Low-Complexity Signal Processing Algorithms for Digital Subscriber Loops

Speaker: Shou-Sheu Lin

Department of Computer and Communication Engineering
National Kaohsiung First University of Science and Technology
Outline

- Introduction
- Interpolated Echo Canceller
- Interpolated Decision Feedback Equalizer and Precoder
- Conclusions
Introduction

The twisted-pair copper wire is known to be the most popular broadband access media in the world.

Several digital subscriber line technologies have been proposed and successfully commercialized.

- ISDN
- HDSL
- ADSL, ADSL2, ADSL2+
- VDSL, VDSL2
Introduction

Major impairments in the DSL environment
- Echo
- ISI
- Noises
Introduction

Signal processing for challenging impairments
- echo cancellation, channel equalization, and precoding.

Motivation
- For next DSL technologies
  - Higher throughput, higher sampling rate
  - Longer echo and ISI
  - Targeting wider service area coverage (better performance needed)
  - High-complexity signal processing requirements
- Propose low-complexity and high-performance signal processing algorithms for wireline applications
Proposed solutions

- Complexity reduction
  - Interpolation (and Extrapolation)
- Performance improvement
  - Turbo signal processing (Not Covered today)
Interpolated Echo Canceller
Introduction - Echo response model

Hybrid:
- For full-duplex transmission over a single pair of wire
- Echo: due to impedance mismatch
Introduction – Echo caused by the unbalance of a hybrid circuit

Hybrid model: echo due to impedance mismatch

Balance condition (without echo)

\[
\frac{Z_i}{R_1 + Z_i} = \frac{Z_b}{R_2 + Z_b}
\]
Introduction –
An example: the echo response of CSA #1
Conventional adaptive echo canceller

- **Echo canceller (EC):**
  - A system identification problem

- **Adaptive EC:**
  - A low-complexity implementation

![Diagram of echo canceller](image)
Previous works

- **FIR echo canceller** [Verhoeckx *et al.*, 1979]
  - Good performance, but high complexity

- **Adaptive interpolated FIR (IFIR) echo canceller** [Aboulnasr *et al.*, 1992]
  - Model long tail portion only or for special echo response
  - Overlook head echo

- **IIR echo canceller** [Kaelin *et al.*, 1995]
  - Adaptive IIR filter to model the long tail portion
  - Low complexity, stability problem, convergence problem
Proposed interpolated echo canceller

- Interpolated echo:

\[ h_h \]

\[ h_r \]

\[ N_h - 1 \]

\[ \alpha - 1 \]

\[ w_1 \]

\[ g \ast w^U_2 \]

\( \text{(Avoid underdetermined problem)} \)
Adaptive interpolated EC:

**Proposed adaptive interpolated echo canceller**

\[ y_k = x_k - \sum_{a} w_{1} h(x_{k-a}) + \sum_{u} w_{2} U + \gamma_{k} + n_k \]

Interpolated echo canceller
Proposed adaptive interpolated echo canceller

Received echo signal:

\[ y_k = h^T x_k + n_k \]

Partition of echo signal (head and tail)

\[ y_k = h_h^T x_h + h_i^T x_i + n_k \]

Interpolated echo canceller:

\[ y_k = w_1^T x_{1,k} + w_2^T \phi_{2,k} \]

where \( \phi_{2,k} = \sum_{i=0}^{(N_g-1)} g_i x_{k-i} \)

Adaptive interpolated echo canceller (LMS):

Filtering

\[ y_k = \begin{bmatrix} w_1^T \\ w_2^T \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} = w^T \phi_k \]

Tap weights update

\[ w_{k+1} = w_k + \mu e_k \phi_k \]
Theoretical analysis – Wiener solution

Weiner solution:

\[ \hat{w}_o = \hat{R}^{-1}\hat{p} \]

where \( \hat{R}, \hat{p} \) is the punctured (at nulled taps) matrices of \( R, p \), respectively.

Original correlation matrices:

\[
R = E \left[ \frac{\phi_k \phi^*_k}{\phi_k} \right] = \begin{bmatrix} R_{x,x_1} & R_{x,\gamma_2} \\ R_{x,\gamma_2}^T & R_{\gamma_2,\gamma_2} \end{bmatrix}
\]

\[
\hat{R} = \begin{bmatrix} R_{x,x_1} & R_{x,\gamma_2} \\ R_{x,\gamma_2}^T & R_{\gamma_2,\gamma_2} \end{bmatrix}
\]

where

\[
R_{x,x_1} = \sigma_x^2 I_{N_1 \times N_1},
\]

\[
R_{x,\gamma_2} = \sigma_x^2 \begin{bmatrix} 0_{\alpha \times (N_1 - \alpha)} & 0_{\alpha \times (N_1' - N_1)} \\ I_{(N_1 - \alpha) \times (N_1 - \alpha)} & 0_{(N_1 - \alpha) \times (N_1' - N_1)} \end{bmatrix} M^T,
\]

\[
R_{\gamma_2,\gamma_2} = \sigma_x^2 M M^T,
\]

\[
p = \sigma_x^2 \begin{bmatrix} h(0 : N_1 - 1) \\ Mh(\alpha : N_h - 1) \end{bmatrix}
\]
Residual echo:
\[
\Delta h = h - (\hat{w}_{1,o} + g \ast \hat{w}_{2,o}^U)
\]

MMSE:
\[
\text{MMSE} = (\Delta h^T \Delta h) \sigma_x^2 + \sigma_n^2
\]

Echo return loss enhancement (ERLE):
\[
\text{ERLE} = 10 \cdot \log_{10} \frac{h^T h}{\Delta h^T \Delta h} \text{ (dB)}
\]
Convergence analysis

Convergence in mean (independence theory):

\[ \varepsilon_k = \hat{w}_k - \hat{w}_o \]

\[ E[\varepsilon_{k+1}] = (I - \mu \hat{R}) E[\varepsilon_k] \]

\[ \hat{w}_k \rightarrow \hat{w}_o \text{ if } 0 < \mu < \frac{2}{\lambda_{\text{max}}}, k \rightarrow \infty \]

where \( \lambda_{\text{max}} \) is the maximum eigenvalue of \( \hat{R} \)

Convergence in MSE:

\[ J_k = E\left[ e_k^2 \right] \]

\[ J_k \rightarrow J_{\infty}, \text{ if } 0 < \mu < \frac{2}{\lambda_{\text{max}}}, \sum_{n=1}^{N_i + N_o - S + 1} \mu \lambda_n / 2(1 - \mu \lambda_n) < 1 \]
Convergence analysis

Steady-state MSE:

\[ J_\infty = \frac{J_{\text{min}}}{N_1 + N_2 - S + 1}, \]

\[ 1 - \sum_{n=1}^{N_1 + N_2 - S + 1} \mu \lambda_n / 2(1 - \mu \lambda_n) \]

where \( J_{\text{min}} \) is the MMSE by Wiener solution

Misadjustment:

\[ \psi = \frac{\sum_{n=1}^{N_1 + N_2 - S + 1} \mu \lambda_n / 2(1 - \mu \lambda_n)}{1 - \sum_{n=1}^{N_1 + N_2 - S + 1} \mu \lambda_n / 2(1 - \mu \lambda_n)} \]

Simulations show that the convergence speed of the proposed LSM/IFIR is somewhat slower than that of the conventional LMS/FIR filter.

This is due to the fact that the input correlation matrix for the IFIR filter is not truly diagonal.
Optimal interpolation filter design

Original echo tail response

What is optimal $g$?

Interpolated echo tail

Minimize sum of the error square in this interval $I_1$ to $I_2$
Define the tail response of a loop as
\[ f = \begin{bmatrix} f_0 & f_1 & \cdots & f_{(N_2-1)M} \end{bmatrix}^T \]
\[ = \begin{bmatrix} h_\alpha & h_{\alpha+1} & \cdots & h_{\alpha+(N_2-1)M} \end{bmatrix}^T \]

Thus, the downsampled of above tail response is given
\[ \hat{f} = \begin{bmatrix} \hat{f}_0 & \hat{f}_1 & \cdots & \hat{f}_{(N_2-1)M} \end{bmatrix}^T \]
\[ = \begin{bmatrix} f_0, 0, L_{M-1}, 0, f_M, 0, L_{M-1}, 0, f_{2M}, 0, L_{M-1}, \ldots, f_{(N_2-1)M} \end{bmatrix}^T \]
\[ = \begin{bmatrix} h_\alpha, 0, L_{M-1}, 0, h_{\alpha+M}, 0, L_{M-1}, 0, h_{\alpha+2M}, 0, L_{M-1}, \ldots, h_{\alpha+(N_2-1)M} \end{bmatrix}^T \]

Interpolated tail response
\[ \hat{f} = \begin{bmatrix} \hat{f}_0 & \hat{f}_1 & \cdots & \hat{f}_{(N_2-1)M} \end{bmatrix}^T \]

where
\[ \hat{f}_k = \sum_{i=0}^{(N_2-1)} g_i \hat{f}_{k-i} \]
Optimal interpolation filter design for a single loop

Cost function:

\[ \xi(g) = \| f_s - \hat{f}_s \|^2 \]

where \( f_s = f(I_1 : I_2) \), \( \hat{f}_s = \hat{f}(I_1 : I_2) \)

In matrix form,

\[ \hat{f}_s = Fg, \quad F = \begin{bmatrix} \hat{f}_{I_1} & \hat{f}_{I_1-1} & L & \hat{f}_{I_1-(N_g-1)} \\ \hat{f}_{I_1+1} & \hat{f}_{I_1} & L & \hat{f}_{I_1-(N_g-2)} \\ M & O & O & M \\ \hat{f}_{I_2} & \hat{f}_{I_2-1} & L & \hat{f}_{I_2-(N_g-1)} \end{bmatrix} \]

The optimal interpolation filter (least-squares method)

\[ \hat{g} = (F^T F)^{-1} F^T f_s \]
Optimal interpolation filter
design for multiple loops

Jointly optimization for multiple loops

\[
\begin{bmatrix}
\frac{\partial}{\partial s_1} f_s \\
\frac{\partial}{\partial s_2} f_s \\
\vdots \\
\frac{\partial}{\partial s_N} f_s
\end{bmatrix} = \begin{bmatrix}
F_1 \\
F_2 \\
\vdots \\
F_N
\end{bmatrix} \cdot \mathbf{g},
\]

where \( f_s, F_i \in i\text{-th loop matrices}

Cost function:

\[
\bar{\xi}(\mathbf{g}) = \left\| \bar{f}_s - \bar{Fg} \right\|^2
\]

where \( \bar{f}_s = \left[ \frac{\partial}{\partial s_1} f_s, \frac{\partial}{\partial s_2} f_s, \ldots, \frac{\partial}{\partial s_N} f_s \right]^T, \bar{F} = [F_1, F_2, \ldots, F_N]^T
\]

Optimal interpolation filter:

\[
\mathbf{g} = (\bar{F}^T \bar{F})^{-1} \bar{F}^T \bar{f}_s
\]

Theoretical performance:

\[
\text{ERLE} = 10 \cdot \log_{10} \frac{\|\mathbf{h}_i\|^2}{\bar{\xi}_{\xi, \text{min}}} \quad \text{(dB)}
\]

where \( \bar{\xi}_{\text{min}} = \left\| \bar{f}_s - \bar{Fg} \right\|^2 = \bar{f}_s^T \left( I - \bar{F} (\bar{F}^T \bar{F})^{-1} \bar{F}^T \right) \bar{f}_s
\]
Optimal interpolation filter for multiple loops

(M=4, S=2, 8 CSA loops)
ERLE performance vs. various interpolation filters

Note: $M=4$, ERLE of CSA #1, Optimal filters for 8 CSA loops
Simulations

A) Loop characteristics and topologies
B) Noise environments: AWGN and NEXT
### Parameters for simulations

<table>
<thead>
<tr>
<th>Modulation</th>
<th>16-PAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tx/Rx filter</td>
<td>3-dB@775 KHz (6-th Butterworth)</td>
</tr>
<tr>
<td>Hybrid</td>
<td>Differential</td>
</tr>
<tr>
<td>Transformer</td>
<td>3 mH (primary inductance)</td>
</tr>
<tr>
<td>Impedance</td>
<td>135 Ohms</td>
</tr>
<tr>
<td>Cutting point</td>
<td>50</td>
</tr>
<tr>
<td>Interpolation factor (M)</td>
<td>4</td>
</tr>
<tr>
<td>Interpolation span (S)</td>
<td>2</td>
</tr>
<tr>
<td>Interpolation filter</td>
<td>Optimal (symmetric)</td>
</tr>
<tr>
<td>Sampling rate</td>
<td>775 KHz</td>
</tr>
<tr>
<td>LMS (variable step-size)</td>
<td>Initial (μ=0.004), 1/2 per segment</td>
</tr>
<tr>
<td></td>
<td>Training: 2,400/seg., total seg.: 5</td>
</tr>
</tbody>
</table>
A) Test loops: CSA loops

[ANSI T1.418, 2002]

Bridge tap

Gauge change

Most bridge taps
A) CSA loops (continue...)
A) SHDSL echo responses for CSA loops at CO side
A) Convergent $w_1$ and $w_2^U$ of the adaptive LMS/IFIR echo canceller

\[ (\alpha = 50, M=4, S=2, \text{OPS filter, CSA \#1}) \]
A) Convergent $w_1$ and $w_2^U$ : tap weight overlapping and nulling

$(\alpha = 50, M=4, S=2, \text{OPS filter, CSA #1})$
A) ERLE performance vs. CSA loops

(\(\alpha = 50, M=4, S=2,\) OPS filter, AWGN: -140 dBm/Hz)
## A) Complexity reduction

<table>
<thead>
<tr>
<th>Operation</th>
<th>Echo emulation</th>
<th>Taps weight update</th>
<th>Complexity ratios</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>+</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>M=2</td>
<td>156 158</td>
<td>151 151</td>
<td>62% 62%</td>
</tr>
<tr>
<td>M=4</td>
<td>114 116</td>
<td>101 101</td>
<td>43% 43%</td>
</tr>
<tr>
<td>M=8</td>
<td>105 107</td>
<td>76 76</td>
<td>36% 36%</td>
</tr>
</tbody>
</table>

Computational complexity ratio for different interpolation factor

$(N_I = 50, M=4, S=2, OPS filter)$
B) Power spectral densities for composite noises (crosstalk+AWGN)

![Power spectral densities graph]

- **Frequency (Hz)**
- **dBM/Hz**

- **Lines and Legends:**
  - Blue: AWGN
  - Green dotted: ISDN
  - Red: HDSL
  - Cyan: T1
  - Magenta: ADSL-US
  - Yellow: ADSL-DS
  - Black dotted: SHDSL

- **Frequency Range:** $10^5$ Hz
B) Simulated ERLE performance at CO side

\[ (N_f = 50, \text{M}=4, \text{S}=2, \text{OPS filter, composite noises}) \]
B) Power spectral densities of echo, residual echo, and composite noises

(10 SHDSL disturbers, CSA #1)
Interpolated Decision Feedback Equalizer and Precoder
ISI channel: due to frequency-selective transfer of a channel
Introduction –
An example: Channel response of CSA #6
Conventional decision feedback equalizer (DFE)

DFE structure:

Problem: long feedback filter
Previous works

- IIR DFE feedback filter [Crespo et al., 1991]
  - Adaptive two-pole IIR (for special channels)
  - Low complexity, numerical stable
  - Poor performance for general channels

- Channel shortening, plus IIR DFE feedback filter [Young., 1991]
  - Fixed 1\textsuperscript{st}-order FIR to shorten channel response (then assume postcursors only)
  - Adaptive one-pole IIR and FIR filter (for postcursors)
  - Low complexity, numerical stable
  - Poor performance for general channels
Previous works

ARMA-model for DFE feedforward and feedback filters [Al-Dhahir et al., 1991]

- Matrix-based multistage method
- Low-complexity for filtering only, but high preprocessing complexity
- Suitable for packet-based system
- Not applicable to DSL channel
DFE Feedback filter interpolation

**Interpolated feedback filter:**

- Conventional feedback filter: $b_k$
- Interpolated feedback head: $b_2$
- Interpolated feedback tail: $g \ast b_1^U$

\[
\alpha
\]

\[
1 \quad \alpha + 1
\]

\[
N_b
\]
The proposed low-complexity adaptive interpolated DFE (IDFE)

**IDFE structure:**

- **ISI channel:**
  - Input: $x_k$
  - Output: $r_k$

- **Decision directed**
  - Input: $x_{k-\Delta}$
  - Output: $\hat{x}_{k-\Delta}$

- **Training**
  - Input: $x_{k-\Delta}$
  - Output: $b_{2,k}$

- **Decision**
  - Input: $x_{k-\Delta-\alpha-1}$
  - Output: $z_{b1,k}$, $z_{b2,k}$

- **Sigma**
  - Input: $z_k$, $z_{f,k}$
  - Output: $y_k$

- **FFF**
  - Input: $y_k$
  - Output: $f_k$

- **FBF**
  - Input: $b_{2,k}$, $z_{b1,k}$, $z_{b2,k}$

Equation:

$$\hat{x}_{k-\Delta} = x_k - \Delta - \ldots$$
The proposed low-complexity interpolated Tomlinson-Harashima Precoder (THP)

Interpolated THP:

\[ x_k \stackrel{\Sigma}{\rightarrow} \text{Modulo} \rightarrow v_k \]

Interpolated Tomlinson-Harashima Precoder

\[ \Sigma \rightarrow b_{1,k} \rightarrow g \rightarrow b_{2,k} \]

ISI channel

\[ r_k \]

Noise

\[ n_k \]

\[ y_k \]

\[ \hat{x}_{k-\Delta} \rightarrow \text{Modulo} \rightarrow f_k \rightarrow \text{FFF} \]
Complexity reduction

\[ \frac{N_{b1}}{N_b} : \text{Feedback tail} \]
\[ \frac{N_{b2}}{N_b} : \text{Feedback head} \]
\[ \frac{N_g}{N_b} : \text{Interpolation} \]

min. C.R. = 24\% @ M=8

\[(DFE, N_f=16, N_b=180)\]
Proposed IDFE (IDFE)

Received signals:

\[
y_k = Hx_k + n_k
\]

Equalized signal (before decision):

\[
z_k = \hat{y}_k - \left( \sum_{i=1}^{2} b_i^T x_{i,k} \right)
\]

where

\[
\hat{y}_k = \begin{bmatrix} f_0 & f_1 & L & f_{N_f-1} \end{bmatrix}^T,
\]

\[
b_1 = \begin{bmatrix} b_{1,1} & b_{1,2} & L & b_{1,N_{b1}} \end{bmatrix}^T,
\]

\[
g = \begin{bmatrix} g_{-(M-1)} & g_{-(M-2)} & L & g_{(M-1)} \end{bmatrix}^T,
\]

\[
b_2 = \begin{bmatrix} b_{2,1} & b_{2,2} & L & b_{2,N_{b2}} \end{bmatrix}^T
\]

\[
x_{1,k} = \begin{bmatrix} f_{k-(\alpha+1)} & f_{k-(\alpha+1)-M} & L & f_{k-(\alpha+1)-(N_{b1}-1)M} \end{bmatrix}^T,
\]

\[
x_{2,k} = \begin{bmatrix} \hat{x}_{k-\Delta-1} & \hat{x}_{k-\Delta-2} & L & \hat{x}_{k-\Delta-N_{b2}} \end{bmatrix}^T
\]

\[
f_{k-(\alpha+1)} = \sum_{i=-(M-1)}^{(M-1)} g_i \hat{x}_{k-\Delta-\alpha-M-i}
\]

\[M: \text{interpolation factor, } g: \text{interpolation filter}\]
Proposed adaptive IDFE

Define:

\[
\begin{bmatrix}
  y_k \\
  x_{1,k} \\
  x_{2,k}
\end{bmatrix},
\begin{bmatrix}
  f_k \\
  -b_{1,k} \\
  -b_{2,k}
\end{bmatrix}
\]

Adaptive IDFE:

Filter output:

\[ z_k = w_k^T u_k \]

Estimation error:

\[ e_k = x_{k-\Delta} - z_k \]

Tap-weight update:

\[ w_{k+1} = w_k + \mu e_k u_k \]

where \( \mu \) is the step-size
Theoretical analysis – Wiener solution

Weiner solution:

\[
\begin{bmatrix}
0 \\
\mathbf{b}_1 \\
\mathbf{b}_2
\end{bmatrix}_{\text{mmse}} = \begin{bmatrix}
\mathbf{R}_{yy} & -\mathbf{R}_{yx_1} & -\mathbf{R}_{yx_2} \\
-\mathbf{R}_{yx_1}^T & \mathbf{R}_{x_1x_1} & \mathbf{R}_{x_1x_2} \\
-\mathbf{R}_{yx_2}^T & \mathbf{R}_{x_2x_1}^T & \mathbf{R}_{x_2x_2}
\end{bmatrix}^{-1} \begin{bmatrix}
p_{yx} \\
0 \\
0
\end{bmatrix}
\]

where

\[
\mathbf{R}_{yy} = \sigma^2_x \mathbf{H} \mathbf{H}^T + \sigma_n^2 \mathbf{I}_{N_f \times N_f}, \quad \mathbf{R}_{x_1x_1} = \sigma^2_x \mathbf{M} \mathbf{M}^T, \quad \mathbf{R}_{x_2x_2} = \sigma^2_x \mathbf{I}_{(M-1)\times(M-1)}
\]

\[
\mathbf{R}_{yx_1} = \sigma^2_x \mathbf{H} \begin{bmatrix}
\mathbf{0}_{(M-1)\times\alpha} & \mathbf{I}_{(M-1)\times(M-1)} \\
\mathbf{0}_{(N_b-M-1)\times\alpha} & \mathbf{0}_{(N_b-M-1)\times(M-1)}
\end{bmatrix},
\]

\[
\mathbf{R}_{yx_2} = \sigma^2_x \mathbf{H} \begin{bmatrix}
\mathbf{0}_{(M-1)\times(\Delta+1)} & \mathbf{I}_{(M-1)\times(M-1)} & \mathbf{0}_{(M-1)\times(N_f+N_b-M-\Delta-1)}
\end{bmatrix}^T,
\]

\[
\mathbf{R}_{yx_1}^T = \sigma^2_x \mathbf{H}^T \begin{bmatrix}
\mathbf{0}_{\Delta\times1} & \mathbf{1} & \mathbf{0}_{1\times(N_f+N_b-\Delta-2)}
\end{bmatrix}^T
\]

\[
\mathbf{p}_{yx} = \sigma^2_x \mathbf{H} \begin{bmatrix}
\mathbf{0}_{1\times\Delta} & \mathbf{1} & \mathbf{0}_{1\times(N_f+N_b-\Delta-2)}
\end{bmatrix}^T
\]

\[
\mathbf{M} = \begin{bmatrix}
g^T, \mathbf{1}^{44 \cdot 2 \cdot 4 \cdot 8}^0 \\
\mathbf{0}, \mathbf{0}, \mathbf{g}^T, \mathbf{0}, \mathbf{0}, \mathbf{0}
\end{bmatrix}_{(N_b-1)\times M}
\]

\[
\mathbf{M} = \begin{bmatrix}
\mathbf{0}, \mathbf{0}, \mathbf{1}^{44 \cdot 2 \cdot 4 \cdot 8}^0, g^T
\end{bmatrix}_{(N_b-1)\times M}
\]
Theoretical analysis - performance

MSE:

\[
J(f, b_1, b_2) = \sigma_x^2 + f^\top R_{yy} f + b_1^\top R_{x_1x_1} b_1 + b_2^\top R_{x_2x_2} b_2
- 2f^\top R_{yx_1} b_1 - 2f^\top R_{yx_2} b_2 + 2b_1^\top R_{x_1x_2} b_2 - 2p_{yx}^\top f
\]

MMSE:

\[
J_{\min, IDFE} = J(f_{mmse}^\top, b_{1, mmse}^\top, b_{2, mmse}^\top)
\]

SNR:

\[
SNR_{IDFE, MMSE} = \frac{\sigma_x^2}{J_{\min, IDFE}}
\]
Convergence analysis

Convergence in mean (independence theory):

\[ \varepsilon_k = w_k - w_o \]

\[ E[\varepsilon_{k+1}] = (I - \mu R_{uu}) E[\varepsilon_k], \text{ where } R_{uu} = E\{u_k u_k^T\} \]

\[ w_k \rightarrow w_o \text{ if } 0 < \mu < \frac{2}{\lambda_{\text{max}}}, k \rightarrow \infty \]

where \( \lambda_{\text{max}} \) is the maximum eigenvalue of \( R_{uu} \)

Convergence in MSE:

\[ J_k = E[\varepsilon_k^2] \]

\[ J_k \rightarrow J_\infty, \text{ if } 0 < \mu < \frac{2^{N_f + N_{p1} + N_{b2}}}{\lambda_{\text{max}}}, \sum_{n=1}^{\mu\lambda_n/2(1-\mu\lambda_n)} < 1 \]
Convergence analysis

Steady-state MSE:

\[ J_\infty = \frac{J_{\text{min}}}{N_f + N_{h1} + N_{h2}}, \]

\[ 1 - \sum_{n=1}^{N_f + N_{h1} + N_{h2}} \mu \lambda_n / 2 (1 - \mu \lambda_n) \]

where \( J_{\text{min}} \) is the MMSE by Wiener solution

Misadjustment:

\[ \psi = \frac{\sum_{n=1}^{N_f + N_{h1} + N_{h2}} \mu \lambda_n / 2 (1 - \mu \lambda_n)}{1 - \sum_{n=1}^{N_f + N_{h1} + N_{h2}} \mu \lambda_n / 2 (1 - \mu \lambda_n)} \]

Simulations show that the convergence speed of the proposed adaptive IDFE is nearly the same as the conventional adaptive DFE.

The interpolation filter results in higher input correlation and less filter order than DFE, thus, these effects on convergence speed will cancel out each other.
Simulations

A) Channel CSA No. 3
B) Different loop topologies
C) Convergence verification
D) Symbol error rate (SER) vs. SNR
# Parameters for simulations

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modulation</td>
<td>8-PAM</td>
</tr>
<tr>
<td>Tx/Rx filter</td>
<td>3-dB@775 KHz (6-th Butterworth)</td>
</tr>
<tr>
<td>Hybrid</td>
<td>Differential</td>
</tr>
<tr>
<td>Transformer</td>
<td>3 mH (primary inductance)</td>
</tr>
<tr>
<td>Impedance</td>
<td>135 Ohms</td>
</tr>
<tr>
<td>Cutting point</td>
<td>0</td>
</tr>
<tr>
<td>Interpolation factor (M)</td>
<td>8</td>
</tr>
<tr>
<td>Interpolation span (S)</td>
<td>1</td>
</tr>
<tr>
<td>Interpolation filter</td>
<td>Linear</td>
</tr>
<tr>
<td>Sampling rate</td>
<td>775 KHz</td>
</tr>
</tbody>
</table>

(Note: Because of joint equalization by FFF and FBF, interpolation filter design is not critical for all CSA loops)
A) Optimal DFE response – CSA No. 3

(M=8, S=1, linear interpolation, Δ=16, SNR_{Rx}=30 dB, SNR_{Eq}=20.4 dB for both)
B) SNR comparison: different loop topologies

(a) Received SNR=40.0 (dB)

(b) Received SNR=20.0 (dB)

(M=8, S=1, linear interpolation, Δ=21)
C) Convergence – CSA No. 6

Adaptive IDFE (Learning Curve)
Adaptive DFE (Learning Curve)
Adaptive IDFE, $J_\infty = -21.5$, $\psi = 4.1\%$
Adaptive DFE, $J_\infty = -21.6$, $\psi = 6.1\%$
Adaptive IDFE, $J_{\text{min}} = -21.7$
Adaptive DFE, $J_{\text{min}} = -21.9$

(LMS, $\mu=0.000638$, Training: 500,000, $\Delta=21$)
D) BER comparison
– CSA No. 6

Symbol Error Rate vs. SNR (dB)

(LMS, \(\mu=0.000638\), Training: 500,000, \(\Delta=21\))
Conclusions

A set of low-complexity algorithms based on filter interpolation have been developed for wireline applications (echo cancellation, precoding, and equalization).

Corresponding computational complexities of proposed algorithms are 43%, and 24% of the conventional methods, respectively.

Proposed algorithms achieve nearly the same performance as conventional algorithms.

Proposed algorithms inherits all the numerical stability advantages of the adaptive FIR algorithms.

Convergence behaviors of proposed adaptive interpolated echo canceller and DFE are analyzed and verified.
Conclusions

- Optimal interpolation filters are derived with a least-squares method.
- Extensive simulations with versatile channels and noise environments demonstrate the robustness and effectiveness of proposed interpolated algorithms.