Computer-Aided Design of Digital Filters

- The FIR filter design techniques discussed so far can be easily implemented on a computer
- In addition, there are a number of FIR filter design algorithms that rely on some type of optimization techniques that are used to minimize the error between the desired frequency response and that of the computer-generated filter

Computer-Aided Design of Digital Filters

- Basic idea behind the computer-based iterative technique
- Let $H(e^{j\omega})$ denote the frequency response of the digital filter $H(z)$ to be designed approximating the desired frequency response $D(e^{j\omega})$, given as a piecewise linear function of ω, in some sense

Computer-Aided Design of Digital Filters

- Objective - Determine iteratively the coefficients of $H(z)$ so that the difference between between $H(e^{j\omega})$ and $D(e^{j\omega})$ over closed subintervals of $0 \leq \omega \leq \pi$ is minimized
- This difference usually specified as a weighted error function

$$\mathcal{E}(\omega) = W(e^{j\omega})[H(e^{j\omega}) - D(e^{j\omega})]$$

where $W(e^{j\omega})$ is some user-specified weighting function

Computer-Aided Design of Digital Filters

- **Chebyshev** or **minimax criterion** - Minimizes the peak absolute value of the weighted error:

$$\varepsilon = \max_{\omega \in R} \left| \mathcal{E}(\omega) \right|$$

where R is the set of disjoint frequency bands in the range $0 \le \omega \le \pi$, on which $D(e^{j\omega})$ is defined

Design of Equiripple Linear-Phase FIR Filters

- The linear-phase FIR filter obtained by minimizing the peak absolute value of

$$\varepsilon = \max_{\omega \in R} |\mathcal{E}(\omega)|$$

is usually called the **equiripple FIR filter**

- After ε is minimized, the weighted error function $\mathcal{E}(\omega)$ exhibits an equiripple behavior in the frequency range R

Design of Equiripple Linear-Phase FIR Filters

- The general form of frequency response of a causal linear-phase FIR filter of length $2M+1$:

$$H(e^{j\omega}) = e^{-jM\omega} e^{j\beta} \breve{H}(\omega)$$

where the amplitude response $\breve{H}(\omega)$ is a real function of ω

- Weighted error function is given by

$$\mathcal{E}(\omega) = W(\omega)[\breve{H}(\omega) - D(\omega)]$$

where $D(\omega)$ is the desired amplitude response and $W(\omega)$ is a positive weighting function

Design of Equiripple Linear-Phase FIR Filters

- **Parks-McClellan Algorithm** - Based on iteratively adjusting the coefficients of $\breve{H}(\omega)$ until the peak absolute value of $\mathcal{E}(\omega)$ is minimized
- If peak absolute value of $\mathcal{E}(\omega)$ in a band $\omega_a \le \omega \le \omega_b$ is ε_o, then the absolute error satisfies

$$\left|\breve{H}(\omega) - D(\omega)\right| \le \frac{\varepsilon_o}{\left|W(\omega)\right|}, \quad \omega_a \le \omega \le \omega_b$$

Design of Equiripple Linear-Phase FIR Filters

- For filter design,

$$D(\omega)=\begin{cases}1, & \text{in the passband}\\ 0, & \text{in the stopband}\end{cases}$$

- $\breve{H}(\omega)$ is required to satisfy the above desired response with a ripple of $\pm\delta_p$ in the passband and a ripple of δ_s in the stopband

Design of Equiripple Linear-Phase FIR Filters

- Thus, weighting function can be chosen either as

$$W(\omega) = \begin{cases} 1, & \text{in the passband} \\ \delta_p / \delta_s, & \text{in the stopband} \end{cases}$$

or

$$W(\omega) = \begin{cases} \delta_s / \delta_p, & \text{in the passband} \\ 1, & \text{in the stopband} \end{cases}$$

Design of Equiripple Linear-Phase FIR Filters

- **Type 1 FIR Filter** - $\breve{H}(\omega) = \sum_{k=0}^{M} a[k]\cos(\omega k)$

  where

  $$a[0] = h[M],\ a[k] = 2h[M-k],\ 1 \le k \le M$$

- **Type 2 FIR filter** -

  $$\breve{H}(\omega) = \sum_{k=1}^{(2M+1)/2} b[k]\cos\left(\omega(k - \tfrac{1}{2})\right)$$

  where

  $$b[k] = 2h[\tfrac{2M+1}{2} - k],\ 1 \le k \le \tfrac{2M+1}{2}$$

Design of Equiripple Linear-Phase FIR Filters

- **Type 3 FIR Filter** - $\breve{H}(\omega) = \sum_{k=1}^{M} c[k]\sin(\omega k)$

where

$$c[k] = 2h[M-k], \;\; 1 \le k \le M$$

- **Type 4 FIR Filter** -

$$\breve{H}(\omega) = \sum_{k=1}^{(2M+1)/2} d[k]\sin\left(\omega(k-\tfrac{1}{2})\right)$$

where

$$d[k] = 2h[\tfrac{2M+1}{2}-k], \quad 1 \le k \le \tfrac{2M+1}{2}$$

Design of Equiripple Linear-Phase FIR Filters

- Amplitude response for all 4 types of linear-phase FIR filters can be expressed as

$$\breve{H}(\omega) = Q(\omega)A(\omega)$$

where

$$Q(\omega) = \begin{cases} 1, & \text{for Type 1} \\ \cos(\omega/2), & \text{for Type 2} \\ \sin(\omega), & \text{for Type 3} \\ \sin(\omega/2), & \text{for Type 4} \end{cases}$$

Design of Equiripple Linear-Phase FIR Filters

and

$$A(\omega) = \sum_{k=0}^{L} \tilde{a}[k]\cos(\omega k)$$

where

$$\tilde{a}[k] = \begin{cases} a[k], & \text{for Type 1} \\ \tilde{b}[k], & \text{for Type 2} \\ \tilde{c}[k], & \text{for Type 3} \\ \tilde{d}[k], & \text{for Type 4} \end{cases}$$

Design of Equiripple Linear-Phase FIR Filters

with

$$L = \begin{cases} M, & \text{for Type 1} \\ \frac{2M-1}{2}, & \text{for Type 2} \\ M-1, & \text{for Type 3} \\ \frac{2M-1}{2}, & \text{for Type 4} \end{cases}$$

$\tilde{b}[k]$, $\tilde{c}[k]$, and $\tilde{d}[k]$, are related to $b[k]$, $c[k]$, and $d[k]$, respectively

Design of Equiripple Linear-Phase FIR Filters

- Modified form of weighted error function

$$\mathcal{E}(\omega) = W(\omega)[Q(\omega)A(\omega) - D(\omega)]$$
$$= W(\omega)Q(\omega)[A(\omega) - \frac{D(\omega)}{Q(\omega)}]$$
$$= \tilde{W}(\omega)[A(\omega) - \tilde{D}(\omega)]$$

where we have used the notation

$$\tilde{W}(\omega) = W(\omega)Q(\omega)$$

$$\tilde{D}(\omega) = D(\omega)/Q(\omega)$$

Design of Equiripple Linear-Phase FIR Filters

- Optimization Problem - Determine $\tilde{a}[k]$ which minimize the peak absolute value ε of

$$\mathcal{E}(\omega) = \tilde{W}(\omega)[\sum_{k=0}^{L} \tilde{a}[k]\cos(\omega k) - \tilde{D}(\omega)]$$

over the specified frequency bands $\omega \in R$

- After $\tilde{a}[k]$ has been determined, corresponding coefficients of the original $A(\omega)$ are computed from which $h[n]$ are determined

Design of Equiripple Linear-Phase FIR Filters

- Alternation Theorem - $A(\omega)$ is the best unique approximation of $\tilde{D}(\omega)$ obtained by minimizing peak absolute value ε of

$$\mathcal{E}(\omega) = W(\omega)[Q(\omega)A(\omega) - D(\omega)]$$

if and only if there exist at least L+2 extremal frequencies, $\{\omega_i\}$, $0 \le i \le L+1$,

in a closed subset R of the frequency range $0 \le \omega \le \pi$ such that $\omega_0 < \omega_1 < \cdots < \omega_L < \omega_{L+1}$

and $\mathcal{E}(\omega_i) = -\mathcal{E}(\omega_{i+1})$, $|\mathcal{E}(\omega_i)| = \varepsilon$ for all i

Design of Equiripple Linear-Phase FIR Filters

- Consider a Type 1 FIR filter with an amplitude response $A(\omega)$ whose approximation error $\mathcal{E}(\omega)$ satisfies the Alternation Theorem
- Peaks of $\mathcal{E}(\omega)$ are at $\omega = \omega_i,\ 0 \le i \le L+1$ where $d\mathcal{E}(\omega)/d\omega = 0$
- Since in the passband and stopband, $\tilde{W}(\omega)$ and $\tilde{D}(\omega)$ are piecewise constant,

$$\frac{d\mathcal{E}(\omega)}{d\omega} = \frac{dA(\omega)}{d\omega} = 0 \text{ at } \omega = \omega_i$$

Design of Equiripple Linear-Phase FIR Filters

- Using $\cos(\omega k) = T_k(\cos\omega)$, where $T_k(x)$ is the k-th order Chebyshev polynomial

$$T_k(x) = \cos(k \cos^{-1} x)$$

- $A(\omega)$ can be expressed as

$$A(\omega) = \sum_{k=0}^{L} \alpha[k](\cos\omega)^k$$

which is an Lth-order polynomial in $\cos\omega$

- Hence, $A(\omega)$ can have at most $L-1$ local minima and maxima inside specified passband and stopband

Design of Equiripple Linear-Phase FIR Filters

- At bandedges, $\omega = \omega_p$ and $\omega = \omega_s$, $|\mathcal{E}(\omega)|$ is a maximum, and hence $A(\omega)$ has extrema at these points
- $A(\omega)$ can have extrema at $\omega = 0$ and $\omega = \pi$
- Therefore, there are at most $L+3$ extremal frequencies of $\mathcal{E}(\omega)$
- For linear-phase FIR filters with K specified bandedges, there can be at most $L+K+1$ extremal frequencies

Design of Equiripple Linear-Phase FIR Filters

- The set of equations

$$\tilde{W}(\omega_i)[A(\omega_i)-\tilde{D}(\omega_i)]=(-1)^i\varepsilon,\ 0\le i\le L+1$$

is written in a matrix form

$$\begin{bmatrix} 1 & \cos(\omega_0) & \cdots & \cos(L\omega_0) & -1/\tilde{W}(\omega_0) \\ 1 & \cos(\omega_1) & \cdots & \cos(L\omega_1) & 1/\tilde{W}(\omega_1) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & \cos(\omega_L) & \cdots & \cos(L\omega_L) & (-1)^{L-1}/\tilde{W}(\omega_L) \\ 1 & \cos(\omega_{L+1}) & \cdots & \cos(L\omega_{L+1}) & (-1)^{L}/\tilde{W}(\omega_{L+1}) \end{bmatrix} \begin{bmatrix} \tilde{a}[0] \\ \tilde{a}[1] \\ \vdots \\ \tilde{a}[L] \\ \varepsilon \end{bmatrix} = \begin{bmatrix} \tilde{D}(\omega_0) \\ \tilde{D}(\omega_1) \\ \vdots \\ \tilde{D}(\omega_L) \\ \tilde{D}(\omega_{L+1}) \end{bmatrix}$$

Design of Equiripple Linear-Phase FIR Filters

- The matrix equation can be solved for the unknowns $\tilde{a}[i]$ and ε if the locations of the $L+2$ extremal frequencies are known a priori
- The **Remez exchange algorithm** is used to determine the locations of the extremal frequencies

Remez Exchange Algorithm

- Step 1: A set of initial values of extremal frequencies $\{\omega_i\}, 0 \le i \le L+1$ are either chosen or are available from completion of previous stage
- Step 2: Value of ε is computed using

$$\varepsilon = \frac{c_0 \tilde{D}(\omega_0) + c_1 \tilde{D}(\omega_1) + \cdots + c_{L+1} \tilde{D}(\omega_{L+1})}{\frac{c_0}{\tilde{W}(\omega_0)} - \frac{c_1}{\tilde{W}(\omega_1)} + \cdots + \frac{(-1)^{L+1} c_{L+1}}{\tilde{W}(\omega_{L+1})}}$$

where

$$c_n = \prod_{\substack{i=0 \\ i \ne n}}^{L+1} \frac{1}{\cos(\omega_n) - \cos(\omega_i)}$$

Remez Exchange Algorithm

- Step 3: Values of $A(\omega)$ at $\omega = \omega_i$ are then computed using

$$A(\omega_i) = \frac{(-1)^i \varepsilon}{\tilde{W}(\omega_i)} + \tilde{D}(\omega_i), \quad 0 \le i \le L+1$$

Remez Exchange Algorithm

- Step 4: One of the L+2 extremal frequencies selected in Step 1 is discarded and the polynomial $A(\omega)$ is determined by interpolating the values of $A(\omega)$ at the remaining L+1 extremal frequencies using the Lagrange interpolation formula

Remez Exchange Algorithm

- For example, if ω_{L+1} is discarded, then $A(\omega)$ is given by

$$A(\omega) = \sum_{i=0}^{L} A(\omega_i) P_i(\cos\omega)$$

where

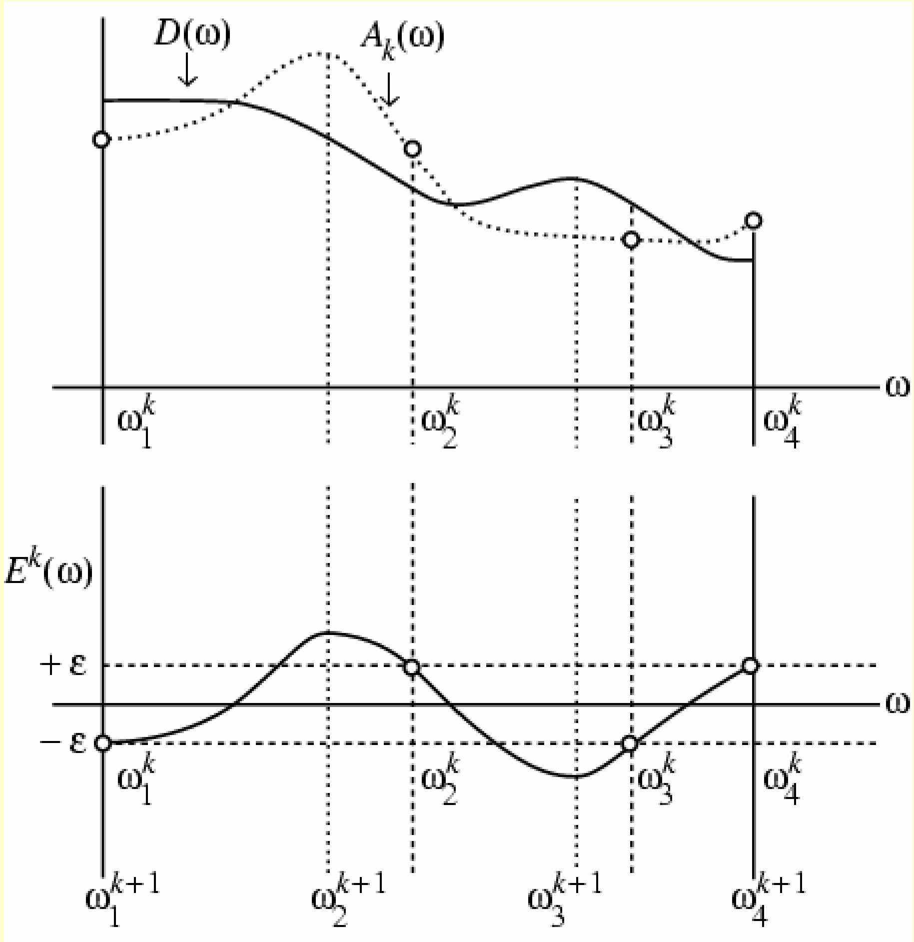
$$P_i(\cos\omega) = \prod_{\substack{\ell=0, \\ \ell \neq i}}^{L} \left(\frac{\cos\omega - \cos\omega_\ell}{\cos\omega_i - \cos\omega_\ell} \right)$$

Remez Exchange Algorithm

- Step 5: The new error function
$$\mathcal{E}(\omega) = \tilde{W}(\omega)[A(\omega) - \tilde{D}(\omega)]$$
is computed at a dense set S ($S \geq L$) of frequencies. In practice $S = 16L$ is adequate. Determine the L+2 new extremal frequencies from the values of $\mathcal{E}(\omega)$ evaluated at the dense set of frequencies.
- Step 6: If the peak values ε of $\mathcal{E}(\omega)$ are equal in magnitude, algorithm has converged. Otherwise, go back to Step 2.

Remez Exchange Algorithm

- Illustration of algorithm



Iteration process is stopped if the difference between the values of the peak absolute errors between two consecutive stages is less than a preset value, e.g., 10^{-6}

Remez Exchange Algorithm

- Example - Approximate the desired function $D(x) = 1.1x^2 - 0.1$ defined for the range $0 \le x \le 2$ by a linear function $a_1 x + a_0$ by minimizing the peak value of the absolute error

$$\max_{x \in [0,2]} \left| 1.1x^2 - 0.1 - a_0 - a_1 x \right|$$

Step 1:

As there are 3 unkowns, a_0 , a_1 , and ε, we need 3 extremal points on x chosen arbirarily as $x_1 = 0,\ x_2 = 0.5,\ x_3 = 1.5$

Remez Exchange Algorithm

- Thus, $D(x_1) = D(0) = -0.1$

$$D(x_2) = D(0.5) = 0.175$$

$$D(x_3) = D(1.5) = 2.375$$

- Next we compute

$$c_n = \prod_{\substack{i=1 \\ i \neq n}}^{3} \frac{1}{x_n - x_i} \quad , \; 1 \leq i \leq 3$$

resulting in $c_1 = 1.333$, $c_2 = -2$ and $c_3 = 0.6667$

Remez Exchange Algorithm

Step 2

- The value of ε is then computed as follows:

$$\varepsilon = \frac{c_1 D(x_1) + c_2 D(x_2) + c_3 D(x_3)}{c_1 - c_2 + c_3}$$

$$= \frac{1.3333 \times (-0.1) + (-2) \times 0.175 + +0.6667 \times 2.375}{1.3333 - (-2) + 0.6667}$$

$$= 0.275$$

Remez Exchange Algorithm

<u>Step 3</u>

- The values of the polynomial $A(x)$ at the initial values of the extremal points on x are then computed as follows:

$$A(x_i) = (-1)^i \quad + D(x_i) = (-1)^i \quad + 1.1(x_i)^2 - 0.1$$

$$1 \le i \le 3$$

Remez Exchange Algorithm

leading to

$$A(x_1) = A(0) = -0.275 - 0.1 = -0.375$$

$$A(x_2) = A(0.5) = 0.275 + 0.175 = 0.45$$

$$A(x_3) = A(1.5) = -0.275 + 2.375 = 2.1$$

Step 4

- Delete the extremal point x_3 (Any one of the other two extremal points could also be deleted).

Remez Exchange Algorithm

- Construct the polynomials $P_1(x)$ and $P_2(x)$:

$$P_1(x) = \left(\frac{x - x_2}{x_1 - x_2}\right) = \left(\frac{x - 0.5}{0 - 0.5}\right) = -2x + 1$$

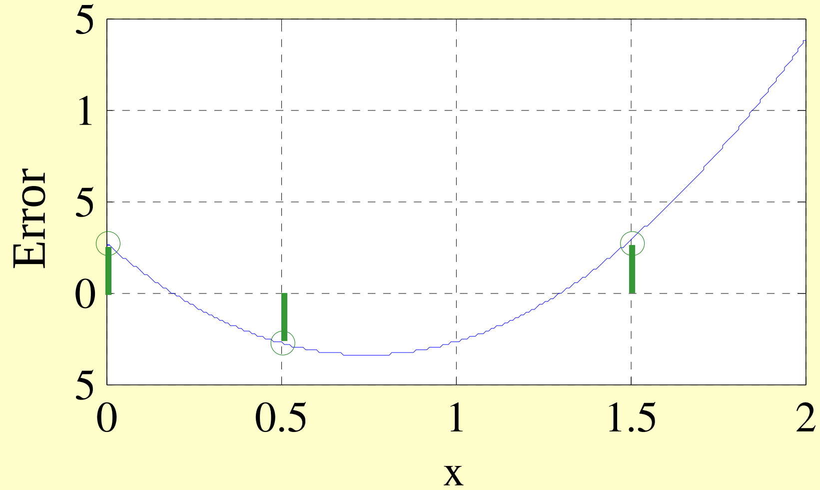
$$P_2(x) = \left(\frac{x - x_1}{x_2 - x_1}\right) = \left(\frac{x - 0}{0.5 - 0}\right) = 2x$$

- The polynomial $A(x)$ is then formed as follows:

$$A(x) = A(x_1)P_1(x) + A(x_2)P_2(x)$$
$$= -0.375(-2x + 1) + 0.45(2x) = 1.65x - 0.375$$

Remez Exchange Algorithm

- Plot of $\mathcal{E}_1(x) = D(x) - A(x)$ $= 1.1x^2 - 1.65x + 0.275$ along with values of error at chosen extremal points shown below



- Note: Errors are equal in magnitude and alternate in sign

Remez Exchange Algorithm

Step 6: As the peak values of $\mathcal{E}_1(x)$ are not equal in magnitude, the next set of extremal points on x are those points where $\mathcal{E}_1(x)$ assumes its maximum absolute values

- These extremal points are given by

$$x_1 = 0, \; x_2 = 0.75, \; x_3 = 2$$

- Thus, $D(x_1) = D(0) = -0.1$

$$D(x_2) = D(0.75) = 0.51875$$

$$D(x_3) = D(2) = 4.3$$

Remez Exchange Algorithm

- We next compute the constants

$$c_n = \prod_{\substack{i=1 \\ i \neq n}}^{3} \frac{1}{x_n - x_i}, \; 1 \leq i \leq 3$$

resulting in

$$c_1 = 0.6667, \;\; c_2 = -1.0667, \;\; c_3 = 0.4$$

- We next go back to Step 2 and repeat the iteration with the new values of x_i

Remez Exchange Algorithm

Step 2

- The new value of ε is computed using

$$\varepsilon = \frac{c_1 D(x_1) + c_2 D(x_2) + c_3 D(x_3)}{c_1 - c_2 + c_3}$$

$$= \frac{0.6667 \times (-0.1) + (-1.0667) \times 0.5188 + 0.4 \times 4.3}{0.6667 + 1.0667 + 0.4}$$

$$= 0.5156$$

Remez Exchange Algorithm

Step 3

- The new values of $A(x)$ at the last set of extremal points on x are given by

$$A(x_1) = A(0) = -0.5156 - 0.1 = -0.6156$$
$$A(x_2) = A(0.75) = 0.5156 + 0.51875 = 1.03435$$
$$A(x_3) = A(2) = -0.5156 + 4.3 = 3.7844$$

Remez Exchange Algorithm

Step 4

- Delete the extremal point x_3 and construct the polynomials $P_1(x)$ and $P_2(x)$:

$$P_1(x) = \left(\frac{x - x_2}{x_1 - x_2}\right) = \left(\frac{x - 0.75}{0 - 0.75}\right) = -\frac{4}{3}x + 1$$

$$P_2(x) = \left(\frac{x - x_1}{x_2 - x_1}\right) = \left(\frac{x - 0}{0.75 - 0}\right) = \frac{4}{3}x$$

Remez Exchange Algorithm

- The polynomial $A(x)$ is now given by

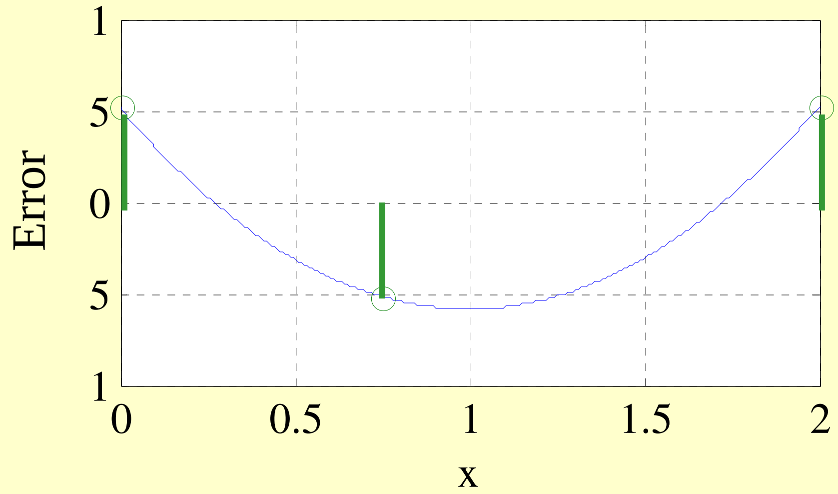
$$A(x) = -0.6156(-\tfrac{4}{3}x + 1) + 1.03435(\tfrac{4}{3}x)$$
$$= 2.2x - 0.6156$$

- The plot of the new error function

$$\mathcal{E}_2(x) = 1.1x^2 - 2.2x + 0.5156$$

and the values of the error at the new set of extremal points are shown in the next slide

Remez Exchange Algorithm



<u>Step 6</u>

- As can be seen the peak value of $\mathcal{E}_2(x)$ are not equal in magnitude

Remez Exchange Algorithm

- We choose the next set of extremal points on x where $\mathcal{E}_2(x)$ assumes its maximum absolute values
- These are now given by $x_1 = 0$, $x_2 = 1$, $x_3 = 2$
- Thus, $D(x_1) = D(0) = -0.1$

$$D(x_2) = D(1) = 1$$

$$D(x_3) = D(2) = 4.3$$

Remez Exchange Algorithm

Step 2

- The new value of ε is then computed using

$$\varepsilon = \frac{c_1 D(x_1) + c_2 D(x_2) + c_3 D(x_3)}{c_1 - c_2 + c_3}$$

$$= \frac{0.5 \times (-0.1) + (-1) \times 1 + 0.5 \times 4.3}{0.5 - (-1) + 0.5}$$

$$= 0.55$$

Remez Exchange Algorithm

<u>Step 3</u>

- The values of $A(x)$ at the new set of extremal points on x are:

$$A(x_1) = A(0) = -0.55 - 0.1 = -0.65$$

$$A(x_2) = A(1) = 0.55 + 1.1 - 0.1 = 1.55$$

$$A(x_3) = A(2) = -0.55 + 4.3 = 3.75$$

Remez Exchange Algorithm

Step 4

- Delete the extremal point x_3 and construct the polynomials $P_1(x)$ and $P_2(x)$:

$$P_1(x) = \left(\frac{x - x_2}{x_1 - x_2}\right) = \left(\frac{x-1}{0-1}\right) = -x+1$$

$$P_2(x) = \left(\frac{x - x_1}{x_2 - x_1}\right) = \left(\frac{x-0}{1-0}\right) = x$$

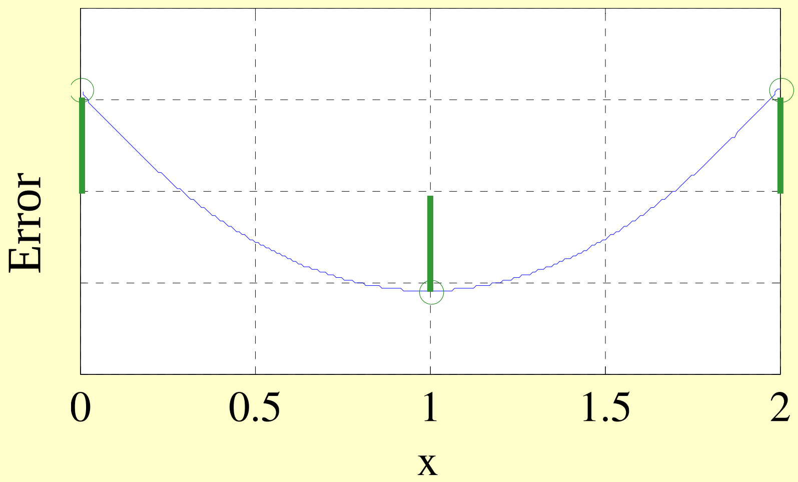
Remez Exchange Algorithm

- The new value of the polynomial $A(x)$ is now given by

$$A(x) = -0.65(-x+1) + 1.55x = 2.2x - 0.65$$

- Plots of $\mathcal{E}_3(x) = 1.1x^2 - 2.2x + 0.55$ along with the values of the error at the new set of extremal points are shown in the next slide

Remez Exchange Algorithm



- Algorithm has converged as ε is also the maximum value of the absolute error