Numerical Hilbert Transform Algorithm for Causal Interpolation of Functions Represented by Cubic and Exponential Splines

DUŠAN N. GRUJIĆ, (Senior Member, IEEE)
School of Electrical Engineering, University of Belgrade, 11000 Belgrade, Serbia
e-mail: dusan.grujic@etf.bg.ac.rs

ABSTRACT
This paper presents an algorithm for numerical Hilbert transform of functions represented by cubic and exponential splines, which is suitable for causal interpolation of data spanning several frequency decades. It does not suffer from excessive number of data points due to large frequency span, and does not exhibit aliasing, both of which are characteristic for Hilbert transform algorithms based on FFT. The proposed algorithm was used for causal interpolation of characteristic impedance of microstrip line over conductive substrate, and the results were compared to the output of CausalCat.

INDEX TERMS
Causality, Hilbert transform, interpolation.

I. INTRODUCTION
Physical systems are causal, and as a consequence real and imaginary parts of physical complex-valued quantities, such as permittivity $\epsilon = \epsilon' + j\epsilon''$, characteristic impedance $Z_0$ etc., are not independent, but are related by Kramers-Kroning relations [1]. Independent interpolation of real and imaginary parts will violate the Hilbert relations and result in a non-causal response. To preserve causality, only one quantity of the Hilbert transform pair should be interpolated, while the other one should be calculated by the Hilbert transform. In this way the Kramers-Kronig relations are enforced and the causality is preserved.

The importance of Hilbert transform resulted in multitude of algorithms [2]–[6]. Most general-purpose Hilbert transform algorithms are based on the Fast Fourier Transform (FFT), which require fixed frequency step, and are prone to aliasing. Since causal interpolation is usually performed on data spanning several frequency decades, the use of fixed frequency step in FFT-based algorithms results in excessive number of data points. Although FFT is very efficient, excessive number of data points can result in slow computation as shown in [6].

Additional pre- and post-processing of data might be necessary when using FFT-based algorithms to prevent aliasing, such as constructing a function which ensures that input data is periodic, as was done in CausalCat [7]. Construction of additional functions is problem-dependent and has to be addressed on case-by-case basis.

The algorithm proposed in this paper allows evaluation of Hilbert transform at any frequency. This property allows efficient evaluation of Hilbert transform on logarithmic frequency scale over several decades, where fixed frequency step algorithms would fail due to excessive number of data points. Furthermore, the proposed algorithm does not exhibit aliasing, so no pre- or post-processing is necessary. It can be viewed as an extension of [6] which allows the use of exponential splines to eliminate the undesired inflexion points which occur if only cubic splines are used for interpolation. As in [6], the Hilbert transform of interpolating function is exact, so the quality of results depends solely on how well the interpolating function describes the input data.

The paper is organized as follows. A short review of exponential splines necessary for derivation of the proposed algorithm is given in Sec. II. Derivation of the proposed algorithm is presented in Sec. III, followed by consideration of special cases in Sec. IV. Example of using the proposed algorithm for causal interpolation of characteristic impedance of microstrip line over conductive substrate and comparison with published results is given in Sec. VI. Conclusion and final remarks are in Sec. VII. Some properties of the Hilbert transform used in derivation are given in the Appendix.

II. REVIEW OF EXPONENTIAL SPLINES INTERPOLATION FUNCTIONS
Exponential splines [8]

$$f_i(\omega) = A_i + B_i(\omega - \omega_i) + C_i e^{\sigma_i(\omega - \omega_i)} + D_i e^{-\sigma_i(\omega - \omega_i)} \quad (1)$$
can be used to avoid inflexion points which occur when polynomial interpolating functions are used. Coefficients of \(i\)-th exponential spline \(A_i, B_i, C_i\) and parameter \(\sigma_i\) are determined by a fitting algorithm, and frequency points \(\omega_i\) determine the frequency range of each segment. Parameter \(\sigma_i\) determines the shape of the interpolating function \(f_i\), and can be used to select the between cubic \((\sigma_i = 0)\), exponential \((0 < \sigma_i < \infty)\) or linear interpolating function \((\sigma_i \rightarrow \infty)\). It has been shown in [8] that there exists a value of parameter \(\sigma_i^* \geq 0\), such that for \(\sigma_i \geq \sigma_i^*\) undesired inflexion points do not occur. The problem of determining the values of \(\sigma_i\) has been thoroughly discussed in [9], and the following text will use the results of it.

The interpolating function \(f_i(\omega)\) in the interval \((\omega_i, \omega_{i+1})\), equivalent to (1), can be expressed as [9]:

\[
f_i(\omega) = \frac{h_i}{\sigma_i v_i} \left[ \alpha_1^{(i)} \cosh(\sigma_i b_i(\omega)) - \alpha_2^{(i)} \sinh(\sigma_i b_i(\omega)) \right],
\]

\[\text{(2)}\]

where

\[
h_i = \omega_{i+1} - \omega_i
\]

\[\text{(3)}\]

\[
b_i(\omega) = (\omega_{i+1} - \omega_i)/h_i
\]

\[\text{(4)}\]

\[
s_i = (y_{i+1} - y_i)/h_i
\]

\[\text{(5)}\]

\[
d_1^{(i)} = s_i - y_i'
\]

\[\text{(6)}\]

\[
d_2^{(i)} = y_{i+1}' - s_i
\]

\[\text{(7)}\]

\[
\sinh(\omega) = \sinh(\omega) - \omega
\]

\[\text{(8)}\]

\[
\cosh(\omega) = \cosh(\omega) - 1
\]

\[\text{(9)}\]

\[
v_i = \sigma_i \sinh(\sigma_i) - 2 \cosh(\sigma_i)
\]

\[\text{(10)}\]

\[
\alpha_1^{(i)} = d_1^{(i)} \sigma_i \cosh(\sigma_i) - (d_1^{(i)} + d_2^{(i)}) \sinh(\sigma_i)
\]

\[\text{(11)}\]

\[
\alpha_2^{(i)} = d_2^{(i)} \sigma_i \sinh(\sigma_i) - (d_1^{(i)} + d_2^{(i)}) \cosh(\sigma_i)
\]

\[\text{(12)}\]

Formulation of interpolating function \(f_i\) given in (2) is convenient because it depends only on the input data \(\omega_i\) and \(y_i\), calculated parameters \(\sigma_i\) and first derivatives \(y_i'\). Parameters \(\sigma_i\) and first derivatives \(y_i'\) can be obtained from input data by invoking the algorithm from [9] which is available as FORTRAN and Matlab code.

It is difficult to calculate the Hilbert transform of the interpolating function \(f_i(\omega)\) in the form of (2), but it can be rearranged to a more manageable form by using identities

\[
\cosh(\omega) = (e^\omega + e^{-\omega})/2
\]

\[\text{and}\]

\[
\sinh(\omega) = (e^\omega - e^{-\omega})/2
\]

\[\text{(26)}\]

\[
f_i(\omega) = \beta_0^{(i)}(\omega - \omega_i) + \beta_1^{(i)}(\omega - \omega_i)/h_i + \beta_2^{(i)} e^{\omega_1(\omega_i - \omega_i)/h_i} + \beta_3^{(i)} e^{-\sigma_i(\omega_i - \omega_i)/h_i}
\]

\[\text{(13)}\]

where coefficients \(\beta_j^{(i)}, j = 0.3\) are given by:

\[
\beta_0^{(i)} = y_{i+1} - h_i \sigma_1^{(i)} + \sigma_i y_i'_{i+1} - \alpha_2^{(i)} / v_i \sigma_i
\]

\[\text{(14)}\]

\[
\beta_1^{(i)} = y_i'_{i+1} - \alpha_2^{(i)} / v_i
\]

\[\text{(15)}\]

\[
\beta_2^{(i)} = h_i \sigma_1^{(i)} - \alpha_2^{(i)} / 2v_i \sigma_i
\]

\[\text{(16)}\]

\[
\beta_3^{(i)} = h_i \sigma_1^{(i)} + \alpha_2^{(i)} / 2v_i \sigma_i
\]

\[\text{(17)}\]

The Hilbert transform of (13) can be calculated, as will be shown in the following section.

The algorithm [9] can output \(\sigma_i = 0\), so it is necessary to determine the form of interpolating function for that case. Letting the value of \(\sigma \rightarrow 0\) results in:

\[
f_i^{(0)}(\omega) = \lim_{\sigma \rightarrow 0} f_i(\omega) = y_2 - h_i y_2 b(\omega)
\]

\[\text{+}(d_1 - 2d_2)b(\omega)^2 + (d_2 - d_1)b(\omega)^3]\]

\[\text{(18)}\]

which is a cubic polynomial. Polynomial (18) can be rearranged to a more convenient form:

\[
f_i^{(0)}(\omega) = y_0^{(0)} + y_1^{(0)}(\omega - \omega_i) + y_2^{(0)}(\omega - \omega_i)^2 + y_3^{(0)}(\omega - \omega_i)^3
\]

\[\text{(19)}\]

where constants \(y_j^{(0)}, j = 0.3\) are given by:

\[
y_0^{(0)} = y_i
\]

\[\text{(20)}\]

\[
y_1^{(0)} = y_i'
\]

\[\text{(21)}\]

\[
y_2^{(0)} = \frac{2d_1^{(0)} - d_2^{(0)}}{h_i}
\]

\[\text{(22)}\]

\[
y_3^{(0)} = \frac{d_2^{(0)} - d_1^{(0)}}{h_i^2}
\]

\[\text{(23)}\]

\[
\text{III. HILBERT TRANSFORM OF INTERPOLATION FUNCTION}
\]

Even or odd function \(f(\omega)\) known at discrete points can be interpolated as:

\[
f(\omega) = f_+(\omega) + (-1)^p f_-(\omega),
\]

\[\text{(24)}\]

where \(f_+(\omega)\) is the interpolation of \(f(\omega)\) for positive values of \(\omega, f_-(\omega) < 0\), and parameter \(p\) determines whether the function \(f\) has even or odd symmetry:

\[
p = \begin{cases} 
1 & \text{odd function} \\
2 & \text{even function}
\end{cases}
\]

\[\text{(25)}\]

Interpolating function \(f_+(\omega)\) can be constructed from \(f(\omega)\) known at points \(\omega_i \geq 0, y_i = f(\omega_i)\), as a sum of \(N-1\) piecewise interpolating functions \(f_i(\omega)\) as:

\[
f_+(\omega) = \sum_{i=1}^{N-1} \chi_i(\omega) f_i(\omega),
\]

\[\text{(26)}\]

where \(\chi_i(\omega)\) is the characteristic function defined by:

\[
\chi_i(\omega) = \chi(\omega_i, \omega_{i+1})(\omega) = \begin{cases} 
0, & -\infty < \omega < \omega_i \\
1, & \omega_i < \omega < \omega_{i+1} \\
0, & \omega_{i+1} < \omega < \infty
\end{cases}
\]

\[\text{(27)}\]

Characteristic function \(\chi_i(\omega)\) windows the piecewise interpolating function \(f_i(\omega)\) to the interval \((\omega_i, \omega_{i+1})\), so \(f_+(\omega) = f_i(\omega)\) for \(\omega_i < \omega < \omega_{i+1}\).
Our goal is to determine the Hilbert transform of (24) in order to perform causal interpolation. Hilbert transform \( \mathcal{H} \) of a function \( f(\omega) \) is defined by [10, pp. 239]:
\[
\mathcal{H}\{f(\mu)\}(\omega) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{f(\mu)}{\mu - \omega} d\mu,
\]
where \( P \) denotes the Cauchy principal value of the integral. To avoid cumbersome notation, we will adopt the short-hand form \( \mathcal{H}f(\omega) \equiv \mathcal{H}\{f(\mu)\}(\omega) \) where it is unambiguous, which denotes that the Hilbert transform of function \( f(\mu) \) is evaluated at \( \omega \). In the following text we will use the interval \((\omega_i, \omega_{i+1})\) and index \( i \) interchangeably so \( \chi(\omega) \equiv \chi(\omega_i, \omega_{i+1}) \).

Using the linearity (71) and reflection (73) properties of the Hilbert transform, listed in Appendix together with other properties of interest, \( F(\omega) = \mathcal{H}f(\omega) \) is then:
\[
F(\omega) = \mathcal{H}\{f(\mu)\}(\omega) = \mathcal{H}(f_+(\mu))(\omega) + (-1)^{p+1}\mathcal{H}(f_-(\mu))(\omega)
\]
(29)

Equation (29) shows that to calculate the Hilbert transform \( F(\omega) = \mathcal{H}f(\omega) \) it is sufficient to calculate \( F_+(\omega) = \mathcal{H}f_+(\omega) \) and evaluate it for \( \omega \) and \(-\omega\).

Taking the Hilbert transform of (26) and using the linearity property results in:
\[
F_+(\omega) = \mathcal{H}\left\{ \sum_{i=1}^{N-1} \chi(\mu)f_i(\mu) \right\} = \sum_{i=1}^{N-1} \mathcal{H}(\chi(\mu)f_i(\mu))(\omega) = \sum_{i=1}^{N-1} F_i(\omega),
\]
(30)
if the sum exists. Special cases when the order of summation and Hilbert transform cannot be exchanged will be considered in Sec. IV. Examining (30) reveals that the problem of calculating the Hilbert transform of \( f(\omega) \) reduces to calculating the Hilbert transform of interpolating functions multiplied by the characteristic function defining the interval of interpolation.

**A. HILBERT TRANSFORM OF \( \chi(\omega)f_i(\omega) \)**

If the value of parameter \( \sigma_i \) is not zero, the \( i^{th} \) interpolating function is an exponential spline of the form (13), which is rewritten for clarity:
\[
f_i(\omega) = \beta_{i0}^{(i)} + \beta_{i1}^{(i)}(\omega - \omega_i) + \beta_{i2}^{(i)} e^{\sigma_i(\omega_{i+1} - \omega)/h_i} + \beta_{i3}^{(i)} e^{-\sigma_i(\omega_{i+1} - \omega)/h_i}.
\]

The Hilbert transform of \( \chi(\omega)f_i(\omega) \) can be calculated term by term using the linearity property. For the first term, \( \chi(\omega) \cdot \text{const} \), the Hilbert transform is [10, pp. 243]
\[
T_0^{(i)}(\omega) = \mathcal{H}\chi(\omega) = \frac{1}{\pi} \log \frac{\omega_{i+1} - \omega}{\omega_i - \omega}
\]
(31)

Hilbert transform of the second term, \( \chi(\omega)(\omega - \omega_i) \), can be obtained by expanding \( \chi(\omega)(\omega - \omega_i) = \chi(\omega)\omega - \chi(\omega)\omega_i \), and using the moment formula (74) on \( \chi(\omega)\omega \):
\[
\mathcal{H}\{\mu \chi(\mu)\}(\omega) = \frac{1}{\pi} \log \frac{\omega_{i+1} - \omega}{\omega_i - \omega} + \omega_{i+1} - \omega_i
\]
(32)

Using the linearity property \( \mathcal{H}(\mu - \omega_i)\chi(\mu))(\omega) = \mathcal{H}(\mu \chi(\mu))(\omega) - \omega_i \mathcal{H}(\chi(\mu))(\omega) \) results in:
\[
T_1^{(i)}(\omega) = \mathcal{H}(\mu - \omega_i)\chi(\mu))(\omega) = \frac{1}{\pi} \left( \log \frac{\omega_{i+1} - \omega}{\omega_i - \omega} + \omega_{i+1} - \omega_i \right)
\]
(33)

The third and fourth term of (13) are of the form \( e^{\pm \sigma_i(\alpha_{i+1} - \alpha)/h_i} \), and their Hilbert transform can be calculated from \( \mathcal{H}\{e^{\mu \chi(\mu)} \} \) and using the linear scale change property:
\[
\mathcal{H}\{e^{\mu \chi(\mu)} \}(\omega) = \frac{1}{\pi} \int_{-\omega_0}^{\omega_{i+1}} \frac{e^{\mu t}}{\mu - \omega} dt
\]
(34)
where the change of variable \( t = - (\mu - \omega) \) was used in the second step. Integral in (34) can be expressed in terms of special functions by noting that:
\[
P \int_{-\omega_0}^{\omega_{i+1}} \frac{e^t}{t} dt = \lim_{c \to \infty} \frac{1}{\pi} \left( P \int_{-c}^{c} \frac{e^{-t}}{t} dt - P \int_{-c}^{-\omega_0} \frac{e^{-t}}{t} dt \right)
\]
(35)
where \( Ei(\omega) \) is the exponential integral function
\[
Ei(\omega) = - P \int_{-\infty}^{\omega} \frac{e^{-t}}{t} dt.
\]
(36)

Using (35) and (36) in (34) the transform \( \mathcal{H}\{e^{\mu \chi(\mu)} \} \) is:
\[
\mathcal{H}\{e^{\mu \chi(\mu)} \}(\omega) = \frac{e^{\omega_0}}{\pi} [Ei(\omega_{i+1} - \omega) - Ei(\omega_i - \omega)].
\]
(37)

The Hilbert transform of \( T_2^{(i)}(\omega) \) (38), as shown at the bottom of the next page can be obtained by using the linear scale change property (72) in (37), where + sign is used for \( T_2^{(i)}(\omega) \), − sign for \( T_3^{(i)}(\omega) \), and
\[
\text{ExpEi}(\omega) = e^{-\omega_0} Ei(\omega).
\]
(39)
The reason for rearranging (38) is that ExpEi(\( \omega \)) can be efficiently approximated with rational Chebyshev functions [11], [12]. Numerical code for calculating ExpEi(\( \omega \)) is available from Netlib repository in the Spectfun FORTRAN package.

Finally, the Hilbert transform of exponential spline interpolating function \( \chi(\omega)f_i(\omega) \) is:
\[
F_i(\omega) = \mathcal{H}\{\chi(\omega)f_i(\omega)\}(\omega) = \sum_{k=0}^{3} \beta_{ik}^{(i)} T_k^{(i)}(\omega)
\]
(40)
B. HILBERT TRANSFORM OF $\chi_i(\omega)\hat{f}_i^0(\omega)$

If the value of parameter $\sigma_i$ is zero, the $i^{th}$ interpolating function is a cubic spline of the form (19), rewritten for clarity:

$$f_i^{(0)}(\omega) = \gamma_i^{(0)}(\omega - \omega_i) + \gamma_i^{(1)}(\omega - \omega_i) + \gamma_i^{(2)}(\omega - \omega_i)^2 + \gamma_i^{(3)}(\omega - \omega_i)^3$$

Hilbert transform of $\chi_i(\omega)\hat{f}_i^0(\omega)$ can be calculated by using the general moment formula (75) when the sum exists, and the sum and Hilbert transform can be exchanged:

$$\mathcal{H}\left\{\chi_i(\mu)\hat{f}_i^0(\mu)\right\}(\omega) = \mathcal{H}\left\{\sum_{k=0}^{3} \gamma_k^{(i)} \chi_i(\mu)(\mu - \omega_i)^k\right\}(\omega) = \sum_{k=0}^{3} \gamma_k^{(i)} \mathcal{H}\left\{\chi_i(\mu)(\mu - \omega_i)^k\right\}(\omega)$$

Special cases will be treated in Sec. IV.

Using Hilbert transform general moment formula (75) for $\omega^i\chi_i(\omega)$ evaluates to:

$$\mathcal{H}\left\{\mu^n\chi_i(\mu)\right\}(\omega) = \frac{1}{\pi} \omega^n \log \left| \frac{\omega_i + 1}{\omega_i - \omega} + \sum_{j=0}^{n-1} \frac{\omega_i^{n-j-1} - \omega_i^{j+1}}{j+1} \right|$$

(41)

Using a binomial formula, Hilbert transform of $\chi_i(\omega)(\omega - \omega_i)^n$ can then be expressed as:

$$\mathcal{H}\left\{(\omega - \omega_i)^n\chi_i(\mu)\right\}(\omega) = \mathcal{H}\left\{\chi_i(\mu)\sum_{k=0}^{n} \binom{n}{k} (-\omega_i)^{n-k} \mu^k\right\}(\omega) = \sum_{k=0}^{n} \binom{n}{k} (-\omega_i)^{n-k} \mathcal{H}\left\{\chi_i(\mu)\mu^k\right\}(\omega)$$

(42)

Using (41) in (42) results in:

$$\mathcal{H}\left\{(\omega - \omega_i)^n\chi_i(\mu)\right\}(\omega) = \frac{1}{\pi} \sum_{k=0}^{n} \binom{n}{k} (-\omega_i)^{n-k} \omega^k \log \left| \frac{\omega_i + 1}{\omega_i - \omega} \right| + \frac{1}{\pi} \sum_{k=0}^{n} \binom{n}{k} (-\omega_i)^{n-k} \sum_{j=0}^{k-1} \omega_i^{k-j-1} \omega_i^{j+1} \frac{1}{j+1}$$

$$= \frac{1}{\pi} (\omega - \omega_i)^n \log \left| \frac{\omega_i + 1}{\omega_i - \omega} \right| + p^{(n-1)}(\omega - \omega_i)$$

(43)

where $p^{(n)}(\omega)$ denotes a polynomial of degree $n$. The Hilbert transform of $\chi_i(\omega)\hat{f}_i^0(\omega)$ is then:

$$\mathcal{H}\left\{\chi_i(\mu)\hat{f}_i^0(\mu)\right\}(\omega) = \frac{1}{\pi} \sum_{k=0}^{n-1} \gamma_k^{(i)} \mathcal{H}\left\{\chi_i(\mu)(\mu - \omega_i)^k\right\}(\omega) = \frac{1}{\pi} \sum_{k=0}^{n-1} \gamma_k^{(i)} (\omega - \omega_i)^k$$

and

$$\mathcal{H}\left\{\chi_i(\omega)^2\hat{f}_i^0(\omega)\right\}(\omega) = \frac{1}{\pi} \sum_{k=0}^{n-1} \gamma_k^{(i)} \chi_i(\omega)(\omega - \omega_i)^k$$

IV. SPECIAL CASES FOR $\omega = \omega_i$

Inspecting the Hilbert transform of exponential (40), (31)-(38), and cubic (45) spline interpolating functions reveals that they contain terms $\log|\omega_i - \omega|$ and $\exp\{\pm \sigma_i(\omega - \omega_i)/h_i\}$ which are singular at $\omega_i$. The singularity at $\omega = \omega_i$ arises because the Hilbert transform was performed over individual interpolation intervals $(\omega_i, \omega_{i+1})$, and taking the principal value of integral in (28) could not remove it. Consequently, for $\omega = \omega_i$ the sum in (30) does not exist, and the order of Hilbert transform and summation cannot be exchanged.

The value of $F(\omega_i)$ can be calculated by observing that only $F_{i-1}$ and $F_i$ contain singular terms, and that calculating the Hilbert transform over two interpolation intervals $(\omega_{i-1}, \omega_{i+1})$ will remove the singularity at $\omega_i$ by taking the principal value of the integral. Using this insight, the sum (30) can be rearranged to:

$$F(\omega_i) = \sum_{j=1}^{N-1} \mathcal{H}\left\{\chi_i(\mu)f_j(\mu)\right\}(\omega_i) + \sum_{j=1}^{N-1} \mathcal{H}\left\{\sum_{j=1}^{i} \chi_i(\mu)f_j(\mu)\right\}(\omega_i)$$

(49)
The Hilbert transform over two intervals is

$$
\mathcal{H} \left\{ \sum_{j=i-1}^{i} \chi_j(\mu) f_j(\mu) \right\}(\omega) = \lim_{\epsilon \to 0} \left[ \int_{\omega_{i-1}}^{\omega_i-\epsilon} f_i(\mu) \frac{d\mu}{\mu - \omega} + \int_{\omega_i+\epsilon}^{\omega_{i+1}} f_i(\mu) \frac{d\mu}{\mu - \omega} \right]
$$

where the Cauchy principal value of integral excludes the singular point $\omega_i$ by splitting the integration interval $(\omega_{i-1}, \omega_{i+1})$ to intervals $(\omega_{i-1}, \omega_i - \epsilon)$ and $(\omega_i - \epsilon, \omega_{i+1})$, and taking the limit $\epsilon \to 0$.

By introducing $\varepsilon F_i(\omega)$ and $\varepsilon F_i(\omega)$, defined as

$$
\varepsilon F_i(\omega) = \frac{1}{\pi} \int_{\omega_{i-1}}^{\omega_i+\epsilon} f_i(\mu) \frac{d\mu}{\mu - \omega}
$$

$$
\varepsilon F_i(\omega) = \frac{1}{\pi} \int_{\omega_i}^{\omega_{i+1}-\epsilon} f_i(\mu) \frac{d\mu}{\mu - \omega}
$$
equation (50) can be expressed as:

$$
\mathcal{H} \left\{ \sum_{j=i-1}^{i} \chi_j(\mu) f_j(\mu) \right\}(\omega_i) = \lim_{\epsilon \to 0} \left[ \varepsilon F_{i-1}(\omega_i) + \varepsilon F_i(\omega_i) \right]
$$

Functions $\varepsilon F_i(\omega)$ and $\varepsilon F_i(\omega)$ can be thought of as the Hilbert transform of interpolating function $f_i(\omega)$ where the interpolation interval is reduced by $\epsilon$ from the lower or upper side. Therefore, $\varepsilon F_i(\omega)$ can be directly obtained from $F_i(\omega)$ by replacing $\omega_i$ with $\omega_i + \epsilon$, while $\varepsilon F_i(\omega)$ is obtained by replacing $\omega_{i+1}$ with $\omega_{i+1} - \epsilon$. Taking the Cauchy principal value of integrals in (51) and (52) is not needed, since the evaluation points $\omega_i$ will always be outside of the integration interval and they will always be well-defined. Values of $\varepsilon F_i(\omega_i)$ and $\varepsilon F_i(\omega_{i+1})$ are needed to evaluate (53), and they are derived for cases of cubic and exponential splines in the following text.

A. CUBIC SPLINE $\varepsilon F_i^0(\omega_i)$ AND $\varepsilon F_i^0(\omega_{i+1})$

Function $\varepsilon F_i^0(\omega_i)$ can be derived from (45) by replacing the lower integration limit $\omega_i$ with $\omega_i + \epsilon$ and substituting $\omega = \omega_i$.

$$
\varepsilon F_i^0(\omega_i) = \frac{1}{\pi} \left[ f_i^0(\omega_i) \log \frac{\omega_{i+1} - \omega_i}{\omega_i + \epsilon - \omega_i} \right] + \sum_{k=0}^{2} \delta_k^{(i)} (\omega_i - \omega_i)^k
$$

and can be rearranged as:

$$
\varepsilon F_i^0(\omega_i) = -\pi^{-1} f_i^0(\omega_i) \log |\epsilon| + \mathcal{F}_i^0(\omega_i)
$$

where the non-singular term $\mathcal{F}_i^0(\omega_i)$ is:

$$
\mathcal{F}_i^0(\omega_i) = \pi^{-1} \left( f_i^0(\omega_i) \log h_i + \delta_0^{(i)} \right)
$$

Similarly, $\varepsilon F_i^0(\omega_{i+1})$ can be obtained by replacing the upper integration limit $\omega_{i+1}$ with $\omega_{i+1} - \epsilon$ and substituting $\omega = \omega_{i+1}$, which results in:

$$
\varepsilon F_i^0(\omega_{i+1}) = \pi^{-1} f_i^0(\omega_{i+1}) \log |\epsilon| + \mathcal{F}_i^0(\omega_{i+1})
$$

where the non-singular term $\mathcal{F}_i^0(\omega_{i+1})$ is:

$$
\mathcal{F}_i^0(\omega_{i+1}) = \pi^{-1} \left( f_i^0(\omega_{i+1}) \log h_i + \sum_{k=0}^{2} \delta_k^{(i)} h_i^k \right)
$$

B. EXPONENTIAL SPLINE $\varepsilon F_i(\omega_i)$ AND $\varepsilon F_i(\omega_{i+1})$

The Hilbert transform of exponential spline (38) contains both logarithmic and exponential integral terms which are singular at $\omega_i$ and $\omega_{i+1}$. Logarithmic terms can be split into singular and non-singular terms, as was previously done for cubic spline case, while the exponential integral terms can be expanded to power series at singular points. Series expansion of $\text{Ei}(\sigma(\omega_i - \omega_i)/h_i)$ around $\omega_i$ is

$$
\text{Ei} \left( \frac{\sigma(\omega_i - \omega_i)}{h_i} \right) = \gamma_E + \log \left( \frac{\sigma(\omega_i - \omega_i)}{h_i} \right) + O(\omega - \omega_i)
$$

where $\gamma_E \approx 0.5772$ is the Euler-Mascheroni constant.

Function $\varepsilon F_i(\omega_i)$ can be obtained from (40) by replacing the lower integration limit $\omega_i$ with $\omega_i + \epsilon$, substituting $\omega = \omega_i$, and using the series expansion (59) around $y = \omega_i$ in (38) for terms containing $\omega - \omega_i$. After collecting singular and non-singular terms the result is:

$$
\varepsilon F_i(\omega_i) = -\pi^{-1} f_i^0(\omega_i) \log |\epsilon| + \mathcal{F}_i(\omega_i)
$$

where the non-singular term $\mathcal{F}_i(\omega_i)$ is:

$$
\mathcal{F}_i(\omega_i) = \pi^{-1} \left( \beta_0^{(i)} \log h_i + \beta_1^{(i)} h_i + \beta_2^{(i)} \text{ExpEi}(-\sigma_i) \right)
$$

$$
+ \beta_3^{(i)} \text{ExpEi}(-\sigma_i) - \left( \log \frac{\sigma_i}{h_i} + \gamma_E \right) \left( \beta_2^{(i)} e^{\sigma_i} + \beta_3^{(i)} e^{-\sigma_i} \right)
$$

(61)

In a similar manner, function $\varepsilon F_i(\omega_{i+1})$ can be obtained from (40) by replacing the upper integration limit $\omega_{i+1}$ with $\omega_{i+1} - \epsilon$, substituting $\omega = \omega_{i+1}$, and using the series expansion (59) around $y = \omega_{i+1}$ in (38) for terms containing $\omega - \omega_{i+1}$. After collecting singular and non-singular terms the result is:

$$
\varepsilon F_i(\omega_{i+1}) = -\pi^{-1} f_i^0(\omega_{i+1}) \log |\epsilon| + \mathcal{F}_i(\omega_{i+1})
$$

where the non-singular term $\mathcal{F}_i(\omega_{i+1})$ is:

$$
\mathcal{F}_i(\omega_{i+1}) = -\pi^{-1} \left( \beta_0^{(i)} \log h_i + \beta_1^{(i)} h_i (\log h_i - 1) \right)
$$

$$
+ \beta_2^{(i)} \text{Ei}(-\sigma_i) + \beta_3^{(i)} \text{Ei}(-\sigma_i) - \left( \log \frac{\sigma_i}{h_i} + \gamma_E \right) \left( \beta_2^{(i)} + \beta_3^{(i)} \right)
$$

(63)

C. EVALUATION AT POINT $\omega_i$

Using the previous results, the Hilbert transform over two
D. N. Grujić: Numerical Hilbert Transform Algorithm for Causal Interpolation

The Hilbert transform of a function interpolated with a mix of splines is given by:

\[ \mathcal{H} \left\{ \sum_{j=-1}^{i} \chi_j(\mu) f_j(\mu) \right\}(\omega) = \frac{1}{\pi} \lim_{\epsilon \to 0} \left( f_{i-1}(\omega_i) - f_i(\omega_i) \right) \log |\epsilon| + F_{i-1}(\omega_i) + F_i(\omega_i) \]  
(64)

Since the interpolating function is assumed to be continuous, it follows that \( f_{i-1}(\omega_i) = f_i(\omega_i) \), and the singular term \( \log |\epsilon| \) vanishes. Finally, the value of the Hilbert transform over two intervals \( (\omega_{i-1}, \omega_{i+1}) \) at point \( \omega_i \) is:

\[ \mathcal{H} \left\{ \sum_{j=-1}^{i} \chi_j(\mu) f_j(\mu) \right\}(\omega_i) = \frac{1}{\pi} \left( F_{i-1}(\omega_i) + F_i(\omega_i) \right) \]  
(65)

Result (65) is valid for any combination of cubic and exponential splines, because both of (56) and (61) contain the term \( -\pi^{-1} f_i(\omega_i) \log |\epsilon| \), and both of (58) and (63) contain the term \( \pi^{-1} f_i(\omega_i) \log |\epsilon| \), which cancel out in (64).

V. ALGORITHM FOR EVALUATION

The Hilbert transform of a function interpolated with a mix of cubic and exponential splines can be expressed as:

\[ F(\omega) = \sum_{\sigma_i > 0}^{N-1} \left[ F_i(\omega) + (-1)^{p+1} F_i(-\omega) \right] \]

Evaluation of (66) can be divided into setup and evaluation steps. In the setup step, shown in Alg. 1, coefficients which do not depend on frequency are evaluated, and needs to be performed only once for a given set of interpolating splines.

Algorithm 2 evaluates the Hilbert transform of function interpolated by a mix of cubic and exponential splines at a given frequency. The algorithm evaluates the contribution of individual interpolating functions at \( \pm \omega \) and accumulates them in variables \( H_{\pm} \). When contributions from all interpolating functions are evaluated, function symmetry set by parameter \( p \) is used to calculate the value of Hilbert transform as \( \pi^{-1} \left( H_+ + (-1)^{p+1} H_- \right) \).

Since interpolating functions can be cubic or exponential splines, there are two ways of evaluating their contribution. Algorithm for evaluating the Hilbert transform of cubic spline is shown in Alg. 3, while the case of exponential spline is given in Alg. 4. Both algorithms check whether the evaluation frequency \( \omega \) is close to singular points \( \omega_{i,i+1} \) and use the appropriate expressions to produce a valid answer.

VI. COMPARISON WITH PUBLISHED RESULTS

The proposed algorithm was compared to published results by using it for causal interpolation of characteristic impedance of...
microstrip transmission line over conductive substrate [13].

The characteristic impedance data used in this example, taken from CausalCat [14], is calculated by power-voltage definition in [15] from full-wave simulation of 5 µm wide microstrip over a 1 µm oxide stacked on top of 100 µm conductive silicon substrate.

Theory of causal interpolation is explained in detail in [14]–[16], and a short overview is given here for completeness. Characteristic impedance is causal and minimum-phase function with magnitude-phase relation:

\[
\mathcal{H}\{\log[Z_0(\mu)](\omega) = \arg(Z_0(\omega)) \}
\]

(67)

Using the property of inverse Hilbert transform (76) results in:

\[
\log[Z_0(\omega)] = -\mathcal{H}\{\arg(Z_0(\mu))\}(\omega) + \log \lambda,
\]

(68)

where \(\lambda\) is a constant. Magnitude of characteristic impedance is then

\[
|Z_0(\omega)| = \lambda e^{-\mathcal{H}[\arg(Z_0(\mu))](\omega)}.
\]

(69)

Causal interpolation of characteristic impedance can be performed by interpolating the phase \(\arg(Z_0(\omega))\) and calculating the magnitude (69). Additionally, using (69) can reveal whether the voltage integration path is correctly chosen [13].

Causal interpolation of characteristic impedance has been implemented in NIST software CausalCat [7], which uses FFT based Hilbert transform, and in [6], which uses analytic solution for piecewise polynomial interpolating function. CausalCat computes the causal characteristic impedance magnitude by interpolating the input data phase, sampling the interpolated function with a fixed frequency step and performing the Hilbert transform by FFT. Using the results from [6] for \(n = 3\) is equivalent to forcing the proposed algorithm to use cubic splines for all interpolating functions by setting \(\sigma_i = 0\).

Hilbert transform is non-local, meaning that the value of Hilbert transform \(\mathcal{H}f(\omega)\) at a given frequency depends on the entire function \(f\). In practice, measured quantity will be known up to some frequency \(\omega_{\text{max}}\), which will require to make assumptions about the frequency behavior above \(\omega_{\text{max}}\). In the case of causal characteristic impedance interpolation, the error at frequency \(\omega\) due to unknown behavior above \(\omega_{\text{max}}\) is bounded by [16]:

\[
\frac{\omega_{\text{max}}^2 - \omega^2}{\omega_{\text{max}}^2} \leq \left| \frac{Z_0}{\bar{Z}_0} \right| \leq \frac{\omega_{\text{max}}^2 - \omega^2}{\omega_{\text{max}}^2},
\]

(70)

where \(Z_0\) is the true characteristic impedance and \(\bar{Z}_0\) is the calculated characteristic impedance. The error due to unknown high frequency behavior can be made arbitrarily small by restricting to frequencies where the error is acceptable.

In the microstrip example, phase of characteristic impedance approaches \(\pm \pi/4\) as \(\omega \to 0\), tends to zero as frequency increases and might exhibit complicated behavior at very high frequencies [14]. Since the frequency range of measurement equipment is not infinite, an assumption about

\[
|Z_0(\omega)| = \lambda e^{-\mathcal{H}[\arg(Z_0(\mu))](\omega)}.
\]

(69)

the phase at frequencies above the measurement range must be made. By assuming that the phase is zero when \(\omega > \omega_{\text{max}}\) an error bounded by (70) is introduced. To avoid aliasing, CausalCat corrects the phase at \(\omega = 0\) by summing the input data with a function \(f = \pm(\pi/4)/(1 + a|\omega|/8)\), so \(\varphi(0) = \varphi(\omega_{\text{max}}) = 0\), and by subtracting \(\mathcal{H}f\) from the result. Such correction is not needed in the proposed algorithm because it does not suffer from aliasing.

Interpolation of \(\arg(Z_0(\omega))\) with cubic and mix of cubic and exponential splines is shown in Fig. 1. In both cases 23 interpolating functions were used, where for the case of

\[
\mathcal{H}\{\log[Z_0(\mu)](\omega) = \arg(Z_0(\omega)) \}
\]

(67)
mixed splines 11 interpolating functions were exponential splines. From zoomed inset in Fig. 1 it can be seen that cubic splines exhibit oscillations due to undesired inflexion points. The use of exponential splines eliminates these oscillations.

Phase of characteristic impedance is an odd function of frequency, so symmetry parameter \( p \) should be set to \( p = 1 \) when calculating its Hilbert transform. Causal interpolation of characteristic impedance magnitude was performed by calculating the interpolated phase Hilbert transform and using (69), and the results are shown in Fig. 2. It can be seen from zoomed inset that using only cubic splines results in oscillations, while using a mix of cubic and exponential splines does not.

Relative error of calculated \( |Z_0| \) for the case of cubic (dashed) and mix of cubic and exponential (solid) splines with CausalCat results as reference is shown in Fig. 3. It can be seen that both cubic and mixed splines interpolation are in good agreement with referent results. The use of exponential splines results in lower overall relative error compared to using only cubic splines. Lower relative error is obtained by eliminating undesired inflexion points introduced by interpolating with cubic splines.

**VII. CONCLUSION**

Algorithm for numerical Hilbert transform of functions represented by cubic and exponential splines is presented in this paper. Allowing the interpolating function to contain both cubic and exponential splines can eliminate undesired inflexion points which can occur if only cubic splines are used. The algorithm calculates exact Hilbert transform of interpolating function, so the accuracy of results depends solely on the quality of interpolation. It is suitable for causal interpolation of data spanning several decades without requiring excessive number of data points, and does not suffer from aliasing. Proposed algorithm was used for causal interpolation of characteristic impedance spanning five frequency decades, and the results are compared to the output of CausalCat.

**APPENDIX A**

**PROPERTIES OF THE HILBERT TRANSFORM**

**Linearity**

\[
\mathcal{H}(af(\mu) + bg(\mu))(\omega) = a\mathcal{H}(f(\mu))(\omega) + b\mathcal{H}(g(\mu))(\omega)
\]  
(71)

Linear scale change property

\[
\mathcal{H}(f(\alpha\mu))(\omega) = \mathcal{H}(f(\mu))(\alpha\omega)
\]  
(72)

**Reflection**

\[
\mathcal{H}(f(-\mu))(\omega) = -\mathcal{H}(f(\mu))(-\omega)
\]  
(73)

**Hilbert transform moment formula**

\[
\mathcal{H}\{f(\mu)\}(\omega) = \omega \mathcal{H}f(\omega) + \frac{1}{\pi} \int_{-\infty}^{+\infty} f(\mu)d\mu
\]  
(74)

**General Hilbert transform moment formula**

\[
\mathcal{H}\{f^{(n)}(\mu)\}(\omega) = \omega^n \mathcal{H}f(n) + \frac{1}{\pi} \sum_{j=0}^{n-1} \omega^{n-j-1} \int_{-\infty}^{+\infty} \mu^j f(\mu)d\mu
\]  
(75)

**Inverse Hilbert transform**

\[
\mathcal{H}\{Hf(\omega)\} = -f(\omega) + c
\]  
(76)

**ACKNOWLEDGMENT**

The author would like to thank Dr. Dylan F. Williams for providing the \( \text{arg}(Z_0) \) data and \( |Z_0| \) calculated by CausalCat, which is used as test case for verification.

**REFERENCES**

[1] A. Djordjević and D. Tosić, “Causality of circuit and electromagnetic-field models,” in Proc. 5th Eur. Conf. Circuits Syst. Commun. (ECCSC), Nov. 2010, pp. 12–21.

[2] O. E. Taurian, “A method and a program for the numerical evaluation of the Hilbert transform of a real function,” Comput. Phys. Commun., vol. 20, no. 2, pp. 291–307, Oct. 1980.

[3] V. C. Vannicola and A. Scott, “Interpolation/extrapolation of frequency domain responses using the Hilbert transform,” IEEE Trans. Microw. Theory Techn., vol. 44, no. 10, pp. 1621–1627, Oct. 1996.

[4] F. W. King, “Efficient numerical approach to the evaluation of Krane-Krönig transforms,” J. Opt. Soc. Amer. B, Opt. Phys., vol. 19, no. 10, pp. 2427–2436, 2002.

[5] R. Bilato, O. Maj, and M. Brambilla, “An algorithm for fast Hilbert transform of real functions,” Adv. Comput. Math., vol. 40, nos. 5–6, pp. 1159–1168, Dec. 2014.

[6] D. N. Grujić, “Numerical Hilbert transform algorithm for causal interpolation of piecewise polynomial even and odd functions.” IEEE Trans. Microw. Theory Techn., vol. 65, no. 6, pp. 2000–2007, Jun. 2017.

[7] CausalCat. Accessed: Sep. 1, 2021. [Online]. Available: https://www.nist.gov/ci/trf-technology-related-software.cfm

[8] H. Späth, “Exponential spline interpolation,” Computing, vol. 4, no. 3, pp. 225–233, Sep. 1969.

[9] R. J. Renka, “Interpolatory tension splines with automatic selection of tension factors,” SIAM J. Sci. Stat. Comput., vol. 8, no. 3, pp. 393–415, May 1987.

[10] A. Erdélyi, W. Magnus, F. Oberhettinger, and F. G. Tricomi, Tables of Integral Transforms, vol. 2. New York, NY, USA: McGraw-Hill, 1954.

[11] W. Cody and H. C. Thacher, “Rational approximations for the exponential integral \( E_i(x) \),” Math. Comput., vol. 23, no. 106, pp. 289–303, 1969.

[12] W. J. Cody and H. C. Thacher, “Rational Chebyshev approximations for the exponential integral \( E_i(x) \),” Math. Comput., vol. 22, no. 103, pp. 641–649, 1968.

[13] D. F. Williams, B. K. Alpert, U. Arz, D. K. Walker, and H. Grabinski, “Causal characteristic impedance of planar transmission lines,” IEEE Trans. Adv. Packag., vol. 26, no. 2, pp. 291–307, Oct. 1980.

[14] D. N. Grujić, “Numerical Hilbert transform algorithm for causal interpolation of piecewise polynomial even and odd functions.” IEEE Trans. Microw. Theory Techn., vol. 44, no. 10, pp. 1621–1627, Oct. 1996.

[15] D. F. Williams and B. K. Alpert, “Characteristic impedance of microstrip on silicon,” in Proc. 8th European Microwave Conf. (EuMC), Sep. 2000, pp. 1159–1168.

[16] D. F. Williams and B. K. Alpert, “Causality and waveguide circuit theory,” IEEE Trans. Microw. Theory Techn., vol. 49, no. 4, pp. 615–623, Apr. 2001.