Additional facilities for displaying multi-dimensional waves in Igor Pro are provided by the Gizmo extension, which create surface plots, slices through volumes and many other 3D plots. To get started with Gizmo, see **3D Graphics** on page II-405.

It is possible to graph a subset of a wave, including graphing rows or columns from a multidimensional wave as traces. See **Subrange Display** on page II-321 for details.

Analysis on Multidimensional Waves

Igor Pro includes the following capabilities for analysis of multidimensional data:

- Multidimensional waveform arithmetic
- Matrix math operations
- Image processing
- Multidimensional Fast Fourier Transform
- The **MatrixOp** operation

There are many analysis operations for 1D data that we have not yet extended to support multiple dimensions. Multidimensional waves do not appear in dialogs for these operations. If you invoke them on multidimensional waves from the command line or from an Igor procedure, Igor treats the multidimensional waves as if they were 1D. For example, the Smooth operation treats a 2D wave consisting of n rows and m columns as if it were a 1D wave with n*m rows. In some cases the operation will be useful. In other cases, it will make no sense.

Multidimensional Wave Indexing

You can use multidimensional waves in wave expressions and assignment statements just as you do with 1D waves (see **Indexing and Subranges** on page II-76). To specify a particular element of a 4D wave, use the syntax:

```
wave[rowIndex][columnIndex][layerIndex][chunkIndex]
```

Similarly, to specify an element of a 4D wave using *scaled* dimension indices, use the syntax:

```
wave(xIndex)(yIndex)(zIndex)(tIndex)
```

To index a 3D wave, omit the chunk index. To index a 2D wave, omit the layer and chunk indices.

rowIndex is the number, starting from zero, of the row of interest. It is an unscaled index. *xIndex* is simply the row index, offset and scaled by the wave's X scaling property, which you set using the SetScale operation (Change Wave Scaling in Data menu).

Using scaled indices you can access the wave's data using its natural units. You can use unscaled or scaled indices, whichever is more convenient. column/Y, layer/Z and chunk/T indices are analogous to row/X indices.

Using bracket notation tells Igor that the index you are supplying is an unscaled dimension index. Using parenthesis notation tells Igor that you are supplying a scaled dimension index. You can even mix the bracket notation with parenthesis notation.

Here are some examples:

```
Make/N=(5,4,3) wave3D = p + 10*q + 100*r
SetScale/I x, 0, 1, "", wave3D
SetScale/I y, -1, 1, "", wave3D
SetScale/I z, 10, 20, "", wave3D
Print wave3D[0][1][2]
Print wave3D(0.5)[2](15)
```

The first Print command prints 210, the value in row 0, column 1 and layer 2. The second Print command prints 122, the value in row 2 (where x=0.5), column 2 and layer 1 (where z=15).