MPULSE RESPONSE

Swept Sine Impulse Response Measurement*

Edgar Berdahl Julius Smith

This work is produced by The Connexions Project and licensed under the Creative Commons Attribution License †

Abstract

The swept sine measurement technique is used for measuring impulse responses. It has advantages when measuring weakly nonlinear systems in that the linear and nonlinear components can be separated to some extent. Free, open-source software is provided in Matlab (or Octave) and Pure Data (PD) for carrying out swept-sine impulse-response measurements using the sound hardware found on a typical personal-computer.

1 Introduction



Figure 1: Linear system to be measured

Figure 1 depicts a linear system characterized by an impulse response h(n), driven by an input signal s(n), and producing the output signal r(n). The **system identification** problem is to estimate h(n) given known input/output signals s(n) and r(n). A practical method for identifying finite impulse responses is the **swept-sine** measurement technique, described below.

2 Sine Sweep Measurement Theory

In some cases, it is desirable to relax the power-maximizing constraint $|s(n)| = 1 \forall n$ in favor of obtaining some other desirable measurement system properties. For example, we may care more about the accuracy of the measurement at lower frequencies compared to higher frequencies, so we would like the excitation signal

^{*}Version 1.4: Aug 19, 2008 6:59 pm -0500

 $^{^{\}dagger}$ http://creativecommons.org/licenses/by/2.0/

s(n) to contain more energy at lower frequencies [1]. We might also be measuring a mechanical or acoustical system in which the motor controlled by s(n) behaves weakly nonlinearly. If the nonlinearity is memoryless and is **NOT** preceded by any filtering, then the system to be measured matches the Hammerstein model shown in Figure 2. The goal is to measure h(n), independently of the motor nonlinearity f(s). Performing the measurement is complicated by the fact that superposition no longer holds.



Figure 2: Hammerstein Model

Mathematically, the Hammerstein system behaves as follows:

$$r(n) = (f(s) * h)(n)$$
(1)

It turns out that we can obtain both of these desirable measurement system properties by using a new excitation signal s(n). This signal is a sine wave with a frequency that is exponentially increased from ω_1 to ω_2 over T seconds [2]:

$$s(n) = \sin\left[K\left(e^{-n/Lf_s} - 1\right)\right] \tag{2}$$

where $K = \frac{\omega_1 T}{ln \frac{\omega_2}{\omega_1}}$ and $L = \frac{T}{ln \frac{\omega_2}{\omega_1}}$. The MATLAB¹ /Octave² code generate_sinesweeps.m³ generates the appropriate sine sweep.

Now we consider how to extract the linearized impulse response from a measurement. In essence, we need to inverse filter⁴ the measurement by the excitation signal.

To this end, we realize that a useful property of s(n) is that the time delay Δt_N between any sample n_0 and a later point with instantaneous frequency N times larger than the instantaneous frequency at $s(n_0)$ is constant:

$$\Delta t_N = T \frac{\ln\left(N\right)}{\ln\frac{\omega_2}{\omega_1}} \tag{3}$$

This characteristic implies that after inverse-filtering the measured response, the signals due to the nonlinear terms in f(s) are located at specific places in the final response signal. Consequently, the linear contribution to the response, which is proportional to h(n) can be separated from the other nonlinear terms. We can thus measure a linear system even if it is being driven by a weakly nonlinear motor.

Because the frequency of s(n) increases exponentially, the system is excited for longer periods of time at lower frequencies. This means that the inverse filter averages measurements at lower frequencies longer, so this measurement technique is better suited to especially low-pass noise sources.

¹http://www.mathworks.com/

²http://www.octave.org/

³http://ccrma.stanford.edu/realsimple/imp_meas/generate_sinesweeps.m

⁴http://ccrma.stanford.edu/~jos/pasp/Inverse Filtering.html

3 Sine Sweep Measurement Procedure

- 1. Generate the sine sweeps⁵ using generate sinesweeps. m^6
- 2. Open the pd patch sinesweeps.pd⁷ in pd.
- 3. Ensure that the patch is not in editing mode, and check the "compute audio" box in the main pd window.
- 4. Adjust the "Output Volume" so that when you click on "Record Response To The Sine Sweeps," the system under test is behaving linearly (i.e. not clipping), but so that the input signal to the sound interface is not too noisy.
- 5. If there is an input volume on the sound interface, adjust it so that the levels approximately match those shown in Figure 3 when you click on "Record Response To The Sine Sweeps." If the sound interface has no input volume, then you will need to adjust the "Output Volume" accordingly.



Figure 3: sinesweeps.pd after making a measurement with an appropriate input level

- 6. Once you are satisfied with the results, click the "Write Responses to Disk" button.
- 7. pd will write the file Resp.wav to disk. Rename this file so that the name matches the measurement you just made. For instance, you might rename it to nonlinear2Resp.wav⁸ if it corresponded to the second time you measured the transfer function of a weakly nonlinear system.
- 8. Run sinesweeps_response('nonlinear2,100,0.4')⁹ in MATLAB¹⁰ or Octave¹¹ to analyze the measured response. This means that the inverse filter will be restricted to a dynamic range of 100 (40 dB), which helps avoid exaggerating problems beneath ω_1 and above ω_2 , where the excitation signal has little energy. 0.4 refers to the length in seconds of the linear impulse response term to be extracted. Plots will be generated, and the file nonlinear2ImpResp.wav¹² will be written to disk.

 $^{^{5} \}rm http://ccrma.stanford.edu/realsimple/imp meas/sinesweeps.wav$

 $^{^{6}} http://ccrma.stanford.edu/realsimple/imp_meas/generate_sinesweeps.m$

⁷http://ccrma.stanford.edu/realsimple/imp_meas/sinesweeps.pd

 $^{^{8}} http://ccrma.stanford.edu/realsimple/imp_meas/nonlinear2Resp.wav$

 $^{^{9}} http://ccrma.stanford.edu/realsimple/imp_meas/sinesweeps_response.m$

¹⁰http://www.mathworks.com/

¹¹http://www.octave.org/

 $^{^{12} \}rm http://ccrma.stanford.edu/realsimple/imp_meas/nonlinear2ImpResp.wav$

Connexions module: m15945

References

- [1] J. Abel and D. Berners. Signal Processing Techniques for Digital Audio Effects. http://ccrma.stanford.edu/courses/424/, 2005.
- [2] Angelo Farina. Simultaneous measurement of impulse response and distortion with a swept-sine technique. Audio Engineering Society Convention, Preprint 5093, Feb. 2000.