VECTOR — A Computer Program That Utilizes SURFER or GRAPHER to Plot a Vector Field

by Ning Lu and Steven Amter

Abstract

VECTOR is a Fortran 77 program that enables SURFER or GRAPHER to rapidly convert a two-dimensional field of vectors into graphical form. SURFER and GRAPHER are both DOS format commercial graphics packages. The resulting graphical output can be plotted on-screen or as hardcopy. Vector arrows and the size of the plot can be either manually or automatically scaled. For input, VECTOR requires only four pieces of data for each point: the x and y coordinates of the point and the x and y components of the vector’s magnitude. VECTOR is an interactive program with defaults and on-screen menus to simplify use.

Introduction

The use of computer models by engineers and scientists to solve ground-water flow problems has become routine, and a vast number of codes are currently in use for this purpose. The output of most models consists of a table listing the head or the specific discharge calculated at each node. Since many flow models do not include options that allow the user to easily represent the output, it is often left to the model user to find the best way to present and analyze the output. Three standard ways to do this are to construct graphical plots of equipotential lines, streamlines, and vectors. At present, microcomputer users, who are doing a significant portion of the modeling work, have a limited number of plotting options available to them. Microcomputer users often resort to plotting equipotential maps and then interpret flow directions manually.

This paper presents a program called VECTOR, which, when used in tandem with the widely used graphical software SURFER or GRAPHER, enables the microcomputer user to convert a field of vectors into a plot in which the vectors are represented by arrows. VECTOR does not calculate vectors; they must be supplied to the program as input.

Program Features

VECTOR produces graphical representations of two-dimensional vector fields. An example of the output is shown in Figure 1. The vector at each point is represented by an arrow pointing in the direction of the flow. The length of the arrow and the size of the arrowhead are proportional to the magnitude of the vector.

Both the length of the vectors and the size of the arrowheads are scaled to the maximum vector that occurs in the field. Vectors whose lengths are very small fractions of the maximum, and are too small to be drawn to actual scale, are represented by only arrowheads which indicate direction. To facilitate interpretation, a legend at the upper right corner of the plot gives the magnitudes of the maximum and minimum vectors occurring in the input data field and the sizes of the arrows used to represent them.

To reduce crowding, the density in which vectors are plotted on the field can also be specified. For example, Figure 2 shows the same field as Figure 1, but vectors have been plotted only for every other nodal point.

The dimensions of the plot are automatically scaled so that the true proportions of the field are maintained. The program assigns a length of one to the field’s longer axis and the shorter axis is scaled to a fractional length of the longer axis. To facilitate determining the coordinates of selected locations on the plot, the minimum and maximum coordinates of the field are shown on the output. For example, in Figure 1 the shorter axis y has been assigned its true length of approximately 26% of the length of the x axis. Thus, the

---

Discussion open until January 1, 1993.
0.5 tick mark on the x axis corresponds to half of the $x_{\text{max}}$ value given on the upper right corner of the plot, or 0.5 (0.18 $\times 10^6$). Since the input data was in cm, this would correspond to 90,000 cm.

Other aspects of the output can be controlled by the modeler. The size of the plot as output is controlled by specifying the length of the longer axis. The vertical axis can be set to point either upwards or downwards. A figure title of up to 80 characters can be entered by the user.

**Description of the Program**

VECTOR is written in standard FORTRAN 77 for use on IBM compatible microcomputers. The source code has been tested with the LAHEY and RM FORTRAN compilers. Some modifications of the code may be necessary with other compilers. The program is dimensioned to accept a maximum of 15,000 data points. Compilation with either the LAHEY or RM FORTRAN compiler yields an executable file that is less than 300,000 K, which should run on most machines. If necessary, the maximum size of the input variable array can be redimensioned easily by changing the dimension statement at line 8 of the source code.

VECTOR can be used with either SURFER or GRAPHER, both of which are widely used commercial graphics programs. For clarity, in the remainder of this paper, it is assumed that SURFER will be used. The code assumes that SURFER will be used and that SURFER is located in the C:/SURFER subdirectory. To use GRAPHER, or if SURFER is in a different subdirectory, statements 350 and 355 of the source code must be changed to contain the appropriate pathname.

As input, VECTOR requires a data file with four pieces
of information for each data point. These are, in order, the horizontal coordinate \( x \), the vertical coordinate \( y \), the \( x \) component of the magnitude, and the \( y \) component of the magnitude. The input data can be of any FORTRAN type (integer, decimal, or exponential form) as long as the data are in the correct order and separated by one or more spaces. Input files must be in ASCII (DOS) format. A sample input file, organized into columns, is shown in Table A1 of Appendix I. Figure A1 shows the vector field that is generated by the sample input.

The program creates an output file that can be used directly by SURFER. The output file can be shown on-screen using SURFER’s VIEW module, or plotted as hard-copy using SURFER’s PLOT module.

**Using VECTOR**

VECTOR is menu driven and interactive. Typing "VECTOR" and pressing ENTER brings up the menu, which is shown in Figure 3. The menu lists and defines the nine commands, and also shows default values of the variables that the program uses. Note that all commands must be entered in capital letters. Details on some of the commands are given below:

**ARROW** — The ARROW command controls the appearance of the vectors. Scaling factors to control the length of the vector and the size of the arrowhead can be specified.

**SIZE** — The SIZE command controls the length of the longest axis of the plot. The smaller axis will be automatically scaled in the correct proportion. The default (and maximum) length is seven inches. Larger or distorted plots can be generated using the options available in the SURFER plot module.

**AXIS** — The AXIS command allows the vertical coordinate system to be switched from the default direction, which is that \( y \) increases upwards, to downwards.

**SKIP** — The SKIP command reduces the number of vectors on the plot by skipping data points. It is used to prevent overcrowded plots. The fraction of data points plotted is controlled by specifying a skip frequency. The default value of 1 specifies that a vector will be drawn at every node point in the field. A frequency of 3 specifies that a vector will be drawn at every third data point. One should be cautious when using the SKIP command. Potentially important vec-

---

**Appendix I**

**Table A1. Sample Input Data File**

<table>
<thead>
<tr>
<th>( x )</th>
<th>( y )</th>
<th>( z )</th>
<th>( w )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>3.3</td>
<td>3</td>
<td>-2</td>
</tr>
<tr>
<td>0.3</td>
<td>2.3</td>
<td>3</td>
<td>-1</td>
</tr>
<tr>
<td>0.3</td>
<td>1.3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>0.3</td>
<td>0.3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>1.3</td>
<td>3.3</td>
<td>1</td>
<td>-3</td>
</tr>
<tr>
<td>1.3</td>
<td>2.3</td>
<td>4</td>
<td>-4</td>
</tr>
<tr>
<td>1.3</td>
<td>1.3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>1.3</td>
<td>0.3</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>2.3</td>
<td>3.3</td>
<td>-1</td>
<td>-3</td>
</tr>
<tr>
<td>2.3</td>
<td>2.3</td>
<td>-4</td>
<td>-4</td>
</tr>
<tr>
<td>2.3</td>
<td>1.3</td>
<td>-4</td>
<td>4</td>
</tr>
<tr>
<td>2.3</td>
<td>0.3</td>
<td>-1</td>
<td>3</td>
</tr>
<tr>
<td>3.3</td>
<td>3.3</td>
<td>-3</td>
<td>-3</td>
</tr>
<tr>
<td>3.3</td>
<td>2.3</td>
<td>-3</td>
<td>-1</td>
</tr>
<tr>
<td>3.3</td>
<td>1.3</td>
<td>-3</td>
<td>1</td>
</tr>
<tr>
<td>3.3</td>
<td>0.3</td>
<td>-3</td>
<td>3</td>
</tr>
</tbody>
</table>

---

**Fig. A1. Vector plot resulting from the sample input data file given in Table 1.**
tors may be missed when skipping. Also, it should only be used when the input to VECTOR is sequentially ordered into rows or columns. Using the SKIP command when the input is unordered may result in the random deletion of vectors on the plot.

**INPUT** — The INPUT command specifies the name of the input data file. The default name is VECTOR.INP. If the input data file is located in a different drive or subdirectory from the VECTOR program, the complete path name (for example, C:\FILES\PUMP.TST) should be provided.

**WRITE** — The WRITE command specifies the name of the output file. The default name is VECTOR.PLT. Output file names should generally have the form NAME.PLT so they can be directly read by the PLOT routine of SURFER. The complete path name should be given to direct the output into a desired drive or subdirectory. For example, the name C:\SURFER\VECTOR.PLT places the output file directly in the SURFER subdirectory.

**PLOT** — The PLOT command executes the main part of the program, generates an output file with the specified parameters, and terminates the VECTOR program. The file is written in a form which can be read by SURFER's PLOT routine.

Once a plot file has been created by VECTOR, it can be viewed in two ways. To see the vector field on-screen, the VIEW routine of SURFER should be used. For example, if an output file from VECTOR called VECTOR.PLT has been placed in the same directory as SURFER, then the command VIEW VECTOR.PLT will bring it on-screen. Alternatively, one can put SURFER in the command path (for example, if the VECTOR program is in a subdirectory called VECTOR, then the appropriate command would be C:\SURFER\VIEW C:\VECTOR\VECTOR.PLT) so that the output can be viewed in the current directory. The same steps are used to print the vector field as hardcopy, except in this case the SURFER PLOT routine is used. The necessary commands would be analogous to those described above for viewing—PLOT VECTOR.PLT or C:\SURFER\PLOT C:\VECTOR\VECTOR.PLT.

Some modelers may wish to convert VECTOR into a subroutine that can be incorporated directly into a flow model. If this is done, it is essential that arrays declared in line 8 of the code have the same size and form as the corresponding arrays in the main program.

An executable copy of VECTOR, the source code, sample input, and sample output are available from the authors for a $15 charge. Please specify disk type preference.

**Acknowledgment**

The authors thank Richard Codell of the Nuclear Regulatory Commission and Michael Wallace of RE/SPEC Inc. for inspiring this work.