

# 3-D Functions v2.2.1

## Visualization of $z = f(x, y)$



By  
Hussam Dandashli  
319 Gage Ave  
Elkhart, IN 46516  
e-mail: [hussamd@skyyenet.net](mailto:hussamd@skyyenet.net)

## Introduction

Functions of two variables are difficult to reproduce on paper because they exist in 3-D space. 3-D Functions supplies the user with different functions to choose from, to add or multiply or divide up to 10 function combinations into a special function. Also supplied is a bunch of preset functions and one nested function.

Version 2 adds the option of contour plots and constraint functions that can be used in optimization problems, where the user has to identify the region where the function is at an minimum value, etc...

Version 2.2 added the feature of producing a quicktime movie of a rotating surface. The current plot position is rotated 360 degrees about the z-axis in one degree increments.

## Functions

The application supports three types of functions:

### (1) Preset Functions:

**Sin(x)\*Cos(y)**  
**Cos(x)\* Cos(y)**  
**Sin(x)\*Sin(y)**  
**Sin(x)\*Exp(y)**  
**Cos(x)\*Exp(y)**  
**Exp[-x^2 - y^2]**  
**Sin(xy)**  
**Cos(xy)**  
**Cos[x + Cos(y)]**  
**Cos[x + Sin(y)]**  
**Sin[x + Cos(y)]**  
**Sin[x + Sin(y)]**

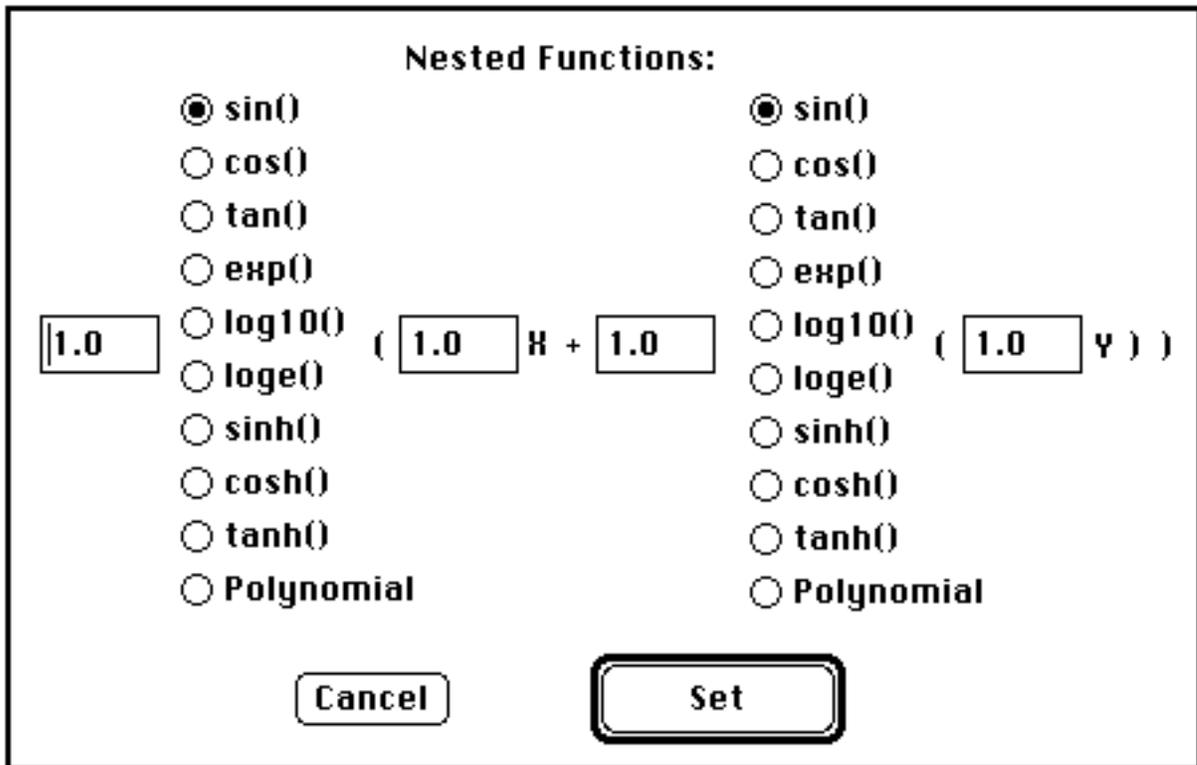
---

**Figure 1.** Preset functions

These functions are supported for a quick plot to visualize these surfaces.

### (2) Nested Function:

This is a step up from the preset functions. The user will be supplied with a dialog box to select the desired function:

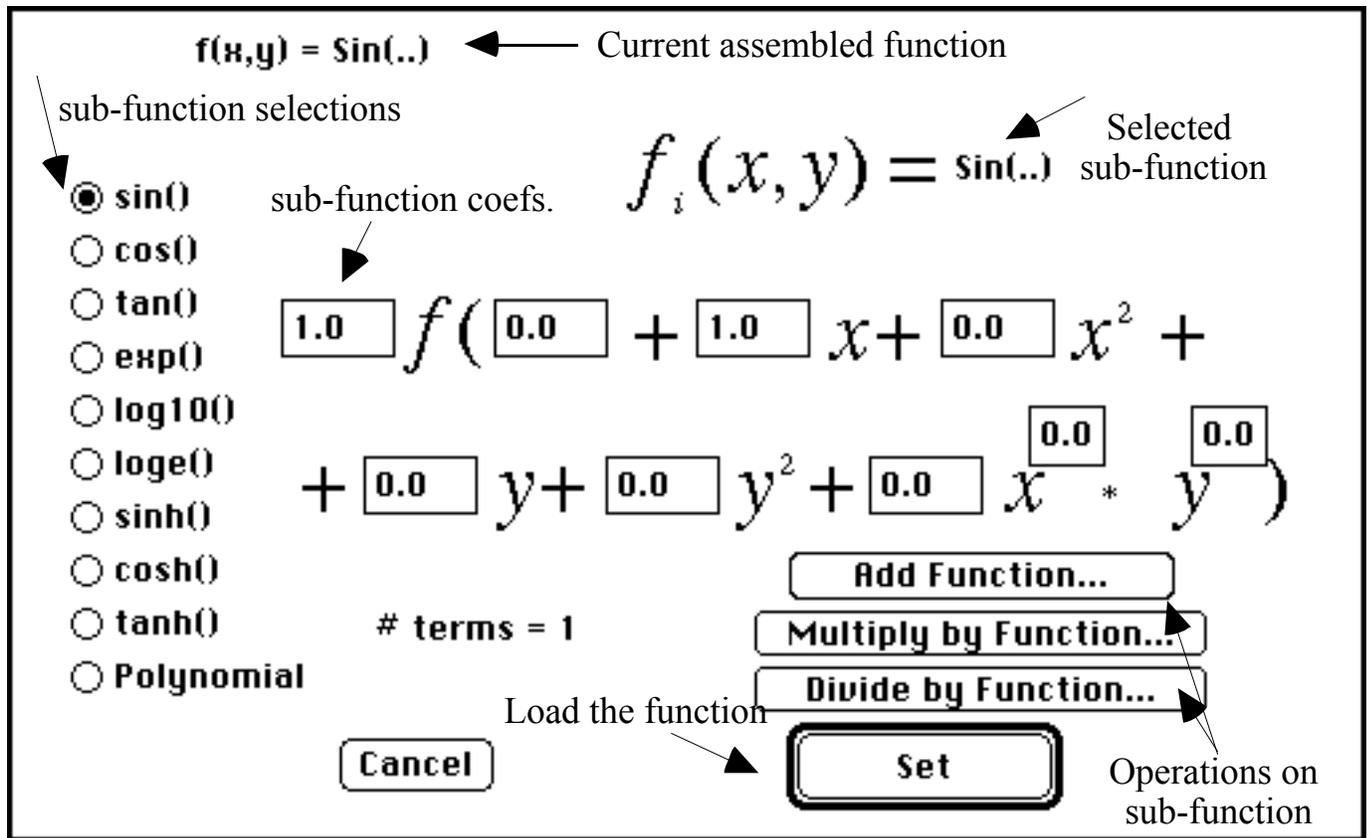


**Figure 2.** Nested function dialog.

The radio buttons will select the two functions, and the coefficients are entered in the text fields. This type of function is still limited as far as selection goes.

**(3) Flexible Function:**

This type of function can comprise of up to ten combinations, added and/or multiplied. The user will be prompted to assemble the function:



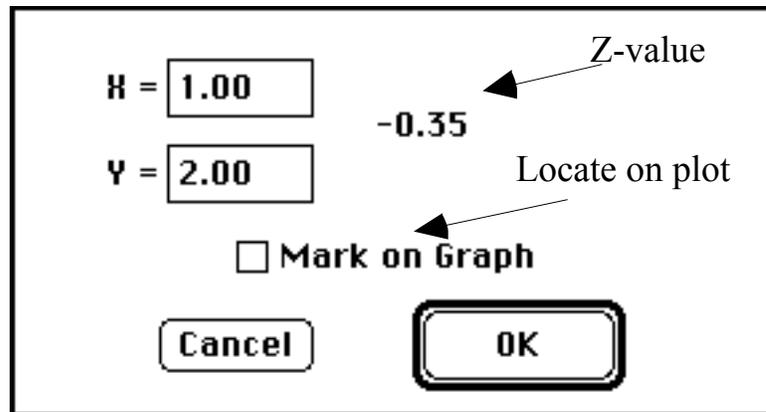
**Figure 3.** Assembling a function

If more functions are desired, select the Add/Multiply... buttons to append another term to the function. Note that the polynomial function is the same as the argument returned. Also note that **Log ( )** and **Log10( )** functions return negative infinity if its argument is zero or negative.

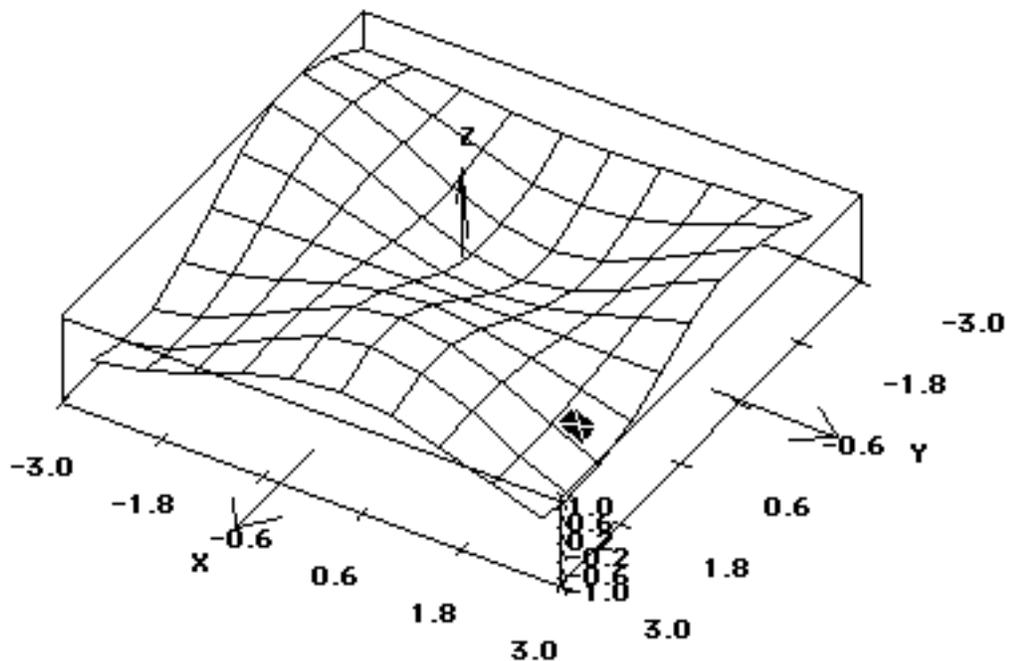
**(4) Constraint Function(s):**

The user will be prompted to enter the number of constraint equations (up to 9 functions) of the form  $f(x, y) = 0$ . The user can enter the required equation in a dialog box similar to the one in Figure 3 above. The only difference is that the constraint function is one part only, i.e. no addition or multiplication with other terms.

At any time, the z-value at any point can be calculated by entering the x and y values in a dialog box from menu "Function", item "Solve for a Z Value...", shown in Figure 4. If the "Mark on Graph" box was checked, a patch will be placed on the plot at the corresponding coordinates as in Figure 5.



**Figure 4.** Calculating a Z Value



**Figure 5.** Marking a point on the plot.

The application allows some flexibility in the way it represents a function. Some of the main editable items are presented in a dialog box from menu "Functions" item "Attributes..." shown in Figure 6. Mesh size, color or gray scale, and limiting z-values are some of the major features. The resulting plot will look similar to Figure 7 (with 20 mesh points (20 x 20 patches)). The higher that number is, the finer the mesh, and the memory consumption. If a screen bigger than 13" is used, more RAM might be needed if the free memory falls below 200K.

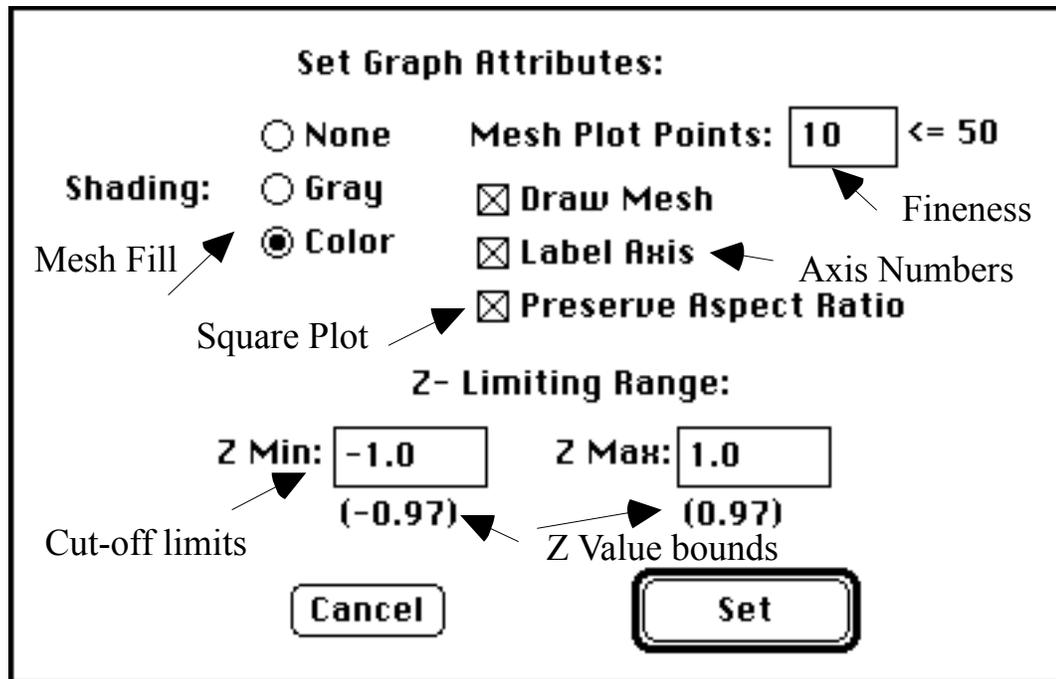


Figure 6. Setting plot attributes.

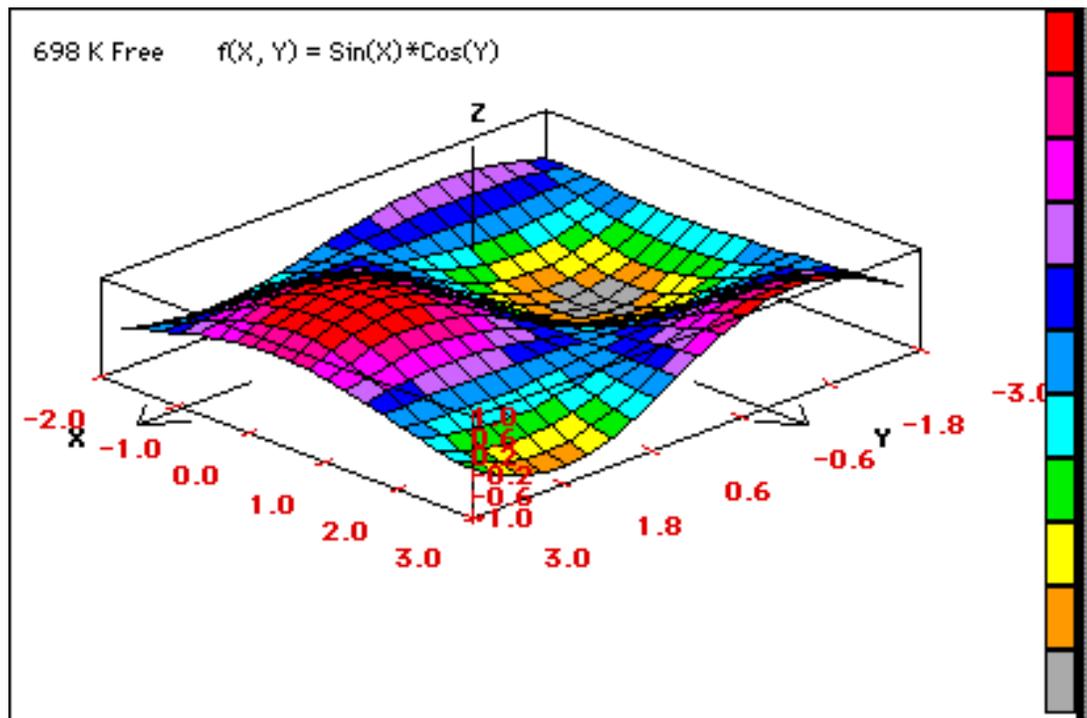


Figure 7. Sample screen with 20 mesh points and color.

The user can specify the desired range of the x and y axis. The dialog box of Figure 8 will come up automatically for new functions. It can also be accessed from the "Functions" menu item "Set X-Y Range..."

**Specify range of X and Y values:**

	Min.	Max.
X-axis:	-3.0	3.0
Y-axis:	-2.0	3.0

**Set**

**Figure 8.** Specifying desired axis range.

### 3-D Aspects

In a 3-D environment, the user must specify four angles:  
View, roll, pitch, and yaw angles to visualize a certain region of the plot.

The view angle has four settings that can be selected from a dialog box in menu "Functions" item "Set View Angle..." shown in Figure 9.

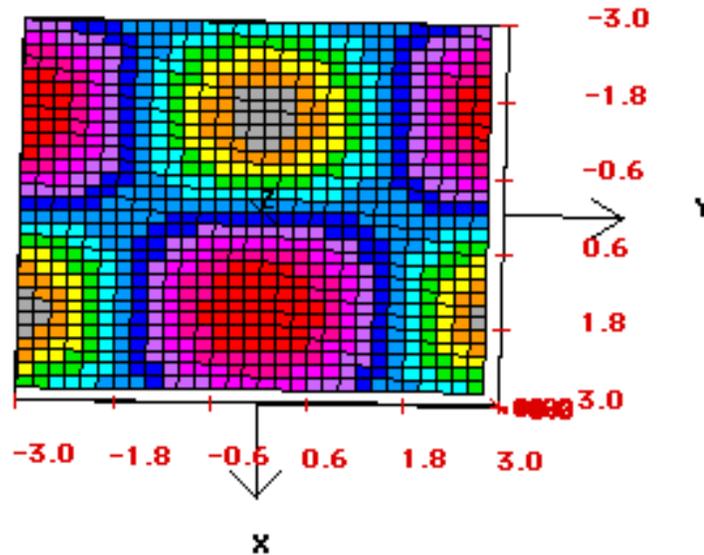
**Set View Angle:**

- No Perspective (0 deg.)**
- Telephoto Lens (10 deg.)**
- Human Eye (25 deg.)**
- Wide Angle Lens (80 deg.)**

**Cancel**      **Set**

**Figure 9.** Setting View Angle.

The yaw angle is fixed for the sake of clarity. The rotations get very confusing when rotating all three axis at the same time. By doing so, the rotations are isolated to rotations about the z-axis (360 degrees in x-y plane) so the user can look at the plot from all directions. The last angle, pitch angle, allows the user to look straight at the plot all the way to a top view. Note that the top view in the color mode can simulate a shaded contour plot showing regions sharing the same z-values, demonstrated in Figure 10.



**Figure 10.** Top view of Figure 7 doubles as a contour plot

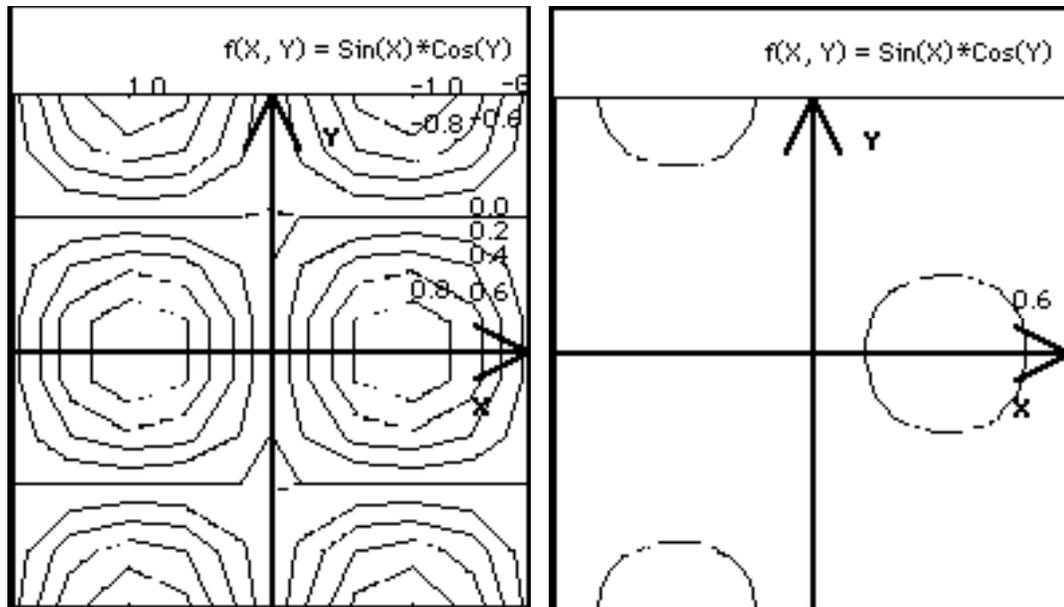
The rotations are done interactively by pressing the mouse button in the plot window and drag the mouse, with the button still down, left and right or up and down or any combination to arrive at the desired view.

If, for any reason, the plot is too small or not centered, this could be remedied from the "Zoom" menu. Note that the "Zoom Rectangle.." item allows the user to specify a region to fill the window with. Also note that due to round off errors some distortion might appear at the edges of the window.

Also the screen background color could toggle between black and white.

## Optimization of a Function

Sometimes it is desirable to find the regions of a function that shares a certain value, or an optimum value. Take as an example a function of the form :  $f = \sin(x) * \cos(y)$ , pictured in Figures 7 and 10. Let's assume that we need to find the regions where the function value is equal to 0.6. The contour plot has the options of drawing multiple function values ("Drawing Mode") like in Figure 11, or just the function desired value (the optimum point) entered in the dialog box of Figure 12. When the application is in the "Mouse Mode", when the user selects a point in the Countour Plot Window, the application finds the x and y values an the corresponding function value, then it procedes to find the regions that share the same function value.



**Figures 11.** Contour Plots in Drawing and Mouse Modes

**Contour Window**

**Color Contours**  
 **Draw Grid**  
 **Label Axis**

**Desired Function Value:** 
Target Value

**Set Function Range:**  
 (ten contours)

**max F:**

**min F:**

**Step Size:** 
Target Value

**Figure 12.** Setting up contours attributes

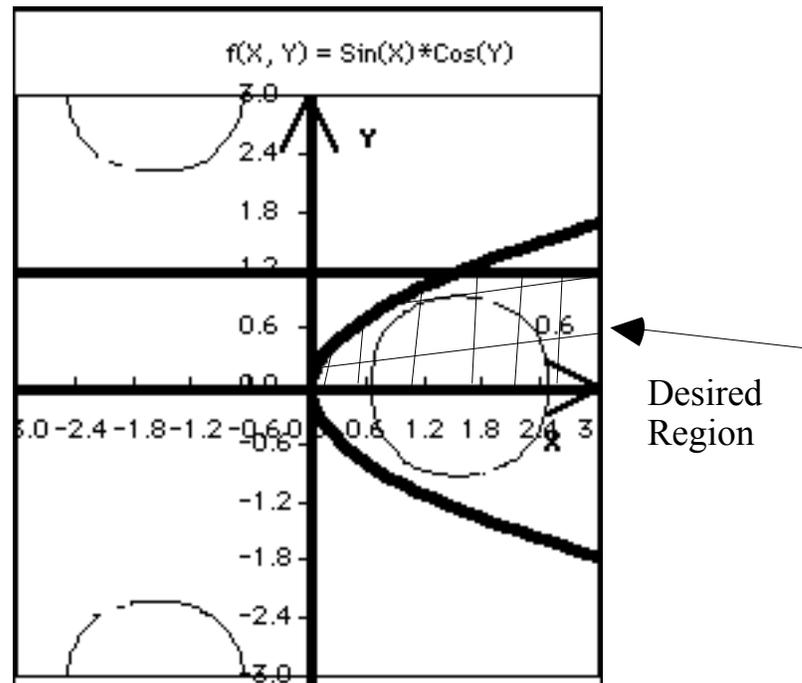
Let's assume that we are interested in finding an optimum (minimum) value of the function subjected to certain constraints. After entering the constraints, the application will draw them and the user can then identify the region that would satisfy the constraints and identify the desired optimum point. Note that to find the minimum of a function, just enter "0" in the field in Figure 12, labeled "Desired Function Value:". Figure 13 demonstrates this concept:

Problem statement: Minimize  $f(x, y) = \sin(x) * \cos(y) = 0.6$ , subject to the following constraints:

C1:  $x - y^2 = 0$

C2:  $x \geq 0$   
C3:  $y \leq 1.2$   
C4:  $y \geq 0$

the contour plot (in Mouse Mode) will reflect these constraints in thick lines and will narrow down the desired region where the optimum value(s) exist (if ever).



**Figure 13.** Finding the optimum solution region

## Input / Output

The plot, or a part of it, can be copied to the clipboard by selecting menu item "Copy..." from "Functions" menu. The user will be prompted to select a region to copy.

The plot window could also be printed or exported as a PICT file to be used by other drawing/painting applications. A movie could also be created to visualize the surface from all directions by rotating it 360 degrees. Use the menu item in the file menu.

Another method to save a plot is to save all the parameters into an input file, which is very small. The user could at any time choose to load that function with all of its attributes and parameters. All of this is done from the "File" menu.

See the menu section for a brief summary of the menus and their functionality.

# Menus

	
<b>About... ⌘?</b>	Show author and credits
<b>Help... ⌘H</b>	Load an on-line manual

<b>File</b>	
<b>Open... ⌘O</b>	Load an input file
<b>Save... ⌘S</b>	Save current parameters as an input function
<b>View...</b>	View any TEXT file in the console window
<b>Copy... ⌘C</b>	Copy a specified region to the clipboard
<b>Print... ⌘P</b>	Print the front window
<b>Export as PICT...</b>	Export the plot window as a PICT file
<b>Rotating Surface Movie...</b>	Create a quicktime movie
<b>Quit ⌘Q</b>	Exit application

<b>Functions</b>	
<b>Graph Attributes...</b>	Specify plot attributes (see Figure 6)
<b>Preset Function... ▶</b>	Choose a function from a list (see Figure 1)
<b>Assemble a Function... ⌘1</b>	Build a custom function (see Figure 3)
<b>Nested Function... ⌘2</b>	Specify nested function (see Figure 2)
<b>Contour Plot... ▶</b>	
<b>Set View Angle...</b>	Choose a View angle (see Figure 9)
<b>Set X-Y Range...</b>	Specify the desired range for the function (Fig. 8)
<b>Solve for a Z Value...</b>	Get a Z value for a specific X and Y values. Also can mark its location on the plot.
<b>Contour Attributes...</b>	Set contour attributes. (See Figure 12)
<b>Mouse Mode</b>	Toggle Mouse/Drawing Mode.
<b>Setup Constraints...</b>	Enter constraint equations (see Figure 3)

## Zoom

<b>Initial Zoom</b>	<b>⌘I</b>
<b>Closer</b>	<b>⌘]</b>
<b>Farther</b>	<b>⌘[</b>
<b>Move UP</b>	<b>⌘U</b>
<b>Move Down</b>	<b>⌘D</b>
<b>Move Left</b>	<b>⌘L</b>
<b>Move Right</b>	<b>⌘R</b>
<b>Zoom Rectangle...</b>	<b>⌘Z</b>

Modify location and size of the plot or the contour

## Window

<b>3-D Function</b>
<b>Contour Plot</b>
<b>Console</b>
<b>Redraw Function</b>
<b>Invert Screen</b>

Bring corresponding window to the front

Recalculate the points of a function (refresh)

Toggle background color between black & white