ABSTRACT
Spectral domain B-spline identification is proposed for acoustic echo cancellation. Two approaches are considered. The first is based on solution of normal equations; we describe an efficient technique for such a solution, which benefits from the sparseness of the system matrix due to B-splines. The second approach is based on using local splines, enabling further simplification. We also show how the proposed techniques can be used for efficient double-talk detection. The echo cancellation performance and complexity of the proposed techniques are compared with that of the cross-spectral technique from [5] and the AP algorithm.

2. SPECTRAL-DOMAIN SPLINE-IDENTIFICATION
In application to acoustic echo cancellation, the identification problem can be described as follows. The microphone signal is

\[ y(t) = u(t) + z(t) \]  

where \( t \) is discrete time, \( u(t) = \sum_{\tau=0}^{L-1} x(t - \tau) h(\tau) \) is the echo signal, \( z(t) \) is the near-end signal and/or white Gaussian noise, \( x(t) \) is the excitation (far-end) signal, and \( h(\tau) \) is the acoustic impulse response to be estimated, and \( L \) is the length of \( h(\tau) \). In the spectral domain, this can be represented as

\[ Y(\omega_k) = X(\omega_k) H(\omega_k) + Z(\omega_k) \]  

where \( X(\omega_k) \) is the spectrum of the excitation signal \( x(t) \), \( Z(\omega_k) \) is the spectrum of the noise and near-end signal \( z(t) \), and \( \omega_k \in \Omega \) form a frequency grid in the frequency bandwidth of interest \( \Omega = [\omega_l, \omega_u] \). The spectra of microphone and excitation signals are calculated over a block of \( N \) samples by using FFT as

\[
Y(\omega_k) = \frac{1}{N} \sum_{i=0}^{N-1} w(i) y(i + t - N + 1) e^{-j\frac{2\pi ki}{N}},
\]

\[
X(\omega_k) = \frac{1}{N} \sum_{i=0}^{N-1} w(i) x(i + t - N + 1) e^{-j\frac{2\pi ki}{N}}
\]

where \( w(i) \) is a window (e.g., the Hamming window). Since \( y(t) \) and \( x(t) \) are real-valued, we are only interested in the first \( N/2 \) frequency bins of the FFTs, \( k = 0, \ldots, N/2 - 1 \).

The frequency response \( H(\omega_k) \) is approximated by a series

\[ \hat{H}(\omega_k) = \sum_{p=1}^{N_p} c_p \varphi_p(\omega_k) \]  

where \( \{ \varphi_p(\omega_k) \} \) are \( N_p \) basis functions. Minimisation of the error

\[ \varepsilon^2 = \sum_{\omega_k \in \Omega} \left| Y(\omega_k) - X(\omega_k) \hat{H}(\omega_k) \right|^2 \]
results in expansion coefficients $c = [c_1, \ldots, c_N]^{T}$ being the solution of normal equations

$$Rc = \xi$$  \hspace{1cm} (7)

where the vector $\xi$ contains elements

$$\xi_q = \sum_{\omega_k \in \Omega} S_{\nu X}(\omega_k) \varphi_q(\omega_k), \hspace{0.5cm} q = 1, \ldots, N_{\varphi} \hspace{1cm} (8)$$

the matrix $R$ contains elements

$$r_{qp} = \sum_{\omega_k \in \Omega} S_{\nu X}(\omega_k) \varphi_q(\omega_k) \varphi^*_p(\omega_k), \hspace{0.5cm} q, p = 1, \ldots, N_{\varphi}, \hspace{0.5cm} (9)$$

$S_{\nu X} = Y(\omega_k)X^*(\omega_k)$ and $S_{\nu X} = X(\omega_k)X^*(\omega_k)$ are respectively the cross-spectrum and auto-spectrum, and $(\cdot)^*$ denotes complex conjugate. To improve the convergence when solving the system (7), the matrix $R$ is regularised as $R \Rightarrow R + \delta \mathbf{I}$, where $\delta > 0$ is a regularisation parameter and $\mathbf{I}$ is the identity matrix.

If basis functions are complex harmonics, the echo path is modeled as a FIR filter with $N_{\varphi}$ filter taps, $R$ is the auto-correlation matrix of the excitation signal, and $\xi$ is the cross-correlation vector of the excitation and microphone signals. This case corresponds to the block LS approach adopted in [3]. Unfortunately, for harmonic basis functions, the matrix $R$ in general is not sparse, which makes solving the system (7) with large $N_{\varphi}$ a complicated problem.

We propose polynomial spline-approximation. Splines provide a low approximation error with a low degree of the polynomial; cubic splines are considered to provide the best trade off between accuracy and complexity [7, 8, 9]. Among spline basis functions, B-splines possess features attractive for implementation [7, 8], e.g., they have the minimum support which leads to a simple calculation of the sparse matrix $R$ and vector $\xi$, and simplify (due to the sparseness) the solution of equations (7). Below we use cubic B-splines

$$\varphi_p(\omega_k) = b_3(\omega - \omega_k - (p - 2)\Delta \omega)$$  \hspace{1cm} (10)

where $\Delta \omega = (\omega_n - \omega_i)/(N_{\varphi} - 3)$ and

$$b_3(\omega) = \begin{cases} \frac{1}{6} (2 - |\omega|)^3 & \text{if } 0 \leq |\omega| < \Delta \omega \\ \frac{1}{6} (2 - \frac{|\omega|}{\Delta \omega})^3 & \text{if } \Delta \omega \leq |\omega| < 2\Delta \omega \\ 0 & \text{otherwise} \end{cases}$$  \hspace{1cm} (11)

It is convenient to choose $\Delta \omega$ as a multiple of the FFT bin: $\Delta \omega = (2\pi/N)D$, where $D$ is an integer. Then samples of basis functions are discrete shifts of $4D$ samples of $b_3(\omega)$ by $D$ FFT frequency bins. Fig.1 shows a few basis B-splines in a part of the frequency range $\Omega$ in the case of $N = 8192$, $D = 7$, and the sampling frequency $F_s = 8$ kHz; circles indicate B-spline values at the FFT frequencies.

The block estimate of the impulse response is obtained by the inverse Fourier transform and truncation

$$\hat{h}_t(\tau) = \frac{1}{N} \sum_{k=0}^{N-1} \hat{H}(\omega_k) e^{j \omega_k \tau}, \hspace{0.5cm} \tau = 0, 1, \ldots, L - 1. \hspace{1cm} (12)$$

The final estimate of the room impulse response $h(\tau)$ is updated as

$$\hat{h}_t(\tau) = (1 - \alpha_t) \hat{h}_{t-M}(\tau) + \alpha_t \hat{h}_t(\tau), \hspace{0.5cm} \tau = 0, 1, \ldots, L - 1. \hspace{1cm} (13)$$

where $M$ is the number of samples between two blocks and $\alpha_t$ is a time-varying forgetting factor determined as described in section 5. The estimate $\hat{h}_t(\tau)$ is used for FIR filtering to model the echo; it remains constant over $M$ samples until the next block processing.

### 3. DCD ALGORITHM

A significant computational problem with such identification is in the solution of the system of equations (7). The use of conventional techniques like the Cholesky decomposition leads to high complexity (see section 7).

We use the recently introduced DCD algorithm [6]. It is based on binary representation of elements of the solution vector with $M$ bits within an amplitude range $[-H, H]$. It starts an iterative approximation of the solution vector $c$ from the most significant bit. Once the most significant bit has been found for all vector elements, the algorithm starts updating the next less significant bit, and so on. If an update happens (such an iteration is called ‘successful’), the vector $\xi$ is also updated. Parameters of the DCD algorithm are the number of bits $M$, representing elements of the vector $c$ within the amplitude range $[-H, H]$ and the maximum number of iterations $N_t$.

As $R$ is real, the system can be solved separately for real $c^{(r)}$ and imaginary $c^{(i)}$ parts of the vector $c = c^{(r)} + j c^{(i)}$ as $R^{(r)} = \xi^{(r)}$ and $R^{(i)} = \xi^{(i)}$, respectively. Then the solution is represented as $c = c^{(r)} + j c^{(i)}$.

In application to the cubic B-spline identification, the DCD algorithm can be described as follows (for the real part of the system).

Initialization: $c^{(r)} = 0$, the step-size $d = H$.

For every $m^{th}$ bit, $m = 1, \ldots, M$, the step-size is reduced as $d = d/2$, the iteration counter $it = 0$, and iterations start:

(1) Indicator of ‘successful’ iterations is reset, Flag = 0, and the iteration counter is incremented, $it = it + 1$.

(2) For $p = 0, \ldots, N_{\varphi} - 1$: if $|\xi^{(r)}_p| > (d/2)r_{pp}$, then the iteration is ‘successful’, Flag = 1, $c^{(r)}_p = c^{(r)}_p + d$, elements of the vector $\xi$ with indexes $q \in \{\max(1, p + n), \min(N_{\varphi}, p + n)\}$ are updated as $\xi^{(r)}_q = \xi^{(r)}_q - \text{sgn}(\xi^{(r)}_q) d r_{pq}$.

(3) If $it = N_{\text{it}}$, then the algorithm stops.
(4) If Flag = 1, then steps (1), (2), and (3) repeat; otherwise iterations start for the next less significant bit \(m = m + 1\) with a reduced step-size \(d = d/2\).

The DCD algorithm guarantees convergence to the true solution if elements of the true solution vector \(\mathbf{e}\) are within the interval \([-H, H]\). If \(H\) is a power of two, then multiplications by factors of power of two are only used; these can be replaced by bit shifts. Thus, the DCD algorithm can be implemented without explicit multiplications, which may be useful in hardware implementation.

### 4. LOCAL SPLINE-APPROXIMATION

Although the DCD algorithm allows efficient solution of normal equations, it may still require a considerable computational load. The use of local splines allows us to avoid such calculations with a slightly higher approximation error [9]. The spline coefficients \(c_a\) can be calculated as follows

\[
c_q = a_0 \xi_q + a_1 (\xi_{q-1} + \xi_{q+1}) + a_2 (\xi_{q-2} + \xi_{q+2}), \quad (14)
\]

\[
\xi_q = \frac{\sum_{\omega_k \in \Omega_q} S_{XY}(\omega_k)}{\sum_{\omega_k \in \Omega_q} S_{XX}(\omega_k)}, \quad (15)
\]

where \(\Omega_q = [\omega_1 + \Delta \omega (q-2) - \Delta \omega / 2, \omega_1 + \Delta \omega (q-2) + \Delta \omega / 2]\).

The weights should satisfy the condition \(a_0 + 2a_1 + 2a_2 = 1\) and they are tuned to obtain the best cancellation performance. In our simulation, the weights are \(a_0 = 1.94, a_1 = -0.58, a_2 = 0.11\).

### 5. DOUBLE-TALK DETECTION

Relationship between the error \(\varepsilon^2\) in (6) and the energy of the microphone signal \(E_y = \sum_{\omega_k \in \Omega} |Y(\omega_k)|^2\) characterizes accuracy of the identification. If the accuracy is low, i.e. \(\varepsilon^2\) is close to \(E_y\) (e.g. in double-talk situation), then the block impulse response estimate is ignored by setting \(\alpha_l = 0\). If \(\varepsilon^2\) is much smaller than \(E_y\), we can update the impulse response estimate by adding the block estimate with the weight \(\alpha_l\) close to 1. This can be considered as a spectral domain implementation of the “two-path” approach [4].

Note that main computations for calculation of \(\varepsilon^2\) and \(E_y\) have already been done and implementation of this approach requires a small extra computational load. In the simulation below, we use the following mapping of the relationship between \(\varepsilon^2\) and \(E_y\) to the forgetting factor \(\alpha_l\):

\[
\alpha_l = \begin{cases} 
\alpha (1), & \varepsilon^2 < \rho (1) E_y \\
\alpha (2), & \rho (1) E_y \leq \varepsilon^2 < \rho (2) E_y \\
\alpha (3), & \rho (2) E_y \leq \varepsilon^2 < \rho (3) E_y \\
0, & \varepsilon^2 \geq \rho (3) E_y.
\end{cases} \quad (16)
\]

The vectors \(\alpha = [\alpha (1), \alpha (2), \alpha (3)]\) and \(\rho = [\rho (1), \rho (2), \rho (3)]\) are chosen to obtain the best cancellation performance. Other dependencies \(\alpha_l\) on \(\varepsilon^2\) and \(E_y\) can also be used. We have found the mapping (16) efficient and simple for implementation.

### 6. NUMERICAL RESULTS

We simulate acoustic echo cancellation in the following scenarios. The acoustic impulse response \(h = [h(0), \ldots, h(L-1)]\) has a length \(L = 512\). The excitation signal is 11-sec female speech sampled at \(F_s = 8\) kHz with a 16-bit resolution. Experimental plots have been obtained by averaging the misalignment \(|h - \hat{h}|^2 / |h|^2\) in 20 trials. In each trial, new excitation speech and noise signals are used. Echo attenuation (ERLE) is calculated over intervals between 2nd and 11th seconds and averaged over the 20 trials. In double-talk scenarios, the near-end speech is applied for 4th and 7th seconds with a power equal to that of the echo signal.

We compare: (1) optimal splines with ideal matrix inversion; (2) optimal splines with the DCD algorithm; (3) local splines; (4) the AP algorithm; and (5) the cross-spectral algorithm. Parameters of the spline identification are: \(N = 8192, M = 2000, D = 7, N_p = 585, \delta = 0.5, \alpha = [0.4, 0.1, 0.05], \rho = [0.015, 0.1, 0.25], \omega_l = 0, \omega_u = \pi F_s\). Parameters of the DCD algorithm are: \(H = 1, M_b = 8, N_t = 20\). In the AP algorithm, the AP order is \(N_{AP} = 8\), the step-size is 0.125, and the regularisation parameter \(\delta = 10^5\). In the cross-spectral algorithm, the average of cross-spectrum and auto-spectrum is performed over intervals of 0.64 sec as in [5].

Fig. 2 shows misalignment for scenarios with noise 30 dB down from the echo \((SNR = 30\) dB). The optimal splines with ideal matrix inversion and with the DCD algorithm demonstrate equal performance, therefore we show one plot only. Local splines provide a slightly higher (by about 1 dB) steady-state misalignment than the optimal splines. The convergence speed of spline algo-
Thus, the optimal spline identification with the DCD algorithm, local splines, the cross-spectral and AP algorithms require 340, 190, 90, and 4600 MACs/sample, respectively. By taking into account the FIR filtering and normalizing to the number of FIR taps \( L \), these can be represented as 1.7, 1.4, 1.1, and 9.0 MACs/sample/tap, respectively. Note that the NLMS algorithm requires 2 MACs/sample/tap; thus, all the spectral-domain algorithms have smaller complexity than the NLMS algorithm.

8. CONCLUSIONS

We have proposed new acoustic echo cancellation algorithms. These are based on time-domain FIR filtering and spectral domain cubic spline identification of the acoustic frequency response. We have considered optimal splines with ideal matrix inversion, optimal splines with the DCD algorithm, and local splines. Ideal matrix inversion and the DCD algorithm result in identical echo cancellation performance, while the DCD algorithm allows significant reduction in the complexity. The DCD-based optimal B-spline identification requires 1.7 MACs/sample/tap, while local splines require 1.4 MACs/sample/tap. The cross-spectral technique, though having a smaller complexity (1.1 MACs/sample/tap), provides poorer cancellation performance, especially at low SNRs. The affine projection algorithm provides a better echo attenuation at high SNRs and comparable at low SNRs; however, it is complex for implementation. The proposed techniques have also been shown to provide efficient double-talk detection.

9. REFERENCES