topograph

Marc Culler

COLLABORATORS			
	TITLE : topograph		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Marc Culler	June 8, 2025	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

topograph

1.1 Topograph - A Function Browser

1.2 overview

Overview
Topograph is an interactive tool for exploring functions of
two variables. It gives both qualitative and accurate numerical
information about the function. You enter an arbitrary formula
defining a function z = f(x,y) and the program displays a color-coded
topographical map of the graph. (This type of plot is sometimes
called a "pseudocolor" or "density" plot.) The graph is interpreted as
follows. The display area represents a rectangle [Xmin,Xmax] x
[Ymin,Ymax] in the x-y plane. Each pixel in the display area
corresponds to a point on a grid in this rectangle. The value of the
function at a grid point is represented by assigning a color to the
corresponding pixel. You may interactively control the domain;

pressing an arrow key causes the domain to shift, while a
click-and-drag operation with the mouse allows you to "zoom in" to a
subrectangle. You may also zoom out (with a menu operation) or select
a specific domain rectangle by entering values for Xmin, Xmax, Ymin,
and Ymax. The numerical information is provided by displaying the
values of x, y and z = f(x,y) at the location represented by the hot
pixel on the pointer.
As you will discover when you use the program, the topographical
features of the surface z = f(x,y) can be seen much more clearly with
this type of plot than with a perspective drawing of the surface in
3-space. It is not necesary to view the graph from different
positions to get a feeling for its shape. Discontinuities or
singularities do not cause strange artifacts since no attempt is made
to triangulate the surface or to use any other "clever" tricks to
reduce the amount of computation; Topograph simply compute a
function value for each pixel. Finally, the accurate numerical data
provided by Topograph would be very difficult to extract
from a perspective drawing of the graph.
The color scheme is as follows. An interval [Zmin,Zmax] on the line
is specified and divided into 28 equal-sized subintervals. These are
assigned colors according to the light spectrum -- reds, oranges,
yellows, greens, blues, indigos and violets. The smaller values
correspond to red and the larger values to violet. Values which are
smaller than Zmin are displayed as black and those larger than Zmax
are displayed as white.

1.3 running

```
To run Topograph double click on the icon or type topograph
from a CLI.

You will need to be running Workbench 2.0 or higher.  It would not be any fun
to use Topograph without a math coprocessor.  So you must have at
least a 68020 CPU chip with a 68881 coprocessor.

This version of Topograph does not parse command line arguments.
Perhaps one day you will be able to specify a function with its domain and
range, or a function file, on the command line.  Probably there will also
be an ARexx port some day.
```

1.4 Controls

The domain and range intervals can be set with the gadgets in the control panel at the top of the display. The domain can be "zoomed" in by clicking-and-dragging the mouse, and out by using the "Graph" menu. The effect of the click-and-drag operation varies according to the setting of the Aspect cycle-gadget.

When the pointer is on the graph the x, y and z = f(x,y) values at the point are displayed in the gadgets at the bottom of the screen.

Pressing one of the arrow keys causes the domain rectangle to be shifted in the corresponding direction.

When the program begins, or when you select the "New" item in the "Project" menu you are prompted to enter the definition of your function in a window in the center of the screen. You end the function definition with a semi-colon, or two returns. The syntax for these expressions is reasonably natural. The program waits for you to set the domain and range with the gadgets at the top of the screen. When you click on Draw the window vanishes and the graph is drawn on the screen.

Topograph does not automatically adjust the range interval when the domain is changed. However, if you click on Rescale the maximum and minimum values of the function on the specified domain will be placed in Zmax and Zmin respectively. The graph will then be redrawn.

Menu

Project Graph

New Zoom Out

Print Find Extrema

Hide

Quit

Gadgets

The control panel at the top of the screen is used to set the domain and range of the function and to start or stop the drawing of the graph. The close button in the title bar causes the program to exit. After changing the domain or range values you must click on Draw to signal the program to redraw the graph.

There are two modes for zooming in, which are indicated by the Aspect cycle-gadget.

Zooming

When the pointer is positioned over the graph, the values of x, y and z=f(x,y) at the position of the pointer are displayed at the bottom of the screen. A click-and-drag on the graph generates a rectangle which will become the new domain when the mouse button is released. You can abort the zoom by moving the pointer back to where it was at the click. The zoom operation works in a very natural way which, however, is different from most implementations of this sort of thing. The point of the click becomes the CENTER of the rectangle, not a corner. So you should point to the part of the graph that you wish to zoom in on, and expand the rectangle to the size you want. The shape of the rectangle behaves differently, depending on how you have set the Aspect.
The control panel window and the display window each become active as soon as the pointer enters the window.

1.5 examples

```
A classic saddle: f(x,y) = x^2-y^2

A monkey's saddle: f(x,y) = (x-y)*(x+y)*y

A mountain lake: f(x.y) = y^3 -2y + x^2

A discontinuity: f(x,y) = xy/(x^2+y^2)

A non-differentiable function with all directional derivatives equal to 0:
      f(x,y) = x^3*y/(x^4 + y^2)

An alien: f(x,y) = y^4-2y^3+y^2-x^2y +x^4
(Domain: [-2,2]x[-0.7,1.8]   Range: [-0.4,0.25])

A psychedelic fish: f(x,y) = cos(x^3-y^2) - sin(x)cos(y)
      (Domain: [-4,4]x[-2.5,2.5])
```

1.6 license

```
                                License

Copyright ©1996 Marc Culler

Topograph version 1.1 uses some freely distributable code
modules from the NewIFFv39.11 distribution.  These modules were
written by Carolyn Scheppner and copyrighted by Commodore Business
Machines.  They were submitted to the Fred Fish collection and appear
on disk number 985.  They are used in Topograph v. 1.1 to write
ILBM IFF files.  The copyright notice above applies to the remainder
of Topograph v. 1.1.
```

Topograph version 1.1 is distributed under the terms of the
GNU General Public License. The distribution consists of the executable
program (Topograph), this file (Topograph.guide), and a copy of the
GNU General Public License (LICENSE). These three files constitute
the "program" for the purposes of the license, and all three must be
included in any redistribution of the program.

To obtain copies of the source code under the terms of the GNU General
Public License, write to

    Marc Culler
    Department of Mathematics (M/C 249)
    University of Illinois at Chicago
    851 S. Morgan St.
    Chicago, IL 60607-7045

or send e-mail to culler@math.uic.edu.

Topograph version 1.1 is provided "as is" without warranty of
any kind. See section 12 of the GNU General Public License for details.

1.7 author

About the Author

I am a Professor of Mathematics at the University of Illinois at
Chicago. My research is primarily on the topology and geometry of
3-dimensional manifolds. I got my first Amiga 1000 in 1985 and have
been an Amiga user ever since. Currently I have an A2500. This
program was written as a tool to help students to understand functions
of several variables.

I can be reached at:

Marc Culler
Department of Mathematics (M/C 249)
University of Illinois
851 S. Morgan Street
Chicago, IL 60607

culler@math.uic.edu

Please write to me if you find an interesting use for Topograph,
or if you find a bug.

1.8 menu-new

Selecting the "New" menu item reopens the input window so you can
enter a new function definition. The old function is forgotten.

1.9 menu-print

The "Print" menu item has three subitems:

Selecting "To Printer" sends the graph to the printer.device as a graphic file. A "cancel" button appears in the title bar at the top of the screen and remains until the printing completes, or is cancelled.

Selecting "As ILBM iff" writes an ILBM iff file with icon. You are given a file requester to choose the file name. The icon is set with MultiView as the default tool if your Kickstart environment variable shows a version above 40. Otherwise the default tool is Display.

Selecting "As PS file" writes a PostScript file. The colors are converted to gray scale. However, it is difficult to distinguish the gray scales in most pictures, so the image is altered to highlight the boundaries between the different grays. The effect is like a level curve map with regions betwen the levels shaded according to height. A minimal effort is made to compress the PostScript bitmap data using a byte-run compression scheme. This scheme was much more effective before I added the feature to highlight the levels, but it is still better than nothing.

1.10 details

Topograph is a multi-threaded application. It runs as two independent tasks. One task manages the control panel and the other does the work of computing and drawing the graph.

Perhaps you have also noticed that it does these calculations rather quickly. Remember, it is actually computing a function value for each pixel in the display area. For example, take f(x,y) = exp(cos(x)*sin(y)). An evaluation of the value of this function at a single point requires a minimum of 1 fmul, 1 fcos, 1 fsin, and 1 fetox instruction. (This ignores such things as converting the double precision answer to a pixel value and writing that pixel to the screen.) According to my 68881 manual, the typical execution times for these instructions (assuming that the source is a floating point register) are:

```
fmul    71 clocks
fcos   391 clocks
fsin   391 clocks
fetox  497 clocks
```

Thus the computation should require well above 1350 clocks per pixel. I have about 60000 pixels in my display area, so it should take a minimum of 810,000,000 clocks to paint the screen. On my 14.3 megaHertz A2500 that would be about 57 seconds. My wrist watch says that it takes 9 seconds to compute the complete graph of this function.

The main thing that accounts for the speed is that Topograph works by incremental compilation. It contains a mini-compiler which parses the expression that defines f(x,y) and generates the code for a

C function that evaluates it. The code is written to dynamically allocated memory and linked into the Topograph program. Most of the graphing programs that I have seen use an interpreter to evaluate the expression. The incremental compilation avoids the cosiderable overhead of the interpreter and maximizes the opportunity for the coprocessor to run in parallel with the CPU.

Finally, I thought it was important to have as much vertical resolution, i.e. as many colors, as possible. (At least as many as possible on my ECS machine.) So I decided to use a low-res screen for displaying the graph. To make the menus and text gadgets look decent I use hi-res screens for the control panel and the window at the bottom that reports x, y and z values. That is why Topograph uses three screens. If you ever find the display looking strange, hit left Amiga M a time or two to find the other parts of the display.

1.11 menu-hide

Selecting the "Hide" menu item causes all three Topograph screens to move to the back in succession. (Yes, there really are three screens. If you want to know why, read about the programming details .

1.12 menu-quit

Selecting the "Quit" menu item does what you expect. You are asked to verify that you want to terminate the program. If you say yes, it quits.

1.13 menu-zoom

You may enlarge the domain by a factor of 2, 4 or 8. The center and aspect ratio of the current domain are preserved by this operation.

1.14 menu-extrema

When the domain is scrolled with the arrow keys, Topograph stops keeping track of the maximum and minimum values of the function. To do so would require checking values over the entire domain, which would slow the scrolling considerably. So when you finish scrolling and would like to know the extrema, select this item.

1.15 domain_range

Click on the text entry gadgets and edit in the usual way to change
the values of Xmin, Xmax, Ymin, Ymax. You can tab through the gadgets.
Meaningless values are ignored. Remember that these gadgets do not
report changes until you hit return or tab.

If you have the Aspect cycle-gadget set to "fixed" then
a change in Xmin or Xmax (Ymin or Ymax) causes Ymin and Ymax (Xmin
and Xmax) to be recomputed to preserve the aspect ratio. This
recomputation preserves the coordinates of the center of the display.

1.16 draw_stop

The draw button starts the computation of the graph using the domain
and range information specified in the control panel. You may stop
the computation by pressing the stop button, which becomes active when
a computation is in progress. If the computation is stopped the portion
of the graph which was not computed is hashed.

1.17 rescale_button

The Rescale button computes the maximum and minimum values of the
function and copies them to Zmin and Zmax. The graph is then redrawn.
This gives maximum resolution in the sense that the graph uses the full
spectrum of colors.

1.18 aspect_gadget

The Aspect cycle gadget has two states - fixed and free. This
controls what happens when you use the mouse to zoom to a smaller
domain. When the aspect is fixed the ratio of height to width for the
zoom box is kept the same as that of the full display area. Thus the
new domain will be similar, in the geometric sense, to the old one.
If you leave the aspect ratio set to 1.0, then you will be seeing an
honest picture of the function.

There are occasional times when some distortion is helpful. In such a
case you can set the cycle gadget to free. You will then be allowed
to select a long skinny domain. The function will be displayed with
the new domain distorted to fill the entire display area.

1.19 syntax

Topograph recognizes the following symbols:

Binary Operators:
+ addition
- subtraction

```
 * multiplication
 / division
 ^ power

Unary Operators:
 -        unary minus
 neg      unary minus
 sin      sine
 cos      cosine
 tan      tangent
 arccos   inverse cosine
 arcsin   inverse sine
 arctan   inverse tangent
 exp      exponential function
 log      (natural) logarithm
 cosh     hyperbolic cosine
 sinh     hyperbolic sine
 tanh     hyperbolic tangent
 arctanh  inverse hyperbolic tangent
 sqrt     square root

Constants
 e        ~2.7182818284590452354
 pi       ~3.1415926535897932385
```

Expressions may also include constants in decimal form.

Expressions may continue on multiple lines. Use a semicolon or two successive carriage returns to terminate the expression.

I have attempted to follow standard mathematical conventions in designing the syntax. Thus parentheses can be omitted when a unary operation is applied to an expression with only one part, and missing binary operators are assumed to be multiplications. For example, the following expressions are equivalent

```
sin(x)*cos(y)
(sin x)(cos y)
sin x cos y
```

Spaces are required to delimit tokens, so expressions such as sinx or xy ↩ will generate error messages.

If you make a syntax error or use an unknown symbol Topograph will indicate the error and allow you to retype the expression. The editing, history and cut-and-paste features of the shell can be used when entering the function.