An Adaptive All-Pass Filter for Time-Varying Delay Estimation

Beth Jelfs, Shuai Sun, Kamran Ghorbani, and **Christopher Gilliam**

School of Engineering, RMIT University, Australia

13th May 2022

Outline

Delay Estimation

Delay Between 2 or More Spatially Separated Sensors

Communications
Delay between mobile and base stations gives location

Sonar
Delay between sensors represents direction of arrival

Radar
Delay receiving reflection of transmitted pulse gives range

Biology
Delay between sensors represents conduction velocity

Wide Range of Different Applications

Time Varying Delay Estimation

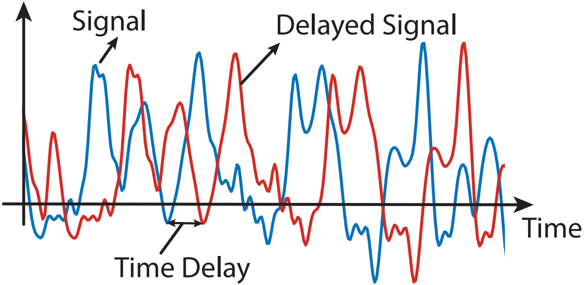
The Problem

At sample time n:

Sensor 1 receives the signal $\longrightarrow$ $x(n)$

Sensor 2 receives a delayed version $\longrightarrow$ $x(n - \tau(n))$

$\looparrowright$ **Need to estimate the time varying delay** $\tau(n)$



Time Varying Delay Estimation

The Problem

At sample time n:

$$\text{Sensor 1 receives the signal} \longrightarrow x(n)$$

$$\text{Sensor 2 receives a delayed version} \longrightarrow x(n - \tau(n))$$

↬ **Need to estimate the time varying delay** $\tau(n)$

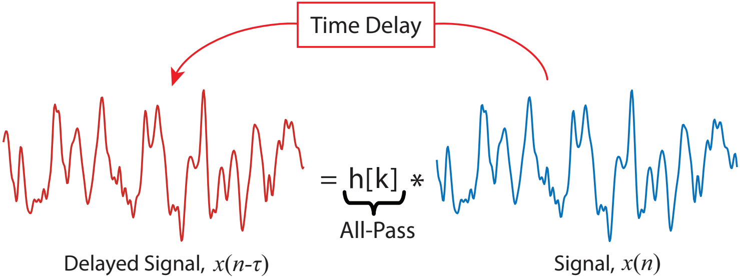
Our Approach:

Normalised Adaptive All-Pass (NAAP) Filter:

- ↬ Versatile and accurate
- ↬ Capable of tracking varying time delays
- ↬ Real time operation

All-Pass Filter Framework - Concept 1

Constant delay $\tau \implies$ Filtering signal with All-Pass Filter h



Shifting in Frequency:

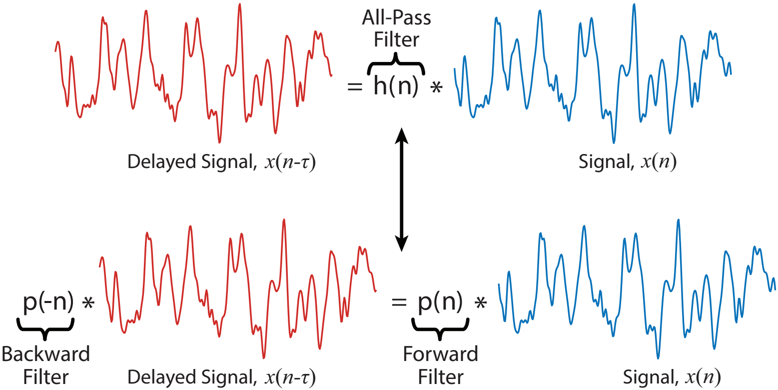
$$\underbrace{X_d(\omega) = X(\omega)\,\mathrm{e}^{-j\tau\omega}}_{=\ \text{Filtering Operation}} \xrightarrow{\text{Define Filter}} \underbrace{H(\omega) = \mathrm{e}^{-j\tau\omega}}_{=\ \text{All-Pass}}$$

All-Pass Filter Framework - Concept 2

Any all-pass filter has a rational structure $\implies H(\omega) = \dfrac{P\left(\mathrm{e}^{j\omega}\right)}{P\left(\mathrm{e}^{-j\omega}\right)}$

All-Pass Filter Framework - Concept 2

Any all-pass filter has a rational structure $\Longrightarrow H(\omega) = \dfrac{P\left(e^{j\omega}\right)}{P\left(e^{-j\omega}\right)}$



All-Pass Filter Framework - Concept 3

Assuming p is an FIR filter of finite support $k \in [0, K]$:

$$p(k) = \begin{cases} a_k, & 0 \leq k \leq K \\ 0, & \text{otherwise,} \end{cases}$$

All-Pass Filter Framework - Concept 3

Assuming p is an FIR filter of finite support $k \in [0, K]$:

$$p(k) = \begin{cases} a_k, & 0 \leq k \leq K \\ 0, & \text{otherwise,} \end{cases}$$

Rewrite:

$$p(-n) * x(n - \tau) = p(n) * x(n)$$

as:

$$\sum_{k=0}^{K} a_k x(n + k - \tau) = \sum_{k=0}^{K} a_k x(n - k)$$

All-Pass Filter Framework - Concept 3

Assuming p is an FIR filter of finite support $k \in [0, K]$:

$$p(k) = \begin{cases} a_k, & 0 \leq k \leq K \\ 0, & \text{otherwise,} \end{cases}$$

Rewrite:

$$p(-n) * x(n - \tau) = p(n) * x(n)$$

as:

$$\sum_{k=0}^{K} a_k x(n + k - \tau) = \sum_{k=0}^{K} a_k x(n - k)$$

Delay estimation solution:

↬ Estimate coefficients a_k

↬ Determine delay: $\hat{\tau} = 2\frac{\sum_k k a_k}{\sum_k a_k}$

Proposed Linear Predictors

- At sample time n, sensor 1 receives $x(n)$ and sensor 2 receives $x(n-\tau)$

Proposed Linear Predictors

- At sample time n, sensor 1 receives $x(n)$ and sensor 2 receives $x(n-\tau)$

Equivalent to setting $a_0 = 1$ and rewriting

$$\sum_{k=0}^{K} a_k x(n+k-\tau) = \sum_{k=0}^{K} a_k x(n-k)$$

as

$$x(n-\tau) - x(n) = \sum_{k=1}^{K} a_k x(n-k) - \sum_{k=1}^{K} a_k x(n+k-\tau)$$

Proposed Linear Predictors

- At sample time n, sensor 1 receives $x(n)$ and sensor 2 receives $x(n-\tau)$

Equivalent to setting $a_0 = 1$ and rewriting

$$\sum_{k=0}^{K} a_k x(n+k-\tau) = \sum_{k=0}^{K} a_k x(n-k)$$

as

$$x(n-\tau) - x(n) = \sum_{k=1}^{K} a_k x(n-k) - \sum_{k=1}^{K} a_k x(n+k-\tau)$$

Predicting current samples based on the other sensor samples:

$$x(n) = \mathbf{x}_+^T(n-\tau)\mathbf{a}$$
$$x(n-\tau) = \mathbf{x}_-^T(n)\mathbf{a}$$

- $\mathbf{a} = [a_1, \ldots, a_K]^T$,
- $\mathbf{x}_-(n) = \left[x(n-1), \ldots, x(n-K)\right]^T$
- $\mathbf{x}_+(n-\tau) = \left[x(n+1-\tau), \ldots, x(n+K-\tau)\right]^T$

Deriving the Adaptive All-Pass Filter

Desired Filter Response

Using linear predictors:

$$d(n) = \mathbf{x}_{-}^{T}(n)\mathbf{a} + \eta_1(n) - \mathbf{x}_{+}^{T}(n-\tau)\mathbf{a} - \eta_2(n)$$

where $\eta_1(n)$ and $\eta_2(n)$ are zero mean i.i.d. noise sources with variance σ_η^2.

$$\mathbf{a} \longrightarrow \text{Optimum filter coefficients}$$

Deriving the Adaptive All-Pass Filter

Desired Filter Response

Using linear predictors:

$$d(n) = \mathbf{x}_-^T(n)\mathbf{a} + \eta_1(n) - \mathbf{x}_+^T(n-\tau)\mathbf{a} - \eta_2(n)$$

where $\eta_1(n)$ and $\eta_2(n)$ are zero mean i.i.d. noise sources with variance σ_η^2.

$\mathbf{a} \longrightarrow$ Optimum filter coefficients

Current Filter Output

$$y(n) = \left[\mathbf{x}_-^T(n) - \mathbf{x}_+^T(n-\tau)\right]\mathbf{w}(n)$$

where $\mathbf{w}(n) = [w_1, w_2, \ldots, w_K]^T$.

$\mathbf{w}(n) \longrightarrow$ Current estimate of $\mathbf{a}$

Deriving the Adaptive All-Pass Filter

Desired Filter Response

Using linear predictors:

$$d(n) = \mathbf{x}_{-}^{T}(n)\mathbf{a} + \eta_1(n) - \mathbf{x}_{+}^{T}(n-\tau)\mathbf{a} - \eta_2(n)$$

where $\eta_1(n)$ and $\eta_2(n)$ are zero mean i.i.d. noise sources with variance σ_η^2.

$\mathbf{a} \longrightarrow$ Optimum filter coefficients

Current Filter Output

$$y(n) = \left[\mathbf{x}_{-}^{T}(n) - \mathbf{x}_{+}^{T}(n-\tau)\right]\mathbf{w}(n)$$

where $\mathbf{w}(n) = [w_1, w_2, \ldots, w_K]^T$.

$\mathbf{w}(n) \longrightarrow$ Current estimate of $\mathbf{a}$

Error Term

Desired response minus filter output:

$$e(n) = \left[\mathbf{x}_{-}^{T}(n) - \mathbf{x}_{+}^{T}(n-\tau)\right]\mathbf{a} - \left[\mathbf{x}_{-}^{T}(n) - \mathbf{x}_{+}^{T}(n-\tau)\right]\mathbf{w}(n) + \eta_1(n) - \eta_2(n)$$

Updating Filter Coefficients

Update filter coefficients using steepest-decent:

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \mu \nabla \mathcal{J}(n)\Big|_{w=w(n)}$$

where μ is the learning rate and $\nabla \mathcal{J}(n)$ is the gradient of the cost function:

$$\mathcal{J}(n) = \left|e(n)\right|^2$$

$$\nabla \mathcal{J}(n)\Big|_{w=w(n)} = -2e(n)\left[\mathbf{x}_-^T(n) - \mathbf{x}_+^T(n-\tau)\right]$$

Updating Filter Coefficients

Update filter coefficients using steepest-decent:

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \mu \nabla \mathcal{J}(n)\Big|_{w=w(n)}$$

where μ is the learning rate and $\nabla \mathcal{J}(n)$ is the gradient of the cost function:

$$\mathcal{J}(n) = |e(n)|^2$$

$$\nabla \mathcal{J}(n)\Big|_{w=w(n)} = -2e(n)\left[\mathbf{x}_-^T(n) - \mathbf{x}_+^T(n-\tau)\right]$$

Our Adaptive All-Pass Filter

$$e(n) = \mathbf{r}^T(n)\mathbf{a} - \mathbf{r}^T(n)\mathbf{w}(n) + \eta(n)$$

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu e(n)\mathbf{r}^T(n)$$

where $\mathbf{r}(n) = \mathbf{x}_-(n) - \mathbf{x}_+(n-\tau)$ and $\eta(n) = \eta_1(n) - \eta_2(n)$

Normalised Adaptive All-Pass Filter

Our filter converges in the mean square error if[1]:

$$0 < \mu < \frac{1}{3\mathrm{tr}[\mathbf{R}]}$$

where:

$$\mathrm{tr}[\mathbf{R}] = \left[\mathbf{x}_{-}(n) - \mathbf{x}_{+}(n-\tau)\right]\left[\mathbf{x}_{-}(n) - \mathbf{x}_{+}(n-\tau)\right]^{T}$$

[1] B. Farhang-Boroujeny, '*Adaptive Filters: Theory and Applications*', Wiley, 2013.

Normalised Adaptive All-Pass Filter

Our filter converges in the mean square error if[1]:

$$0 < \mu < \frac{1}{3\mathrm{tr}[\mathbf{R}]}$$

where:

$$\mathrm{tr}[\mathbf{R}] = \left[\mathbf{x}_-(n) - \mathbf{x}_+(n-\tau)\right]\left[\mathbf{x}_-(n) - \mathbf{x}_+(n-\tau)\right]^T$$

Normalised Adaptive All-Pass (NAAP) Filter

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{\rho}{\|\mathbf{x}_-(n) - \mathbf{x}_+(n-\tau)\|_2^2 + \varepsilon} e(n)\mathbf{r}(n),$$

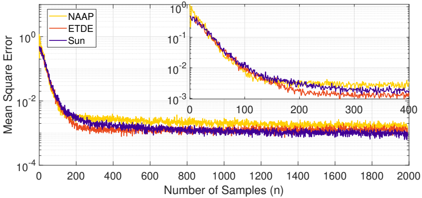
where $0 < \rho < 1/3$ and ε is a small positive regularisation constant.

↬ Delay estimate obtained from $\mathbf{w}(n)$

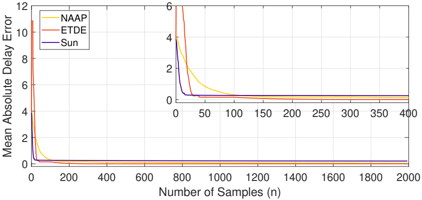
[1] B. Farhang-Boroujeny, '*Adaptive Filters: Theory and Applications*', Wiley, 2013.

Estimating a Constant Delay

Estimating a constant delay $\tau(n) = 5.85$ samples:



Evolution of the Mean Square Error



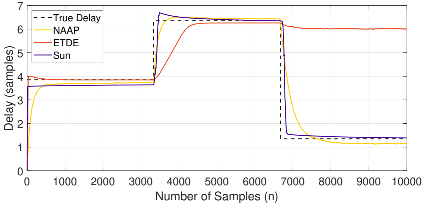
Evolution of the mean absolute delay error

- NAAP ($\rho = 0.08$), ETDE[1] ($\mu = 0.04$) and Sun[2] ($\mu = 0.02$)
- Averages were obtained using 100 realisations of the synthetic signals

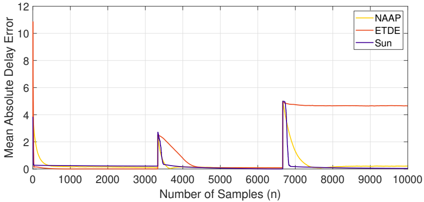
[1] H. So, P. Ching, and Y. Chan, '*A new algorithm for explicit adaptation of time delay*,' IEEE Trans. Signal Process., 1994.
[2] X. Sun and S. Douglas, '*Adaptive time delay estimation with allpass constraints*,' in Proc. Asilomar Conf. Signals, Systems, and Computers, 1999

Tracking a Time Varying Delay

Estimating a piecewise constant delay in noise SNR$= 20$dB:



Average evolution of the delay estimate



Evolution of the mean absolute delay error

- NAAP ($\rho = 0.01$), ETDE[1] ($\mu = 0.02$) and Sun[2] ($\mu = 0.008$)
- Averages were obtained using 100 realisations of the synthetic signals

[1] H. So, P. Ching, and Y. Chan, '*A new algorithm for explicit adaptation of time delay*,' IEEE Trans. Signal Process., 1994.
[2] X. Sun and S. Douglas, '*Adaptive time delay estimation with allpass constraints*,' in Proc. Asilomar Conf. Signals, Systems, and Computers, 1999

Tracking a Time Varying Delay

Average mean absolute delay errors for different SNR values

	Small Step Change				Large Step Change			
SNR (dB)	5	10	20	30	5	10	20	30
NAAP ($\rho = 0.01$)	0.496	0.313	0.153	0.124	0.528	0.337	0.228	0.219
ETDE ($\mu = 0.02$)	0.112	0.074	0.052	0.047	1.805	1.700	1.661	1.663
Sun ($\mu = 0.008$)	0.249	0.235	0.235	0.234	0.243	0.230	0.233	0.233

'Small Step Change' $\longrightarrow$ Changes of $+0.75$ and -1.50 samples

'Large Step Change' $\longrightarrow$ Changes of $+2.50$ and -5.00 samples

↬ NAAP is more robust to small changes in the learning rate

[1] H. So, P. Ching, and Y. Chan, '*A new algorithm for explicit adaptation of time delay*,' IEEE Trans. Signal Process., 1994.
[2] X. Sun and S. Douglas, '*Adaptive time delay estimation with allpass constraints*,' in Proc. Asilomar Conf. Signals, Systems, and Computers, 1999

Conclusions

- All-pass filter framework for time delay estimation
  - All-pass filtering equivalent to a time delay
  - Time delay estimated using the filter coefficients
- Adaptive all-pass filter algorithm
  - Proposed novel linear predictors of current samples
  - Formulated a LMS style algorithm to estimate the filter coefficients
- Evaluated adaptive filter using synthetic data
  - Accurate and capable of tracking varying time delays
  - More versatile estimate than alternative methods

The End

Thank you for listening