A Toepplitz Formulation of a Real-Time Algorithm for Time Decoding Machines *

Aurel A. Lazar, Ernő K. Simonyi and László T. Tóth

November 5, 2005

Abstract

Time encoding is a real-time asynchronous mechanism for encoding the amplitude information of an analog bandlimited signal into a time sequence, or time codes, based on which the signal can be reconstructed. Using a Toepplitz formulation we propose an efficient real-time reconstruction procedure. As an illustration, time encoding is carried out by an asynchronous sigma-delta modulator. The proposed method is confirmed by numerical simulations.

1 Introduction

Time-encoding is a real-time asynchronous mechanism of mapping the amplitude information of a bandlimited signal $x(t), t \in \mathbb{R}$, into a set of time codes (TCs) $\{t_k\}, k \in \mathbb{Z}$, where $\mathbb{R}$ and $\mathbb{Z}$ denote the sets of real numbers and integers, respectively. The TCs are generated by Time Encoding Machines (TEMs) driven by $x(t)$. The TEMs are simple nonlinear asynchronous analog circuits with typically low power consumption. Usually the TEM output, $z(t)$, is an asynchronous binary signal or pulse-train based on which the TCs can be identified.

Known nonlinear asynchronous analog circuits can be used as TEMs. The first example of a TEM (see [9] and the references therein), also shown in Fig. 1, was an asynchronous sigma-delta modulator. Other TEMs include integrate-and-fire neurons [10] and frequency modulators [11]. Based on the TCs, $x(t)$ can be reconstructed by algorithms commonly referred to as Time Decoding Machines (TDMs) if certain Nyquist-type rate conditions on the TCs are met. Although methods used in frame theory [2, 8] and irregular sampling [4, 15] are needed to establish these conditions [11, 10], the algorithms are often easy to find and are reduced to solving consistent but (typically) ill-conditioned set of linear equations.

The first example of a TEM [9, 11] was an asynchronous sigma-delta modulator [7, 14] shown in Figure 1. The TEM consists of integrator and a symmetrically-centered noninverting Schmitt trigger in a negative-feedback arrangement where \( \kappa, \delta \) and \( b \) are circuit parameters. As shown, the zero-crossings of the asynchronous binary output \( z(t) \) define the TCs. Furthermore, the input signal is bounded both in amplitude and (angular) frequency as

\[
|x(t)| \leq c < b \quad \text{and} \quad X(\omega) = 0 \quad \text{if} \quad |\omega| < \Omega, \tag{1}
\]

where \( X(\omega) \) denotes the Fourier transform of \( x(t) \).

The operation of this TEM is simple. Since \( z(t) \) takes either \( b \) or \( -b \) values, the input to the integrator is either \( x(t) + b \) or \( x(t) - b \). Since \( |x(t)| \leq c < b \), the integrator output \( y(t) \) is a strictly increasing or decreasing function for \( t \in (t_k, t_{k+1}) \) and thus either \( y(t_k) = \delta \) or \( y(t_k) = -\delta \). A simple analysis [11] gives

\[
\int_{t_k}^{t_{k+1}} x(t) dt = (-1)^k (2\kappa\delta - b(t_{k+1} - t_k)) \tag{2}
\]

for all \( k \in \mathbb{Z} \). The original TDM of [9] can be found by assuming that \( x(t) \) is expressed as

\[
x(t) = \sum_{\ell \in \mathbb{Z}} c_{\ell} g(t - s_{\ell}), \quad s_{\ell} = \frac{t_{\ell} + t_{\ell+1}}{2}, \quad \text{and} \quad g(t) = \frac{\sin(\Omega t \pi t)}{\pi t} \tag{3}
\]

is the impulse response of an ideal lowpass filter (LPF) with cutoff frequency \( \Omega \). The goal is to find the coefficients \( c_{\ell} \). Substituting (3) into (2) gives:

\[
\sum_{\ell \in \mathbb{Z}} c_{\ell} \int_{t_k}^{t_{k+1}} g(t - s_{\ell}) dt = (-1)^k (2\kappa\delta - b(t_{k+1} - t_k)). \tag{4}
\]

With the definitions of the matrix \( G \) and, vectors \( q \) and \( c \) introduced in (4), the unknown \( c \) verify the linear equations \( Gc = q \). It can be shown [11] that this formulation gives perfect reconstruction if the condition \( 2\kappa\delta/(b - c) < \pi/\Omega \) is satisfied. Note that \( G, q \) and \( c \) have infinite dimensions.
1.1 Toeplitz Formulation

In the case of irregular sampling, an efficient approximation was proposed by transforming
the linear equations $Gc = q$ into a system represented by a Hermitian Toeplitz matrix [5].
This approach was generalized to time encoding in [12] in two steps.

In the first step a reformulation of the reconstruction technique of Section 1 is introduced.
If $x(t)$ is $\Omega$-bandlimited, then so is its indefinite integral and therefore similarly to (3) as:

$$\int_{-\infty}^{t} x(u)du = \sum_{\ell \in \mathbb{Z}} c_{\ell} g(t - t_{\ell}), \quad (5)$$

where the $c_{\ell}$'s are to be determined. Subtracting (5) evaluated at $t = t_{k}$ from the same
evaluated at $t = t_{k+1}$ gives:

$$\int_{t_{k}}^{t_{k+1}} x(u)du = \sum_{\ell \in \mathbb{Z}} c_{\ell} (g(t_{k+1} - t_{\ell}) - g(t_{k} - t_{\ell})).$$

Using (2) and rewriting the left-hand-side (LHS) using Kronecker’s notation gives:

$$(-1)^{k}(2\kappa\delta - b(t_{k+1} - t_{k})) = \sum_{\ell \in \mathbb{Z}} c_{\ell} \sum_{m \in \mathbb{Z}} [G]_{m,\ell} \sum_{\delta_{k+1,m} - \delta_{k,m}} [P]_{k,m} \sum_{\omega_{n}} [d]_{n}.$$

Using the matrices and vectors introduced above gives

$$q = PGc,$$

or, equivalently

$$P^{-1}q = Gc, \quad (6)$$

where

$$|P^{-1}|_{\ell,k} = \begin{cases} -1 & \text{if } \ell \leq k \\ 0 & \text{if } \ell > k. \end{cases} \quad (7)$$

In the second step an approximation for $g(t)$ introduced in (3) is given by:

$$g(t) \simeq \alpha \sum_{n=-N}^{N} e^{jn \frac{\Omega}{2N} t} = \alpha \frac{\sin \left(\frac{(2N+1)\Omega}{2N} t\right)}{\sin \left(\frac{\Omega}{2N} t\right)} \quad \text{with } \alpha = \frac{\Omega}{(2N + 1)\pi}. \quad (8)$$

When $N$ tends to infinity this function converges to $g(t)$. At the same time, the approxi-
mating function is both $\Omega$-bandlimited and periodic with a period $2N\pi/\Omega$.

Substituting (8) into (5) gives the approximation

$$\int_{-\infty}^{t} x(u)du \simeq \sum_{\ell \in \mathbb{Z}} c_{\ell}\alpha \sum_{n=-N}^{N} e^{jn \frac{\Omega}{2N} (t-t_{\ell})}$$

$$= \sum_{n=-N}^{N} e^{jn \frac{\Omega}{2N} t} \alpha \sum_{\ell \in \mathbb{Z}} c_{\ell} e^{-jn \frac{\Omega}{2N} t_{\ell}}. \quad (9)$$
Defining the elements of the matrix $S$ by

$$[S]_{n,\ell} = e^{-jn\frac{\Omega}{N}t},$$  \hspace{1cm} (10)

the vector $d$ introduced in (9) can be expressed as

$$d = \alpha Sc.$$ \hspace{1cm} (11)

Taking the derivative of both sides of (9) gives the approximation

$$x(t) \simeq j\frac{\Omega}{N} \sum_{n=-N}^{N} ne^{jn\frac{\Omega}{N}t}[d]_n$$ \hspace{1cm} (12)

for $x(t)$. Note that the periodic approximation is a Fourier-series expansion with a finite number of coefficients. This suggests the use of the Fast Fourier Transform (FFT) as discussed in Section 2.6.

Based on equations (6) and (8), and given the definition of the vector $d$ introduced in (9) we have

$$[P^{-1}q]_k = \sum_{\ell \in \mathbb{Z}} [G]_{k,\ell}[c]_\ell = \sum_{\ell \in \mathbb{Z}} g(t_k - t_\ell)c_\ell$$

$$\simeq \sum_{\ell \in \mathbb{Z}} c_\ell \alpha \sum_{n=-N}^{N} e^{jn\frac{\Omega}{N}(t_k - t_\ell)}$$

$$= \sum_{n=-N}^{N} e^{jn\frac{\Omega}{N}t_k} \sum_{\ell \in \mathbb{Z}} c_\ell e^{-jn\frac{\Omega}{N}t_\ell} = \sum_{n=-N}^{N} e^{jn\frac{\Omega}{N}t_k} [d]_n$$

In matrix form

$$P^{-1}q \simeq S^H d,$$  \hspace{1cm} (13)

where $S$ was defined in (10) and superscript $H$ denotes conjugate transposition. Introducing the diagonal matrix (see [12])

$$D = \text{diag}(t_{k+1} - t_k),$$ \hspace{1cm} (14)

and multiplying both sides of (13) by $\alpha SD$ gives $\alpha SDP^{-1}q = \alpha SDS^H d$. As a result, with

$$T = \alpha SDS^H = \alpha \sum_{k \in \mathbb{Z}} (t_{k+1} - t_k)e^{i(m-n)\frac{\Omega}{N}t_k}$$ \hspace{1cm} (15)

the set of linear equations

$$\alpha SDP^{-1}q = Td$$ \hspace{1cm} (16)

needs to be solved for $d$. Note that $P^{-1}$ and $\alpha$ are given (see (7) and (8)), and $S$, $D$, $T$ and $q$ are determined by the TCs. Note also that $T$ is a Hermitian Toeplitz matrix.

Therefore, instead of the exact reconstruction in (3) with corresponding linear equations $Gc = q$ represented by a not well structured matrix $G$ (see (4)), we now have an approximate reconstruction in (12) together with the linear equations in (16) represented by a structured matrix $T$. The “exact reconstruction” using infinite-dimensional vectors and matrices is most often intractable, and finite-dimensional matrices and vectors are used in practice.
2 The Proposed Real-Time TDM

Using the original reconstruction method of [11], also outlined in Section 1, a multiresolution algorithm is presented in [13] to stitch finite-dimensional approximations together by using appropriate window functions. We present below an alternative solution based on the Toeplitz formalism discussed in Section 1.1.

2.1 Finite Dimensional Covering

Our simulation experience shows that using the finite range $[t_k, t_\ell]$ with $k < \ell$, i.e. using the finite set of TCs $\{t_k, t_{k+1}, \ldots, t_\ell\}$, an accurate approximation can be achieved in a reduced range $[t_{k+M}, t_{\ell-M}]$ both for the original [11] and the Toeplitz-based reconstruction [12]. Here $M \in \mathbb{Z}$ typically ranges between 2 and 5. In particular, we define the finite-dimensional versions of the vectors and matrices introduced in Section 1.1 as:

\[
[q_{k,\ell}]_{i-k+1}^i = (-1)^i(2\kappa \delta - b(t_{i+1} - t_i))
\]

\[
D_{k,\ell} = \text{diag}(t_{i+1} - t_i)
\]

\[
[S_{k,\ell,N}]_{n+N+1,i-k+1} = e^{-jn \frac{\Omega}{N} t_i}
\]

\[
[P^{-1}]_{r-k+1,i-k+1} = \begin{cases} -1 & \text{if } r \leq i \\ 0 & \text{if } r > i \end{cases}
\]

\[
[T_{k,\ell,N}]_{m+N+1,n+N+1} = \alpha \sum_{i=k}^\ell (t_{i+1} - t_i)e^{j(m-n)\frac{\Omega}{N} t_i}
\]

for all integers $i, r = k, \ldots, \ell$ and $m, n = -N, \ldots, N$. Then, the correspondence of the linear equations in (16) becomes

\[
\alpha S_{k,\ell,N} D_{k,\ell} P^{-1}_{k,\ell} q_{k,\ell} = T_{k,\ell,N} d_{k,\ell,N}
\]

for the unknown (finite-dimensional) vector $d_{k,\ell,N}$. Since matrix $T_{k,\ell,N}$ of size $(2N+1) \times (2N+1)$ can easily be ill-conditioned, a familiar (minimum-norm) solution is given by

\[
d_{k,\ell,N} = \alpha T^+_{k,\ell,N} S_{k,\ell,N} D_{k,\ell} P^{-1}_{k,\ell} q_{k,\ell}
\]

where $T^+_{k,\ell,N}$ denotes the pseudo inverse of $T_{k,\ell,N}$ [1]. Using the solution $d_{k,\ell,N}$ in (12) gives

\[
x_{k,\ell,N}(t) = j \frac{\Omega}{N} \sum_{n=-N}^N n e^{jn \frac{\Omega}{N} t} [d_{k,\ell,N}]_n
\]

as a Fourier-series approximation for $x(t)$ for $t \in [t_{k+M}, t_{\ell-M}]$ using finite number of coefficients.
2.2 A Multiresolution Algorithm for Signal Recovery

Consider the set of window functions, \( w(t) \), forming a partition of unity

\[
\sum_{n \in \mathbb{Z}} w(t - nT_0) = 1 \quad (20)
\]

with some known period \( T_0 \). Several window functions of this type are available in the literature. For example, partition of unity is a requirement for so called admissible scaling functions in Wavelet Theory (see, e.g., [16, 3]). To simplify the analysis we assume that \( w(t) \) is even and has a compact support, i.e., \( w(t) = w(-t) \) and for some known \( T_w \) we have \( w(t) = 0 \) if \( t \not\in [-T_w, T_w] \). Here \( T_w \) is determined by \( T_0 \) depending on the (known) shape of \( w(t) \) as illustrated Figure 2.

Now, from the obvious relationship (see also (20))

\[
x(t) = x(t) \sum_{n \in \mathbb{Z}} w(t - nT_0) = \sum_{n \in \mathbb{Z}} w(t - nT_0)x(t) \quad (21)
\]

we can see (see also Figure 2) that

\[
w(t - nT_0)x(t) = 0 \quad \text{for} \quad t \not\in [nT_0 - T_w, nT_0 + T_w]
\]

holds.

Therefore, if a good approximation of \( x(t) \) can be achieved for \( t \in [nT_0 - T_w, nT_0 + T_w] \), then no problem will arise if the approximation is not good for \( t \not\in [nT_0 - T_w, nT_0 + T_w] \). The approximation discussed in Section 2.1 can be matched to the window functions as illustrated in Figure 3. As shown, since the times \( nT_0 - T_w \) are known, \( k_n \in \mathbb{Z} \) and \( \ell_n \in \mathbb{Z} \) are defined as:

\[
k_n := \{ t_{k_n} \leq nT_0 - T_w \quad \text{and} \quad t_{k_n+1} > nT_0 - T_w \}
\]

\[
\ell_n := \{ t_{\ell_n-1} \leq nT_0 + T_w \quad \text{and} \quad t_{\ell_n} > nT_0 + T_w \}. \quad (22)
\]
Comparing this results with the formulation of Section 2.1 we have that $k = k_n - M$ and $\ell = \ell_n + M$. Therefore, with the notation of (19) the approximating signal corresponding to $w(t - nT_0)$ is given by:

$$x_{kn-M,\ell_n+M,N}(t) = j\frac{\Omega}{N} \sum_{m=-N}^{N} me^{j\frac{\Omega}{N}t}[d_{kn-M,\ell_n+M,N}]_m.$$  \hspace{1cm} (23)

Substituting this relationship into (21) gives:

$$\sum_{n \in \mathbb{Z}} w(t - nT_0)j\frac{\Omega}{N} \sum_{m=-N}^{N} me^{j\frac{\Omega}{N}t}[d_{kn-M,\ell_n+M,N}]_m.$$  

Introducing

$$d(M, N)_{m,n} = m[d_{kn-M,\ell_n+M,N}]_m$$  \hspace{1cm} (24)

gives the overall approximation for $x(t)$:

$$x_{M,N}(t) = \sum_{n \in \mathbb{Z}} \sum_{m=-N}^{N} d(M, N)_{m,n} w(t - nT_0)e^{j\frac{\Omega}{N}t}$$  \hspace{1cm} (25)

### 2.3 Example

With $c = 0.3$ and $\Omega = 2\pi \times 40$ krad/s, the input signal was created as a sum 20 sinusoids with amplitudes, frequencies, and phases randomly selected within $[-c, c]$, $[0, \Omega/2\pi]$, and $[0, 2\pi]$, respectively. In numerical simulations for the TEM in Figure 1, 123 TCs together with the signals shown were determined with high accuracy. The input signal $x(t)$, the integrator output $y(t)$, and the overall TEM output $z(t)$ are shown in Figure 4 ranging $t$ from zero to $875.5 \mu s$.

Using the settings $T_0 = T_w = \frac{2\pi}{\Omega}$, the window functions were selected as:

$$w(t) = \begin{cases} 
\cos^2 \left( \frac{\pi t}{2Tw} \right) & \text{if } t \in [-T_w, T_w] \\
0 & \text{if } t \notin [-T_w, T_w]
\end{cases}$$
Therefore the Fourier transform of \( w(t) \) is given by

\[
W(\omega) = \frac{\pi^2 \sin(\omega T_w)}{\pi^2 \omega - T_w^2 \omega^3}
\]

Figure 5 shows \( w(t) \) and \( W(\omega) \) in scaled form. Two time-shifted window functions, \( w(t - 10T_0) \) and \( w(t - 22T_0) \), are also shown in Figure 6 in dashed gray line.

With (see Figure 3) \( M = 5 \) the proposed reconstruction procedure was implemented. Using the periodic functions in (23) simulation results are shown in Figure 6 for \( N = 8 \). It can be seen that \( x_{26,44,8}(t) \) and \( x_{68,86,8}(t) \) give good approximations for \( t \in [nT_0 - T_w, nT_0 + T_w] \), but outside this range the periodic approximations are poor. However, this is not a problem since \( w(t - nT_0) = 0 \) hence \( w(t - nT_0)x_{k_n-M,\ell_n+M,N}(t) = 0 \) for \( t \not\in [nT_0 - T_w, nT_0 + T_w] \). In the figure, \( \mathcal{E}_n \) denotes the root-mean-square (RMS) value of the error functions defined as \( e_n(t) = x_{k_n-M,\ell_n+M,N}(t) - x(t) \) and evaluated over the support of \( w(t - nT_0) = 0, t \in [nT_0 - T_w, nT_0 + T_w] \).

The quality of the overall reconstruction in (25) is quantified by the error function

\[
e_{M,N}(t) = x_{M,N}(t) - x(t)
\]

for \( t \in [T_{\min}, T_{\max}] \). Here \( T_{\max} \) and \( T_{\min} \) are appropriate simulation-dependent bounds. The RMS value of \( e_{M,N}(t) \) in [dB] is defined by:

\[
\mathcal{E}_{M,N} = 10 \log \left( \frac{\int_{T_{\min}}^{T_{\max}} e_{M,N}^2(t) dt}{T_{\max} - T_{\min}} \right)
\]
Figure 6: Approximating periodic functions given by (23) with $N = 8$.

Figure 7: Scaled error function $e_{5,8}(t)$ with RMS value $\mathcal{E}_{5,8}$ evaluated using $T_{\text{min}} = 75 \mu s$ and $T_{\text{max}} = 800 \mu s$.

Figure 8: Scaled error function $e_{5,10}(t)$ with RMS value $\mathcal{E}_{5,10}$ evaluated using $T_{\text{min}} = 75 \mu s$ and $T_{\text{max}} = 800 \mu s$.

Figure 7 shows $e_{5,8}(t)$ and $\mathcal{E}_{5,8}$ corresponding to the case $N = 8$, i.e., the case when the approximations in Figure 6 were obtained. The RMS error shows good agreement with those shown in Figure 6. Figure 8 shows $e_{5,10}(t)$ and $\mathcal{E}_{5,10}$ corresponding to the case $N = 10$. 
Thus, increasing \( N \), i.e., generating a better approximation for \( g(t) \) in (8) increases the accuracy. However, the conditioning of the system matrices gets worse by increasing \( N \) \[12\]. The designer has a number of parameters to tune including the window width \( T_w \), the window type, \( N \), and \( M \). Generally, increasing \( T_w \) requires larger value for \( N \).

### 2.4 Postfiltering

The bandwidth of \( x_{M,N}(t) \) certainly exceeds \( \Omega \). In particular, let \( \Omega_w \) be such that \( W(\omega) \approx 0 \) for \(|\omega| > \Omega_w \). For example, with parameters of Section 2.3 using Figure 5, \( \Omega_w = 3\Omega \) appears to be a safe choice. Since the bandwidth of \( x_{k_0-M,\ell_0+M,N}(t) \) is \( \Omega \) (see (23)), the bandwidth of the product \( x_{k_0-M,\ell_0+M,N}(t)w(t-nT_0) \), and thus that of \( x_{M,N}(t) \) is \( \Omega + \Omega_w \). Using broader \( w(t) \) in the time domain narrows \( W(\omega) \) in the frequency domain, hence decreases \( \Omega_u \). However, with broader \( w(t) \) more TCs are covered by \((nT_0 - T_w, nT_0 + T_w)\) which generally needs larger \( N \). The enlarged size of the Toeplitz matrices \( T_{k_n-M,\ell_n+M,N} \) increases the computational load for calculating the pseudo inverses \( T_{k_n-M,\ell_n+M,N}^+ \) in (18).

By appropriately choosing \( w(t) \), \( \Omega_w \) can be decreased for fixed \( T_w \) hence \( T_0 \). For example, if \( w(t) \) is chosen as that in Figure 5, then both \( w(t) \) and its derivative are continuous for all \( t \in \mathbb{R} \). Then, as shown in Figure 5, a good enough frequency localization for \( W(\omega) \) can be achieved.

Passing \( x_{M,N}(t) \) through a lowpass filter with cutoff-frequency \( \Omega \) restores the original bandwidth of the input signal. If digital signal processing is required on the reconstructed signal, the samples \( x_{M,N}(nT_s), T_s \geq \pi/(\Omega + \Omega_w) \), can be processed by a discrete-time LPF with (digital) cutoff frequency \( \pi/(1 + \Omega_w/\Omega) \). Since the reconstruction error spreads over the range \( \omega \in (\Omega_w - \Omega, \Omega_w + \Omega) \), lowpass filtering in either analog or discrete-time domain further improves the overall accuracy \[13\].

### 2.5 Pseudo-Inverse Recursion

For a real-time TDM, the pseudo inversions \( T_{k_0,\ell_0,N}^+ \) is a critical factor in terms of both computational complexity and accuracy. As in \[13\], instead of calculating \( T_{k_n-M,\ell_n+M,N}^+ \) individually for each \( n \), recursive solutions can be developed taking advantage of the fact that \( T_{k_0-M,\ell_0+M,N} \) and \( T_{k_0+1-M,\ell_0+1+M,N} \) have a number of common elements. One possible solution is based on the general result in \[1\] (page 50, Corollary 3.3.1):

\[
(A + \text{uv}^H)^+ = A^+ - \frac{1}{\beta}A^+\text{uv}^HA^+ \\
\text{where } \beta = 1 + \text{v}^HA^+\text{u} \neq 0
\]

Therefore, if an arbitrary matrix \( A \) is modified by a rank-one matrix, \( \text{uv}^H \), then the pseudo inverse is also modified by a rank-one matrix, \( (A^+\text{u})(\text{v}^H A^+)\beta \).

Any matrix can be decomposed as a sum of rank-one matrices in several ways. For our
Toeplitz matrices

\[
[T_{k_n-M,\ell_n+M,N}]_{m+N+1,n+N+1} = \alpha \sum_{i=k_n-M}^{\ell_n+M} (t_{i+1} - t_i) e^{i(m-r)\frac{\Omega}{N} t_i}
\]

with \(m, r = -N, \ldots, N\) a decomposition as a sum of rank-one Hermitian matrices is possible after defining

\[
[u_i]_{r+N+1} = \sqrt{\alpha(t_{i+1} - t_i)} e^{-j\frac{\Omega}{N} t_i}
\]

with \(r = -N, \ldots, N\) and \(i = k_n - M, \ldots, \ell_n + M\) as:

\[
T_{k_n-M,\ell_n+M,N} = \sum_{i=k_n-M}^{\ell_n+M} u_i u_i^H.
\] (27)

Therefore for the “next” \(n\) (replacing \(n\) by \(n+1\)):

\[
T_{k_n+1-M,\ell_n+1+M,N} = \sum_{i=k_{n+1}-M}^{\ell_{n+1}+M} u_i u_i^H.
\] (28)

Since \(k_{n+1} > k_n\) and \(\ell_{n+1} > \ell_n\), combining (27) and (28) gives:

\[
T_{k_n+1-M,\ell_n+1+M,N} = T_{k_n-M,\ell_n+M,N} + \sum_{i=\ell_n+M+1}^{\ell_{n+1}+M} u_i u_i^H - \sum_{i=k_n-M}^{\ell_{n+1}-M-1} u_i u_i^H.
\]

Whenever a rank-one matrix \(u_i u_i^H\) is added to or subtracted from \(T_{k_n-M,\ell_n+M,N}\), the result in (26) can be used recursively for calculating \(T_{k_n+1-M,\ell_n+1+M,N}\) in terms of \(T_{k_0-M,\ell_0+M,N}\). If for \(n = 0\) some initial pseudo inverse \(T_{k_0-M,\ell_0+M,N}\) is given, then no further pseudo inversion is needed in the recursion for \(n > 0\). Note however that, not even this initial pseudo inversion is needed if a short initial “transient” type of error can be tolerated in the overall reconstruction. Since in each recursion step new rank-one matrices are added and old rank-one matrices are subtracted, the “effect” of any initial value, say

\[
T_{k_0-M,\ell_0+M,N} = \sum_{i=k_0-M}^{\ell_0+M} u_i u_i^H = I
\]

disappears after a finite number of steps, denoted by \(\hat{n}\), where \(I\) stands for an identity matrix of dimensions \((2N+1) \times (2N+1)\). Then, for \(n \geq \hat{n}\) the recursion for \(T_{k_{n+1}-M,\ell_{n+1}+M,N}\), and thereby its pseudo inverse \(T_{k_{n+1}-M,\ell_{n+1}+M,N}\), becomes accurate. This method was confirmed by simulations. After the disappearance of the initial errors the same error function was obtained as that shown in Figure 7 for \(N = 8\) We note that for large \(N\) the method exhibits sensitivity to the order of the consecutive subtractions and additions of \(u_i u_i^H\).
2.6 Reconstruction by Using FFT

If further evaluation using digital signal processing is needed on the reconstructed signal, it has to be sampled uniformly. We show that under mild conditions the uniformly-taken samples can be calculated efficiently via FFT. Denoting the sampling period by $T_s$, the reconstruction in (25) becomes:

$$x_{M,N}(kT_s) = \sum_{n \in \mathbb{Z}} w(kT_s - nT_0)D(M, N)_{k,n}$$

where:

$$D(M, N)_{k,n} = \sum_{m=-N}^{N} d(M, N)_{m,n}e^{jm\Omega N k T_s}.$$  \hfill(29)

Denoting the Nyquist period of $x(t)$ by $T$ ($\Omega = \pi/T$) and with appropriate positive integers $N_w, N_0$, and $N_s$ assuming

$$T_w = N_w T_s, \quad T_0 = N_0 T_s \quad \text{and} \quad T = N_s T_s$$  \hfill(30)

we have:

$$x_{M,N}(kT_s) = \sum_{n \in \mathbb{Z}} w(kT_s - nN_0 T_s)D(M, N)_{k,n}$$  \hfill(31)

Since $w(t) \neq 0$ holds only for $-T_w \leq t \leq T_w$, $w(kT_s - nN_0 T_s) \neq 0$ only when $-T_w \leq kT_s - nN_0 T_s \leq T_w$ or equivalently (see (30)) when $N_w + k \leq nN_0 \leq -N_w + k$ holds. Therefore, (31) becomes:

$$x_{M,N}(kT_s) = \sum_{n=[-N_w]}^{[N_w]} w(kT_s - nN_0 T_s)D(M, N)_{k,n}$$  \hfill(32)

Rewriting $D(M, N)_{k,n}$ in (29) gives:

$$D(M, N)_{k,n} = e^{-j\Omega k T_s} \sum_{\ell=0}^{2N} d(M, N)_{\ell-N,n}e^{j\ell\Omega N k T_s}.$$  

Using (30) and the familiar notation

$$W_K = e^{j\frac{2\pi}{K}}$$

gives:

$$D(M, N)_{k,n} = W_{2N_s}^{-k} \sum_{\ell=0}^{2N} d(M, N)_{\ell-N,n}W_{2N N_s}^{\ell k}$$

12
Since the bandwidth of the reconstructed signal is greater than \( \Omega \) (see Section 2.4) \( T_s < T \) certainly holds. Then, however, \( N_s > 1 \), and thereby \( 2NN_s > 2N + 1 \) also holds. Therefore, with the zero-padded sequence

\[
\begin{align*}
    h(M, N)_{\ell,n} &= \begin{cases} 
    d(M, N)_{\ell-N,n} & \text{if } \ell \in [0, 2N] \\
    0 & \text{if } \ell \in (2N, 2NN_s - 1]
    \end{cases} \\
\end{align*}
\] (33)

we have

\[
    D(M, N)_{k,n} = W_{2NN_s}^{-k} \sum_{\ell=0}^{2NN_s-1} h(M, N)_{\ell,n} W_{2NN_s}^{\ell k}.
\] (34)

The summation in (34) can directly be calculated by using FFT.

\section{Conclusions and Future Work}

We summarize some of the potentials of the proposed method.

- Since apart from the partition-of-unity requirement no constraints are imposed on the window functions \( w(t) \), several scaling functions used in the theory of Wavelets, e.g. [3], can be tested and compared in terms of accuracy, computational complexity, and insensitivity to finite-precision arithmetic. Our immediate goal is to consider orthogonal scaling functions.

- Note that the reconstruction in (25) is essentially a Gabor-system representation of \( x(t) \), see e.g. [6, 3]. Our goal is to investigate Gaussian windows when Gabor systems become Gabor frames. Although Gaussian windows are not compactly supported, they might have other advantages.

- The key formula in (26) for pseudo-inverse recursion does not take advantage of the fact that we are dealing Hermitian Toeplitz matrices. Our goal is to simplify the recursion in Section 2.5 by taking this property into account.

\section*{References}


