Journal of Structural Geology, Vol. 9, No. 4 pp. 507 to 509, 1987
Printed in Great Britain

0191–8141/87 $03.00 + 0.00

The use of computer animation techniques in the teaching of stereographic projection

PAUL D. RYAN

Department of Geology, University College, Galway, Ireland

Abstract—A suite of 10 programs, written for the BBC Model B, uses computer animation techniques to aid in the teaching of the principles of stereographic projection. The representation of the net remains static in the background whilst plotting and rotation occur on a foreground plane. This mimics the actual operations that are required using a net and tracing paper. Plotting of planes, lines, intersection and rotation problems are covered by this package. The methods of computer animation are reviewed.

INTRODUCTION

THE TEACHING of stereographic projection requires the student to become familiar with both the principles of projection and also with the various plotting procedures. Such familiarity is usually gained by using worked examples from handouts, and hence is non-interactive and usually involves rehearsing the solutions to a few specific problems. To overcome this difficulty a package has been written for the BBC Model B microcomputer that uses computer animation techniques to illustrate all steps involved in the plotting and solving of user-defined problems on the stereonet.

The implementation of such a package on the BBC micro places four main constraints upon the programmer: (1) the use of high resolution graphics (mode 0, 640 × 256 pixels) limits the amount of machine memory available; (2) a rapid technique is needed to plot a faithful representation of the net; (3) algorithms are needed to calculate and plot poles and great circles; and (4) an animation procedure is required to perform all the necessary rotations of the tracing paper on the screen.

MEMORY USAGE

Any unexpanded 8 bit micro has only 64K of memory available. In the BBC Model B the operating system and language require 32K, the high resolution screen another 20K and the disc filing system, buffers and workspace 6.25K, thus leaving only 5.75K for the programmer. Cheap 'shadow memory' boards, which are standard on recent models may be fitted that release the screen memory allowing c25K in all modes. However, as this package was written for the basic machine it had to be broken up into 10 short programs which were compacted by putting several statements on one line, using short variable names and omitting all but the essential spaces in the listing. Finally the systems variable 'PAGE' was reset from its default setting of &1900 on a disc based machine to &1300 which saves a further 1.5K. This technique can only be used if programs, not data, are read from the disc.

THE EQUAL-AREA NET

In research applications it is probably satisfactory to represent the net by the Primitive with a N arrow and a centre point. However, teaching requires a more realistic representation (Fig. 1). Generating such a net repeatedly from within a program by calculating the co-ordinates of each point would be very slow. Hence, these co-ordinates were calculated beforehand and stored in a data statement in the first program, "LAMBNET". These values, which range from 0 to 250 and can be stored as 1 byte numbers (0–255), are poked into a suitable space in the machine's memory (&A00–&AC8 for a disc based machine). Only one quadrant of the net need be digitized, the entire net being plotted by a simple routine that peeks the appropriate values from the machine's memory and alters the signs accordingly (Fig. 1). This method produces an adequate representation of the net in under 2.2 s and uses minimal memory to retain the co-ordinates (&C8 or 198 bytes). If a net of different scale is required these one byte co-ordinates can be multiplied by a scale factor before plotting.

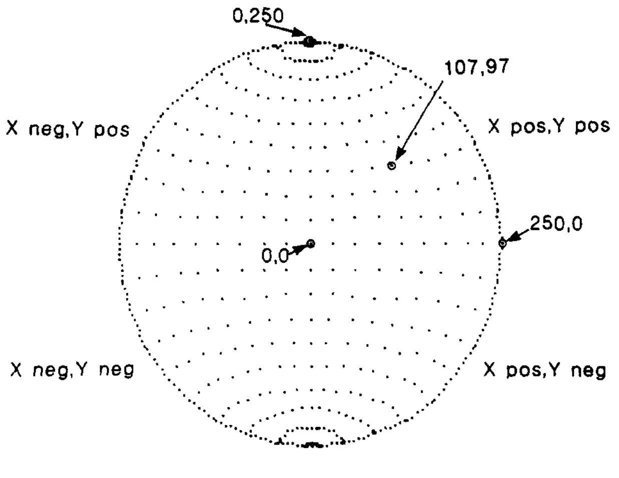

Fig. 1. Screendump of the representation of the equal-area net showing the sign conventions used for plotting the quadrants of the net. Representative co-ordinate pairs of various points (circled) are shown.

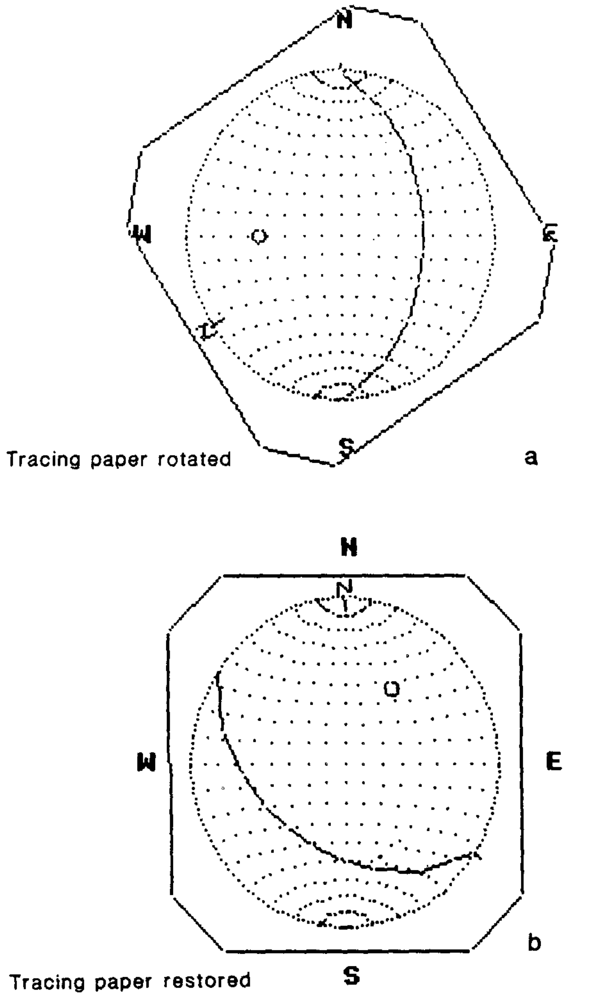

Fig. 2.(a) Mode 4 screendump of net with tracing paper on which the great circle and the pole of the plane 123° 45° (left-hand rule) have been plotted. (b) As (a) but with the tracing paper restored to its original position.

ALGORITHMS FOR PLOTTING DATA

Geometric problems are solved using standard vector methods (see, for example, Liebeck 1969). A matrix is used to store the co-ordinates of all points plotted on the tracing paper. Lines are stored as the co-ordinates of a small square centred on the point of projection and planes as the co-ordinates of points on a great circle with pitches of 0–180° in steps of 20° measured in a clockwise sense (Fig. 2a).

The Cartesian co-ordinates (X, Y) of a point representing the equal area projection of a line are given by:

$$X = r \sin (A)$$
$$Y = r \cos (A),$$

where

$$r = \sqrt{2}R \sin (45 - I/2),$$

R = the radius of the net, I = the plunge and A = the plunge azimuth. The co-ordinates of the point representing a pole are yielded by the same equations where:

$$I = 90 - d,$$
$$A = (180 + a)\text{MODULO } 360,$$

where d = the dip and a = the dip azimuth. All angles are quoted in degrees.

One problem is that these co-ordinates are calculated with respect to north on the net, that is as they would plot when the tracing paper is restored to its initial position (see Fig. 2b). If the tracing paper is rotated so that 'N' no longer coincides with north on the underlying net (e.g. Fig. 2a) these co-ordinates must be transformed by the appropriate rotation matrix before they are plotted.

ANIMATION ROUTINES

Animation of the rotation of plotted points is easily achieved by two methods. The first is to plot an image in the foreground colour and then remove it by replotting it in the background colour. The co-ordinates are then changed and the process repeated. The second method involves selecting a background colour that is the inverse of the foreground colour (e.g. black background with white foreground). The foreground image is then plotted in the inverse of the background colour. It can be removed by replotting the same image in its inverse colour which returns it to background. For example, if a black background was originally present an image plotted in inverse colour would be white. If that image is replotted in inverse colour it will become black again and disappear. McGregor & Watt (1984) give a useful account of these techniques.

In this package the net is stored in a background plane, which is held in the screen memory, and the tracing paper in a foreground plane, which is a numeric matrix within the BASIC program. The second method of animation is used since it allows graphics in the background plane to be 'remembered' as the colour of the line in the foreground image will always be the inverse of the background. Thus when it is again inverted the original background pattern is restored.

The matrix of co-ordinates of all the features is read and plotted. Then it is removed, transformed by the rotation matrix in 10° increments, and replotted in the new position. The BBC micro is sufficiently fast to allow reasonable, if a little jerky, animation to be obtained using this method.

DESCRIPTION OF THE PROGRAMS

The 10 programs listed below form the core of a computer aided learning package covering all the basic operations on the equal-area net.

"LAMBNET": pokes the co-ordinates of the net into memory;

"TUTMENU": is a menu program that selects one of the following;

"LAMBTUT": basic procedures for plotting a great circle;

"LAMPOLE": procedures for plotting a great circle and pole;

"LAMLINE": plots a line given the plunge and azimuth;
"LAMPTCH": plots a line given the plane and the pitch;
"LAMBFIT": finds the plane containing two user defined lines;
"LAMBINT": finds the intersection of two planes;
"LAMBROT": rotation about horizontal axis;
"LAMBINC": rotation about inclined axis.

A facility is included so that an instructor's voice-over, using a standard tape recorder, can be included. These programs were developed as part of a Joint Study Program (JSP-85-244-IRL) held jointly with D. Sanderson, Queen's University, Belfast, who has written a related package for the Apple II.

REFERENCES

Liebeck, H. 1969. *Algebra for Scientists and Engineers*. Wiley, London.

McGregor, J. & Watt, A. 1984. *The Art of Microcomputer Graphics—for the BBC Micro/Electron*. Addison-Wesley, Wokingham, Berkshire, U.K.