can be described simply by several regular dimensions. For example, a right rectangular prism can be described by its width, depth and height. A right circular cylinder can be described by its height and the radius of its base, etc. Furthermore, these dimensions of different primitive geometrical shapes can be simply described by no more than four specific points (see Fig.1). These points, not only can determine the dimensions, but also can determine the locations of the primitive geometrical shapes.

For instance, the triangular prism (Fig. 1b), the distance between A and B determines the radius of the circle which circumscribes the triangle, the distance between C and D determine the height of the prism, the X and Z coordinates of A and the Y coordinate of C determine the location of the prism, and the X and Z coordinates of B determine the location of one apex of the triangle. If a set of programs for each primitive geometrical shape has been established, then when these four points are given, the computer can draw the three orthogonal views and an axonometric projection of the geometrical shape immediately. Through this approach, input procedures and the drawing process are simplified.

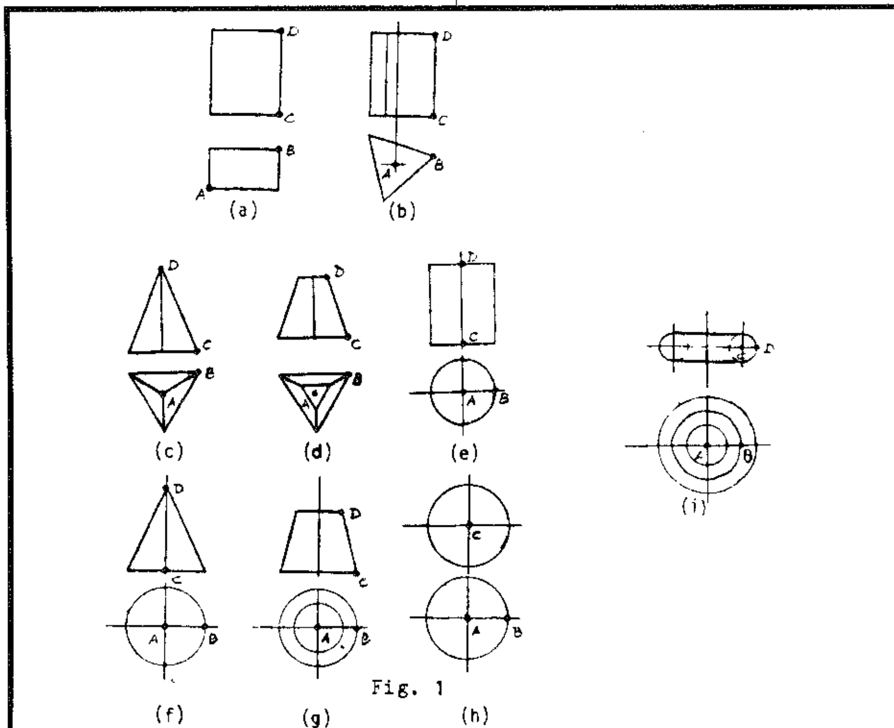


Figure 1

When ADDING or SUBTRACTING a primitive geometrical shape to or from an original object, some lines have to be deleted or changed to dotted lines. For example, Fig. 2 shows that when a primitive geometrical shape A is ADDED to B, the common border lines 1-2 and 3-4 of the two plane faces which lie on the same plane (Fig. 2) are deleted. In Fig. 2b the tangent lines 1-3 and 2-4 of the plane faces tangent to the curved face and the hidden lines in the axonometric projections have to be deleted. Fig. 3 shows a primitive geometrical shape A SUBTRACTED from B, the coincident lines 1-2, 1-3, 1-4, 3-4, 4-5, and 4-6 and the hidden lines in the axonometric projections are deleted.

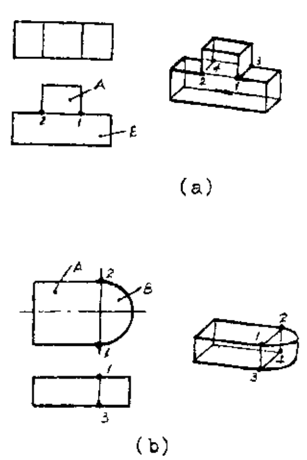


Figure 2

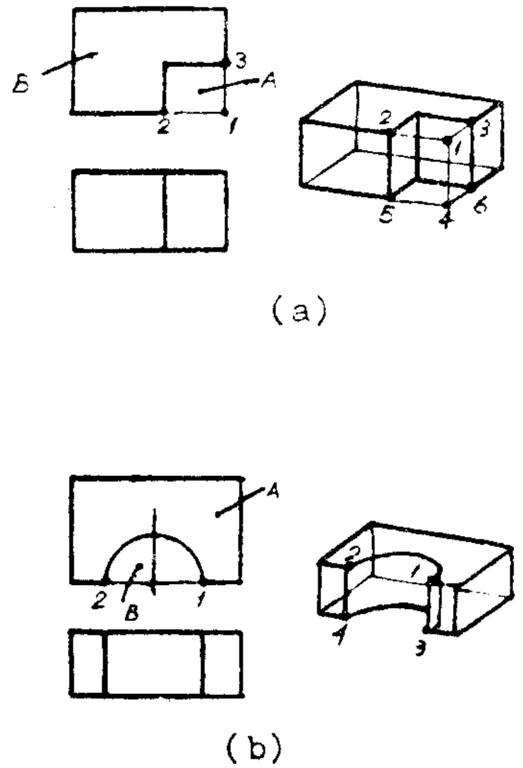


Figure 3

Fig. 4 shows the hidden lines 1-2 and 3-4 in the orthogonal views have to be changed to dotted lines. Traditional interactive graphic computers can also be used to do this but only line by line after the lines are drawn as solid lines. But, by use of the ADDING or SUBTRACTING concept, programs are comparatively easy to set up for doing this automatically by the computer. Therefore, only four points are needed to be inputted when a primitive geometrical shape is going to be ADDED to or SUBTRACTED from an object and will result in a typical drawing on the screen (CRT) of the computer after input. Furthermore, if the appropriate angles made with a horizontal line and the projections of the line of sight is given in advance, different kinds of axonometric projections can be displayed.

In our first approach, we only used a right rectangular prism with its surfaces parallel to the principle projection planes and a right circular cylinder with its center-line perpendicular to one of the principle projection planes. Two computer generated illustrations are shown in Figures 5a and 5b as examples.Figure 6 shows the sequence and locations of inputting four points for a rectangular prism, in which lower left point 1 and upper right point 2 on the front view of the prism for determining its width and height, lower right point 3 and upper right point 4 on the top view of the prism for determining its depth. The X, Z coordinates of point 1 and the Y coordinate of point 3 determine the location of the prism.

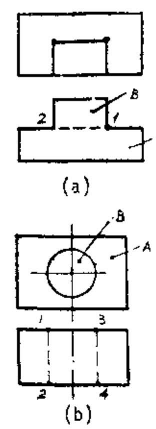


Figure 4

Cylinders can be divided into three major categories: whole, semi and quarter, which often occur in drawings and three kinds of conditions can be specified, that is the center line of the cylinder is perpendicular to different projection planes Fig. 7 shows the sequences and locations of points for distinguishing the different types of cylinders and conditions.

Whether the ADDING or SUBTRACTING operation is performed, the input procedure is the same. Fig. 9 shows ADDING a cylinder to a prism, Fig. 10 shows subtracting a rectangular slot from a prism. Fig. 9a and 10a illustrate inputting of points, Fig. 9b and 10b are the results.

COMPARISON

In order to determine which line or line segment should be deleted (coincidental) or changed to dotted line (or hidden edge), we must compare each line of the primitive geometrical shape with each line of the original object. In orthogonal views, this is done directly by comparing the coordinates of each of two lines. In axonometric projection, one has to also consider the angles e_1 and e_2, which are the angles made with a horizontal projections of the line of sight S of the axonometric projection respectively, see Fig. 11. Let us consider the following three conditions:

1. Coincident lines or line segments of two primitive geometrical shapes (see line 1-2 in Fig. 2a) can be determined directly by first

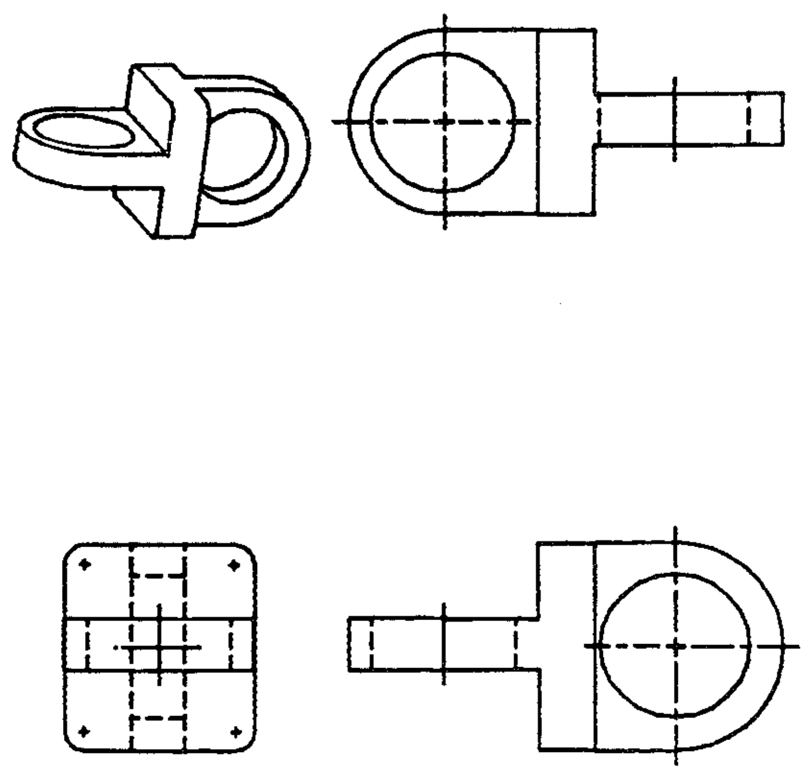


Figure 5

comparing the Z coordinates then the Y coordinates of these two lines to see if they have the same value. If they do this it means that they are coincident. Finally, compare the X coordinates of the starting points and then the end points of both lines to determine the segments 1-2 of the two shapes which have to be deleted. The sequence of comparison according to X, Y, Z is different from different projections and different directions of lines, and therefore this can simplify the comparison programs.

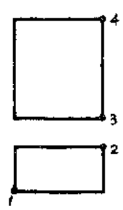


Figure 6

2. Hidden lines or line segments are the lines of the new primitive geometrical shape which are hidden by the visible surface of the original object or vice versa. For example, in orthogonal views, the sequence of comparison is as follows: (a) remove a line from the new primitive geometrical shape, for instance, the horizontal line EF in Fig. 12, and compare with the horizontal lines of the original object. Compare the Y coordinate of EF with the lower visible horizontal line AB of the original object. (b) If $EF_Y > AB_Y$, that means EF is behind AB, than compare the Z coordinate of these two lines. (c) If $EF_Z > AB_Z$, that means EF is higher than AB, then compare the Z coordinate of EF and CD. (d) If $EF_Z > AB_Z$, that means EF is higher than AB, then compare the X coordinate of starting pints E and A, and then the end points F and B. If $E_X > A_X$, $F_X < B_X$, that means EF is fully hidden by the visible surface ABCD (Fig. 12a). The whole line EF has to be changed to a hidden edge line. If $E_X > A_X$, $F_X > B_X$, then compare the X coordinate of point B and E, if $E_X < B_X$ that means segment EK is invisible, and has to be changed to a hidden edge. Obviously, the X coordinate of K is the same as B. In axonometric projections (see Fig. 13), for instance, (a) compare the Y coordinate of EF and AC. (b) If EF is behind AC, then compare A_X+X (see Fig. 13b) with F_X and E_X, if $(A_X+X)(A_X+X)<E_X$, then compare EF with HI, if $(A_X+X)<F_X$, and $(A_X+X)>E_X$, then (c) compare (E_Z+Z) with A_Z and C_Z, if $(E_Z+Z)>A_Z$, then EF is visible, if $(E_Z+Z)<C_Z$, then compare EF with CG, if $(E_Z+Z)>A_Z$ and $(E_Z+Z)<C_Z$, then EK is visible, where $K_X=(A_X+X)$. The comparison of other kinds of lines is similar to the above. For instance, the comparison of circle 0 and line AC in Fig. 14a, (a) if circle0 is behind line AC, then determine the intersection point K, L of circle 0 and line AC, (b) if $(K_Z+Z1)>A_Z$, $(K_Z+Z1)<C_Z$, then K is one of the separate points of the visible and invisible segment of the circle.

3. In the case of SUBTRACTING, the comparison in three views is the same as above, but in axonometric projection, after a primitive geometrical shape has been subtracted, some lines of the slot or hole will be visible (see Fig. 15). Lines AC, CG, CE and arc CF or only line AK can be determined as above. The end point F of arc CF can be determined as shown in Fig. 16, in which, the direction of line of sight of axonometric projection is reversed. In the case of a circular hole, the method of determining the two end points of the arc is the same.

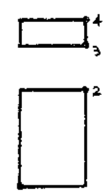


Figure 7

Continued on Page 11

Chairman from p. 5

Educators, like all people live in the present. However, we need to anticipate trends in engineering graphics education that affect our students. Several observable trends are offered for our reflection and thought.

1. The days are over in which a manually-drawn drawing was the end point of the teaching process. Hand-done graphics must now interface with computer graphics. Our task is to make this interface both smooth and meaningful.

2. Our graduates will use graphics in the broad term more than ever. They will use graphical concepts as they interact with computers and develop designs.

3. Extensive manually done drafting will be done more and more by skilled drafters who provide their valuable services without the need for a baccalaureate degree. Graduate engineers will use the concepts and not necessarily the skills to evolve the design.

4. Teaching of CAD is time intensive. As educators try to include a greater percentage of CAD within the overall time available, they must become more efficient in teaching the basic concepts underlying the discipline.

5. A merger is evolving to include traditional manual graphics <u>and</u> CAD within graphics education. The use of the computer to teach <u>concepts</u> will be slow to evolve because of a lack of good software However, the use of the computer to demonstrate <u>applications</u> of concepts will steadily increase.

It is apparent that our area of teaching is not stagnant, but rather full of potential and challenge. There is ample room in engineering graphics for the finest minds in education! Don't be modest. Share your experiences and thoughts!

Robert J Foster
Chairman

Damin from p. 10

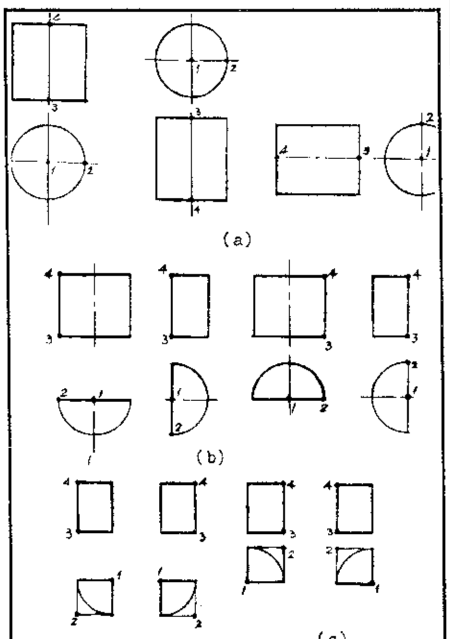


Figure 8

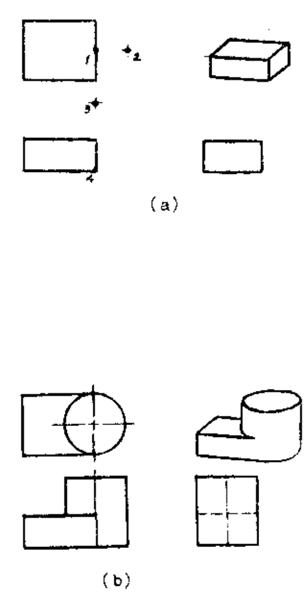


Figure 9

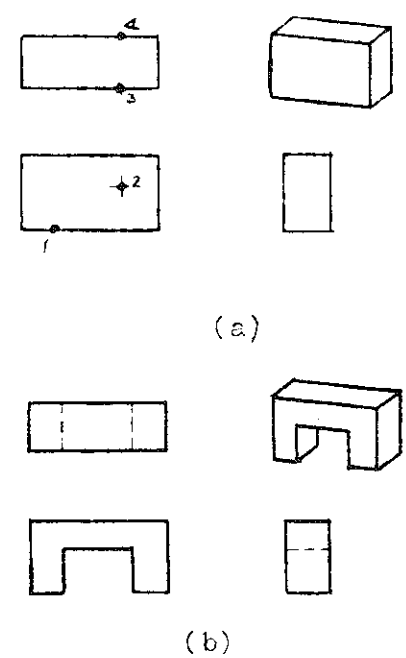


Figure 10

Continued on Page 47

Letters from page 4

Additional letters, continuing the saga of the Charles Moore article (Volume 49, No. 2, pp15-19)

From Abe Rotenberg:

It is true that a "more judicious use of terms" (your note in EDGJ, Autumn 1985, p.22) often may remove confusion, however introducing new terms to replace adequate existing ones is unlikely to help. I noticed that you have changed the wording from "the perspective ellipse which is not a symmetrical orthogonal ellipse" in your earlier Editor's note (EDGJ, Spring 1985) to "a non-symmetrical Perspective envelope", yet you refer to "Abe Rotenberg's confusion" without actually responding to Abe Rotenberg's comments. I suspect that some of your and your reader's difficulties result from confusing drawing methods with drawings. Please note also that Moore's paper deals with the perspective projections of ellipses, and NOT of projections of Circles.

Confusion often depends on your station point..........ed

And in support of Abe Rotenberg, from Pat Kelso:

Regarding LETTERS TO THE EDITOR , FALL, 1985; the point of Abe Rotenberg's letter , it would seem, was simply put, that there is no distinctive entity as an "orthogonal" ellipse which the editor spoke of (a term regrettably also edited by The Journal into a letter published over my own name) verses a "perspective" ellipse which the editor also spoke of. As the poet would say, an ellipse is an ellipse is an ellipse. Abe's point has nothing to do with "a circle in perspective" (which, happily, the editor seems to have acknowledged is not an ellipse after all). It is another point entirely. And when one of the most eminent projective graphicians on our planet, Abe Rotenberg, tells the Journal he is puzzled by the editor's note, perhaps the Journal should not interpret this as implying that it is Abe who is confused.

An ellipse by any other name should appear so elliptical...ed

From Larry Goss on the matter of ellipses and other things:

I would propose that the following test be attempted by Professor Moore's proponents and critics alike:

a. Start with two ellipses tangent to each other at points other than the ends of the major or minor axes.

b. Enclose the ellipse in rectangles and draw or project the rectangles into a perspective scheme of your choice.

c. Using Professor Moore's method, construct two ellipses inside the perspective rectangles.

d. See if the two resulting ellipses are still precisely tangent at the same geometric location as on the planar figures.

What our colleagues fail to recognize, I believe, is that perspective planes are subjected to a non-linear, non-symmetric distortion. This distortion occurs in all forms of perspective, is always present, and is independent of any control we may try to exercise over it. Its effects can be minimized by limiting the included angle in the cone of vision and by placing the perspective plane close to normal to the line of sight. Any geometric shape which is projected on the perspective plane will be subjected to the distortion of the plane itself. For purposes of illustration, the projection of a closed curve such as a circle or an ellipse is beneficial in demonstrating the degree of distortion which occurs, but it literally makes no difference what the shape is that is being used as a test. A square or triangle would work just as well. it's just that the distortion is more readily apparent with curved figures.

I won't get into a discussion of the construction of the human eye and the relationship between it and various camera lenses, but I will say that if the fovea of the human eye was much larger than it is, we as humans would find our normal vision subject to the same perspective distortion that we fight all the time on the drawing board.

It appears that we are destined to make the same mistakes over and over...ed

From EDGD Chairman Bob Foster:

ASEE headquarters has become concerned about the bad schedule of our Journal. Members are calling ASEE wondering why they are not receiving the Journal. This is the kind of publicity our Division doesn't need, but then you are as aware of the problem as anyone. Your suggestions as to how we might avoid this situation next year can help.

I wish I had a nickle for each time an irate EDGD member has called me complaining that they aren't receiving their Journal. First off, as Editor I have no control over whether or not you get your Journal; you either are or are not on the appropriate ASEE file depending on several factors. This includes whether or not you allowed your membership to lapse. I have a feeling that you are dropped immediately and reinstated slowly. You probably should expect discontinuous Journal service if you let your subscription lapse.

The Journal is published three times/year in the Winter, Spring, and Fall. Since we operate on an academic year, you can expect to receive your three copies between December and June, though not at the same time each volume. Because all editorial functions are purely donated, scheduling is highly variable. Because the publisher prints the Journal at a considerable discount, any publishing timetable is also highly variable. Rest assured, you will receive three issues of the EDGJ each academic year! ...ed

1984 北京

Presented at the International Conference on Engineering and Computer Graphics

TEACHING COMPUTER GRAPHICS TO ARCHITECTS AND GRAPHIC ARTISTS

William J. Mitchell and
Robin S. Liggett

Graduate School of
Aarchitecture and Urban
Planning
University of California
Los Angeles

INTRODUCTION

There are at least four very different ways to teach computer graphics. One is to introduce it from systems programming viewpoint, and concentrate on the question of how to engineer computer graphics systems. The focus with this is upon graphics devices and their characteristics,data structures for representing pictures, algorithms for operating upon these data structures, the design of user interfaces, questions of efficiency, and so on. Two excellent texts are available: Newman and Sproull's **Principles of Interactive Computer Graphics**, and Foley and Van Dam's **Fundamentals of Interactive Computer Graphics.**

A second approach is to largely ignore the details of implementation, and to concentrate instead upon underlying mathematical principles. Here the focus is upon matrix transformations of coordinate data, and the mathematical representation of curves and surfaces. There are some useful introductory texts for this, too: Rogers and Adams' Mathematical Elements for Computer Graphics, and Chasen's Geometric Principles and Procedures for Computer Graphics Applications. These two approaches should be regarded, of course, as complementary rather than mutually exclusive. But emphasis upon one of the other will give quite different flavours to introductory computer graphics courses.

Yet another approach is to take some existing turnkey computer graphics system as given, to provide some user training on it, then to let students use it to complete practical graphics projects. This is encouraged by system vendors, since it develops a market for their products. It is also finding some favor, in art and architecture schools, where students are popularly supposed to be incapable of, or at least uninterested in, approaching the subject at any other level. This may be the appropriate approach for a technical college that is concerned with training system operators to occupy specific, technician-level positions in industry, and it is certainly important at a university level to make good computer graphics systems freely available for the everyday use of art and architecture students. But a course which takes this user-training approach has little more genuine intellectual content than a course in how to use a word processor. Furthermore, the system upon which students train is likely to be obsolete by the time that they graduate...if not before.

a carefully graduated sequence of exercises.

In this paper we describe a fourth approach, that we have found to be especially appropriate to the needs of architecture and graphic art students who want to gain a sound, fundamental understanding of computer graphics as a design medium. It forms the basics of a very popular introductory course that we have taught at the Graduate School of Architecture and Urban Planning UCLA, for a number of years. A version is also taught, to the general public, through UCLA Extension (Department of the Arts).

BASIC ASSUMPTIONS

We take as our starting point the **Pascal** programming language, plus a primitive to draw a **vector** on the screen of some display device. These facilities can be provided within many computational environments, right down to the level of home computers with cheap raster display, so the approach may be followed in a wide range of different contexts.

The approach is to conduct students through a carefully graduated sequence of exercises in writing Pascal programs to generate drawings. We assume no previous background in computing, and no previous knowledge of Pascal. We begin with very simple drawings, and introduce the basic Pascal constructs necessary to write programs to generate them. Then we progress through a sequence of increasingly complex and sophisticated types of drawings. At each step, we discuss and analyze the compositional principles that structure a certain type of drawing, show how these principles correspond to Pascal constructs, andcarry out an exercise of writing a program, which uses the constructs in questions, to generate such a composition. The objective, always, is to write a concise, elegant and well-structured program that not only generates the desired result, but also clearly expresses the structure of the drawing. The final project is to write a program to generate a large, ambitious drawing of an important work of architecture. An exhibition of these drawings is held. Students are graded, finally, **both** on the elegance and clarity of the programs that they write **and** on the graphic quality of the drawings that they produce.

Thus we treat programming as an art form, (much like musical composition, choreography, or play writing) which involves a distinction between specifying a work in some notation system, and performing that work.* The programmer is the composer, and the computer is a very exact and meticulous performer.

The central objective, that we accomplish with the course, is to teach students to think about graphic composition in computational terms. This is what designers and artists need to be capable of doing in order to use any computer graphics system creatively. It is knowledge of very general applicability, which does not go out of date easily. And it not only gives insight into computing; it also gives students the opportunity to see issues of architectural and graphic composition in a new and revealing way.

*See Goodman(1976) for an important philosophical analysis of this issue.

THE FIRST STEP: VECTOR-BY-VECTOR DRAWINGS

The capability to draw a vector is added to Pascal by providing two external procedures:

MOVE (X,Y)

which moves the electron beam invisibly to coordinates (X,Y), and:

DRAW (X,Y)

which draws a vector from the current position of the electron beam to (X,Y). A screen coordinate system, in raster units, is used.

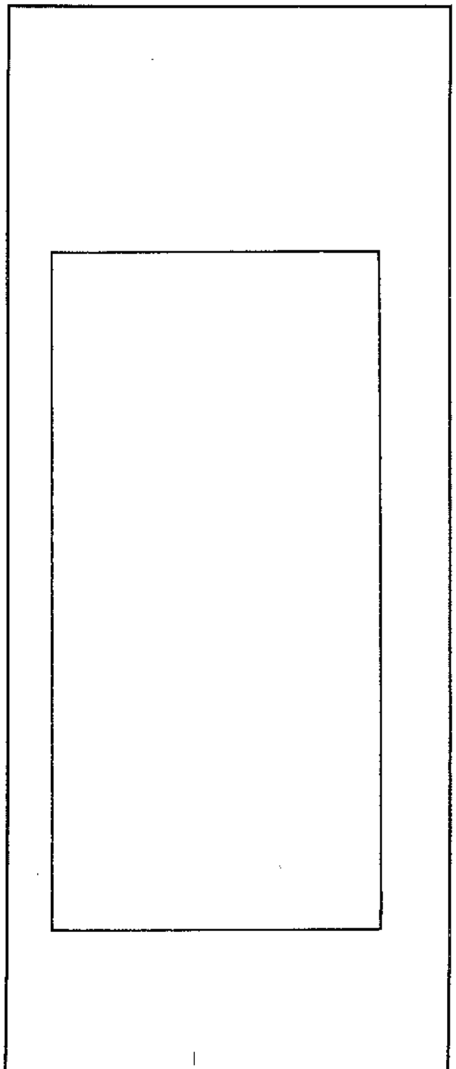


Figure 1

The first exercise is to write a program that generates a simple drawing on the screen by executing MOVEs and DRAWs. The only knowledge of Pascal that this requires is of the basic organization of a Pascal program. It can be successfully completed by students after their first class. An example of a typical result is shown in Figure 1.

THE SECOND STEP:
COORDINATE CALCULATIONS

When we consider slightly more complicated figures, it becomes evident that it is often useful to do some arithmetic to calculate coordinates within a program. The concepts of assignment and Pascal arithmetic expressions are introduced at this point. More ambitious calculations suggest the need for trigonometric and other functions, so the concept of a function, and the Pascal standard functions, are introduced too.

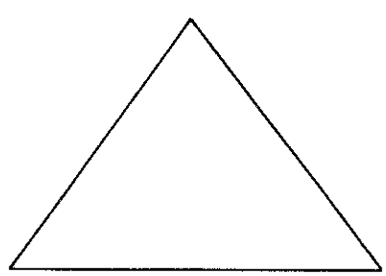


Figure 2

The exercise that follows is to write a program to draw a figure that requires some coordinate calculations. An example is shown in Figure 2.

THE THIRD STEP:
PROCEDURES TO INSTANTIATE VOCABULARY ELEMENTS.

Next, we analyze more complex drawings still, and show that they are usually made up out of instances of certain types of figures. This leads to the idea of a graphic vocabulary as a set of such types. Each element of such a vocabulary may have parameters to control shape and parameters to control position.

The corresponding Pascal construct is a parameterized procedure that generates an instance of a vocabulary element. Declaration of such procedures establishes a graphic vocabulary. Invocation of these procedures creates a composition. The next exercise, then, is to define a simple graphic vocabulary and use it to build up a composition. Figure 3 shows a typical result.

This exercise raises some non-trivial questions of design theory. How should the essence of an architectural or graphic type be defined?

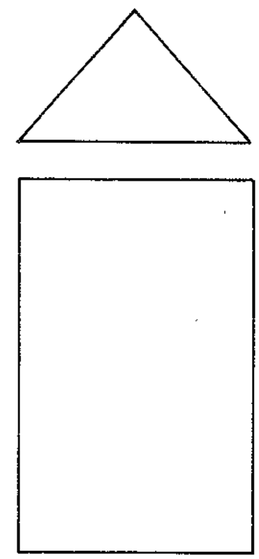


Figure 3

How should this essence be expressed as code within a procedure? What shape and position properties do you want to vary from instance to instance? How can you best formalize these design variables as procedure parameter?

THE FOURTH STEP:
REPETITION.

Once we have established the idea of a graphic vocabulary, we can go on to consider the ways that vocabulary elements may be combined. That is, we begin to consider principles of composition. . . . repetition, symmetry and so on.

The most elementary principle of composition is regular repetition, and the corresponding Pascal construct is a loop within which a vocabulary element (or elements) is instantiated. Shape, or position, or both may be varied at each iteration. Several loops in sequence, or nested loops, may be used to generate more complex repetitive compositions. The exercise, now, is to use loops to generate a repetitive composition. Figure 4 illustrates this.

THE FIFTH STEP:
CONDITIONALS.

Every designer knows, though, that compositions are rarely strictly regular. Under certain conditions the pattern may be broken. In a regular column grid, for example, the corner columns may be made a different size and shape from the interior columns.

The idea of Pascal conditionals, that is if and case statements, can be introduced at this point, and it can be shown that surprisingly subtle and apparently complex compositions can often be generated by concise programs that use loops and conditionals.

The next exercise is to introduce conditionals into a program that generates a repetitive composition. Figure 5 shows a result.

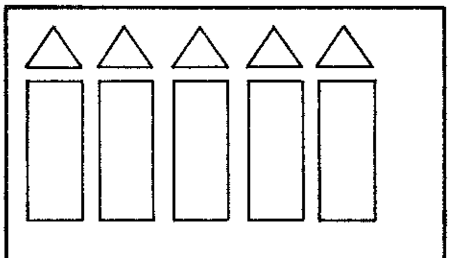


Figure 4

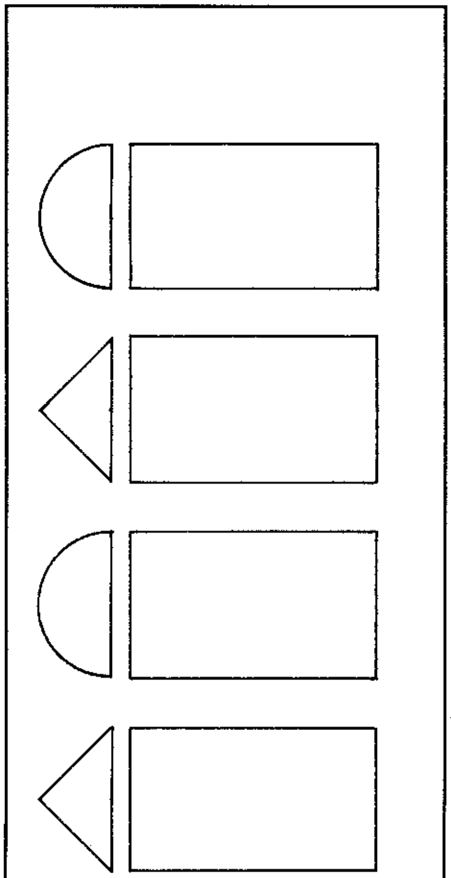


Figure 5

THE SIXTH STEP:
NESTING.

Upon close inspection, most interesting drawings turn out to have a hierarchical structure. Vectors are put together in certain ways to form vocabulary elements, then vocabulary elements are put together to form sub-compositions ,sub-compositions are put together to form high level sub-compositions, and so on to the complete drawing.

This logic can be reflected by the nesting of Pascal procedures. So the next exercise (figure 6) is to generate such a drawing

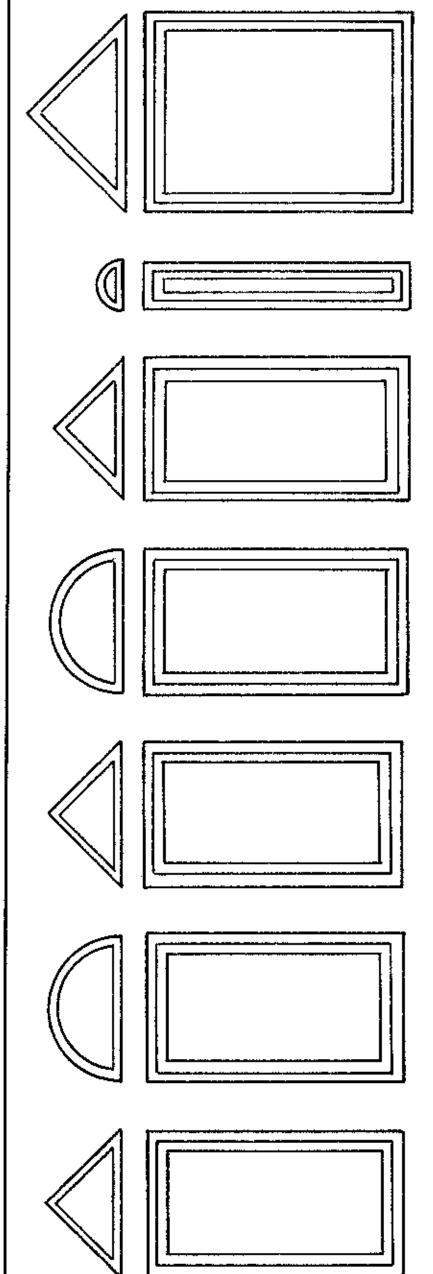


Figure 6

using appropriately nested procedures.

THE SEVENTH STEP:
DATA STRUCTURES AND TRANSFORMATIONS.

Up until this point, a heavily procedural view of graphics has been taken. Vectors have been thought of as DRAW operations, graphic vocabulary elements are procedures, and drawings as programs. Now we discuss the possibility of thinking of drawings not as programs, but as data, and the idea of a data structure that encodes a drawing is introduced. We concentrate mostly upon the very simple sort of structure shown in Figure 7.

X	Y	BEAM
200	200	OFF
500	200	ON
500	680	ON
200	680	ON
200	200	ON

```
MOVE (200,200);
DRAW (500,200);
DRAW (500,680);
DRAW (200,680);
DRAW (200,680);
```

Figure 7

Once we have a drawing represented in such a data structure, we can transform it in various ways by applying procedures to the data structure. The standard geometric transformations are introduced at this point, and their matrix formulation is discussed.

The next exercise is to rewrite some earlier program so that it now puts coordinate values into a data structure rather than directly executing MOVEs and DRAWs. (This emphasizes the logical equivalence of thinking of a drawing as a procedure and thinking of a drawing as a set of values in a data structure.) Students are provided with code to perform geometric transformations, and asked to write a program which generates a composition out of transformed instances of vocabulary elements.

THE EIGHTH STEP:
THE FINAL PROJECT.

The students now know enough to write concise, well-structured programs to generate surprisingly complex and sophisticated drawings. Further concepts can be introduced at this point (in particular, curve description and recursion), but these are not strictly necessary in an introductory course.

The final exercise is to select some important work of architecture, and to write a well-structured program which draws it in elevation or plan. Figure 8 shows some typical results.

THE LIMITATIONS OF THE APPROACH.

As the examples of final projects show, the concepts of graphic vocabulary, repetition, conditionals, nested picture structure, and geometric transformation together constitute a very powerful graphic tool-kit, and students are able to use it to produce very interesting results. However, we certainly would not want to suggest this is the only way to look at picture structure, or that Pascal provides the only way to express algorithms that generate drawings.

First, we must keep in mind that the whole approach is based upon the assumption that a drawing is a set of vectors. This is a reasonable and appropriate assumption for our purposes, but it can be challenged on two levels. One may suggest that something other than a vector should be taken as the primitive. And one may argue that a drawing should not necessarily be thought of as a finite set of discrete elements. Some very interesting alternative approaches to analyzing picture structure may be developed upon different foundations from the one that we have chosen for our purposes here.

Secondly, there are more powerful ways to describe picture structure than that which we employ. In particular, it is possible to write formal grammars to generate pictures.* In this

*See Stiny(1980), and Stiny and Mitchell (1978) for details of this approach.

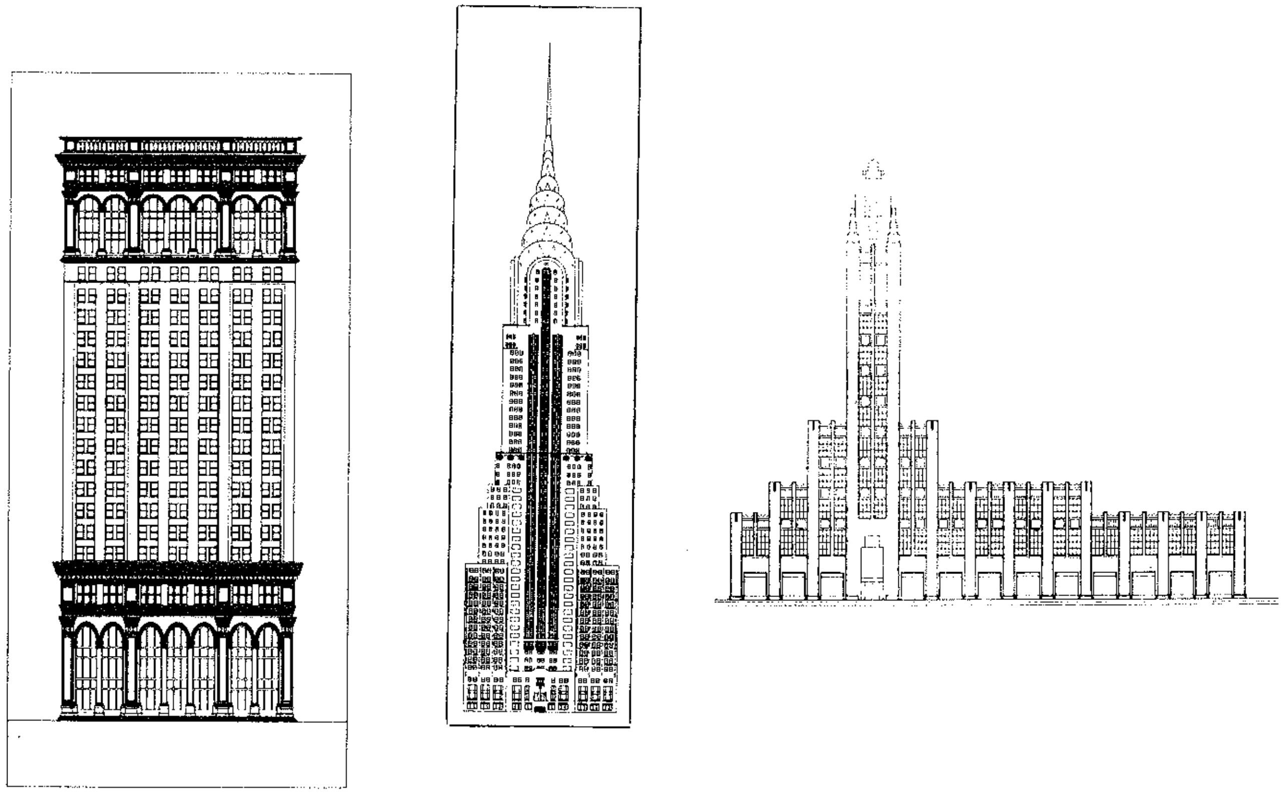


Figure 8

case, rules of architectural or graphic composition are expressed as productions, and a picture is generated by recursively applying a production system to a data structure. In principle this is a very attractive approach, but in practice there is no appropriate and widely available software to support its use in teaching elementary computer graphics.

Thirdly, we should recognize that Pascal code expresses algorithms to generate pictures at a particular level. We could use a language that expressed them at a lower level, or at a higher level. The level of Pascal seems appropriate for an introductory course, though, since it is the level at which most practical computer graphics application software currently is being written and discussed in text books, and the level at which introductory programming courses are customarily taught.

CONCLUSIONS.

It is common to make the assumption that artists and designers do not think is a structured way, cannot handle mathematics and logic very well, and cannot therefore be expected to develop a sophisticated grasp of the principles of computation. In fact, talented graphic artists and designers have deep insights into the structure of drawings that they manipulate. The approach to teaching computer graphics that has been outlined here builds upon those insights. We have found that students are uniformly enthusiastic about it, that few get into serious difficulties with the material, and that the better students rapidly achieve a very good grasp of it. There is a high correlation of success in design studio courses with success in this course.

Computer scientists, of course, also have insights into the structures of things, and students who complete this course gain from the concepts of structure which form the basis of Pascal a new way to analyze architectural and graphic compositions. Architectural teachers are fond of observing that you do not really understand a building until you draw it carefully yourself. We may add that you do not really understand it until you can write a concise, expressive algorithm to generate that drawing. And you have a still deeper level of understanding if you can structure that algorithm, and parameterize its procedures, in such a way that you can easily generate interesting variants of the original drawing by changing the values that are initially assigned to some of the variables that control shape, position and repetition of vocabulary elements and subassemblies.

REFERENCES.

S.H. Chasen, Geometric Principles and Procedures for Computer Graphics and Applications, Prentice-Hall, Englewood Cliffs, NJ, 1978.

James D. Foley and Andries Van Dam, Fundamentals of Interactive Computer Graphics, Addison-Wesley, Reading, MA, 1982.

Nelson Goodman, Languages of Art (Second Edition), Hacket, Indianapolis, IN 1976.

William M. Newman and Robert F. Sproull, Principles of Interactive Computer Graphics (Second Edition), McGraw-Hill, New York, NY 1979.

D.F. Rogers and J.A. Adams, Mathematical Elements for Computer Graphics, McGraw-Hill, New York, NY 1979.

George Stiny, "Introduction to Shape and Shape Grammars", Environment and Planning B, vol. 7, 1980, pp. 343-351.

George Stiny and William Mitchell, "The Palladian Grammar", Environment and Planning B, Vol. 5, 1978, pp. 5-18.

George Stiny and William Mitchell, "Counting Palladian Plans", Environment and Planning B, Vol. 5, pp. 189-198, 1978.

INVENTORY MANAGEMENT USING VECTOR GRAPHICS

J. S. Duggal
Texas A&M University

Introduction

In this article a parallel is drawn between graphical analysis of forces and replenishment and depletion of inventory in industrial situations. Vectors representing quantities, such as number of bolts, bottles of soda etc., can be used to solve inventory problems in similar manner to vectors representing forces used in solving shear and moment problems in statics. Graphic statics[1,2] deals with forces whereas inventory systems[3] deal with quantity of material. This quantity of material is dealt with as a force and the time span is considered as a beam. The graphical aspect is emphasized in this article as it is easier to visualize by eye inspection the condition of inventory at any point in time. Before proposing the theoretical models it would be essential to acquaint the reader with force system which has been used to create the models for inventory systems.

Resultant of Coplanar Parallel Force Systems

This system of forces is a special case of non-concurrent force system. The line of action of the forces in such a system lie in one plane and are parallel to each other.

As shown in Figure 1, there are three basic types of parallel force systems.

1. The resultant is a single force as shown in Figure1(a).

2. The resultant is a couple (moment) as shown in Figure 1(b).

3. The resultant is equal to zero. This means no resultant force and no resultant moment as shown in Figure 1(c).

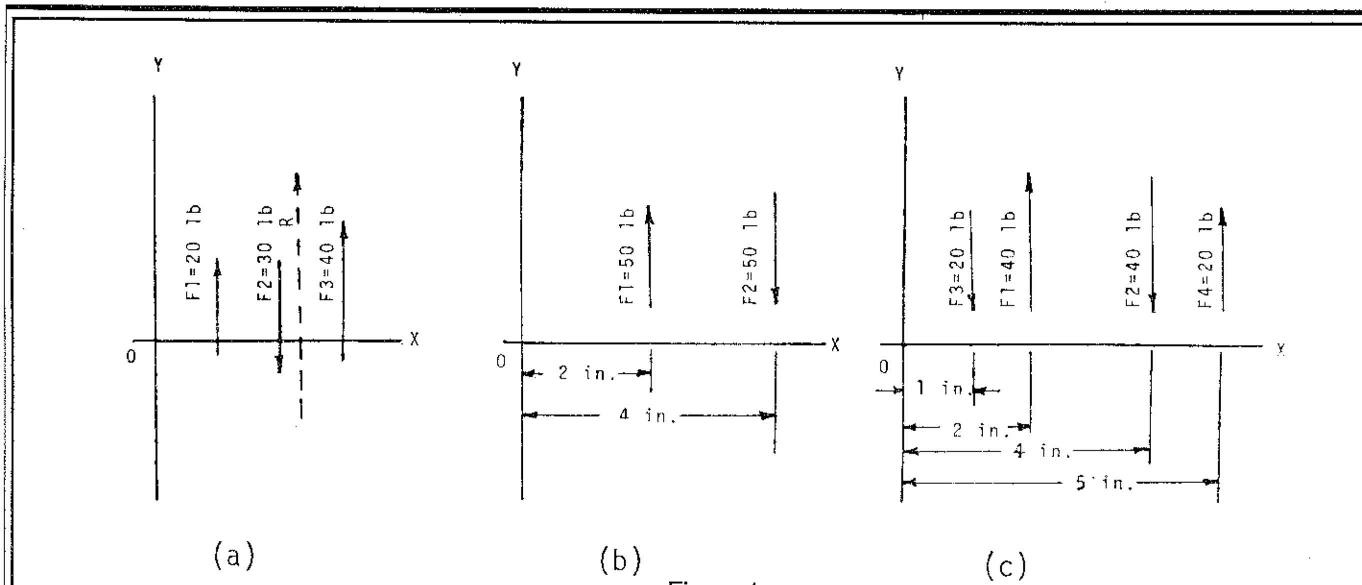


Figure 1

Similarities Between the Inventory System and Coplanar Parallel Force System.

Inventory systems deal with quantities of material and their distribution and/or availability over a period of time. This period of time is a linear quantity and is measured as days, weeks, months, etc. The amount of material varies at different times and is referred to as replenishment and depletion quantities. Whereas a structural force system deals with magnitude (quantity) of forces and their distribution along a structural member. For parallel forces this distribution is linear and is measured as feet, meters, etc. The direction of the force could be upward or downward. Or in other words, a set of inventory quantities over a span of time could be dealt with as a beam with set of parallel forces. This comparison is illustrated in Figure 2. Figure 2 (a) shows a conventional force/distance relationship in statics whereas Figure 2 (b) shows an equivalent quantity/time inventory system.

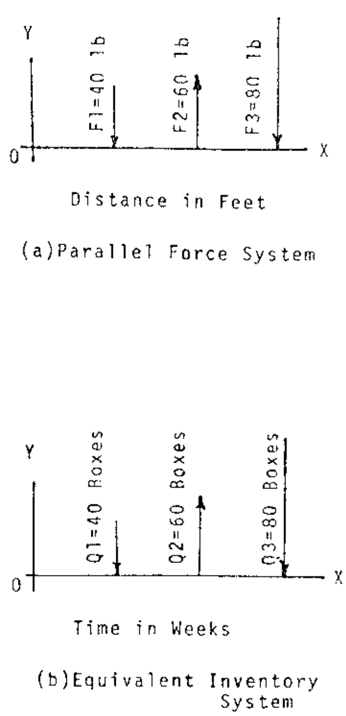


Figure 2

The resultant inventory at any point in time is what a production control man is seeking, whereas a resultant force and its line of action is what an engineer is looking for in designing a beam. This resultant force and its line of action can be determined using analytical or graphical procedures. The two systems can best be compared in terms of their resultants (R_1,R_2,R_3,R_4) in Table 1.

From the above discussion it is apparent that the location of resultant inventory vector and quantity of it, both have to be examined to access the inventory situation. The position of resultant inventory vector being too much inside the time span and being too much outside the time span is unfavorable. If the resultant inventory vector is upward and is too much inside of time span an excessive inventory amounts and reflected and hence a higher cost of carrying inventories. On the other hand, if the resultant inventory vector is downward and is too much inside of the time span it would indicate the use of safety stock and eventually a complete depletion of inventory, which would have the implication of stopped production.

The Seven Models

Seven models (Figures 3-9) are now presented to describe various inventory situations. The last three of these models have only a theoretical implication. The graphical procedure in constructing the vector diagrams is illustrated in each model. References at the end can provide details on the graphical procedure of finding resultant of parallel forces[1,2]. The models will be studied with certain underlying assumptions. These assumptions have been made to illustrate, in the simplest possible manner, the models being proposed. The seven models described here, for illustration purposes, have the following asssumptions associated with them:

1. Two weeks of inventory is studied because it is the minimum number to derive the resultant.
2. The upward arrow indicates replenishment of inventory.
3. The downward arrow indicates depletion of inventory.
4. For simplicity the pound is used as a measure of quantity of inventory. It would be number of boxes, tons of steel, spools of cables, etc.
5. A week of safety stock inventory is available at all times and is 100 lb.
6. No single depletion will be larger in amount than the safety stock, i.e., 100 lb.

inventory is easier to visualize at any point in time

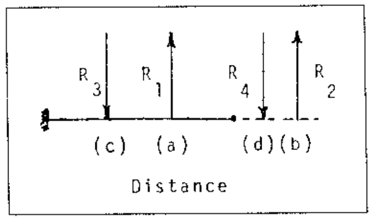
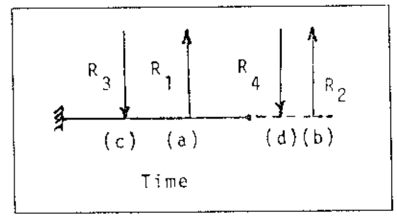
Parallel Force System	Equivalent Inventory System
	
(a) If R1 is upward and falls within the beam, beam is being pushed up at point a with a force of magnitude R1.	(a) If R1 is upward and falls within the same time span being studied, enough inventory is on hand equal to the quantity of R1.
(b) If R2 is upward and falls outside of the beam, the beam has to be extended to be replaced by a single force of magnitude R2. The beam will then be pushed up.	(b) If R2 is upward and falls outside the time span being studied, enough inventory will be on hand beyond the time being studied equal to the quantity R2.
(c) If R3 is downward and falls within the beam, beam is being pushed down at point c with a force magnitude R3.	(c) If R3 is downward and falls within the time span being studied, inventory is already depleated at point c by a quantity of R3.
(d) If R4 is downward and falls outside of the beam, the beam has to be extended to be replaced by a single force of magnitude R4. The beam will then be pushed down.	(d) If R4 is downward and falls outside the time span being studied, inventory will be depleated before the time at point d.

Table 1: Comparison of Force System and Inventory System

Model 1. Replenishment Before Depletion

This model illustrates that if at the end of week one 100 lb of inventory is added and by the end of week two only 30 lb of inventory is depleted the resultant effect is 70 lb. The location of this resultant is of significance as it points out that inventory is good for at least one and one half of a week. No safety stock has been used.

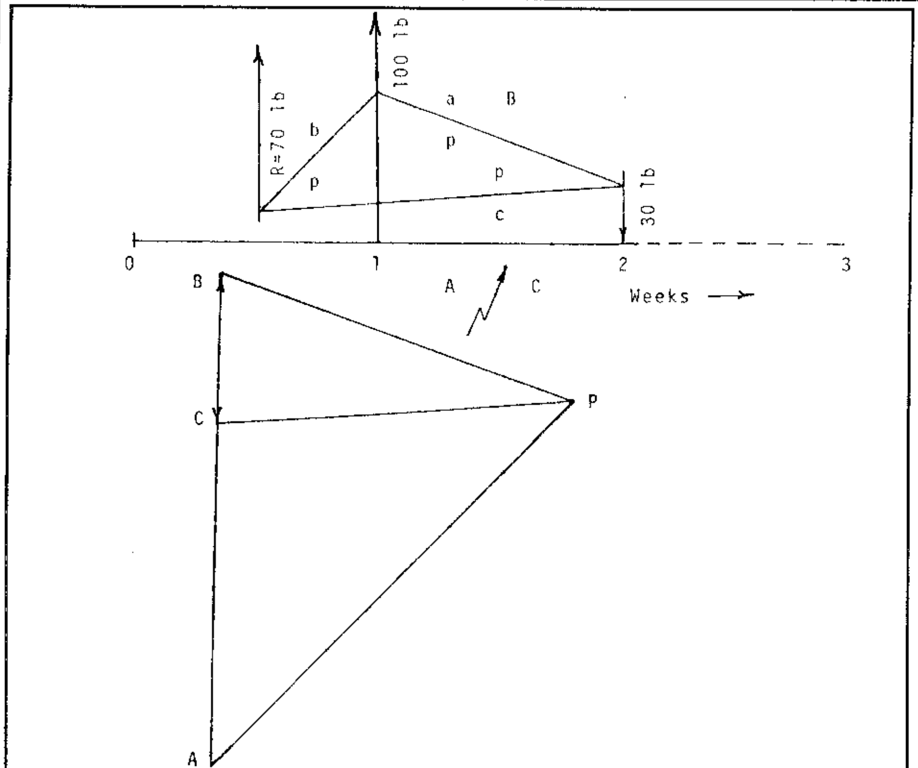


Figure 3

Model 2: Lesser DepletionBefore Larger Replenishment

This model describes that if at the end of week one 30 lb is depleted (of course this depletion will be made from the safety stock of 100 lb) and at the end of week two 100 lb of inventory is added, week three will have a resultant of 70 lb. Monitoring the location of this resultant with respect to anticipated depletions and replenishment is of significance in assessing the future inventory quantities.

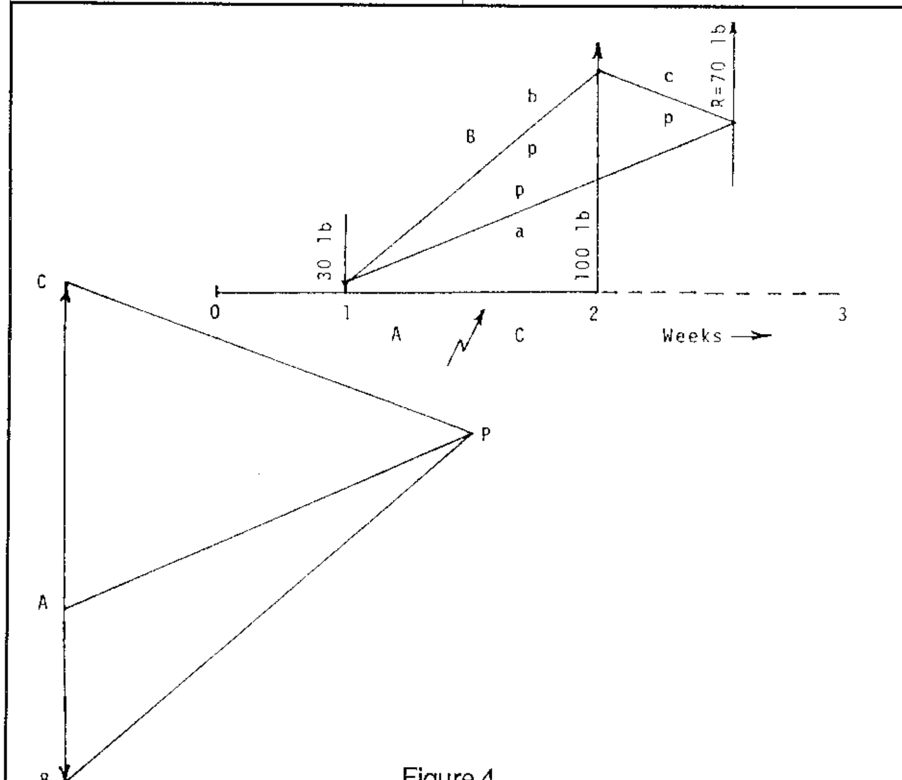


Figure 4

Model 3: Excessive Depletion After Lesser Replenishment

This model illustrates that if at the end of week one 30 lb of inventory is replenished and by the end of week two 100 lb of it is depleted the resultant will be 70 lb downward and in the third week segment. How far away it is in the third week describes the depth of situation. As described in the assumptions that one week of safety stock is available, hence model indicates the use of safety stock.

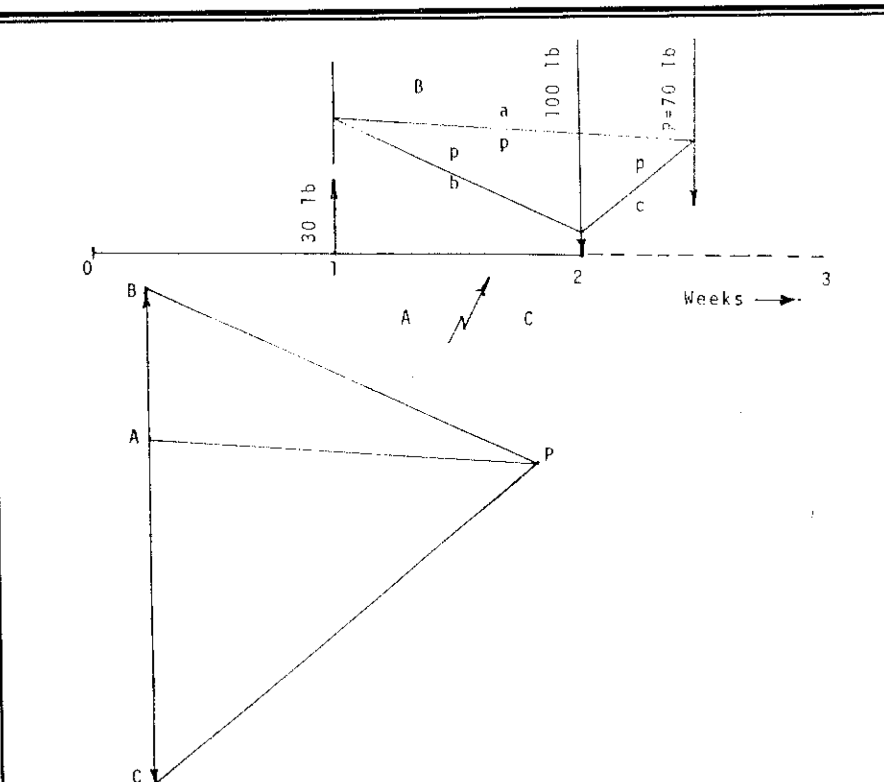


Figure 5

Model 4: Excessive Depletion Before Smaller Replenishment

This model illustrates that if at the end of week one 100 lb. of inventory is depleted and at the end of week two only 30 lb. will be replaced, the resultant moves into the week one and hence immediate depletion of safety stock is indicated and before the middle of the first week a complete exhaustion of inventory including the safety stock is anticipated. This situation would require immediate action.

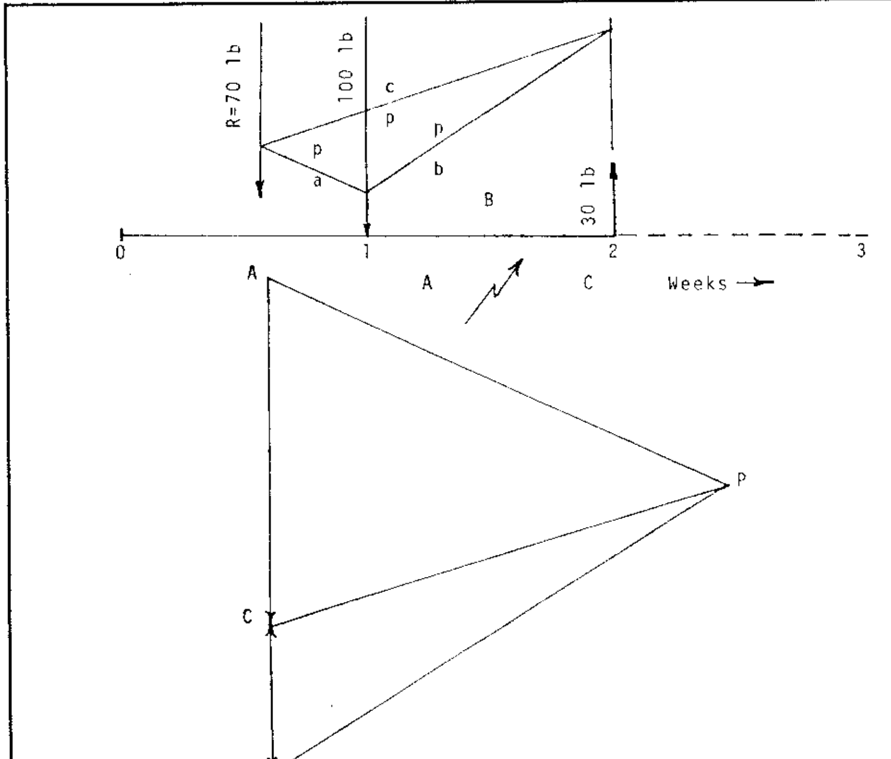


Figure 6

Model 5: Constant Supply Depletion After Replenishment

This is a theoretical model and describes the ideal situation. At the end of week one 100 lb. of inventory is replenished and at the end of week two 100 lb. is depleted and the cycle goes on. There is no resultant and only 100 lb.-week of clockwise moment exists.

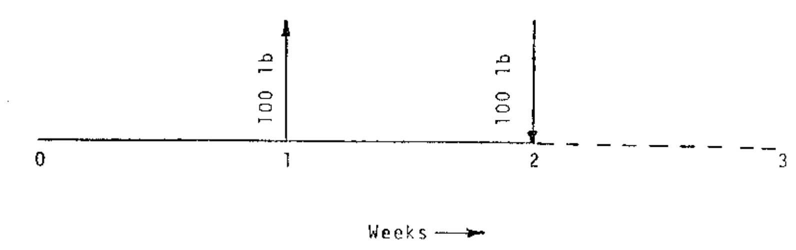


Figure 7

Model 6: Constant Supply Depletion Before Replenishment

This also is a theoretical model and describes the ideal situation. At the end of week one 100 lb. of inventory (of course this depletion will be made from safety stock) is depleted and at the end of week two 100 lb. is replenished. There is no resultant and only 100 lb-week of counter-clockwise moment exists.

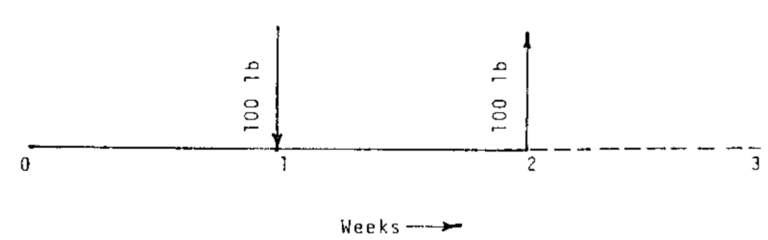


Figure 8

Model 7: Constant Supply Replenishment and Depletion in Good Balance

This model is also a theoretical model with no resultant and no moment with the use of safety stock replenishment and depletion cycle can be maintained.

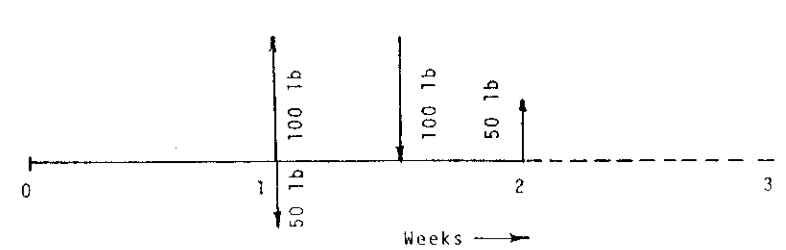


Figure 9

continued on page 33

MATHEMATICAL PRINCIPLES OF OBLIQUE PROJECTION WITH APPLICATIONS FOR COMPUTER GRAPHICS

Ming H. Land
Appalachian State University

Introduction

Among the various types of pictorial projections that find applications in engineering illustration, oblique projection is used to illustrate objects composed of cylindrical shapes in regular or sectional views. Unlike axonometric projection, which is a form of orthographic projection, oblique projection is based on a different set of principles A review of existing engineering and computer graphics literature reveals that very little has been written regarding the mathematical principles of oblique projection. Newman and Sproull (1979) and Giloi (1978) have covered the mathematical principles of perspective projection extensively. Rogers and Adams (1976) have exensive treatment on the mathematical principles of pictorial projection for computer graphics applicaton except oblique projection. Bunk (1984) presented an analysis of the oblique projection theory in terms of the so-called plan and profile approach angles in the Sjpring 1984 issue of Engineering Design Graphics Journal. Unfortunately, these two angles as well as the projection angle were not properly printed out in the article.

In order to provide a better understanding of the theory of oblique projection, this paper will present an analysis of the mathematical principles of oblique projection and techniques of applying these principles to computer graphics.

Oblique Projection

Oblique projection is a method of constructing a pictorial with the observer locating at an infinite distance from the object and the projectors parallel to each other and oblique to the plane of projection. The object is generally placed with one of its principal faces parallel to the plane of projection.

As shown in Fig. 1, the projectors, or lines of sight, are oblique to the plane of projection of an object given in the top and side views. The angle made by the projectors with the plane of projection is α in the top view and β in the side view. The frontal face that is parallel to the projection plane appears in true size and shape. The surfaces of the object that are not parallel to the projection plane do not project in true size and shape. The angle made by the receding axis of the right surface with the horizontal plane is θ in the oblique projection.

scales can easily be built in the computer program ......

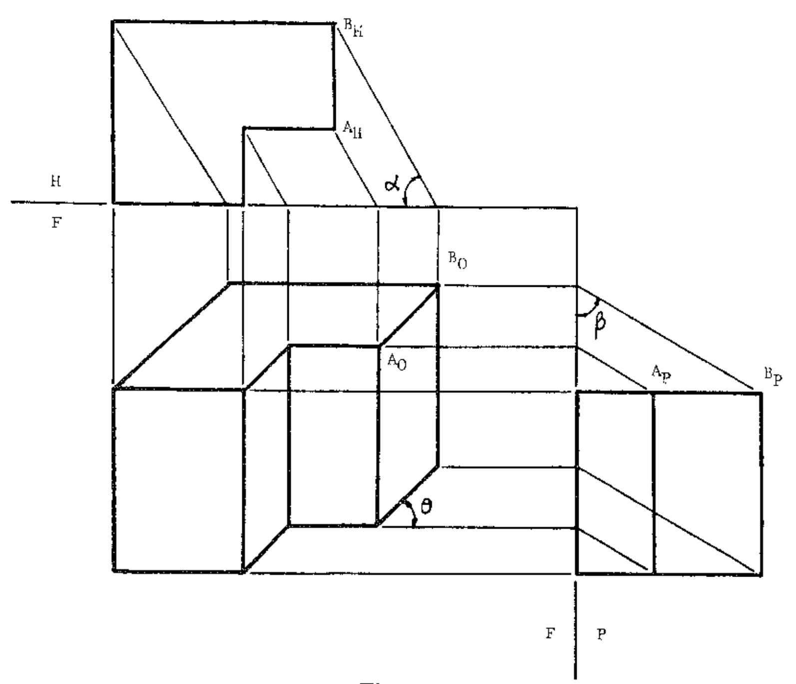


Figure 1

Land from page 28

```
]LIST

5   REM OBLIQUE PROJECTION
10  REM LOCATE ORIGIN
20  T1=119
25  T2=69
30  T3=0
33  REM PROJECTION ANGLE
35  Q=-30/180*3.14159
40  DIM P(45,3),P1(45,3)
45  DIM C(3,3)
50  FOR X=1 TO 45
55  READ P1,P2,P3
60  P(X,1)=P1
65  P(X,2)=P2
70  P(X,3)=P3
75  NEXT X
80  DATA
```

DATA

```
95  REM     TRANSFORMATION
96  REM MATRIX FULL SCALE
97  REM CAVALIER PROJECTION
100 C(1,1)=1
110 C(1,2)=0
120 C(1,3)=0
130 C(2,1)=0
140 C(2,2)=1
150 C(2,3)=0
160 C(3,1)=COS(Q)
170 C(3,2)=SIN(Q)
180 C(3,3)=1
200 FOR X=1 TO 45
210 P1(X,1)=T1+P(X,1)*C(1,1)+
     P(X,2)*C(2,1)+P(X,3)*C(3,1)
220 P1(X,2)=T2+P(X,1)*C(1,2)+
     P(X,2)*C(2,2)+P(X,3)*C(3,2)
230 P1(X,3)=T3+P(X,1)*C(1,3)+
     P(X,3)*C(2,3)+P(X,3)*C(3,3)
240 NEXT X
250 HGR : HCOLOR=3
255 HPLOT P1(1,1),P1(1,2)
260 FOR X=1 TO 44
270 HPLOT  TO P1(X,1),P1(X,2)
280 NEXT X
300 REM   DRAWS CIRCLES
310 F=1.1:PI=3.14159
320 LET N=100
330 LET XO=139:LET YO=84
335 LET R=8
340 A1=0:A2=2*PI
350 INC=(A2-A1)/ N
360 HPLOT XO+R,YO
370 FOR I=A1 TO A2+.01 STEP
     INC
380 HPLOT TO X0+R*COS(I), YO
     +R*SIN(I)/F
390 NEXT I
400 REM  DRAWS CENTER LINES
410 HCOLOR=3
420 HPLOT 139,74 TO 139,94:
     HPLOT 129,84 TO 149, 84
430 HCOLOR=0
440 HPLOT 139,80 TO 139,81:
     HPLOT 139,87 TO 139,88
450 HPLOT 136,84 TO 137,84:
     HPLOT 141,84 TO 142,84
500 REM  DRAWS ARCS
505 HCOLOR=3
510 F=1.1:PI=3.14159
520 XO=147:YO=76
530 R=8
540 A1=105*PI / 180
550 A2=160*PI / 180
560 INC=(A2-A1) / N
570 FOR I=A1 TO A2 +.01 STEP
     INC
580 HPLOT XO+R*COS(I), YO
     +R*SIN(I) / F
590 NEXT I
600 REM  DRAW BORDER
610 HPLOT 5,5 TO 275,5, 155 TO
     5, 155 TO 5,5
620 END
```

Damin from page 47

of an inclined surface on primitive shapes; (c) the addition of other kinds of primitive geometrical shapes, etc.

REFERENCES

1. Steve M. Slaby, "The Fundamentals of Three Dimensional Descriptive Geometry."

2. Zhu Fuxi and others, "Architectural Engineering Drawing." 1982.

3. Yehonathan Hazony, "Interactive Computer Methods in Design and Analysis." 1982.

4. David F. Rogers and J. Alan Adams, "Mathematical Elements for Computer Graphics." 1982.

5. I.D. Faux and M.J. Pratt, "Computational Geometry for Design and Manufacture." 1979.

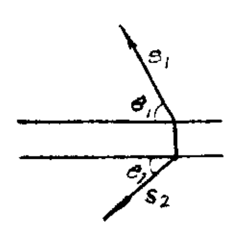


Figure 11

Continued inside back cover

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:

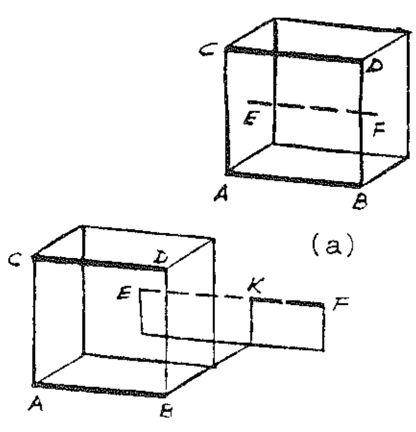
Jon M. Duff
Editor, EDGJ
355 Knoy Hall of Technology
Purdue University
West Lafayette, IN 47907

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.

Damin Figures from p. 47



(b) Figure 12

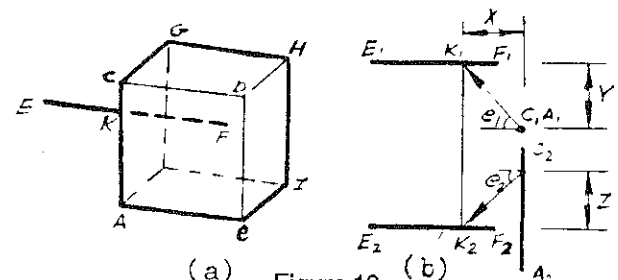


Figure 13

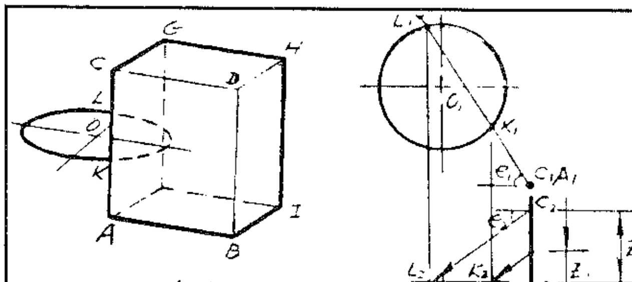


Figure 14

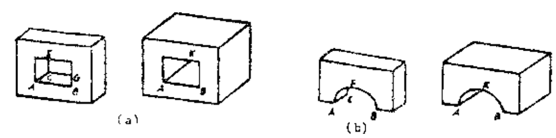


Figure 15

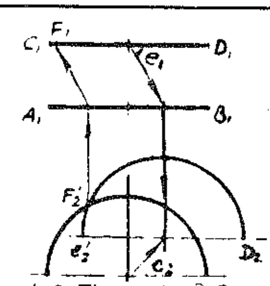


Figure 16

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

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