Signal Processing 130 (2017) 47–56

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Method for finding optimal exponential decay coefficient in numerical Laplace transform for application to linear convolution

Jean-Michel Attendu, Annie Ross

Mechanical Engineering Department, Polytechnique Montreal, CP 6079 Station Centre-ville, Montréal, Que., Canada H3C 3A7

ARTICLE INFO

Article history:
Received 9 November 2015
Received in revised form
24 February 2016
Accepted 31 March 2016
Available online 3 April 2016

Keywords:
Linear convolution
Weighted circular convolution
Generalized Fourier transform
Numerical Laplace transform

ABSTRACT

In this paper, a method based on the numerical Laplace transform is used for calculating the full linear convolution of real or complex signals. An algorithm for obtaining the last N values of the convolution is presented, along with a method for finding an optimal value for the decay coefficient of the transform. It is shown that the use of the numerical Laplace transform formulation allows the calculation of each half of the linear convolution independently, which has computational benefits. The numerical Laplace transform is expressed as the fast Fourier transform of signals that have been premultiplied by a decreasing exponential window characterized by decay coefficient c. The error of the resulting linear convolution depends on the value of the decay coefficient; undervalue results in the generation of wrap-around error whereas overvalue causes amplification of Gibbs phenomenon. In this paper, a formula that optimizes the value of the decay coefficient is developed. A trade-off value for c is obtained and error analysis shows that it outperforms other coefficients proposed in the literature when applied to the calculation of linear convolution. The relative errors obtained are of the order of 10^{-6}% and 10^{-9}% for single and double precisions.

1. Introduction

THE numerical Laplace transform (NLT) was introduced by Wilcox [1] for the analysis of linear systems. It has been used for such motives by others since, notably in the analysis of transients [2–5]. In recent years, a renewed interest has occurred towards the NLT for its ability to improve the computation of linear convolutions.

Linear convolution is usually calculated by applying circular convolution on signals doubled with zero-padding [6]. However, this method increases significantly the amount of data to process, and consequently the computational complexity, especially in the case of 2D or 3D convolutions. Algorithms using the generalized discrete Fourier transform (GDFT) were used in order to obtain full linear convolution at reduced computational cost [7,8]. Babic and Mandic proposed a similar method for obtaining full linear convolution with one circular convolution, by embedding the first non-wrapped samples in the high-range of the full dynamic range given by finite arithmetic precision, and the last N wrapped samples in the low-range, at the cost of sacrificing numerical accuracy [9]. The NLT, a specific case of the GDFT, was also used for such calculation by Martinez [7], but only to obtain the first N convolution values.

In this paper, a method for obtaining full $2N-1$ values of the linear convolution with the NLT is presented. The main asset of the NLT is that it allows a form of control over the periodical repetitions that occur due to the discrete nature of the signal. With this property, it is possible to suppress wrap-around error from circular convolutions. This allows the computation of linear convolution without doubling signals with zero-padding, which is computationally beneficial. With the NLT, it is also possible to obtain the first or last N values of the linear convolution with the calculation of only one circular convolution, unlike the GDFT which requires two calculations when convolving signals with complex values.

A succinct review of the numerical Laplace transform is presented along with some properties and its application to linear convolution. The main benefits of using the NLT for such application are described, along with a method for calculating the full linear convolution of complex signals using the NLT. The core of this paper deals with the elaboration of a simple formula to evaluate the optimized exponential decay coefficient used in the NLT. This coefficient is a trade-off value between the reduction of the wrap-around error, errors due to computer machine and Gibbs phenomenon. Wilcox [1], Wedepohl [3] and Inoue [10] have also worked in order to find such trade-off value. However, their analysis was not aiming at optimizing its application to linear

Abbreviations: NLT, numerical Laplace transform; GDFT, generalized discrete Fourier transform; DTFT, discrete-time Fourier transform; FFT, fast Fourier transform; DFT, discrete Fourier transform

E-mail addresses: jean-michel.attendu@polymtl.ca (J.-M. Attendu), annie.ross@polymtl.ca (A. Ross).

http://dx.doi.org/10.1016/j.sigpro.2016.03.029
0165-1684/© 2016 Elsevier B.V. All rights reserved.

convolution (no such work was found in the literature); rather, it aimed at optimizing the representation of analytical functions in the Laplace domain. Since similar numerical phenomena are involved in both applications, our results are analyzed and compared to those obtained with their methods.

The proposed method is applied to the resolution of an impulsive excitation of a 1D vibrational system, and to the convolution of different simple mathematical signals to bring out the impact of Gibbs phenomenon.

2. Numerical Laplace transform

2.1. Relation to the generalized Fourier transform

The numerical Laplace Transform (NLT) is a special case of generalized discrete Fourier transform (GDFT). For discrete signal $x[n]$, $n=\{0, \ldots, N-1\}$, GDFT is defined as [11]

$$F_\alpha(x[n]) = X_\alpha[k] = \sum_{n=0}^{N-1} x[n]e^{i\alpha n/N}e^{-2\pi ink/N}. \tag{1}$$

Its inverse is

$$F_\alpha^{-1}(X_\alpha[k]) = x[n] = e^{-i\alpha n/N}\sum_{n=0}^{N-1} X_\alpha[k]e^{2\pi ink/N}, \tag{2}$$

where $\alpha\in\mathbb{C}$ is the modulation coefficient, and $k=\{0, \ldots, N-1\}$ is the frequency vector. Eq. (1) is equivalent to the regular discrete Fourier transform (DFT) of signal $x[n]$ initially multiplied with a complex exponential. Similarly, (2) is analogous to the inverse DFT demodulated using the complex exponential of opposite sign. The standard DFT case is obtained with $\alpha = 0$.

As discussed by Martinez et al. [7], Narasimha [8], Babic and Mandic [9] and Inoue et al. [10] in the case of real signals, full $2N-1$ values of linear convolution using GDFT with complex modulation can be obtained by performing only one circular convolution of length N, with specific modulation parameter α in (1). For example, with $\alpha=\pi/2$, concatenation of the real and imaginary parts of the resulting circular convolution produces the linear convolution. This is true only if the convolved signals are real. It results in an improvement of the computational complexity of about 50% in comparison with standard zero-padding method. However, when convolving complex signals, such as when the impulse response of a physical system is complex, two circular convolutions are required to obtain the linear convolution.

2.2. Mathematical formulation

NLT is obtained when the modulation coefficient is chosen to be positive and imaginary ($\alpha = i\bullet c$, $c\in\mathbb{R}, c>0$). Consequently, values of the complex exponential in (1) are real and decrease as n increases. For this reason, c is referred as the “damping” or “decay” coefficient [12] because multiplication of resulting decreasing exponential causes an attenuation of the signal with respect to time.

NLT, expressed with discrete Fourier transform (DFT), is defined as [13]

$$\begin{aligned} NLT(x[n]) = X_c[k] &= \sum_{n=0}^{N-1}\left(x[n]e^{-cn/N}\right)e^{-2\pi ink/N} \\ &= DFT\left(x[n]e^{-cn/N}\right). \end{aligned} \tag{3}$$

The inverse NLT is

$$\begin{aligned} NLT^{-1}(X_c[k]) = x[n] &= e^{cn/N}\sum_{n=0}^{N-1} X_c[k]e^{2\pi ink/N} \\ &= e^{cn/N}DFT^{-1}(X_c[k]). \end{aligned} \tag{4}$$

A main asset of the NLT (and the GDFT) is that by being very close to the DFT, it can be computed using fast Fourier transform (FFT) algorithms, making the process very efficient.

2.3. Property 1: Reduction of the truncation error

Another asset of the NLT is that it attenuates truncation effects in the time domain in the case of any causal signal $x(t) = 0$ for $t<0$. It thereby acts as a substitute to windowing. This property is brought up by expressing the discretization of the analytical signal and applying Laplace transform.

Multiplication of analog signal $x(t)$ with a Dirac comb $\Pi[n]$ is performed to obtain discretization. The Dirac combs in the time and frequency domains are expressed as

$$\Pi(t) = \sum_{n\in\mathbb{Z}}\delta(t-n\Delta t),$$

$$\Pi(\omega) = \frac{2\pi}{\Delta t}\sum_{k\in\mathbb{Z}}\delta\left(\omega-\frac{2\pi k}{\Delta t}\right). \tag{5}$$

By applying the convolution property of the Fourier transform, we obtain the following equation for $x(t)$:

$$\int_{-\infty}^{\infty} X(\omega-\omega')\cdot\Pi(\omega')d\omega' = 2\pi\int_{-\infty}^{\infty} x(t)e^{-\frac{ct}{T}}e^{-i\omega t}\Pi(t)dt. \tag{6}$$

Integrations are simplified by the Dirac combs, and Eq. (6) reduces to:

$$\frac{2\pi}{\Delta t}\sum_{k\in\mathbb{Z}} X\left(\omega-\frac{2\pi k}{\Delta t}\right)=2\pi\sum_{n\in\mathbb{Z}} x(n\Delta t)e^{-\frac{cn}{N}}e^{-i\omega n\Delta t}. \tag{7}$$

As shown by Inoue [10], $X(\omega)$ can be isolated which brings out two error terms:

$$X(\omega) = \hat{X}(\omega) + E_1 + E_2. \tag{8}$$

Here, $\hat{X}(\omega)$ is analogous to the numerical Laplace transform of x:

$$\hat{X}(\omega)=\Delta t\sum_{n=0}^{N-1} x(n\Delta t)e^{-\frac{cn}{N}}e^{-i\omega n\Delta t}. \tag{9}$$

Factor E_1 is the discretization error (or aliasing):

$$E_1(\omega)=-\sum_{k=1}^{\infty}\left[X\left(\omega-\frac{2\pi k}{\Delta t}\right)+X\left(\omega+\frac{2\pi k}{\Delta t}\right)\right]. \tag{10}$$

The periodical repetitions of $X(\omega)$ brought out in (10) are therefore a consequence of discretization in the time domain. The repetitions are spaced proportionally to the sampling frequency.

In most engineering applications, function $X(\omega)$ has a spectral content that tends to zero for high frequency values. Thus, by taking a sufficiently high sampling frequency, E_1 can be neglected.

Factor E_2 is the truncation error:

$$E_2(\omega) = \sum_{n=N}^{\infty} x(n\Delta t)e^{-\frac{cn}{N}}e^{-i\omega n\Delta t}. \tag{11}$$

From (11), it is clear that a high value of c attenuates the effects of the truncation errors.