SINE FUNCTION IDENTIFICATION AND REMOVAL

By Tom Irvine
Email: tomirvine@aol.com

August 14, 2000

Introduction

There are several methods for identifying the amplitude and frequency of sine functions in measured data.

One method is the direct inspection. Another method is the Fourier transform.

A third method is a curve-fit method that is performed in the time domain. This method is useful for identifying the amplitude, frequency, and phase angle of a sine function. It is potentially more accurate than either the direct inspection method or the Fourier transform.

The time-domain curve-fit method is also useful for removing spurious sine signals from data. These spurious signals are usually due to ground loops, eddy currents, and other electrical interference effects. In addition, analog tape records often introduce spurious sine signals.

The purpose of this report is to demonstrate the time-domain curve-fit method.

Method

The method is semi-automatic. The user must estimate the sine frequency, or frequencies. A computer program then uses random number generation to determine the best fit in terms of amplitude, frequency, and phase. The computer program is called sinefind. The source code is given in Reference 1.

The steps are in summary:

1. The user estimates one or more sine frequencies. The user also specifies the number of iterations.
2. The computer program reads in the measured data signal and calculates its standard deviation amplitude.
3. The program generates a trial amplitude, frequency, and phase angle using a random number routine.
4. The program generates a trial signal based on the parameters in step 3.
5. The program subtracts the trial signal from the measured data signal and computes the error.
6. The program repeats steps 2 through 5 many times per the user's initial instructions.
7. The program then selects the amplitude, frequency, and phase angle which gave the lowest error.

As an important note, this method works best when the measured data has a sine function with fairly constant amplitude. Also, it is appropriate for small segments of data, with less than 10,000 data points. These guidelines are not absolute requirements, however.

Example 1

The following data is measured data taken on a floor adjacent to a wafer polishing machine in a semiconductor facility. The measured time history is shown in Figure 1.

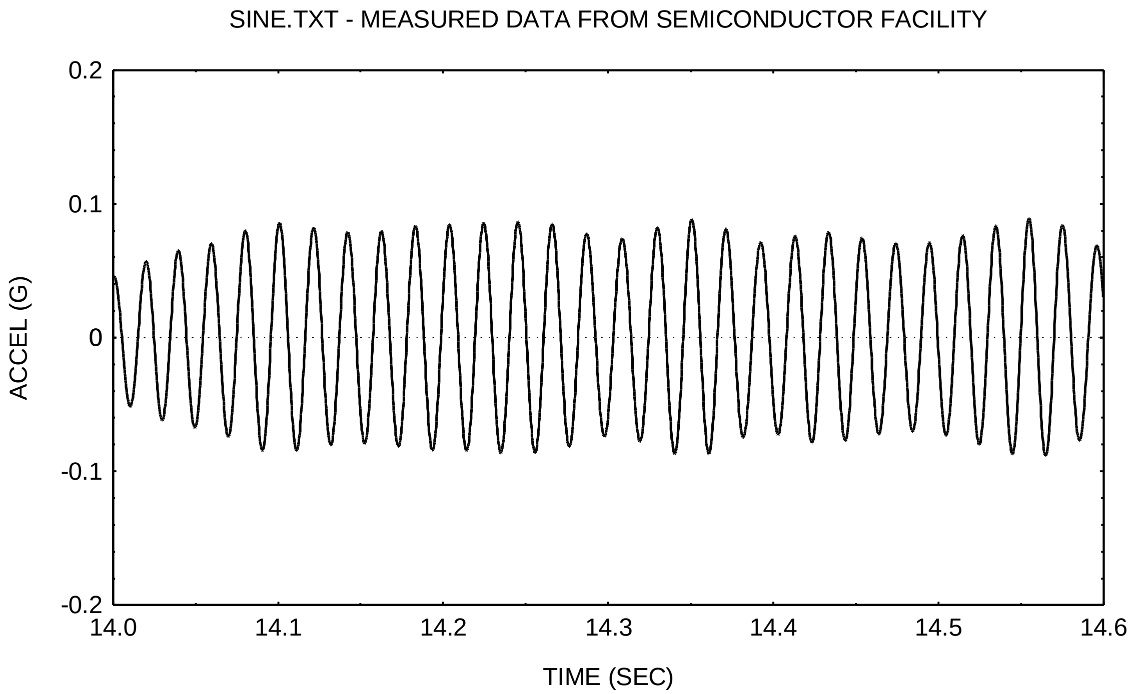

Figure 1.

The dominant frequency is approximately 50 Hz, as estimated by counting the peaks and then dividing by the duration. Use the curve-fitting method to determine the amplitude, frequency, and phase angle.

The following estimate is thus obtained using program sinefind:

$$y(t) = A \sin\left(2\pi f\ t - \phi \right)$$

where

$$A = 0.0812 \ \ G$$
$$f \ = 48.381 \ \ Hz$$
$$\phi \ = 6.173 \ \ radians \qquad (1)$$

The input parameter to program sinefind were:

40000 trials per frequency,
nominal frequency = 50 Hz with a tolerance of ±5 Hz.

Equation (1) is shown superimposed on the sine.txt time history in Figure 2. The agreement is good. The error signal is shown in Figure 3.

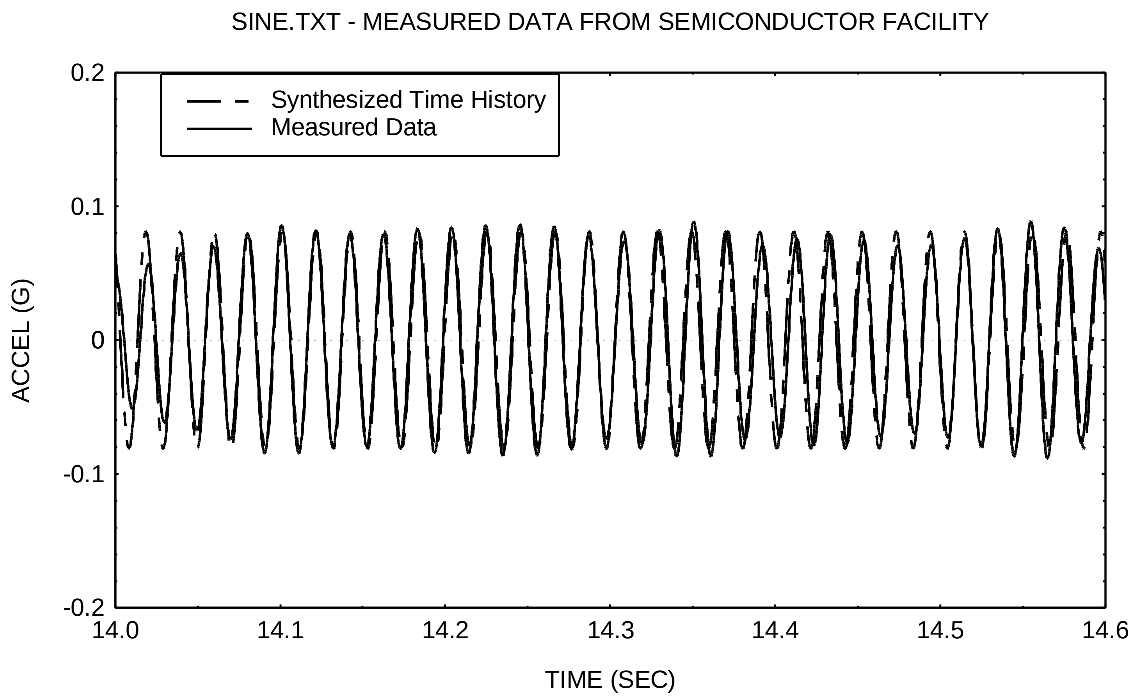

Figure 2.

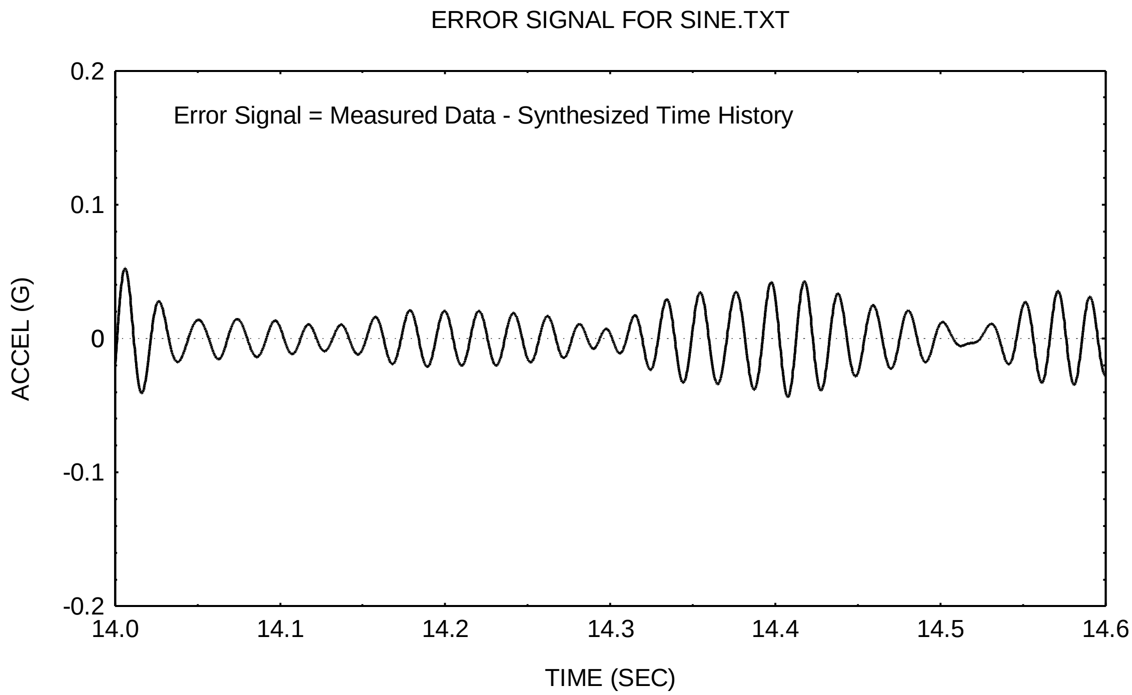

Figure 3.

Example 2

Note that the time history in Figure 1 has a slight beat frequency effect. Thus, repeat the analysis using two frequencies.

The following estimate is obtained using program sinefind.exe.

$$y(t) = \sum_{i=1}^{2} A_i \sin\left(2\pi f_i t - \phi_i\right)$$

where

$$\begin{aligned} A_1 &= 0.0812 \text{ G} & A_2 &= 0.0179 \text{ G} \\ f_1 &= 48.381 \text{ Hz} & f_2 &= 46.176 \text{ Hz} \\ \phi_1 &= 6.173 \text{ radians} & \phi_2 &= 3.394 \text{ radians} \end{aligned} \tag{2}$$

The input parameter to program sinefind.exe were:

40000 trials per frequency,
nominal frequency 1 = 50 Hz with a tolerance of ±5 Hz,

nominal frequency 2 = 50 Hz with a tolerance of ±10 Hz.

Equation (2) is shown superimposed on the sine.txt time history in Figure 4. The agreement is very good. The error signal is shown in Figure 5.

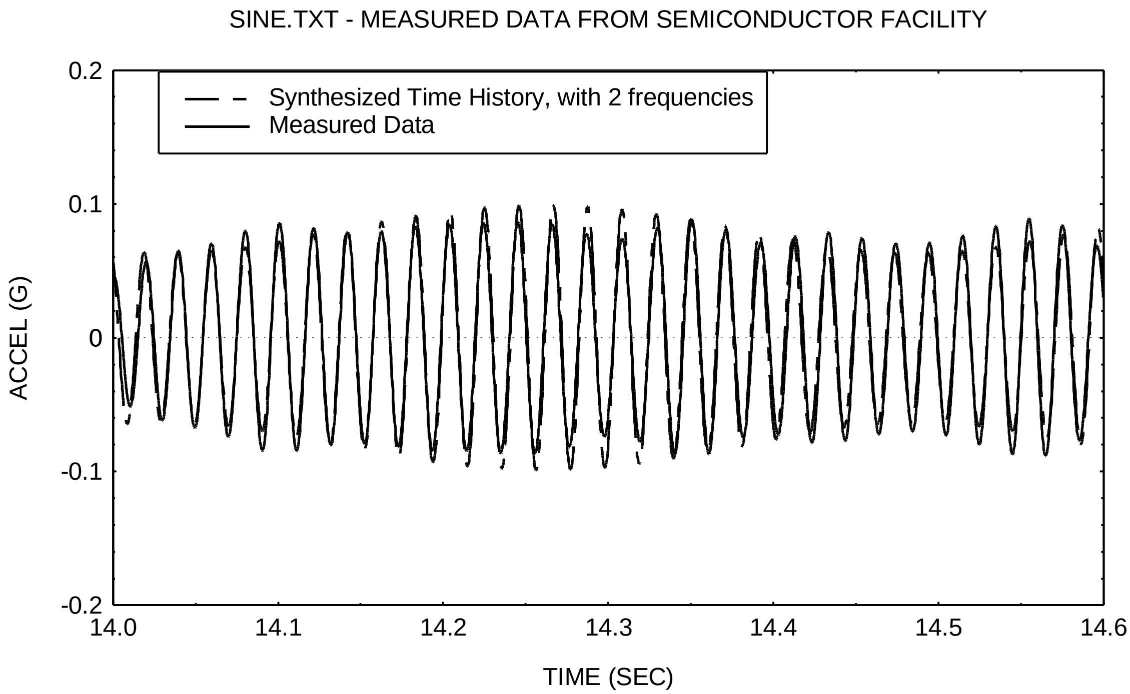

Figure 4.

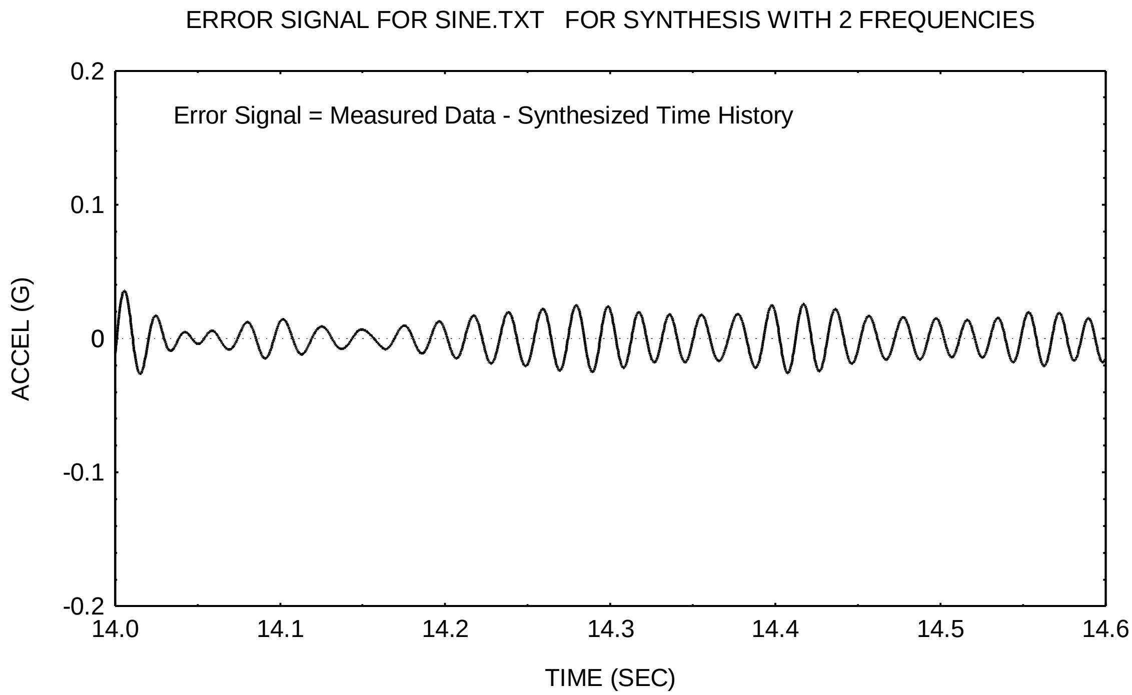

Figure 5.

Obviously, this synthesis process could be repeated for additional frequencies in order to reduce the error.

Reference

1. T. Irvine, Program: sinefind. Source code and executable program posted at: http://www.vibrationdata.com/software.htm