Available online at www.sciencedirect.com
ScienceDirect
Mechanical Systems and Signal Processing 21 (2007) 165–176
Mechanical Systems and Signal Processing
www.elsevier.com/locate/jnlabr/ymssp

# Effects of aliasing on numerical integration

Timothy S. Edwards*

*Sandia National Laboratories*[1]*, MS0557, P.O. Box 5800, Albuquerque, NM 87185-0557, USA*

Received 14 February 2005; received in revised form 21 June 2005; accepted 6 August 2005
Available online 23 September 2005

**Abstract**

During the course of processing acceleration data from mechanical systems it is often desirable to integrate the data to obtain velocity or displacement waveforms. However, those who have attempted these operations may be painfully aware that the integrated records often yield unrealistic residual values. This is true whether the data has been obtained experimentally or through numerical simulation such as Runge–Kutta integration or the explicit finite element method. In the case of experimentally obtained data, the integration errors are usually blamed on accelerometer zero shift or amplifier saturation. In the case of simulation data, incorrect integrations are often incorrectly blamed on the integration algorithm itself. This work demonstrates that seemingly small aliased content can cause appreciable errors in the integrated waveforms and explores the unavoidable source of aliasing in both experiment and simulation—the sampling operation. Numerical analysts are often puzzled as to why the integrated acceleration from their simulation does not match the displacement output from the same simulation. This work shows that these strange results can be caused by aliasing induced by interpolation of the model output during sampling regularisation.


*Keywords:* Aliasing; Cubic spline; Integration; Zero shift; Interpolation; Runge–Kutta; Matlab

## 1. Introduction

When integrating acceleration data to obtain estimates of velocity or displacement, errors from a variety of sources can cause the integrated waveforms to drift from the true values, regardless of whether the data was collected experimentally or from a simulation. When experimental data shows strange integration, the drift is usually blamed on the transducer or amplifier saturation. Numerical analysts often suffer the same integration problems. Many have integrated the acceleration output from their model and wondered why it didn't match the displacement output at the same node. Some mistakenly blame the numerical integration algorithm itself. Although the list of potential sources of low frequency error is long, it is the intent of this work to demonstrate an as-of-yet unexplored source of low frequency errors, aliasing.

*Tel.: +505 844 0044.
*E-mail address:* tsedwar@sandia.gov.
[1]Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

0888-3270/$ - see front matter 
doi:10.1016/j.ymssp.2005.08.009

The work presented in this paper demonstrates that aliasing can introduce low-frequency errors in the acceleration waveform whether it is collected experimentally or through simulation. These aliasing errors will cause the integrated waveforms to drift.

If a continuous waveform is represented by a finite collection of samples, aliasing is impossible to prevent. All experimentally collected waveforms of finite duration, and that have been discretised by an analog to digital converter, will contain aliased content to some degree. If the anti-aliasing filter is well-chosen, and the sample rate is high compared to the filter cut-off frequency, then the aliased content can be minimised but not eliminated. Simulation data is also sampled. It, therefore, also, contains aliased content. The aliased content from explicit dynamics finite element simulation can be minimised by requesting the output at every solution increment [1].

Even after correctly requesting the output, the results can still be corrupted by aliasing. Since the explicit solver uses a variable time step, the results are often interpolated to the smallest solution increment to regularise the data. Matlab's implementation of the Runge–Kutta integration also uses a variable time step [2]. The user can request the solution at regular intervals, but this has no effect on the time step. The output at the user-specified increments is an interpolated version of the solution [2].

This work shows that the interpolation of irregularly spaced data will lead to aliasing that can cause the integrated waveforms to drift. This is the likely cause for results where the integration of simulated acceleration data does not match the velocity and displacement output from the same model.

## 2. Source of aliased content

It is never possible to eliminate aliasing completely from measured data. As will be discussed below, this effect comes about from three sources: the infinite bandwidth induced by finite record length, and the spectrum replication caused by representation of the data by its discrete-time samples, and in the case of experimentally measured data, the finite attenuation provided by non-ideal anti-aliasing filters.

It is impossible to experimentally record a signal of infinite duration. Therefore, all recorded signals must have frequency content that approaches infinity due to the convolution theorem. Observing a signal over finite time duration is equivalent to multiplication of that signal with a rectangular window function. According to the convolution theorem [3, p. 26], multiplication of the signal by the rectangular window function in the time domain is equivalent to the inverse transform of the convolution of the signal's transform with the transform of the window function. In this case, the transform of the window function is the sinc function, which is infinite in bandwidth [3, p. 20]. The Fourier transform of the finite-length signal is then given by

$$X_W(f) = X_c(f) * W(f), \tag{1}$$

where $(*)$ denotes convolution. Since $W(f)$ is infinite, the convolution of $W(f)$ with a spectrum of finite length produces a spectrum of infinite length. The magnitude of the content may become small as frequency increases, but it persists nonetheless. Therefore, the act of observing an event over finite time duration induces infinite support in its Fourier spectrum.

The operation of sampling a continuous-time signal, $x_c(t)$, at discrete instances of time with period $T_s$ is equivalent to modulation of the signal with an impulse train. The transform of the sampled signal is then the convolution of the transforms of the impulse train and the continuous-time signal yielding [4, p. 143]

$$X_s(f) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X_c(f - kf_s) \tag{2}$$

due to the sifting property of the impulse train.

Eq. (2) assumes that the area of the impulses in the impulse train is unity. However, in the digital representation, the height of the impulses is unity while the area is equivalent to the sampling interval $T_s$. A true impulse in the digital domain has height $T_s$ so that the area is unity. Because the area of the digital sampling impulses is $T_s$, Eq. (2) becomes [5, p. 68]

$$X_s(f) = \sum_{k=-\infty}^{\infty} X_c(f - kf_s). \tag{3}$$