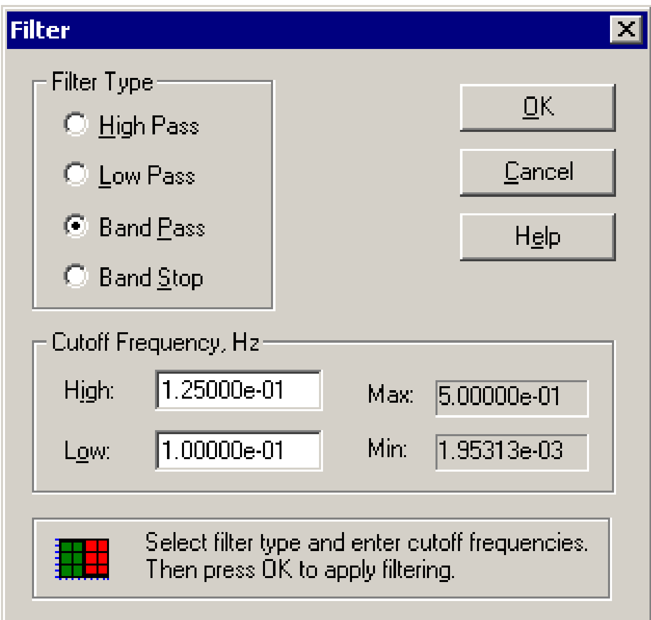
The Stable32 Filter Function

W.J. Riley
Hamilton Technical Services

• Introduction

Stable32 Version 1.54 and above includes a Filter function that can apply low pass, high pass, band pass, and band stop filtration to time domain phase or frequency data. This filtering can be used to preprocess the data before analysis and to otherwise investigate its properties. The filtering operation is very simple, requiring only the selection of a filter type and the desired cutoff frequency or frequencies, as shown in the following figure.



Control	Type	Description
Filter Type	Group	Filter types
High Pass	Radiobutton	Select high pass filter
Low Pass	Radiobutton	Select low pass filter
Band Pass	Radiobutton	Select band pass filter
Band Stop	Radiobutton	Select band stop filter
Cutoff Freq	Group	Cutoff frequencies, Hz
High	Edit	Enter upper cutoff freq
Low	Edit	Enter lower cutoff freq
Max	Text	Max Fourier frequency
Min	Text	Min Fourier frequency
OK	Pushbutton	Perform filtration
Cancel	Pushbutton	Cancel filtration
Help	Pushbutton	Show help
Prompt	Icon & Text	User information

Stable32 Filter Function Dialog Box

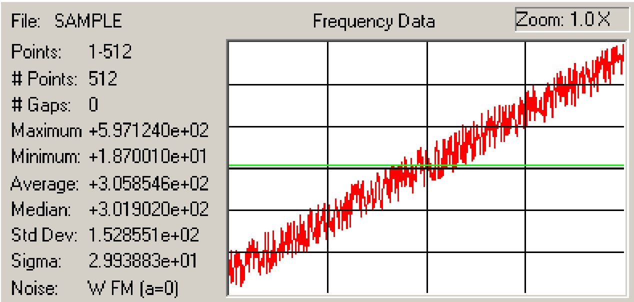
• Purpose

The primary purpose of the Stable32 filtering function is exploratory as an additional way to characterize phase and frequency data and thereby gain further insight into their properties. Filtering can also provide a means of preprocessing data before analysis, but it should be realized that such preprocessing is relatively unusual and must therefore be used with care. For example, high pass filtration can remove drift from frequency data but it is generally better to remove it by means of a more common functional fit that provides better information. Nevertheless, it is hoped that the filtering capability will prove useful.

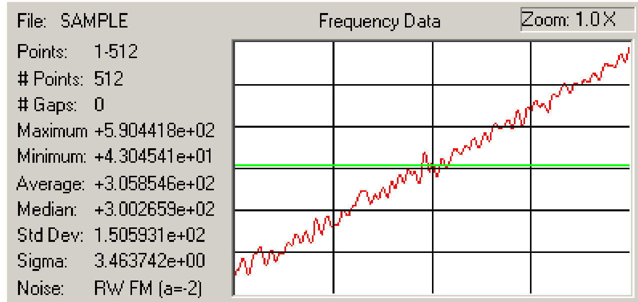
• Examples

The following figures show examples of the four Stable32 filter types applied to the SAMPLE.DAT frequency data file included with the Stable32 software package. The filtering parameters apply to the original data set, not the extended data described below.

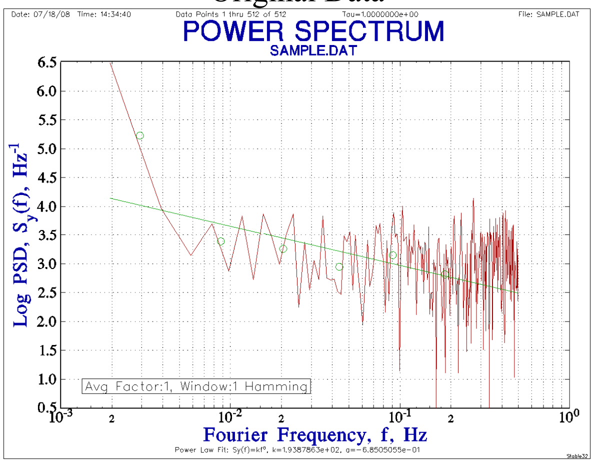
Low Pass Filtration with $f_{cutoff} = 0.1$ Hz $= 0.098\ f_{nyquist}$



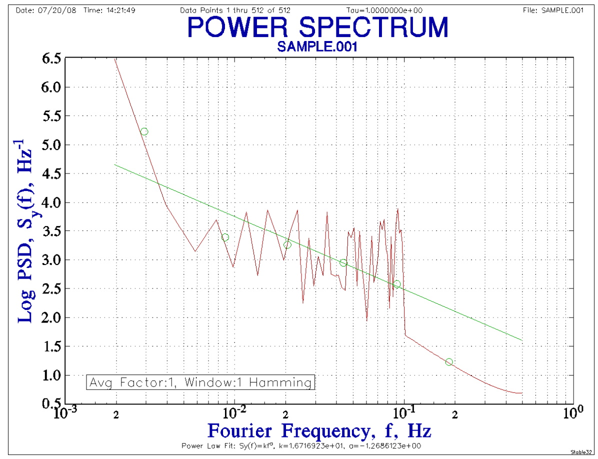
Original Data



Filtered Data

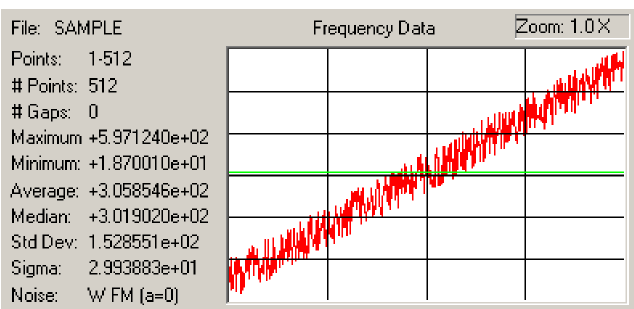


Original Spectrum

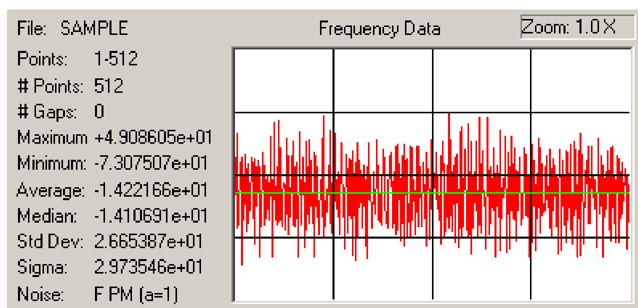


Filtered Spectrum

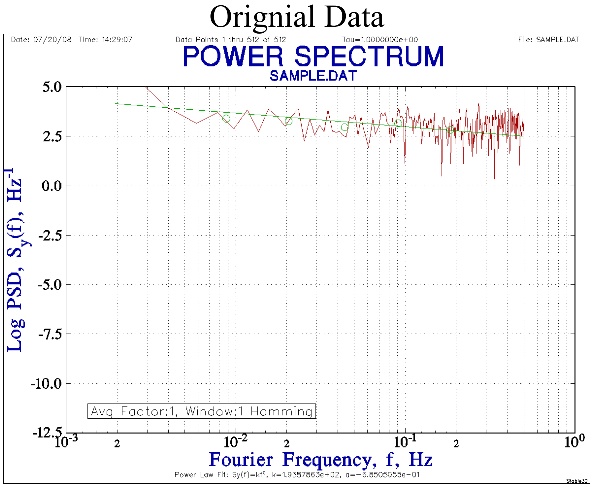
High Pass Filtration with $f_{cutoff} = 0.1$ Hz $= 0.098\ f_{nyquist}$



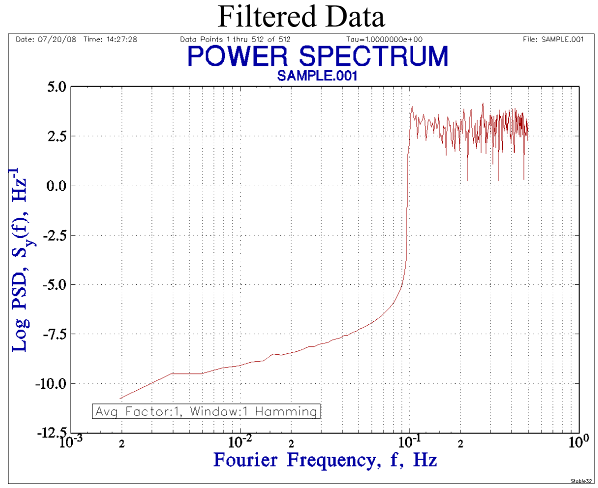
Origrial Data



Filtered Data

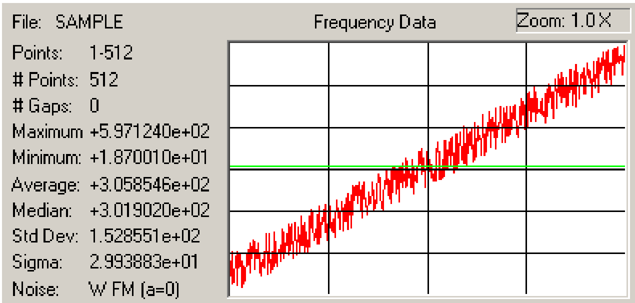


Original Spectrum

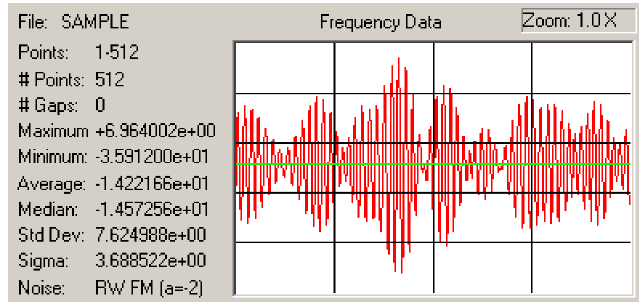


Filtered Spectrum

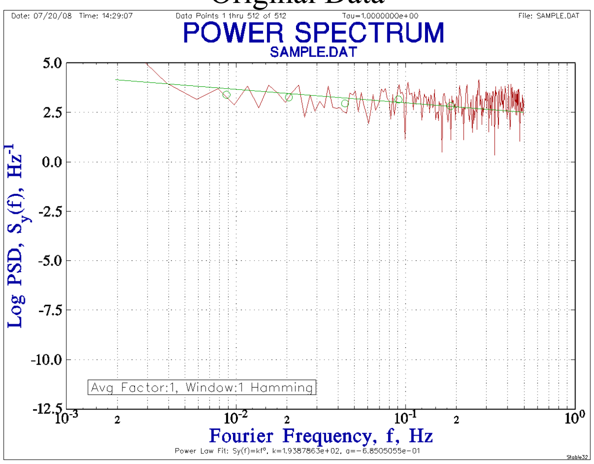
Band Pass Filtration with $f_{lower\ cutoff} = 0.1$ Hz $= 0.098\ f_{nyquist}$ and $f_{upper\ cutoff} = 0.125$ Hz $= 0.122\ f_{nyquist}$



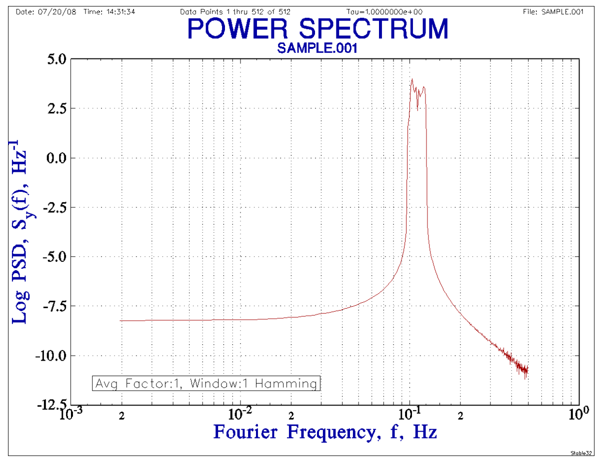
Original Data



Filtered Data

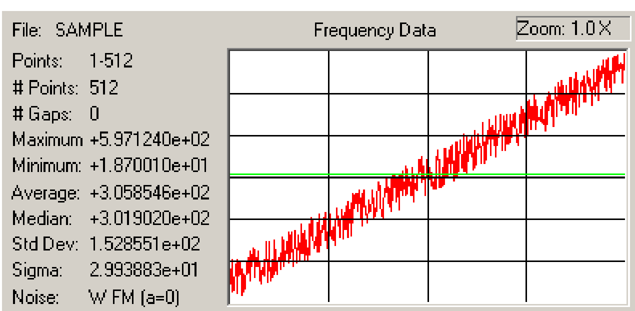


Original Spectrum

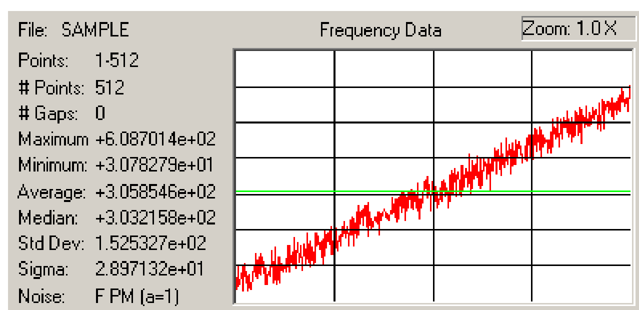


Filtered Spectrum

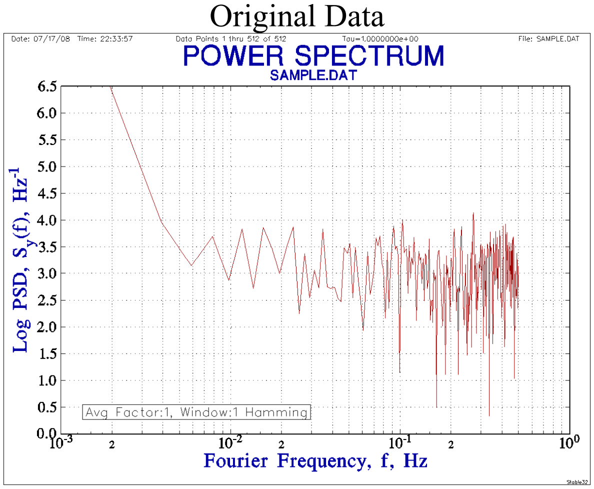
Band Stop Filtration with $f_{lower\ cutoff} = 0.1$ Hz $= 0.098\ f_{nyquist}$ and $f_{upper\ cutoff} = 0.2$ Hz $= 0.195\ f_{nyquist}$



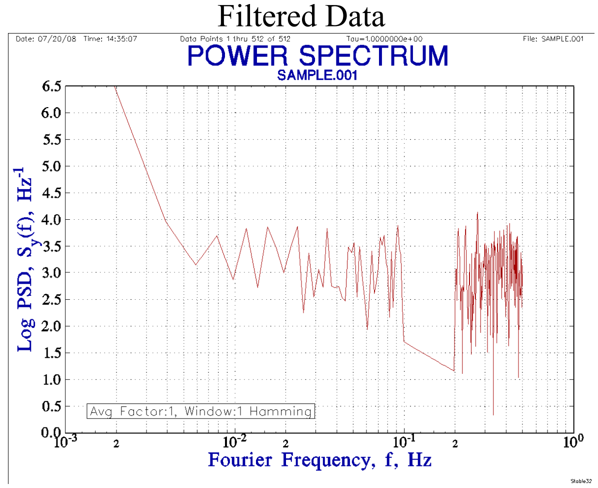
Original Data



Filtered Data



Original Spectrum



Filtered Spectrum

• Applications

Low pass filtration can be useful for removing high frequency noise that may otherwise obscure underlying variations in the data. Its effect is similar to data averaging, but does not lengthen the sampling interval or reduce the number of data points.

High pass filtration can be useful for removing large amplitude low frequency fluctuation in the data due to divergent noise, drift or wandering in order to better see and analyze the high frequency noise. This is particularly effective when the drift or wandering does not fit a function to allow its removal.

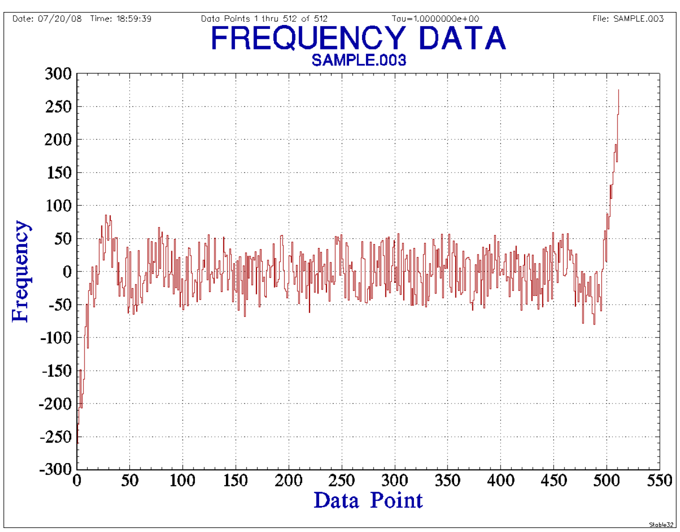
Band pass filtration can be useful for analyzing the amplitude variations of a discrete interfering component. Its function resembles that of a classic wave analyzer.

Band stop filtration can be useful for removing a discrete interfering component. By repeating this operation, multiple components may be removed without significantly affecting the underlying behavior.

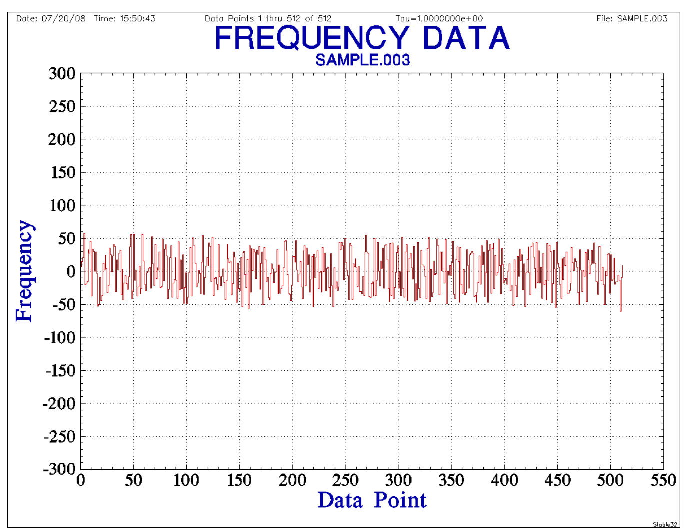
• Implementation

The Stable32 Filter function accomplishes its filtration in the frequency domain by performing a Fast Fourier Transform (FFT), zeroing the conjugate symmetric stop band frequencies, and performing an inverse FFT to obtain the filtered time domain data. The discrete Fourier transform imposes an implicit periodicity on the time domain data, which, if violated, can cause ringing because of the Gibbs phenomenon. That problem is most often minimized by the use of a windowing function to taper the data to zero at its endpoints, but windowing is inappropriate for frequency domain filtering. Instead, the Stable32 Filter function removes a linear trend from the data to force its endpoints to be equal (like the Endpoints drift removal method for phase data, and similar to an early version of the Total variance.) before performing the FFT. The trend line is subsequently restored for the cases of low pass and band stop filtration.

The efficacy of endpoint matching before performing the FFT is shown in the example below for high pass filtration of the same sample data set with a cutoff frequency of 0.02 Hz. The data in the two plots are normalized to zero mean on the same scale for easy comparison.



Filtered data without trend removal before FFT



Filtered data with trend removal before FFT

The range of Fourier frequencies for the phase or frequency data set extends from the frequency bin size of $f_{min} = 1/N\cdot\tau$ to the Nyquist frequency $f_{max} = 1/2\cdot\tau$, where τ is the sampling interval of the data and N is the number of data points, extended as necessary by zero padding to the next larger power-of-2. Low pass filtration has a pass band that extends from f_{min} to an upper cutoff frequency f_{high}, while high pass filtration has a pass band that extends from a lower cutoff frequency f_{low} to f_{max}. Band pass filtration has a pass band between f_{low} to f_{high}, while band stop filtration has a stop band between those frequencies. No specific filter function is required, and there is no deliberate transition region between the pass and stop bands.

File: The Stable32 Filter Function.doc
W.J. Riley
Hamilton Technical Services
July 20, 2008