

Curve Fitting Features in Stable32

W.J. Riley

Hamilton Technical Services

- **Introduction**

The Stable32 program includes several curve fitting features to enhance its usability. Some of these use standard techniques like linear least squares linear regression, while others are unique to specific frequency stability analysis purposes such as power law noise fits.

- **Drift Fits in Phase and Frequency Data Plots**

The Stable32 Phase Plot Lines and Frequency Plot Lines functions support the drawing of various lines on a phase or frequency data plot [1].

For phase data, these lines are endpoint, average, regression and marker lines, where the endpoint lines are drawn between designated end points, the average is a horizontal line is drawn at the mean of the phase data, the regression line is a least squares quadratic fit to the phase data, and the marker is a vertical line at a certain time. The average and regression lines can be drawn between selected times. The phase data quadratic regression fit is of the form $x(t) = a + bt + ct^2$, where the a , b , and c coefficients are determined by closed-form equations [2, 3].

For frequency data, these lines are also endpoint, average, regression and marker lines, where the regression fit is a line of the form $y(t) = a + bt$, where the a and b coefficients are determined by standard closed-form least squares linear regression equations.

In addition, the Phase Plot Options and Frequency Plot Options functions provide several specific curve fits appropriate to their respective data types.

For phase data, these lines are called quadratic and average, and are identical to the quadratic and average lines described above for the Phase Plot Lines function.

These three methods commonly used to analyze frequency drift in phase data are implemented as follows:

1. A least squares quadratic fit to the phase data:

$$x(t) = a + bt + ct^2, \text{ where } y(t) = x'(t) = b + 2ct, \text{ slope} = y'(t) = 2c.$$

This continuous model can be expressed as $x_n = a + \tau_0 bn + \tau_0^2 cn^2$ for $n=1, 2, 3, \dots, N$ and τ_0 is the sampling interval for discrete data, where the a , b and c coefficients have units of sec, sec/sec and sec/sec² respectively, and the frequency drift slope and intercept are $2c$ and b respectively. The fit coefficients can be estimated by the following expressions [2, 3]:

$$\hat{a} = \left(A \sum_{n=1}^N x_n + \tau_0 B \sum_{n=1}^N n x_n + \tau_0^2 C \sum_{n=1}^N n^2 x_n \right) / G$$

$$\hat{b} = \left(B \sum_{n=1}^N x_n + \tau_0 D \sum_{n=1}^N n x_n + \tau_0^2 E \sum_{n=1}^N n^2 x_n \right) / G \tau_0$$

$$\hat{c} = \left(C \sum_{n=1}^N x_n + \tau_0 E \sum_{n=1}^N n x_n + \tau_0^2 F \sum_{n=1}^N n^2 x_n \right) / G \tau_0^2$$

where the A - F terms are as follows:

$$A = 3[3N(N+1)+2]$$

$$B = -18(2N+1)$$

$$C = 30$$

$$D = 12(2N+1)(8N+11) / [(N+1)(N+2)]$$

$$E = -180 / (N+2)$$

$$F = 180 / [(N+1)(N+2)]$$

$$G = N(N-1)(N-2)$$

A quadratic fit to the phase data is the optimum model for white PM noise.

2. The average of the 2nd differences of the phase data:

$$y(t) = [x(t+\tau) - x(t)] / \tau, \text{ slope} = [y(t+\tau) - y(t)] / \tau = [x(t+2\tau) - 2x(t+\tau) + x(t)] / \tau^2.$$

This method is optimum for random walk FM noise. The calculation itself is trivial.

3. A 3-point fit at the start, middle and end of the phase data:

$$\text{slope} = 4[x(\text{end}) - 2x(\text{mid}) + x(\text{start})] / (M\tau)^2, \text{ where } M = \# \text{ data points.}$$

It is the equivalent of the bisection method for frequency data. The calculation itself is trivial.

For frequency data, these lines are called line, log, diffusion, and average. The line and average lines are identical to those described above for the Frequency Plot Lines function, while log and diffusion lines are nonlinear fits useful for characterizing frequency stabilization (drift that becomes smaller with time). The log fit has the form $y(t) = a \ln(bt+1) + c$ as described in MIL-PRF-55310E [4]. The a , b , and c coefficients of the log fit are determined by an iterative least squares algorithm. Similarly, the diffusion fit has the form $y(t) = a + b(t+c)^{1/2}$, where the \sqrt{t} dependence is associated with a diffusion process. The a , b , and c coefficients of the diffusion fit are also determined by an iterative least squares algorithm. These log and diffusion fits are also available in the Stable32 Drift function.

These four methods commonly used to analyze frequency drift in frequency data are implemented as follows:

1. A least squares linear regression to the frequency data:

$$y(t) = a+bt, \text{ where } a = \text{intercept, } b = \text{slope} = y'(t).$$

Linear frequency drift can be estimated by a linear least squares fit to the frequency data, $y(t) = a + bt$. That continuous model can be expressed as $y_n = a + \tau_0 b n$ for $n = 1, 2, 3, \dots, M$ where M is the # of frequency data points, τ_0 is the sampling interval for the discrete data. The frequency drift intercept and slope are a and b , and they have units of sec and sec/sec respectively. The fit coefficients can be estimated by the following expressions:

$$\hat{b} = \frac{M \sum_{n=1}^M n y_n - \sum_{n=1}^M n \sum_{n=1}^M y_n}{M \sum_{n=1}^M n^2 - \left(\sum_{n=1}^M n \right)^2}$$

and:

$$\hat{a} = \frac{\sum_{n=1}^M y_n}{M} - b \cdot \frac{\sum_{n=1}^M n}{M}$$

A linear fit to the frequency data is the optimum model for white FM noise.

2. The frequency averages over the first and last halves of the data:

slope = $2 [y(\text{2nd half}) - y(\text{1st half})] / (N\tau)$, where $N = \#$ points.

This bisection method is optimum for white and random walk FM noise. The calculation itself is trivial.

3. A log model that applies to frequency stabilization [4]:

$y(t) = a \cdot \ln(bt+1)$, where slope = $y'(t) = ab/(bt+1)$

Determining the nonlinear log fit coefficients requires an iterative procedure. This involves setting b to an initial value, linearizing the equation, solving for the other coefficients and the sum of the squared error, comparing that with an error criterion, and iterating until a satisfactory result is found. The key aspects to this numerical analysis process are establishing a satisfactory iteration factor and error criterion to assure both convergence and small residuals.

4. A diffusion (\sqrt{t}) model of the form:

$y(t) = a + b(t+c)^{1/2}$, where slope = $y'(t) = \frac{1}{2} \cdot b(t+c)^{-1/2}$.

Determining the nonlinear diffusion fit coefficients also requires an iterative procedure. This involves setting c to an initial value, linearizing the equation, solving for the other coefficients and the sum of the squared error, comparing that with an error criterion, and iterating until a satisfactory result is found. Again, the key aspects to this numerical analysis process are establishing a satisfactory iteration factor and error criterion to assure both convergence and small residuals.

References:

1. These fitting functions are part of the FrequenC Library from Hamilton Technical Services.
2. J.A. Barnes, "The Measurement of Linear Frequency Drift in Oscillators", *Proc. 15th PTTI Meeting*, pp. 551-582, Dec. 1983.
3. J.A. Barnes, "The Analysis of Frequency and Time Data", Austron, Inc., Dec. 1991.
4. [MIL-PRF-55310E](#), Oscillators, Crystal, General Specification For, March 2006.

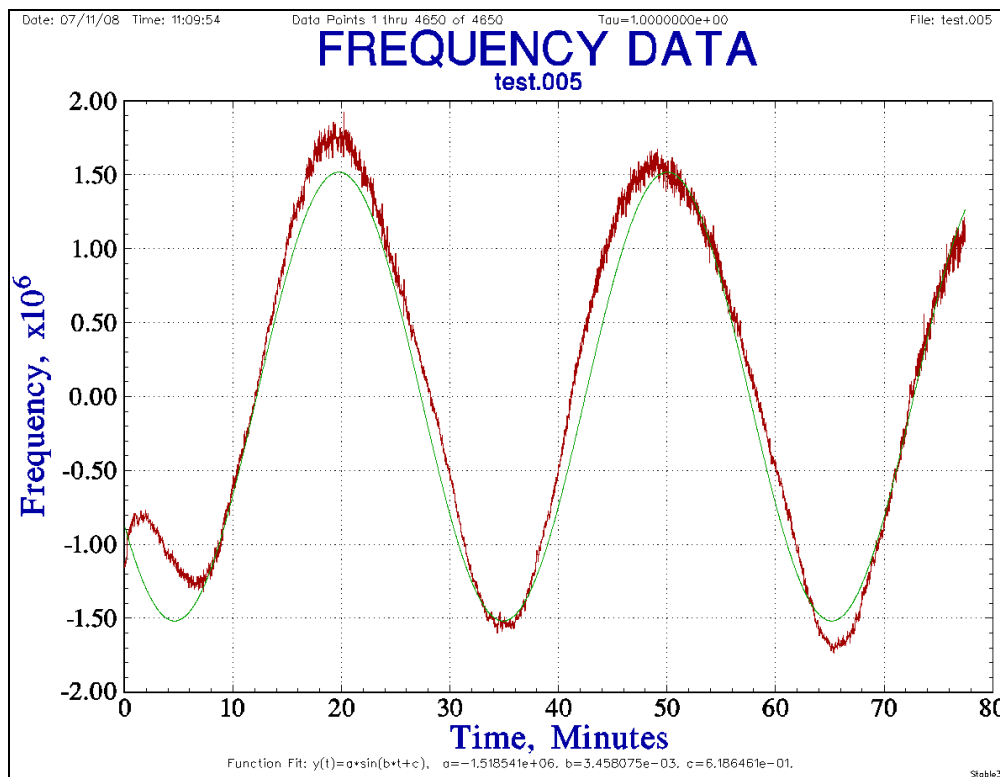
• **Other Fits in Phase and Frequency Data Plots**

Stable32 (Version 1.52 and higher) supports two additional fitting methods for phase and frequency data, polynomial fits and general functional fits.

The polynomial fit feature extends the available polynomial fits up to 9th order. While these usually have little physical significance, they can be used to examine the character of the data in more detail.

The general functional fit feature allows fitting the data to an arbitrary function. The function expression comprises the basic arithmetic operators, mathematical functions such as log() and exp(), and a maximum of 10 parameters. The supported mathematical functions are sin(), cos(), tan(), asin(), acos(), atan(), sinh(), cosh(), tanh(), exp(), log(), log10(), sqrt(), floor(), ceil(), abs() and hypot(). Function fitting uses the Levenberg-Marquardt algorithm for nonlinear least squares curve fitting, which may require entry of appropriate initial parameter values.

An example of a sinusoidal fit of the form $y(t) = a \cdot \sin(b \cdot t + c)$ to frequency data having a strong discrete component is shown in the plot below:



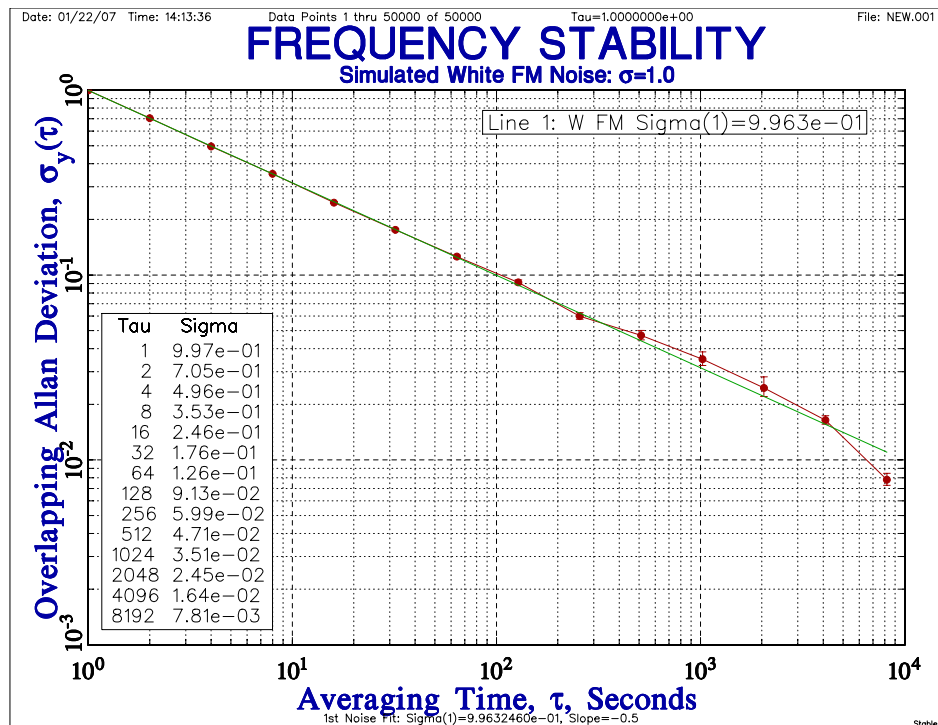
References:

1. K. Levenberg, "A Method for the Solution of Certain Non-Linear Problems in Least Squares", *Quarterly of Applied Mathematics*, Vol. 2-2, pp. 164-168, July 1944.
2. D.W. Marquardt, "An Algorithm for the Least-Squares Estimation of Nonlinear Parameters", *SIAM Journal of Applied Mathematics*, Vol. 11-2, pp. 431-441, June 1963.
3. K. Madsen, H.B. Nielsen and O. Tingleff, [Methods for Non-Linear Least Squares Problems](#), Technical University of Denmark, April 2004.

● **Power Law Noise Fits in Stability Plots**

The Stable32 Run Plot Lines function supports the drawing of power law noise lines on a stability plot. These lines are weighted fits to certain power law noise slopes based on the stability data and their error bars [1].

An example of an overlapping Allan deviation stability plot for simulated white FM noise is shown below. The green power law noise fit line has a slope of -0.5, as appropriate for this noise type, and a $\sigma(1)$ intercept of 0.996, essentially equal to the nominal value of the simulated noise.



The Stable32 power law noise fits of the Run Plot Lines function stability plots are weighted according to the error bars associated with each stability point. More specifically, the weight applied to each point is scaled inversely by w , the span of its confidence interval (error bars) divided by the nominal sigma. This is theoretically desirable, and also provides better fits, especially for stability data comprising more than one power law noise type. Because the line slope is fixed for a particular user-selected noise type, the noise or aging lines are simply fitted to the weighted average of expressions that represent them on log-log coordinates.

$$\hat{\sigma}(1) = \frac{\sum_{i=1}^N \frac{\sigma(\text{nominal})_i}{w_i \cdot \tau^{\mu/2}}}{\sum_{i=1}^N 1/w_i}$$

where:

N = # stability data points

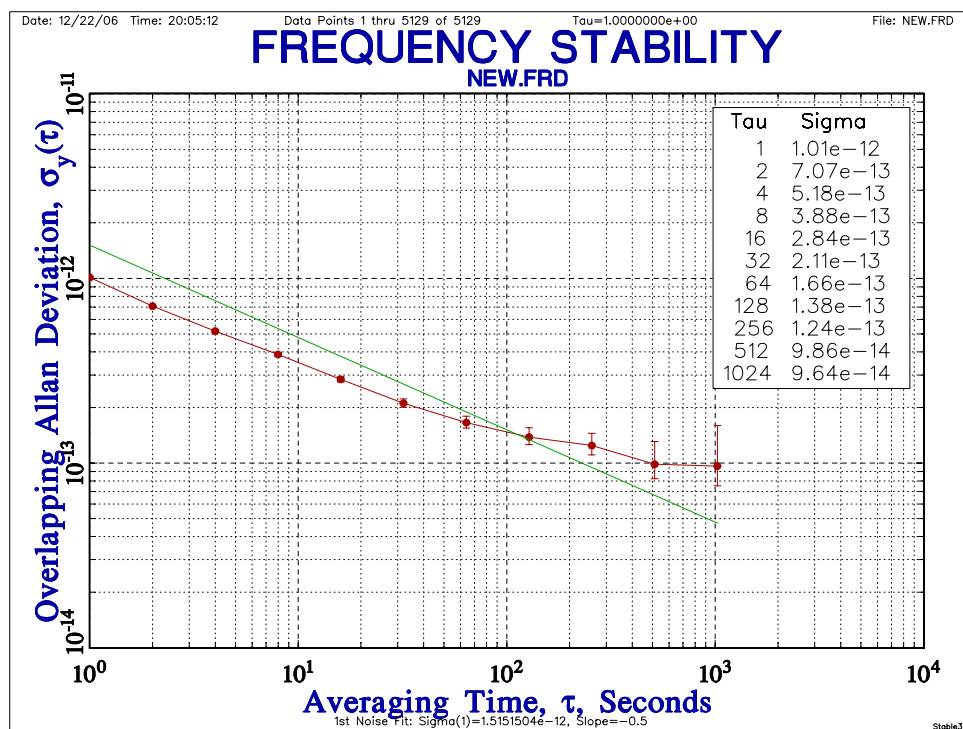
$w_i = \frac{\sigma(\text{max})_i - \sigma(\text{min})_i}{\sigma(\text{nominal})_i}$ = weighing function

μ = variance slope: $\sigma_y^2(\tau) \propto \tau^\mu$, $-2 \leq \mu \leq 2$ for unmodified variances

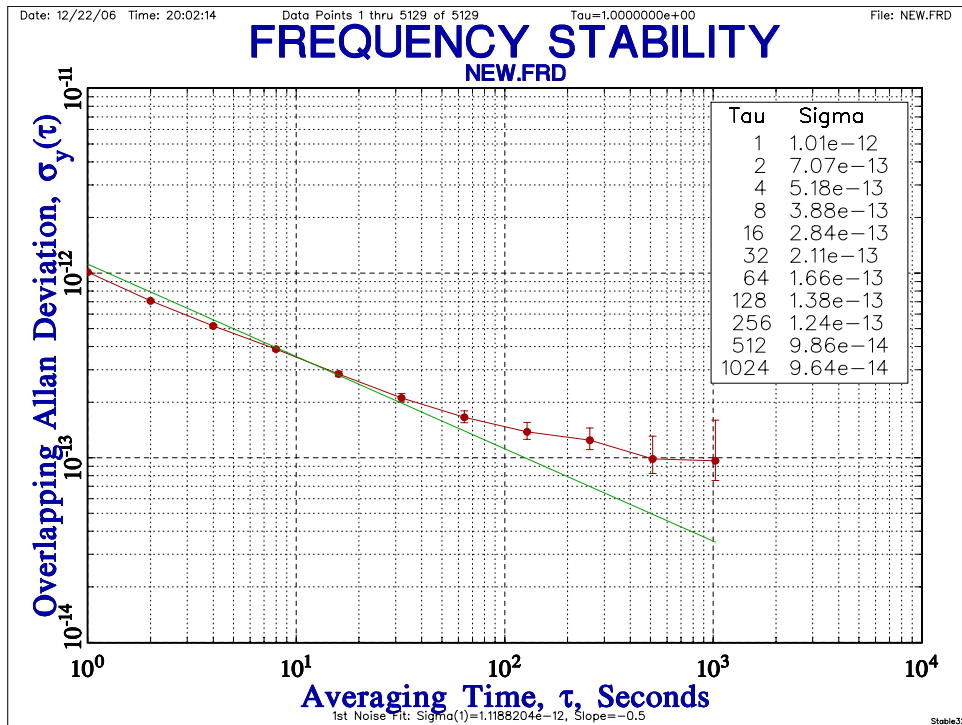
$\sigma_y^2(\tau) \propto \tau^{\mu'}$, $-3 \leq \mu' \leq 2$ for modified variances

Noise Type	$\sigma(\tau)$ Slope	$\sigma(1)$ Equation	Remarks
W PM	-1.5	$(\sigma \cdot \tau \cdot \sqrt{\tau})/w$	Applies only to modified sigma types
F PM	-1.0	$(\sigma \cdot \tau)/w$	Also applies to W PM for unmodified sigma types
W FM	-0.5	$(\sigma \cdot \sqrt{\tau})/w$	
F FM	0.0	$(\sigma)/w$	
RW FM	+0.5	$(\sigma/\sqrt{\tau})/w$	
Aging	+1.0	$(\sigma/\tau)/w$	

An example of overlapping Allan deviation stability plots with unweighted and weighted white FM noise fits for a set of simulated clock data having the combination of white FM at a level of $1 \times 10^{-12} \tau^{-1/2}$ plus flicker FM at a level of 1×10^{-13} is shown below. This type of stability record is quite typical for rubidium passive atomic frequency standards. The white FM noise fit that applies to the shorter averaging times is obviously better for the weighted case.

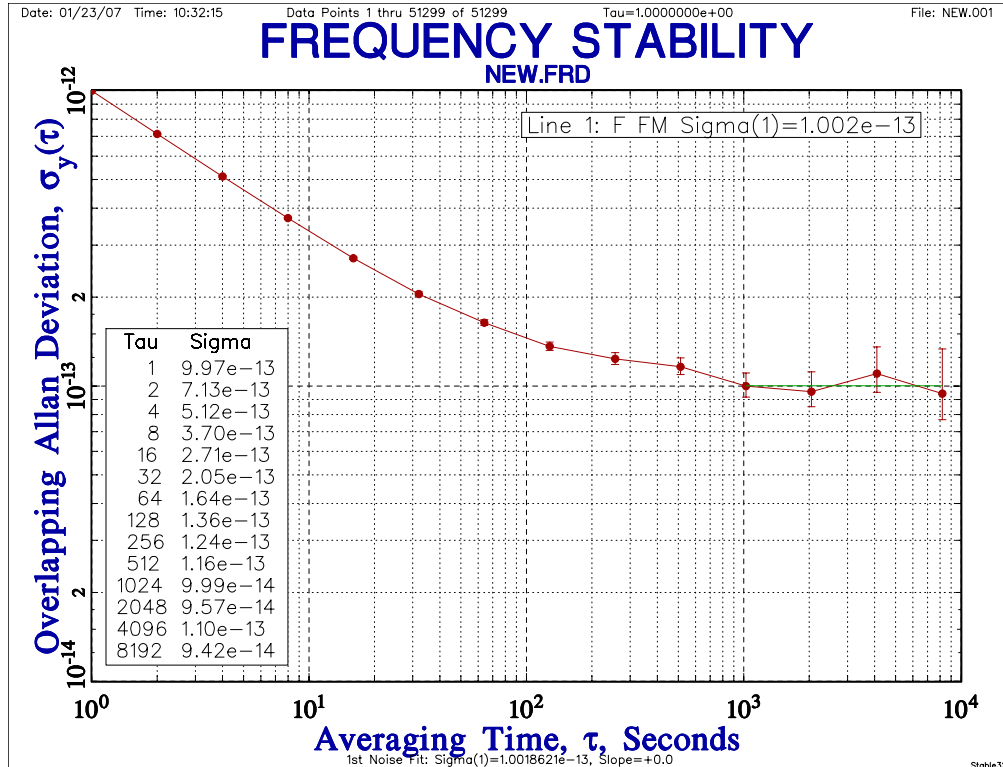


Unweighted White FM Noise Fit



Weighted White FM Noise Fit

The Stable32 stability plot noise lines can be fitted to a selected tau range, which is appropriate when the noise type changes as a function of averaging time. An example of this for the flicker FM portion of a larger data set of the same nominal $1 \times 10^{-12} \tau^{-1/2}$ W FM plus 1×10^{-13} F FM noise is shown below:



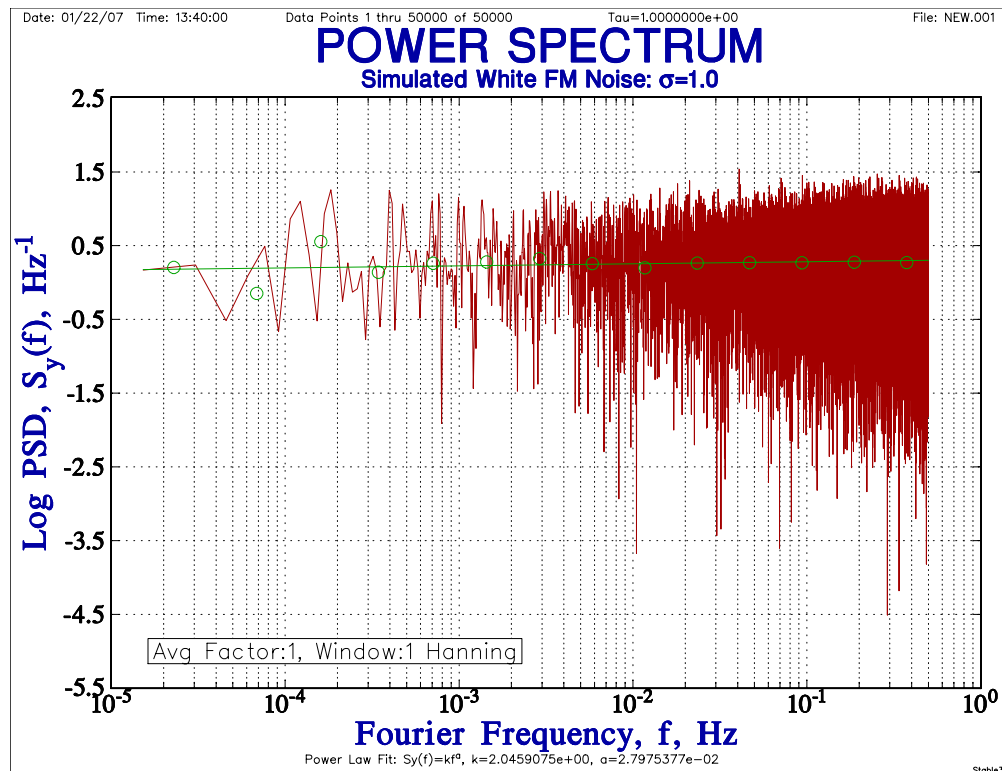
Reference:

1. The author wishes to thank Robert Lutwak for suggesting the weighted noise line feature.

- **Power Law Noise Fits in PSD Plots**

The Stable32 Power function that plots several types of power spectral densities supports an option to fit a line to the PSD data. That line is a weighted least squares linear fit to octave band averages of the PSD data. The weighting is based on the # of unaveraged PSD points used to determine each octave band average (shown as circles on the PSD plot). The slope of the resulting line, provided that it is close to an integer value, provides an estimate to a power law noise fit.

An example of a $S_y(f)$ PSD plot for same simulated white FM noise is shown below. The octave band averages are shown as green circles, and the resulting power law fit as a green line. The power law exponent, α , is about 0.028, very close to the expected value of 0 for white FM noise. For comparison, the lag 1 autocorrelation method reports $\alpha = 0.00$ at AF=1, and a lag 1 scatter slope of -0.001 (see below) for this 50,000-point data set. These data have AF=1 overlapping Allan and standard deviation values of 0.997 and 0.996 respectively, and their near equality is another indication that the data are white FM noise.



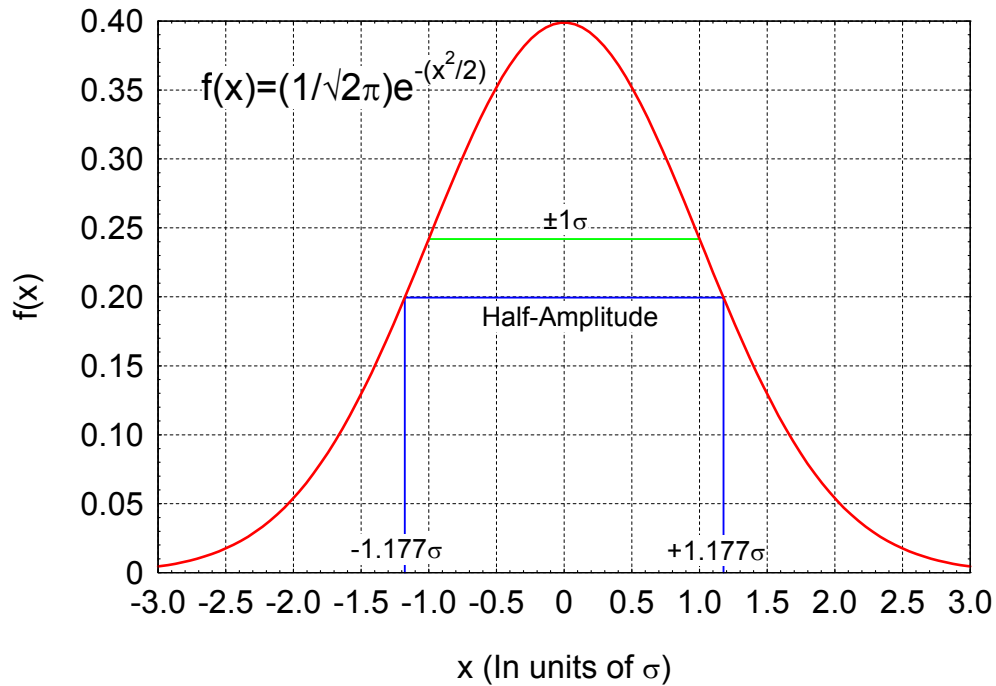
Reference:

1. W.H. Press, B.P. Flannery, S.A. Teukolsky and W.T. Vetterling, [*Numerical Recipes in C*](#), Cambridge University Press, 1988, ISBN 0-521-35465-X, Chapter 14, `fit()`.

- **Normal Distribution Fits in Histograms**

The Stable32 Histogram function supports an option to show a normal distribution curve fit on the plotted histogram. Information about the normal probably distribution is shown in the figure below:

Standardized Normal Distribution ($\mu=0, \sigma=1$)

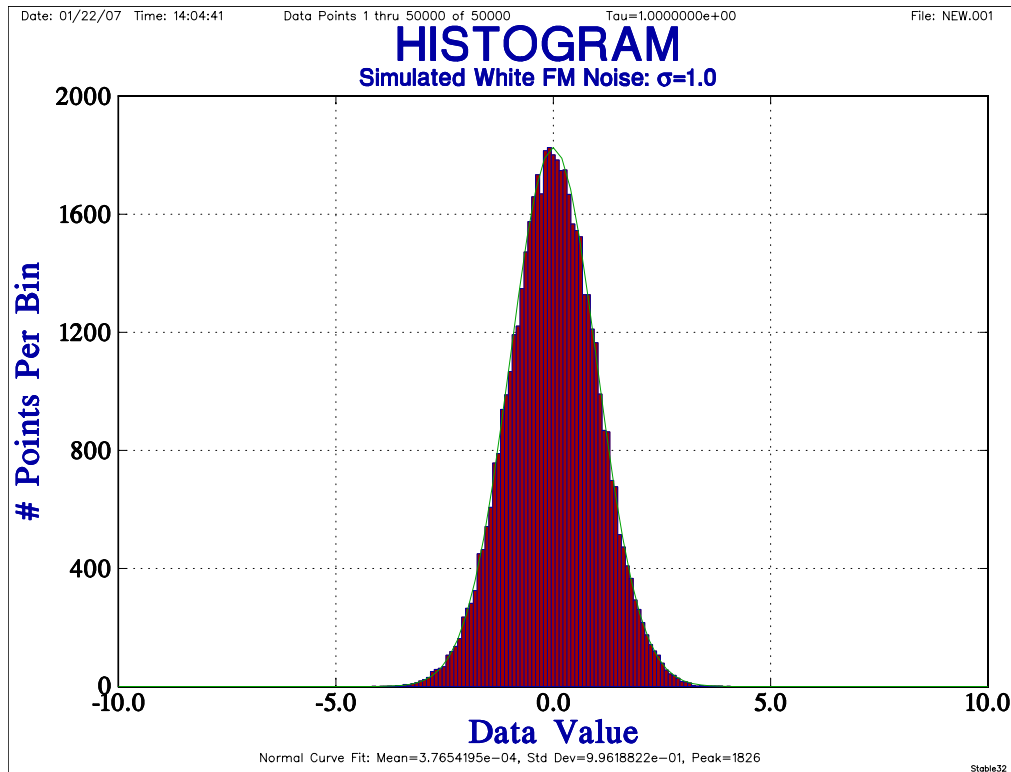


The probability distribution of a normal distribution with mean μ and standard deviation σ is a Gaussian function given by [1]:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Stable32 simply calculates the mean and standard deviation of the histogram data, and uses a discrete version of this equation to add a normal distribution fit the histogram plot.

An example of a fit to a normal distribution function on a Stable32 histogram plot is shown in the figures below. This histogram shows some simulated white FM noise as above, having a nominal Allan (and standard) deviation of 1.0. The histogram has a normal distribution with a total width of about ± 3 , a mean value of about zero, and a standard deviation of 0.996.

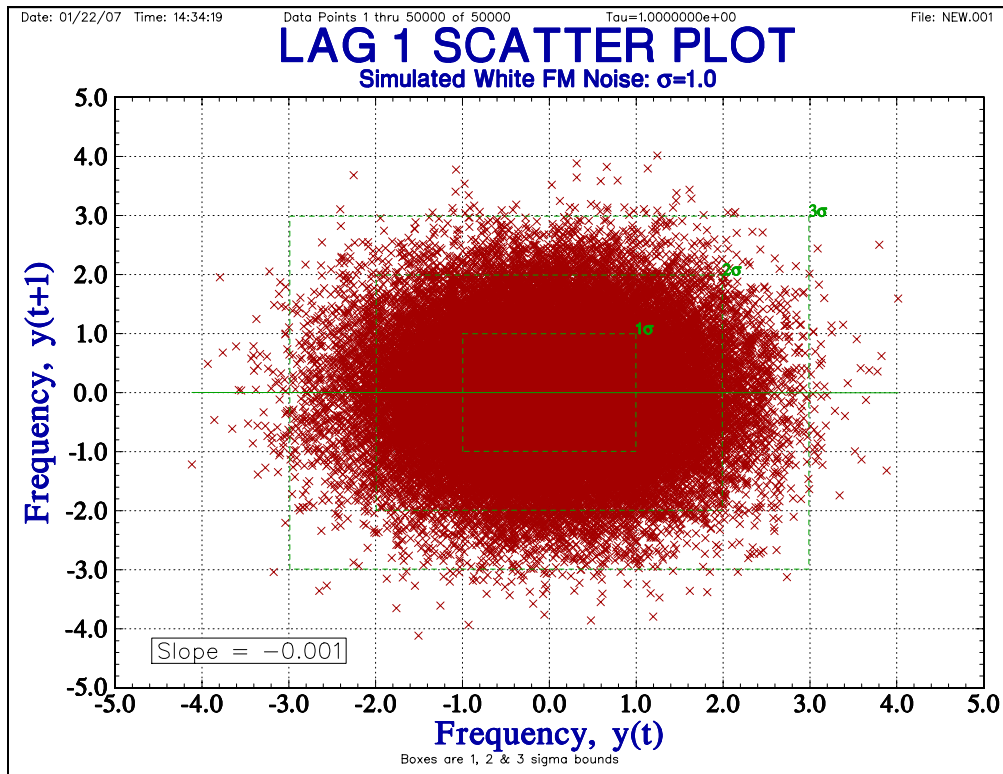


Reference:

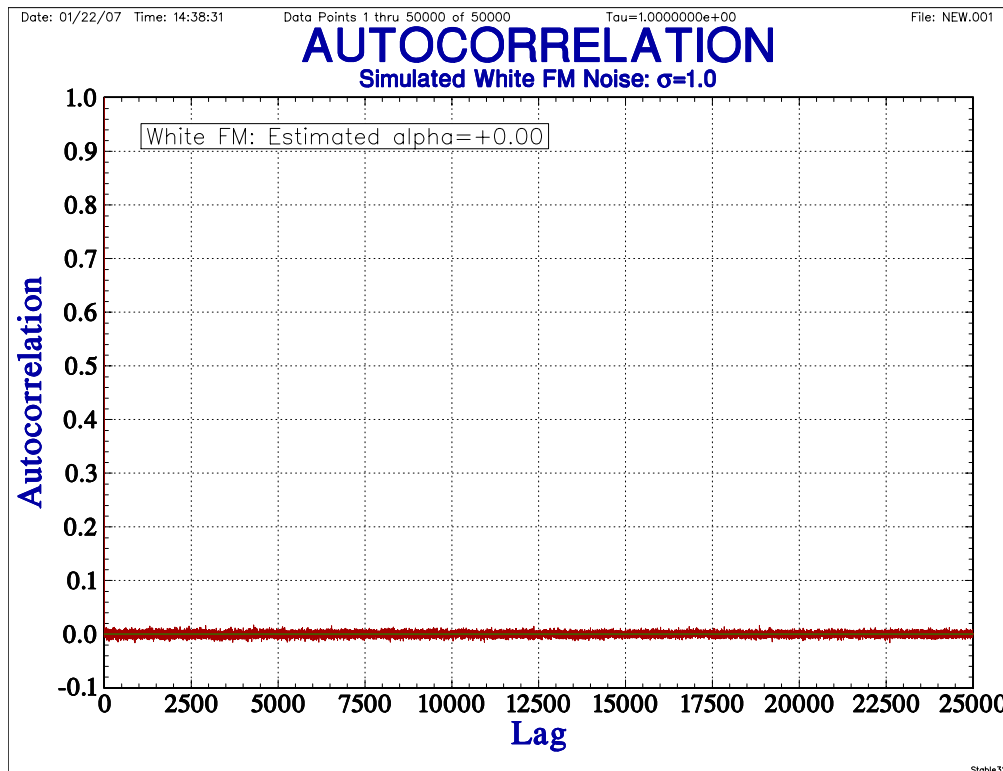
1. http://en.wikipedia.org/wiki/Normal_distribution

- **Slope Fits in Lag 1 Autocorrelation Scatter Plots**

The Stable32 Autocorrelation function includes a Lag 1 Scatter Plot that plots $y(t+1)$ versus $y(t)$ and thus shows the correlation between adjacent points by its average tilt or slope, as well as being an indicator of periodic components in the data [1, 2]. The Lag 1 scatter plot can be shown either as an insert in the autocorrelation plot, or separately as shown below. It includes a linear least squares fit to its plotted points, whose slope is displayed in a text box. The Lag 1 Scatter Plot also includes boxes showing its 1, 2 and 3σ regions.



The near-zero slope indicates that these data are uncorrelated (white), as is also shown in the corresponding autocorrelation plot:



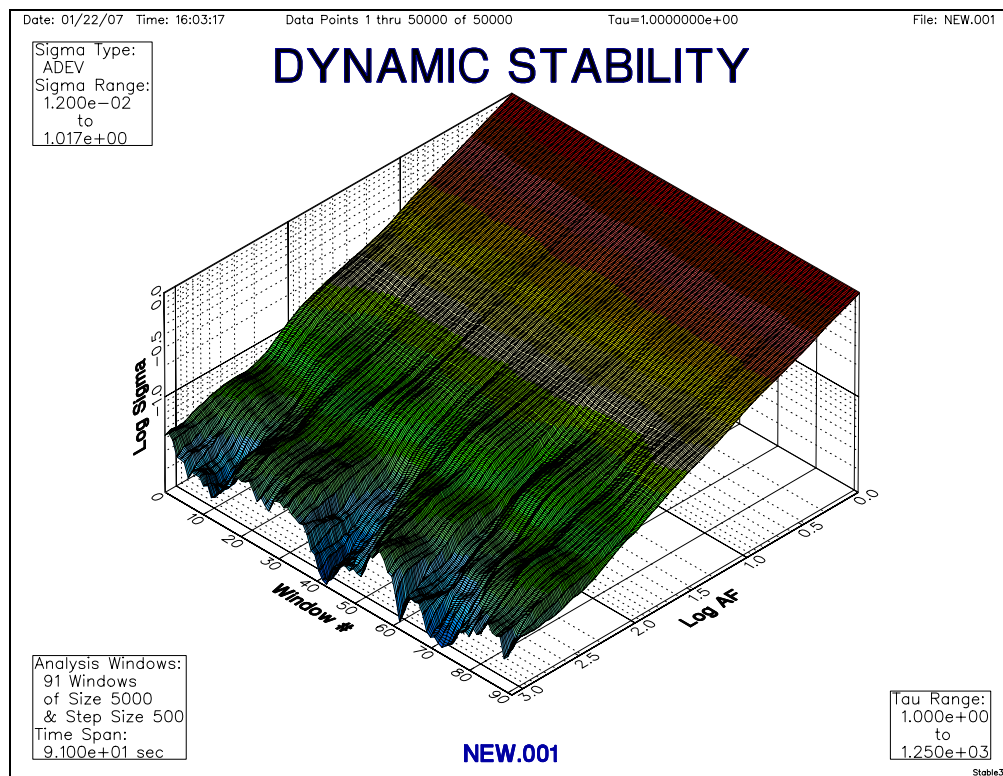
The autocorrelation is essentially zero at all lags (except 0, of course, where it is 1). The green line in this plot is simply drawn at zero.

References:

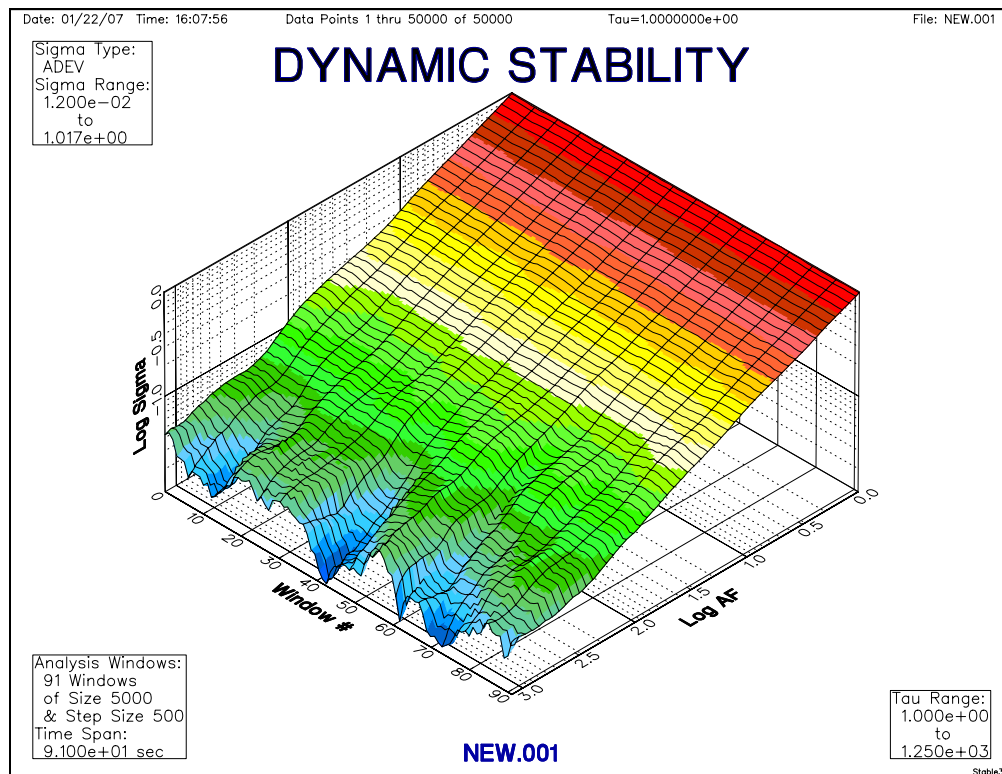
1. D.B Percival and A.T. Walden, [*Spectral Analysis for Physical Applications*](#), Cambridge University Press, ISBN 0-521-35532-X, 1993
2. W.J. Riley and C.A. Greenhall, "[Power Law Noise Identification Using the Lag 1 Autocorrelation](#)", *Proc. 18th European Freq. and Time Forum*, April 2004.

• **Dynamic Stability 3D Surface Plots**

The Stable32 Dynamic Stability function generates a 3D surface plot that shows the stability variations over the span of the data record [1, 2]. The surface is represented by a uniform mesh with x and y intersections at increments of $100 \cdot \log AF$ and window # respectively, and a z value that varies as $\log \sigma$. Typical grid values are several hundred x points by one hundred y points. At the smaller averaging factors, it is necessary to interpolate between sigma values to obtain a z value at every grid intersection. That is done by simple linear interpolation.



Full Grid Dynamic Stability Plot



1/5 Grid Dynamic Stability Plot

The Stable32 DAVAR function includes a option to set the grid spacing. The grid factor does not change the underlying mesh interval, only its visible markings, so there is little difference in the time required to redraw these plots. The full grid shows the stability results better, and the only advantage of the reduced grid is to make the colors brighter.

References:

1. L. Galleani and P. Tavella, "[The Characterization of Clock Behavior with the Dynamic Allan Variance](#)", *Proc. 2003 Joint FCS/EFTF Meeting*, pp. 239-244, 2003.
2. L. Galleani and P. Tavella, "[Tracking Nonstationarities in Clock Noises Using the Dynamic Allan Variance](#)", *Proc. 2005 Joint FCS/PTTI Meeting*.