

As before, the coefficient wave can be either real or complex, and the independent variables can be real or complex. The independent variables must be either all real-valued or all complex-valued.

### Complex All-At-Once Fitting Function

In certain cases, a fit function may need to produce all model values at one time. You can write an all-at-once fitting functions that implement complex-valued functions:

```
Function AAOFitFunc(Wave pw, Wave/C yw, Wave xw) : FitFunc  
    yw = <expression involving pw and xw>  
End
```

Unlike the basic format, the function does not return a complex value and is not declared with Function/C. Instead, the all-at-once fit function returns the complex model values to Igor via the yw wave which you must declare as complex using /C.

As with the basic format, you can make the coefficient wave pw complex or real, and you can make the independent variable wave xw complex or real, as long as the type matches the waves you pass to the FuncFit operation.

All-at-once fitting functions can be multivariate functions, too, by adding more x input waves. As with the basic format, independent variable inputs must be all either real or complex.

### Other Waves

A variety of other waves may be used during a fit: weighting, masking, epsilon, and residuals waves, and a destination wave to receive the model values of the fit solution. All of these waves must be complex or real depending on the nature of the fitting function.

Weighting applied to the dependent variable values must always be complex because a complex fitting function by definition returns complex dependent variable values. Residual and destination waves must be complex for the same reason.

If you use an epsilon wave, it must match your coefficient wave.

Mask waves must be real; they select data points, and cannot select just the real or imaginary part of your input data.

If you are fitting with errors in both dependent and independent variables, the X weighting waves must match the type of your independent variable inputs.

## Curve Fitting with Multiple Processors

If you are using a computer with multiple processors, you no doubt want to take advantage of them. Curve fitting can take advantage of multiple processors in two ways:

- Automatic multithreading
- Programmed multithreading

### Curve Fitting with Automatic Multithreading

If you use the built-in fit functions or thread-safe user-defined fit functions in the basic or all-at-once formats, Igor automatically uses multiple processors if your data set is “large enough”. The number of points required to be large enough is set by the **MultiThreadingControl** operation.

There are two keywords used by MultiThreadingControl for curve fitting. The **CurveFit1** keyword controls automatic multithreading with built-in and basic format user-defined fit functions. The **CurveFitAllAtOnce** keyword controls automatic multithreading with user-defined fit functions in the all-at-once format. The appropriate setting of these limits depends on the complexity of the fitting function and the nature of your problem. It can be best determined only by experimentation.