Operations that Work on Functions

Some Igor operations work on functions rather than data in waves. These operations take as input one or more functions that you define in the Procedure window. The result is some calculation based on function values produced when Igor evaluates your function.

Because the operations evaluate a function, they work on continuous data. That is, the functions are not restricted to data values that you provide from measurements. They can be evaluated at any input values. Of course, a computer works with discrete digital numbers, so even a "continuous" function is broken into discrete values. Usually these discrete values are so close together that they are continuous for practical purposes. Occasionally, however, the discrete nature of computer computations causes problems.

The following operations use functions as inputs:

- **IntegrateODE** computes numerical solutions to ordinary differential equations. The differential equations are defined as user functions. The IntegrateODE operation is described under **Solving Differential Equations** on page III-322.
- **FindRoots** computes solutions to f(x)=a, where a is a constant (often zero). The input x may represent a vector of x values. A special form of FindRoots computes roots of polynomials. The FindRoots operation is described in the section **Finding Function Roots** on page III-338.
- **Optimize** finds minima or maxima of a function, which may have one or more input variables. The Optimize operation is described in the section **Finding Minima and Maxima of Functions** on page III-343.
- **Integrate1D** integrates a function between two specified limits. Despite its name, it can also be used for integrating in two or more dimensions. See **Integrating a User Function** on page III-336.

Function Plotting

Function plotting is very easy in Igor, assuming that you understand what a waveform is (see **Waveform Model of Data** on page II-62) and how X scaling works. Here are the steps to plot a function.

1. Decide how many data points you want to plot.
2. Make a wave with that many points.
3. Use the SetScale operation to set the wave's X scaling. This defines the domain over which you are going to plot the function.
4. Display the wave in a graph.
5. Execute a waveform assignment statement to set the data values of the wave.

Here is an example.

```
Make/O/N=500 wave0
SetScale/I x, 0, 4*PI, wave0     // plot function from x=0 to x=4π
Display wave0
wave0 = 3*sin(x) + 1.5*sin(2*x + PI/6)
```