Considering Transient Effect in Spectrum Analyis

Jürgen Reuter
Karlsruhe,
Germany,
www.soundpaint.org

Abstract

Signal processing with discrete Fourier transform (DFT) works well in standard settings, but is unsatisfying for rapid changes in signal spectra. We illustrate and analyze this phenomenon, develop a novel transform and prove its close relation to the Laplace transform. We deploy our transform for deriving a replacement for the sliding window DFT. Our approach features transient effect and hence shows more natural response to rapid spectral changes.

Keywords

signal processing, DFT, sliding window technique, spectral analysis

1 Introduction

In the 17th century, Christiaan Huygens postulated that each point of an advancing wave front can be viewed as source of a new wave (Fig. 1).

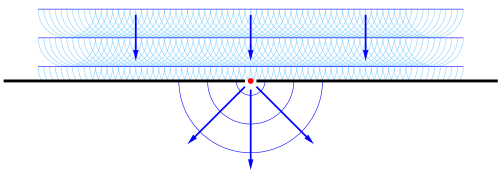

Figure 1: Cutting Out an Elementary Wave from a Wave Front with an Extremely Narrow Slit

This principle together with the principle of superposition suggests that a complex wave can be *decomposed* into elementary waves. However, decomposition is generally ambiguous, as the following equation demonstrates:

$$\cos(x) + \cos(y) = 2\cos(\frac{x-y}{2})\cos(\frac{x+y}{2}) \quad (1)$$

The terms on either side of the equation represent different decompositions for the same wave. On the left, we have a decomposition into two waves of different frequencies x and y, but both with an amplitude of 1 that remains constant over time. The term on the right can be interpreted as a single wave with frequency $\frac{x+y}{2}$ and with amplitude $2\cos(\frac{x-y}{2})$, i.e. with an amplitude that changes periodically over time at a frequency of $x - y$. In acoustics, the interpretation of this equation is known under the term *beats*: If two very similar voices sound at almost the same pitch, the effect is equivalent to having one of the voices sounding with a loudness that oscillates at very low frequency.

If we assume that the amplitudes of the waves remain constant over time, we decide for the left hand side of the equation. It was Fourier who formally proved that, with this additional assumption of *steady state*, there is indeed a unique decomposition, known as Fourier series (for periodic signals) or the Fourier transform (for aperiodic signals).

When in the 1950's techniques for broadcasting audio and video over very high frequency carriers emerged, electrical engineers found that a purely static view (i.e. the steady state) is insufficient. When modeling the effects of printed board circuit layout for highest frequency applications, the *change* of the spectrum over time can no longer be neglected. The Laplace transform turned out as a useful tool for modeling such dynamic environments. Still, the Laplace transform was originally developed to solve higher-order differential equations and never has been adapted for modeling properties of technical signals.

Our mission is to develop a "better" Fourier transform for audio applications that considers transient effect. Our novel *spectral transform* turns out to be closely related to the Laplace transform. For algorithmic approximation, we derive a formula that is very similar to the sliding window DFT technique. Our formula essentially differs only in an additional *decay factor* that smoothly fades out the past content of the signal, rather than subtracting the signal content that falls off from the window, as the DFT sliding window does.

1.1 Paper Outline

After reviewing the state of art (Sect. 2), we explore generic characteristics of linear signal processing at the example of a low pass filter (Sect. 3). Thereupon, we highlight those characteristics that the Fourier transform does not respect by construction (Sect. 3.6). We consequently modify the formula of the Fourier transform to meet our desired characteristics, and thus come up with a new transform, the *spectral transform* (Sect. 4). For checking the sanity of our novel transform, we examine its basic properties (Sect. 4.1). In particular, we prove that our spectral transform boils down to the Laplace transform combined with a linear transform (Sect. 4.2). This way, we kill two birds with one stone: firstly, on a sudden it becomes much clearer, that in electrical engineering, it is indeed often the right choice to use the Laplace transform instead of Fourier transform; secondly, we are much more confident that our spectral transform works as expected, since it is strongly related to what electrical engineers have been doing for half a century. Still, our goal is to exploit the spectral transform for retrieving new or better algorithms in signal processing. For this purpose, we derive a recursive formula, that serves as the pendant for the sliding window DFT recursive formula (Sect. 4.3). We evaluate the differences between the two formulas and their impact (Sect. 5). Still, our work is far away from being complete, since we expect the spectral transform to eventually gain much broader application (Sect. 6). In particular, we expect to derive new or enhanced signal processing algorithms (e.g. filters) on the basis of the spectral transform. We close with a short summary of our findings (Sect. 7).

2 State of the Art

The sliding window DFT algorithm [1; 2] implements a discrete approximation of the Fourier transform and delivers the coefficients for the series of the partials of a periodic wave. Since real world acoustic signals are usually not periodic, a *window function* is used to mask a frame of limited time and ignore the signal outside of this time frame (Fig. 2a). Instead, the signal within the frame is extrapolated by periodic continuation suitable for the DFT (Fig. 2b). However, the discontinuities at the continuation points would add artificial harmonic content with a base frequency determined by the size of the window. Therefore, the signal is usually faded out towards the window borders, e.g. by multiplying the signal with a Gauss distribution curve (Fig. 2c).

The sliding window DFT works fairly well for the *static case*: As long as the spectrum of the signal changes slowly over time, it does not much matter where exact the spectral analysis starts and where it ends. Moreover, assuming a sufficiently long window size that covers deep frequencies, the Gauss distribution curve attenuates the amount of all frequencies almost evenly. However, in the *dynamic case*, the sliding window responds slowly to rapid changes in the spectrum, since the Gaussian filter attenuates particularly the most recent samples.

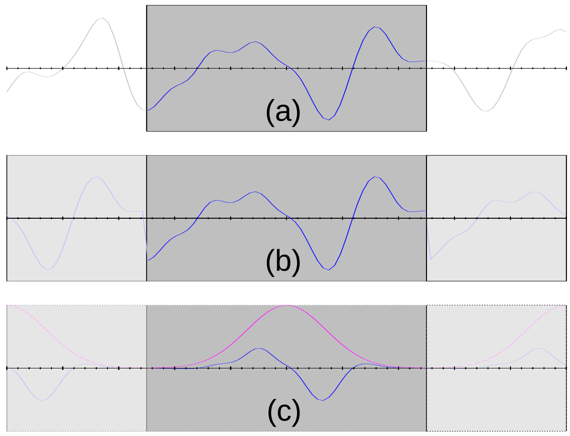

Figure 2: Preparation of a Signal for DFT

On last year's LAC, FFITCH [3; 4] presented a Csound implementation of the sliding window DFT based on the recursive function $F_{t+1}(n) = (F_t(n) - f_t + f_{t+N})e^{2\pi i \frac{n}{N}}$. This function still inherits the steady state assumption from the Fourier transform. Our goal is to develop and implement a transform for spectral analysis that responds quickly to rapid changes and behaves more closely to linear filters.

3 Case Study: RC Low Pass Filter

We need to know more about the properties of signal spectra. Intuitively, we consider the spectrum of a complex signal as decomposition of the signal into a weighted (possibly infinite) sum of primitive components (the *spectral lines*). That is, more formally, the primitive components are base vectors, and the complex signal is a linear combination of them. Therefore, there is a linear relation between signals and spectra: if signal $f_1(t)$ has spectrum $\mathcal{S}_1(t)$ and $f_2(t)$ has spectrum $\mathcal{S}_2(t)$, then we expect

$f_{1+2}(t) := f_1(t) + f_2(t)$ to have the spectrum $\mathcal{S}_{1+2}(t) := \mathcal{S}_1(t) + \mathcal{S}_2(t)$.

Consequently, we choose a linear filter to study the characteristics of linear signal processing. To keep mathematics feasible, we choose a very simple linear filter. The probably most simple yet non-trivial linear filter is the *RC low pass filter*.

3.1 Filter Diagram

Fig. 3 shows the diagram of the electrical implementation of an RC low pass filter with resistor R and capacitor C. We denote in short $\tau = \mathrm{RC}$ if we do not want to differentiate between R and C. For further simplification, we assume that the incoming signal U_{in} has negligible low impedance, and that the load connected to the output of the filter has negligible high impedance. In practice, this behavior can be fairly approximated by adding amplifiers into the signal flow.

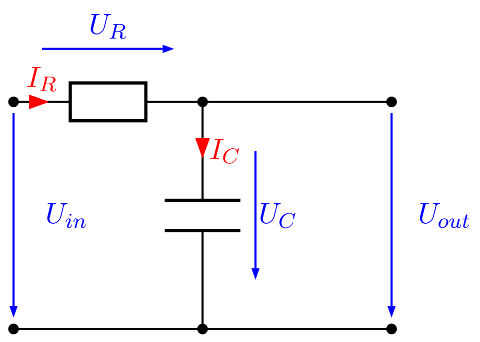

Figure 3: RC Low Pass Filter

3.2 Natural Response Function

We want to express the filter's output signal $\mathrm{U_{out}}(t)$ in terms of its input signal $\mathrm{U_{in}}(t)$. With $\mathrm{U_{out}}(t) = \mathrm{U_{in}}(t) - \mathrm{U_R}(t)$, $\mathrm{U_R}(t) = \mathrm{RI_R}(t)$, $\mathrm{I_R}(t) = \mathrm{I_C}(t) = \mathrm{dQ_C}(t)/\mathrm{d}t$, and $\mathrm{Q_C}(t) = \mathrm{CU_C}(t) = \mathrm{CU_{out}}(t)$, we get in summary $\mathrm{U_{out}}(t) = \mathrm{U_{in}}(t) - \tau\frac{\mathrm{dU_{out}}(t)}{\mathrm{d}t}$, that is

$$\mathrm{U}'_{\mathrm{out}}(t) = \frac{1}{\tau}(\mathrm{U_{in}}(t) - \mathrm{U_{out}}(t)).$$

This is an ordinary linear differential equation of first order. A solution for this equation can be found in standard literature on differential equations [5]: Assuming there is some initial value $\mathrm{U_{out}}(t_0), t_0 \leq t_1$ given (i.e. the capacitor's charge at some earlier point in time), the equation has the only solution

$$\mathrm{U_{out}}(t_1) = e^{\frac{t_0 - t_1}{\tau}}\mathrm{U_{out}}(t_0) + \frac{1}{\tau}\int_{t_0}^{t_1} \mathrm{U_{in}}(t) e^{\frac{t-t_1}{\tau}} \mathrm{d}t \tag{2}$$

Note that for computing $\mathrm{U_{out}}(t_1)$ it is sufficient to know the output $\mathrm{U_{out}}(t_0)$ at some earlier point in time $t_0 < t_1$ and the input signal in the range of time (t_0, t_1). That is, the capacitor's charge represents all of the signal's past history as to what extent it is required for low pass filtering.

For the special case $t_0 \to -\infty$, the equation simplifies as follows:

$$\mathrm{U_{out}}(t_1) = \frac{1}{\tau}\int_{-\infty}^{t_1} U_{\mathrm{in}}(t) e^{\frac{t-t_1}{\tau}} \mathrm{d}t \tag{3}$$

3.3 Steady State Transfer Function

We examine the transfer function of the RC low pass by computing the natural response of the filter when feeding a sine wave into its input. As sine wave, we could use the function $\mathrm{U_{in}}(t) = \rho \sin(\phi t + \varphi)$ with sine amplitude ρ and phase shift φ. However, since calculating with trigonometric functions is cumbersome, we prefer the complex notation $\mathrm{U_{in}}(t) = \rho(i \sin(\phi t + \varphi) + cos(\phi t + \varphi)) = \rho e^{i(\phi t + \varphi)}$. Since we examine the steady state, we do not want to consider any initial value of the filter. Consequently, we compute the transfer function based on Eqn. 3 rather than on Eqn. 2:

$$\begin{aligned}
\mathrm{U_{out}}(t_1) &= \frac{1}{\tau}\int_{-\infty}^{t_1} \rho e^{i(\phi t + \varphi)} e^{\frac{t-t_1}{\tau}} \mathrm{d}t \\
&= \frac{1}{\tau}\rho e^{i\varphi - t_1/\tau} \left[\frac{\tau}{i\phi\tau + 1} e^{\frac{t(i\phi\tau+1)}{\tau}}\right]_{t=-\infty}^{t=t_1} \\
&= \frac{1}{\tau}\rho e^{i\varphi - t_1/\tau} \frac{\tau}{i\phi\tau + 1} e^{\frac{t_1(i\phi\tau+1)}{\tau}} \\
&= \frac{1}{i\phi\tau + 1}\rho e^{i(\phi t_1 + \varphi)} \\
&= \frac{1}{i\phi\tau + 1}\mathrm{U_{in}}(t_1) = \ldots = \\
&= \frac{e^{-i\arctan(\phi\tau)}}{\sqrt{1 + (\phi\tau)^2}}\mathrm{U_{in}}(t_1)
\end{aligned} \tag{4}$$

That is, the signal is attenuated by factor $\sqrt{1 + (\phi\tau)^2}^{-1}$ and phase-shifted by $\arctan(\phi\tau)$. (The last step in the calculation comprises a complex partial fraction expansion and some other conversions, that we have omitted here for space restrictions.)

Fig. 4 shows the input sine wave @ 1kHz and the resulting steady state output sine wave for $\tau = 3ms$ with observable decreased amplitude and phase shift.

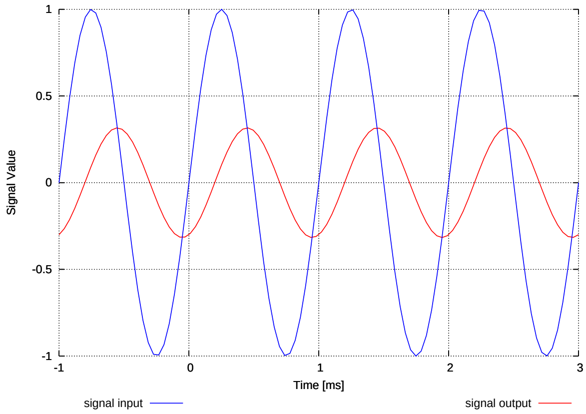

Figure 4: RC Low Pass Steady State Transfer

3.4 Transient Effect

Imagine that after a long period of a null signal, a sine wave suddenly sets in. That is, we examine the RC filter response to the function:

$$\mathrm{U_{in}}(t) := \begin{cases} \sin(t) & \text{if } t \geq 0, \\ 0 & \text{otherwise.} \end{cases} \quad (5)$$

Since $\mathrm{U_{in}}(t) = 0 \forall t < 0$, the integral in Eqn. 3 is 0 $\forall t_1 < 0$, that is, $\mathrm{U_{out}}(t_1) = 0\ \forall t_1 < 0$. For the remaining case $t_1 \geq 0$ we derive

$$\begin{aligned} \mathrm{U_{out}}(t_1) &= \frac{1}{\tau} e^{\frac{-t_1}{\tau}} \int_{-\infty}^{t_1} \mathrm{U_{in}}(t) e^{\frac{t}{\tau}} \mathrm{d}t \\ &= \frac{1}{\tau} e^{\frac{-t_1}{\tau}} \int_{0}^{t_1} \sin(t) e^{\frac{t}{\tau}} \mathrm{d}t \quad (6) \\ &= \ldots \text{ (twice integration by parts) } \ldots \\ &= -e^{\frac{-t_1}{\tau}} \left[\frac{1}{\tau} \cos(t) e^{\frac{t}{\tau}} - \frac{1}{\tau^2} \sin(t) e^{\frac{t}{\tau}} \right]_{t=0}^{t=t_1} \\ &\quad - e^{\frac{-t_1}{\tau}} \frac{1}{\tau^3} \int_{0}^{t_1} \sin(t) e^{\frac{t}{\tau}} \mathrm{d}t. \quad (7) \end{aligned}$$

By isolating the integral in Eqns. 6 and 7, it can be eliminated, leading to $\mathrm{U_{out}}(t) = \frac{\tau}{\tau^2+1}(\frac{1}{\tau}\sin(t) - \cos(t) + e^{\frac{-t}{\tau}})$ for $t \geq 0$. In summary, we have

$$\mathrm{U_{out}}(t) = \begin{cases} \frac{\sin(t) - \tau\cos(t) + \tau e^{\frac{-t}{\tau}}}{\tau^2+1} \forall t \geq 0, \\ 0 \text{ otherwise.} \end{cases} \quad (8)$$

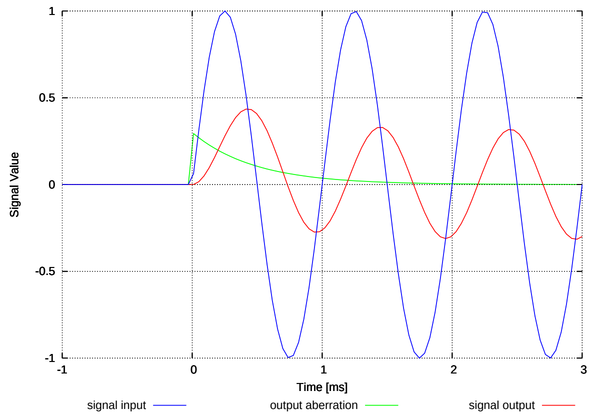

Figure 5: RC Low Pass Transient Effect

Fig. 5 illustrates the transient effect: In the steady state, the output signal is negative, when the sine wave traverses the zero crossing upwards. However, when the sine wave suddenly sets in, the charge of the capacitor is initially zero rather than negative. Consequently, the output signal starts somewhat too high, but approaches asymptotically to the steady state. In Fig. 5, we additionally depict this output aberration, that asymptotically drops to zero. Similarly, the phase of the output signal approaches asymptotically to the steady state.

3.5 Discretization

For algorithmic implementation of the RC low pass, we assume that the input signal is approximated by a discrete series of floating point values $U_{in}(t_0), U_{in}(t_1), U_{in}(t_2), \ldots, U_{in}(t_n)$ that are equidistant over time: $t_1 - t_0 = t_2 - t_1 = \ldots = t_n - t_{n-1} = \Delta t$. Approximately, we assume $\mathrm{U}_{\mathrm{in}\_approx}(t) \equiv \mathrm{U_{in}}(t_i) \forall t \in [t_i, t_{i+1})$. Then, from Eqn. 2 follows:

$$\begin{aligned} &\mathrm{U}_{\mathrm{out}\_approx}(t_1) \\ =& e^{\frac{t_0 - t_1}{\tau}} \mathrm{U}_{\mathrm{out}\_approx}(t_0) \\ &+ \frac{1}{\tau} \int_{t_0}^{t_1} \mathrm{U}_{\mathrm{in}\_approx}(t) e^{\frac{t-t_1}{\tau}} \mathrm{d}t \\ =& e^{-\Delta t/\tau} \mathrm{U}_{\mathrm{out}\_approx}(t_0) + \frac{1}{\tau} \int_{t_0}^{t_1} \mathrm{U_{in}}(t_0) e^{\frac{t-t_1}{\tau}} \mathrm{d}t \\ =& e^{-\Delta t/\tau} \mathrm{U}_{\mathrm{out}\_approx}(t_0) + \mathrm{U_{in}}(t_0) \frac{1}{\tau} \left[\tau e^{\frac{t-t_1}{\tau}} \right]_{t=t_0}^{t=t_1} \\ =& e^{-\Delta t/\tau} \mathrm{U}_{\mathrm{out}\_approx}(t_0) + \mathrm{U_{in}}(t_0)(1 - e^{-\Delta t/\tau}) \\ =& \alpha \mathrm{U}_{\mathrm{out}\_approx}(t_0) + (1-\alpha)\mathrm{U_{in}}(t_0) \quad (9) \end{aligned}$$

for $\alpha = e^{-\Delta t/\tau}$. Therein, $\omega = 2\pi f = 1/\tau$ represents the cut-off frequency f [Hz] of the low pass and Δt the time [s] between two adjacent samples.

For an even more precise approximation, we can *linearly* interpolate $\mathrm{U_{in\_}}_{approx}(t), t_i \leq t \leq t_{i+1}$ between the adjacent samples $\mathrm{U_{in\_}}_{approx}(t_i) = \mathrm{U_{in}}(t_i)$ and $\mathrm{U_{in\_}}_{approx}(t_{i+1}) = \mathrm{U_{in}}(t_{i+1})$ rather than assuming $\mathrm{U_{in\_}}_{approx}(t) \equiv \mathrm{U_{in}}(t_i) \forall t \in [t_i, t_{i+1})$. This way, we get

$$\begin{aligned}
&\mathrm{U_{out_}}_{approx}(t_1) \\
=&e^{\frac{t_0-t_1}{\tau}} \mathrm{U_{out_}}_{approx}(t_0) \\
&+ \frac{1}{\tau} \int_{t_0}^{t_1} (\mathrm{U_{in}}(t_0) + \\
&\frac{t-t_0}{t_1-t_0} (\mathrm{U_{in}}(t_1) - \mathrm{U_{in}}(t_0))) e^{\frac{t-t_1}{\tau}} \mathrm{d}t \\
=&\ldots \text{ (quite lengthy calculation) } \ldots \\
=&e^{\frac{t_0-t_1}{\tau}} \mathrm{U_{out_}}_{approx}(t_0) \\
&- \left(e^{\frac{t_0-t_1}{\tau}} + \frac{\tau(e^{\frac{t_0-t_1}{\tau}} - 1)}{t_1 - t_0} \right) \mathrm{U_{in}}(t_0) \\
&+ \left(1 + \frac{\tau(e^{\frac{t_0-t_1}{\tau}} - 1)}{t_1 - t_0} \right) \mathrm{U_{in}}(t_1) \qquad (10)
\end{aligned}$$

for a marginally better approximation.

3.6 Discussion

Having the steady state transfer function in mind, the RC low pass filter can be viewed as a linear operation not only on the signal itself, but also on its spectrum. Performing such linear operations is often realized by transforming a signal into frequency space, then applying the operation, and then transforming back to signal space. This is, where the Fourier integral (and its inverse) is typically used:

$$\mathcal{F}(f(t), \phi) := \int_{-\infty}^{+\infty} f(t) e^{-2\pi i \phi t} \mathrm{d}t \qquad (11)$$

However, the RC low pass filter function

$$\mathrm{U_{out}}(t_1) = \frac{1}{\tau} \int_{-\infty}^{t_1} U_{\mathrm{in}}(t) e^{\frac{t-t_1}{\tau}} \mathrm{d}t$$

differs from the Fourier integral in some significant properties:

Time as Parameter. The Fourier transform is applied on an input function as a whole and has a frequency as parameter, but not a point of time. That is, the Fourier transform assumes a static spectrum that does not change over time. Instead, a window is introduced and the signal in this window is periodically extrapolated to retrieve a spectrum that is bound to a limited range in time. In contrast, the RC low pass filter output is a function over time.

Past Time Contribution. Since the Fourier transform is designed to compute the static spectrum over the complete signal, it requires the future course of the signal. The window technique constructs a future signal by periodically extrapolating the signal. In contrast, the output of the RC low pass filter can for some point $t = t_1$ of time be sufficiently expressed in terms of a function over the input signal ranging from $t = -\infty$ to $t = t_1$. It is *not* necessary to know or extrapolate the *future* course of the signal.

Exponential Decay. The Fourier transform weights the input function with factor $e^{2\pi i \phi t} =: w(t)$. Since $|w(t)| = 1 \forall t, \phi \in \mathbb{R}$, all function values of the signal contribute equally to the transform, regardless of what point of time they represent. In contrast, the signal's contribution to the RC low pass output decays exponentially over time: the signal's recent history has much more impact on the filter output than the signal's history long ago.

4 The Spectral Transform

We present and examine the *spectral transform* that modifies the Fourier transform such that it eliminates all deficiencies discussed in Sect. 3.6:

$$\begin{aligned}
&\mathcal{S}(f(t), \phi, t_0) := \\
&(\mu - 2\pi i)\phi \int_{-\infty}^{t_0} f(t) e^{(\mu - 2\pi i)\phi(t-t_0)} \mathrm{d}t \qquad (12)
\end{aligned}$$

The parameter t_0 represents the point of time of the spectrum. The upper bound of the integral is t_0, such that the future signal is not involved. Lastly, the e function, that performs only a complex rotation on the input signal in the Fourier transform, is augmented with the positive real value μ, such that the signal is also attenuated to honor exponential decay as we observed in the RC low pass. The transform is scaled by factor $(\mu - 2\pi i)\phi$ to simplify some of the subsequent expressions.

For simplicity, we often normalize $f(t)$ such that t_0 becomes 0:

$$\mathcal{S}(f(t),\phi) := (\mu - 2\pi i)\phi \int_{-\infty}^{0} f(t) e^{(\mu-2\pi i)\phi t} \mathrm{d}t. \quad (13)$$

The normalized form is related to the full form as follows:

$$\mathcal{S}(f(t),\phi) = \mathcal{S}(f(t-t_0),\phi,t_0). \quad (14)$$

Similar to the recursive RC low pass (Eqn. 2) expression, the spectral transform can be expressed recursively as follows:

$$\begin{aligned}
&\mathcal{S}(f(t),\phi,t_1)\\
=&(\mu - 2\pi i)\phi \int_{-\infty}^{t_0} e^{(\mu-2\pi i)\phi(t-t_1)} f(t)\mathrm{d}t\\
&+ (\mu - 2\pi i)\phi \int_{t_0}^{t_1} e^{(\mu-2\pi i)\phi(t-t_1)} f(t)\mathrm{d}t\\
=&e^{(\mu-2\pi i)\phi(t_0-t_1)}\\
&(\mu - 2\pi i)\phi \int_{-\infty}^{t_0} e^{(\mu-2\pi i)\phi(t-t_0)} f(t)\mathrm{d}t\\
&+ (\mu - 2\pi i)\phi \int_{t_0}^{t_1} e^{(\mu-2\pi i)\phi(t-t_1)} f(t)\mathrm{d}t\\
=&e^{(\mu-2\pi i)\phi(t_0-t_1)} \mathcal{S}(f(t),\phi,t_0)\\
&+ (\mu - 2\pi i)\phi \int_{t_0}^{t_1} e^{(\mu-2\pi i)\phi(t-t_1)} f(t)\mathrm{d}t. \quad (15)
\end{aligned}$$

4.1 Basic Properties

We give some basic properties of the spectral transform. For space restrictions, we omit all (not so difficult) proofs.

Linearity.

$$\mathcal{S}(a_1 f_1(t) + a_2 f_2(t),\phi,t_0) = a_1\mathcal{S}(f_1(t),\phi,t_0) + a_2\mathcal{S}(f_2(t),\phi,t_0). \quad (16)$$

Convolution.

$$\mathcal{S}((f_1 * f_2)_{t_0}(t),\phi,t_0) = \frac{1}{2\pi\phi}\mathcal{S}(f_1(t),\phi,-t_0)\mathcal{S}(f_2(t),\phi,t_0). \quad (17)$$

Differentiation.

$$\mathcal{S}(f^{(n)}(t),\phi,t_0) = ((2\pi i - \mu)\phi)^n \mathcal{S}(f^{(0)}(t),\phi,t_0) + 2\pi\phi \sum_{j=0}^{n-1} ((2\pi i - \mu)\phi)^{n-1-j} f^{(j)}(t_0). \quad (18)$$

Time Shifting.

$$\mathcal{S}(f(t+\xi),\phi,t_0) = e^{-(\mu-2\pi i)\phi\xi}\mathcal{S}(f(t),\phi,t_0) + (\mu - 2\pi i)\phi \int_{t_0}^{\xi} e^{(\mu-2\pi i)\phi(r-\xi-t_0)} f(r)\mathrm{d}r \quad (19)$$

$\forall \xi \in \mathbb{R}, \xi \geq 0.$

Scaling.

$$\mathcal{S}(f(t),\frac{\phi}{a},t_0) = \mathcal{S}(g(t),\phi,t_0), \quad (20)$$

with $g(t) = f((t-t_0)a + t_0) \forall a \in \mathbb{R}, a > 0.$

Frequency Shifting.

$$\mathcal{S}(f(t),\phi+\alpha,t_0) = \frac{\phi+\alpha}{\phi}\mathcal{S}(e^{(\mu-2\pi i)\alpha(t-t_0)} f(t),\phi,t_0). \quad (21)$$

4.2 Relation to Laplace Transform

The spectral transform is closely related to the Laplace transform:

$$\mathcal{L}(g(r),s) = \frac{1}{(\mu - 2\pi i)\phi}(\mathcal{S}(f(t),\phi)) \quad (22)$$

for $s = (\mu - 2\pi i)\phi$, $r = -t$, and $g(r) \equiv f(t)$.

Proof.

$$\begin{aligned}
\mathcal{L}(g(r),s) &= \int_0^{\infty} e^{-sr} g(r)\mathrm{d}r\\
&= \int_0^{\infty} e^{-(\mu-2\pi i)\phi r} g(r)\mathrm{d}r\\
&= -\int_0^{-\infty} e^{(\mu-2\pi i)\phi t} g(r)\mathrm{d}t\\
&= \int_{-\infty}^{0} e^{(\mu-2\pi i)\phi t} g(r)\mathrm{d}t\\
&= \int_{-\infty}^{0} e^{(\mu-2\pi i)\phi t} f(t)\mathrm{d}t\\
&= \frac{1}{(\mu - 2\pi i)\phi}(\mathcal{S}(f(t),\phi)).
\end{aligned}$$

This close relation to Laplace explains on the one hand why electric engineers are so successful in deploying the Laplace transform (although our results suggests that they require some additional pre- and post-transform). On the other hand, we are now confident that we are on the right track with our transform, since we inherit those important features from Laplace, that electric engineers claim to be essential when considering the non-steady state.

We also can now compute the inverse spectral transform from the inverse Laplace transform by substituting $r = -t$, $g(r) = f(t)$, $s = (\mu - 2\pi i)\phi$, $ds = (\mu - 2\pi i)d\phi$:

$$\begin{aligned}
g(r) =& \frac{1}{2\pi i}\int_{c-i\infty}^{c+i\infty} e^{sr}\mathcal{L}(g(r), s)\mathrm{d}s \\
=& \frac{1}{2\pi i}\int_{c-i\infty}^{c+i\infty} e^{sr}\frac{1}{(\mu - 2\pi i)\phi}(\mathcal{S}(f(t), \phi))\mathrm{d}s \\
f(t) =& \frac{1}{2\pi i}\int_{c-i\infty}^{c+i\infty} e^{(\mu-2\pi i)\phi(-t)} \\
& (\frac{1}{(\mu - 2\pi i)\phi}(\mathcal{S}(f(t), \phi)))(\mu - 2\pi i)\mathrm{d}\phi \\
=& \frac{1}{2\pi i}\int_{c-i\infty}^{c+i\infty} e^{-(\mu-2\pi i)\phi t}\frac{(\mathcal{S}(f(t), \phi))}{\phi}\mathrm{d}\phi
\end{aligned} \tag{23}$$

Transfer Function for $\sin(\alpha t)$**.** $\mathcal{S}(\sin(\alpha t), \phi)$ evaluates (after twice integration by parts) to $\frac{-(\mu-2\pi i)\phi\alpha}{(\mu-2\pi i)^2\phi^2+\alpha^2}$. Alternatively, knowing from Laplace transform tables that $\mathcal{L}(g(r), s) = \frac{-\alpha}{s^2+\alpha^2}$ for $g(r) = -\sin(\alpha r)$, with Eqn. 22, $s = (\mu - 2\pi i)\phi$, $r = -t$ and $g(r) = -\sin(\alpha r) = -\sin(\alpha(-t)) = \sin(\alpha t) = f(t)$, we easily get the same result $\mathcal{S}(\sin(\alpha t), \phi) = \frac{-(\mu-2\pi i)\phi\alpha}{(\mu-2\pi i)^2\phi^2+\alpha^2}$.

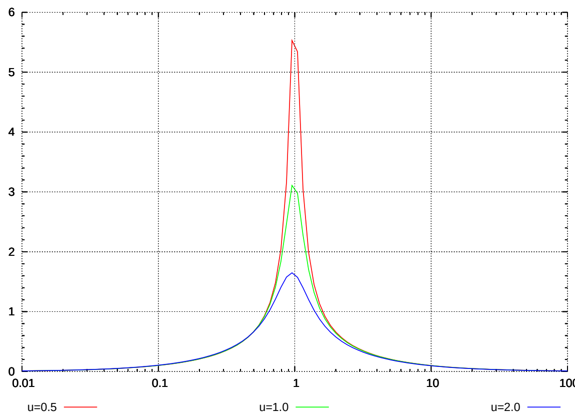

Figure 6: Spectral Transform of $sin(2\pi t)$

Fig. 6 shows the absolute values of $\mathcal{S}(\sin(\alpha t), \phi)$ for various μ and $\alpha = 2\pi$. A high value for μ increases the sharpness of the frequency peak, but also results in faster decay of the signal. After all, we may tune μ such that the peaks fit well into a given set of discrete frequency lines.

4.3 Discretization

Similar to the RC low pass, we discretize the spectral transform to derive an algorithmic approximation. For two successive samples at times t_0 and t_1 with $\Delta t = t_1 - t_0$ we derive

$$\begin{aligned}
\mathcal{S}(f(t), \phi, t_1) =& e^{-(\mu-2\pi i)\phi\Delta t}\mathcal{S}(f(t), \phi, t_0) + \\
& (1 - e^{-(\mu-2\pi i)\phi\Delta t})f(t_0),
\end{aligned} \tag{24}$$

assuming that $f(t) \equiv f(t_0)\ \forall t : t_0 \le t < t_1$.

Proof. The discretization follows immediately from the recursive form of the spectral transform (Eqn. 15):

$$\begin{aligned}
& \mathcal{S}(f(t), \phi, t_1) \\
=& e^{(\mu-2\pi i)\phi(t_0-t_1)}\mathcal{S}(f(t), \phi, t_0) \\
& + (\mu - 2\pi i)\phi\int_{t_0}^{t_1} e^{(\mu-2\pi i)\phi(t-t_1)}f(t)\mathrm{d}t \\
=& e^{-(\mu-2\pi i)\phi\Delta t}\mathcal{S}(f(t), \phi, t_0) \\
& + (\mu - 2\pi i)\phi f(t_0)\left[\frac{e^{(\mu-2\pi i)\phi(t-t_1)}}{(\mu - 2\pi i)\phi}\right]_{t=t_0}^{t=t_1} \\
=& e^{-(\mu-2\pi i)\phi\Delta t}\mathcal{S}(f(t), \phi, t_0) \\
& + f(t_0)(1 - e^{-(\mu-2\pi i)\phi\Delta t})
\end{aligned} \tag{25}$$

Interestingly, Douglas and Soh [6] scale the signal in their sliding window DFT implementation with factor $0 \ll \hat{\lambda} < 0$, that resembles our $e^{\mu\phi t}$ for $t < 0$. Still, their focus is on numerical stability of the DFT rather than transient effect. Also, given an L-sized window, they remove the sample that falls off the sliding window, scaled by $\hat{\lambda}^L$. Instead, we do not have a limited window size, such that $\lim_{L\to\infty}\hat{\lambda}^L = 0$ and there is nothing to remove.

5 Evaluation

For evaluating our work, we developed a DSP library on a Linux system with JDK 1.6 that implements the spectral transform. The source code is available at `http://www.soundpaint.org/spectral-transform/`.

Unfortunately, we have not yet developed a sufficiently persuasive implementation of the inverse spectral transform, presumably for issues similar to that of the inverse sliding DFT [3]. Instead, we compare the graphical representation of both, the sliding DFT and our spectral transform (Fig. 8, 9) for a slap noise (Fig. 7) recorded at 44.1kHz. We chose a DFT window size of $N := 512$, and $\log(\mu) := (N-1)/N$.

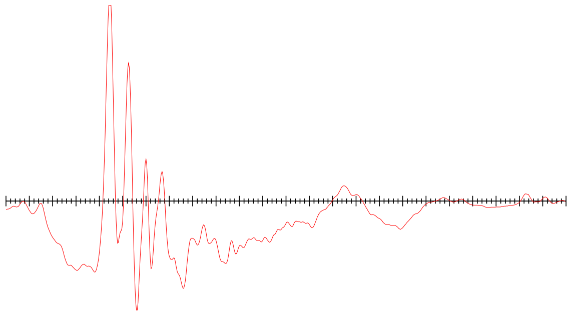

Figure 7: Slap Noise

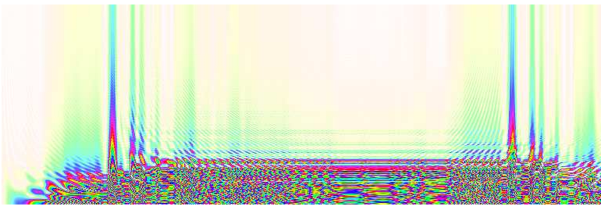

Figure 8: Sliding DFT of Slap Noise

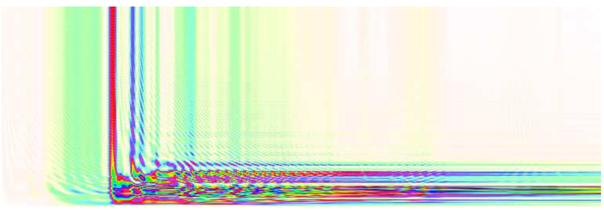

Figure 9: Spectral Transform of Slap Noise

No attempt was made to apply the Gaussian filter to either DFT or spectral transform (Sect. 2), hence the sliding DFT also reacts promptly to rapid changes, as we demanded. However, the DFT tends to produce overly sharp lines, that lead to a speckled spectrum. Moreover, when the initial samples fall off from the DFT window, the initial spectrum peak is echoed. In contrast, our spectral transform produces more balanced frequency areas of excitation and fades out smoothly, while the initial peak is even broader than in the DFT.

6 Future Work

The presumably most annoying issue is that our transform is still missing its discrete inverse implementation. Signal manipulation in the frequency domain appears only useful, when the modified signal can be reconstructed from the frequency domain. We are confident that we will find an adequate solution, since our transform is closely related to sliding DFT and Laplace, both of which are known to be invertible.

7 Conclusion

Originally inspired by the wave theory of light, we developed our spectral transform and implemented it under Linux as a small Java library for the field of audio signal processing, but expect broad applicability in various fields of physics. The transform shifts uncertainty from signal to frequency domain by constructing a dynamically adapting spectrum for a any sharply defined signal at any sharp point of time. Compared with DFT, the spectral transform gives more natural results in environments when the spectrum changes rapidly. The drawback is that it lacks of determining sharp spectral lines, as can be constructed from steady state signals. This behavior is a natural tribute to the dynamically changing spectrum. Rather, our transform encounters the full signal history with exponential decay, not just the steady state extrapolation of a signal section, as the DFT does.

References

[1] Lawrence R. Rabiner and Bernard Gold. *Theory and Application of Digital Signal Processing*. Prentice Hall, 1975.

[2] John J. Shynk. Frequency-domain and multirate adaptive filtering. *IEEE Signal Processing Magazine*, 9(1), January 1992.

[3] Russell Bradford, Richard Dobson, and John ffitch. Sliding is smoother than jumping. In *Proceedings of the ICMC 2005*, pages 287–290, Tampere, Finland, September 2005. SuviSoft Oy Ltd.

[4] John ffitch. Sliding DFT for fun and musical profit. In *Proceedings of the 6th International Linux Audio Conference (LAC2008)*, Köln, Germany, Feb–Mar 2008. Kunsthochschule für Medien Köln.

[5] Wolfgang Walter. *Gewöhnliche Differentialgleichungen: eine Einführung*. Springer Lehrbuch. Springer Verlag, Berlin, Heidelberg, New York, 4th edition, 1989.

[6] S.C. Douglas and J.K. Soh. A numerically-stable sliding-window estimator and its applicationto adaptive filters. In *Conference Record of the Thirty-First Asilomar Conference on Signals, Systems & Computers*, volume 1, pages 111–115, November 1997.