A Unified Approach to the Design of Interpolated and Frequency-Response-Masking FIR Filters

Wu-Sheng Lu
University of Victoria
Victoria, Canada

Takao Hinamoto
Hiroshima University
Higashi-Hiroshima, Japan

May 2016

Outline

- Early Work
- Filter Structures
- Convex-Concave Procedure (CCP)
- Design of Interpolated FIR (IFIR) Filters
- Design of Frequency-Response Masking (FRM) FIR Filters
- Design Examples

1. Early Work

Interpolated FIR (IFIR) Filters

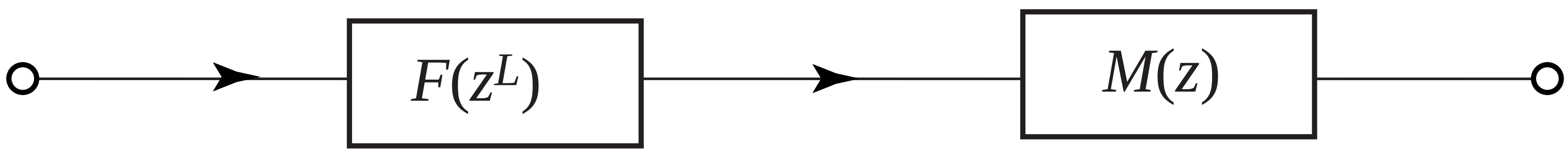
- Y. Neuvo, C. Y. Dong, and S. K. Mitra, 1984.
- T. Saramäki, Y. Neuvo, and S. K. Mitra, 1988.

Frequency-Response Masking (FRM) FIR Filters

- Y. C. Lim, 1986.
- Y. C. Lim and Y. Lian, 1993.
- Many variants since 1990s.

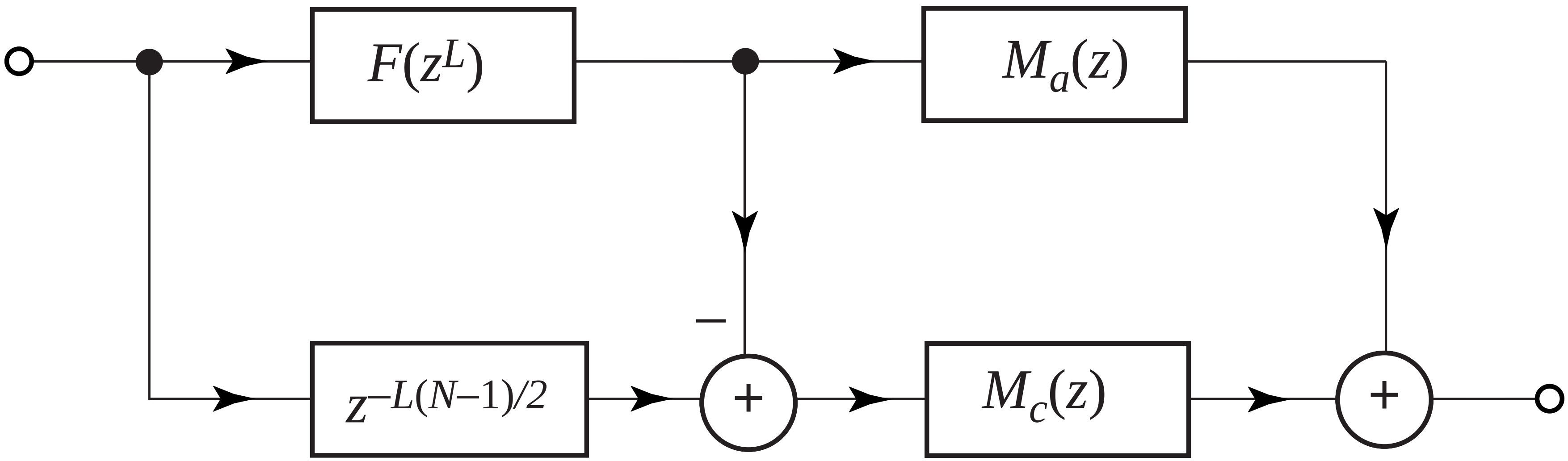
2. Filter Structures

Interpolated FIR (IFIR) Filters



$$H(z) = F(z^L)M(z)$$

Frequency-Response Masking (FRM) FIR Filters



$$H(z) = F(z^L)M_a(z) + \left[z^{-L(N-1)/2} - F(z^L) \right] M_c(z)$$

3. Convex-Concave Procedure (CCP)

CCP refers to a heuristic method to solve a general class of *nonconvex* problems of the form

$$\begin{aligned} &\text{minimize} \quad f(\boldsymbol{x}) - g(\boldsymbol{x}) \\ &\text{subject to:} \quad f_i(\boldsymbol{x}) \le g_i(\boldsymbol{x}) \ \text{ for } \ i = 1, 2, \ldots, m \end{aligned}$$

where $f(\boldsymbol{x})$, $g(\boldsymbol{x})$, $f_i(\boldsymbol{x})$, and $g_i(\boldsymbol{x})$ for $i = 1, 2, \ldots, m$ are convex. The basic CCP algorithm is an iterative procedure including two steps:

(i) Convexify the objective function and constraints by replacing $g(\boldsymbol{x})$ and $g_i(\boldsymbol{x})$, respectively, with their affine approximations

$$\begin{aligned} \hat{g}(\boldsymbol{x}, \boldsymbol{x}_k) &= g(\boldsymbol{x}_k) + \nabla g(\boldsymbol{x}_k)^T (\boldsymbol{x} - \boldsymbol{x}_k) \\ \hat{g}_i(\boldsymbol{x}, \boldsymbol{x}_k) &= g_i(\boldsymbol{x}_k) + \nabla g_i(\boldsymbol{x}_k)^T (\boldsymbol{x} - \boldsymbol{x}_k) \ \text{ for } \ i = 1, 2, \ldots, m \end{aligned}$$

(ii) Solve the convex problem

$$\text{minimize} \quad f(\boldsymbol{x}) - \hat{g}(\boldsymbol{x}, \boldsymbol{x}_k)$$
$$\text{subject to:} \quad f_i(\boldsymbol{x}) \le \hat{g}_i(\boldsymbol{x}, \boldsymbol{x}_k) \ \text{ for } \ i = 1, 2, \ldots, m$$

- **Property 1**

  ◊ If $\boldsymbol{x}_0$ is feasible for the original problem, $\boldsymbol{x}_0$ is also a feasible point for the convexified problem.

  ◊ If $\boldsymbol{x}_{k+1}$ is produced by solving the convexified problem, then $\boldsymbol{x}_{k+1}$ is also feasible for the original problem.

- **Property 2**

  CCP is a descent algorithm, namely, $\{f(\boldsymbol{x}_k), k = 0,1,\ldots\}$ decreases monotonically.

- **Property 3**

  Iterates $\{\boldsymbol{x}_k, k = 0,1,\ldots\}$ converge to a critical point of the original problem.

4. Design of Interpolated FIR (IFIR) Filters

Frequency response of an IFIR filter:

$$H(e^{j\omega}) = F(e^{jL\omega})M(e^{j\omega})$$

Its zero-phase frequency response:

$$H_0(\boldsymbol{x},\omega) = \left[\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega)\right]\left[\boldsymbol{a}_m^T \boldsymbol{t}_m(\omega)\right]$$

where $\boldsymbol{a}_f$ and $\boldsymbol{a}_m$ are coefficient vectors determined by the impulse responses of $F(z)$ and $M(z)$ respectively, and $\boldsymbol{t}_f(\omega)$ and $\boldsymbol{t}_m(\omega)$ are vectors with trigonometric components determined by the filter lengths and types

Let $H_d(\omega)$ be the desired zero-phase response of the IFIR filter, the frequency-weighted minimax design of an IFIR filter amounts to finding $\boldsymbol{a}_f$ and $\boldsymbol{a}_m$ that solve the nonconvex minimax problem

$$\underset{\boldsymbol{a}_f,\, \boldsymbol{a}_m}{\text{minimize}} \;\; \max_{\omega\in\Omega} \; w(\omega)\left|\left[\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega)\right]\left[\boldsymbol{a}_m^T \boldsymbol{t}_m(\omega)\right] - H_d(\omega)\right|$$

where $w(\omega) > 0$ is a frequency-selective weight over $\omega \in \Omega$.

◊ Converting the problem to:

$$\begin{aligned} &\text{minimize} && \delta \\ &\text{subject to:} && \left[\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega)\right]\left[\boldsymbol{a}_m^T \boldsymbol{t}_m(\omega)\right] \le \delta_w + H_d(\omega) \\ & && -\left[\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega)\right]\left[\boldsymbol{a}_m^T \boldsymbol{t}_m(\omega)\right] \le \delta_w - H_d(\omega) \end{aligned}$$

◊ Convexifying the problem by adding $\frac{1}{2}p(\boldsymbol{x},\omega)$ with

$$p(\boldsymbol{x},\omega) = \left[\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega)\right]^2 + \left[\boldsymbol{a}_m^T \boldsymbol{t}_m(\omega)\right]^2$$

hence the constraints become

$$\left[\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega) + \boldsymbol{a}_m^T \boldsymbol{t}_m(\omega)\right]^2 \le p(\boldsymbol{x},\omega) + 2\delta_w + 2H_d(\omega)$$

$$\left[\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega) - \boldsymbol{a}_m^T \boldsymbol{t}_m(\omega)\right]^2 \le p(\boldsymbol{x},\omega) + 2\delta_w - 2H_d(\omega)$$

which fit nicely into a CCP, hence the convexification is done by linearizing $p(\boldsymbol{x},\omega)$ on the right-hand sides of the constraints.

◊ Summarizing, the *k*-th iteration in the CCP solves the convex problem

$$\begin{array}{ll} \text{minimize} & \delta \\ \text{subject to:} & \begin{bmatrix} \eta_i(\boldsymbol{x}, \boldsymbol{x}_k, \omega) & \boldsymbol{t}_i^T(\omega)\boldsymbol{x} \\ \boldsymbol{t}_i^T(\omega)\boldsymbol{x} & 1 \end{bmatrix} \succeq \boldsymbol{0} \quad \text{for } \omega \in \Omega_d,\ i = 0,1 \end{array}$$

where

$$\eta_i = p(\boldsymbol{x}_k, \omega) + \nabla p(\boldsymbol{x}_k, \omega)^T (\boldsymbol{x} - \boldsymbol{x}_k) + 2\delta_w + (-1)^i 2H_d(\omega),$$

$$\boldsymbol{t}_i(\omega) = \begin{bmatrix} \boldsymbol{t}_f(L\omega) \\ (-1)^i \boldsymbol{t}_m(\omega) \end{bmatrix}$$

$$\Omega_d = \{\omega_j,\ j = 1, 2, \ldots, K\} \subseteq \Omega$$

In words, the *k*-th iteration of the design algorithm solves an SDP problem involving a total of 2K 2-by-2 matrices that are required to be positive semidefinite.

5. Design of Frequency-Response Masking (FRM) FIR Filters

Frequency response of an FRM filter:

$$H(e^{j\omega}) = F(e^{jL\omega})M_a(e^{j\omega}) + \left[e^{-jL(N-1)\omega/2} - F(e^{jL\omega})\right]M_c(e^{j\omega})$$

Its zero-phase frequency response:

$$H(\boldsymbol{x},\omega) = \left[\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega)\right]\left[\boldsymbol{a}_a^T \boldsymbol{t}_a(\omega) - \boldsymbol{a}_c^T \boldsymbol{t}_c(\omega)\right] + \boldsymbol{a}_c^T \boldsymbol{t}_c(\omega)$$

where $\boldsymbol{a}_f$, $\boldsymbol{a}_a$ and $\boldsymbol{a}_c$ are coefficient vectors determined by the impulse responses of $F(z)$, $M_a(z)$ and $M_c(z)$, respectively.

Let $H_d(\omega)$ be the desired zero-phase response of the IFIR filter, the frequency-weighted minimax design of an IFIR filter amounts to finding $\boldsymbol{a}_f$ $\boldsymbol{a}_a$, and $\boldsymbol{a}_c$ that solve the nonconvex minimax problem

$$\underset{\boldsymbol{a}_f,\, \boldsymbol{a}_a,\, \boldsymbol{a}_c}{\text{minimize}} \ \max_{\omega \in \Omega} \ w(\omega)\left|H(\boldsymbol{x},\omega) - H_d(\omega)\right|$$

where $w(\omega) > 0$ is a frequency-selective weight over $\omega \in \Omega$.

◊ Converting the problem to:

$$\begin{aligned} &\text{minimize} \quad && \delta \\ &\text{subject to:} \quad && H(\boldsymbol{x},\omega) - \delta_w - H_d(\omega) \le 0, \quad \omega \in \Omega \\ & && -H(\boldsymbol{x},\omega) - \delta_w + H_d(\omega) \le 0, \quad \omega \in \Omega \end{aligned}$$

◊ Convexifying the problem by adding the term

$$p(\boldsymbol{x},\omega) = \left[\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega)\right]^2 + \tfrac{1}{2}\left[\boldsymbol{a}_a^T \boldsymbol{t}_a(\omega)\right]^2 + \tfrac{1}{2}\left[\boldsymbol{a}_c^T \boldsymbol{t}_c(\omega)\right]^2$$

hence the constraints become

$$u(\boldsymbol{x},\omega) \le p(\boldsymbol{x},\omega)$$

$$v(\boldsymbol{x},\omega) \le p(\boldsymbol{x},\omega)$$

where

$$u(\boldsymbol{x},\omega) = p(\boldsymbol{x},\omega) + H(\boldsymbol{x},\omega) - \delta_w - H_d(\omega)$$

$$v(\boldsymbol{x},\omega) = p(\boldsymbol{x},\omega) - H(\boldsymbol{x},\omega) - \delta_w + H_d(\omega)$$

are convex because

$$p(\boldsymbol{x},\omega)+H(\boldsymbol{x},\omega)=\tfrac{1}{2}[\boldsymbol{a}_f^T \;\; \boldsymbol{a}_a^T]\boldsymbol{M}_0\begin{bmatrix}\boldsymbol{a}_f\\ \boldsymbol{a}_a\end{bmatrix}+\tfrac{1}{2}[\boldsymbol{a}_f^T \;\; \boldsymbol{a}_c^T]\boldsymbol{N}_1\begin{bmatrix}\boldsymbol{a}_f\\ \boldsymbol{a}_c\end{bmatrix}+\boldsymbol{a}_c^T\boldsymbol{t}_c$$

$$p(\boldsymbol{x},\omega)-H(\boldsymbol{x},\omega)=\tfrac{1}{2}[\boldsymbol{a}_f^T \;\; \boldsymbol{a}_a^T]\boldsymbol{M}_1\begin{bmatrix}\boldsymbol{a}_f\\ \boldsymbol{a}_a\end{bmatrix}+\tfrac{1}{2}[\boldsymbol{a}_f^T \;\; \boldsymbol{a}_c^T]\boldsymbol{N}_0\begin{bmatrix}\boldsymbol{a}_f\\ \boldsymbol{a}_c\end{bmatrix}-\boldsymbol{a}_c^T\boldsymbol{t}_c$$

with positive semidefinite $\boldsymbol{M}_i=\boldsymbol{m}_i\boldsymbol{m}_i^T$ and $\boldsymbol{N}_i=\boldsymbol{n}_i\boldsymbol{n}_i^T$ where

$$\boldsymbol{m}_i=\begin{bmatrix}\boldsymbol{t}_f(L\omega)\\ (-1)^i\boldsymbol{t}_a(\omega)\end{bmatrix} \text{ and } \boldsymbol{n}_i=\begin{bmatrix}\boldsymbol{t}_f(L\omega)\\ (-1)^i\boldsymbol{t}_c(\omega)\end{bmatrix} \text{ for } i=0,1$$

The above form of constraints fits nicely into a CCP, hence convexification can be done by linearizing $p(\boldsymbol{x},\omega)$ on the right-hand sides of the constraints, i.e.,

$$u(\boldsymbol{x},\omega)\le\tilde{p}(\boldsymbol{x},\boldsymbol{x}_k,\omega) \text{ for } \omega\in\Omega$$

$$v(\boldsymbol{x},\omega)\le\tilde{p}(\boldsymbol{x},\boldsymbol{x}_k,\omega) \text{ for } \omega\in\Omega$$

where $\tilde{p}(\boldsymbol{x},\boldsymbol{x}_k,\omega)=p(\boldsymbol{x}_k,\omega)+\nabla p(\boldsymbol{x}_k,\omega)^T(\boldsymbol{x}-\boldsymbol{x}_k)$ with

$$\nabla p(\boldsymbol{x}_k, \omega) = \begin{bmatrix} 2[\boldsymbol{a}_f^T \boldsymbol{t}_f(L\omega)]\boldsymbol{t}_f(L\omega) \\ [\boldsymbol{a}_a^T \boldsymbol{t}_a(\omega)]\boldsymbol{t}_a(\omega) \\ [\boldsymbol{a}_c^T \boldsymbol{t}_c(\omega)]\boldsymbol{t}_c(\omega) \end{bmatrix}$$

◊ Summarizing, the *k*-th iteration in the CCP solves the convex problem

$$\begin{aligned} &\text{minimize} && \delta \\ &\text{subject to:} && u(\boldsymbol{x}, \omega_j) \le \tilde{p}(\boldsymbol{x}, \boldsymbol{x}_k, \omega_j) \;\; \text{for } j = 1, \ldots, K \\ & && v(\boldsymbol{x}, \omega_j) \le \tilde{p}(\boldsymbol{x}, \boldsymbol{x}_k, \omega_j) \;\; \text{for } j = 1, \ldots, K \end{aligned}$$

where $\{\omega_j,\, j = 1, 2, \ldots, K\} \subset \Omega$.

In words, the *k*-th iteration of the design algorithm solves an SOCP problem minimizing a linear objective subject to a total of 2K quadratic constraints.

6. Design Examples

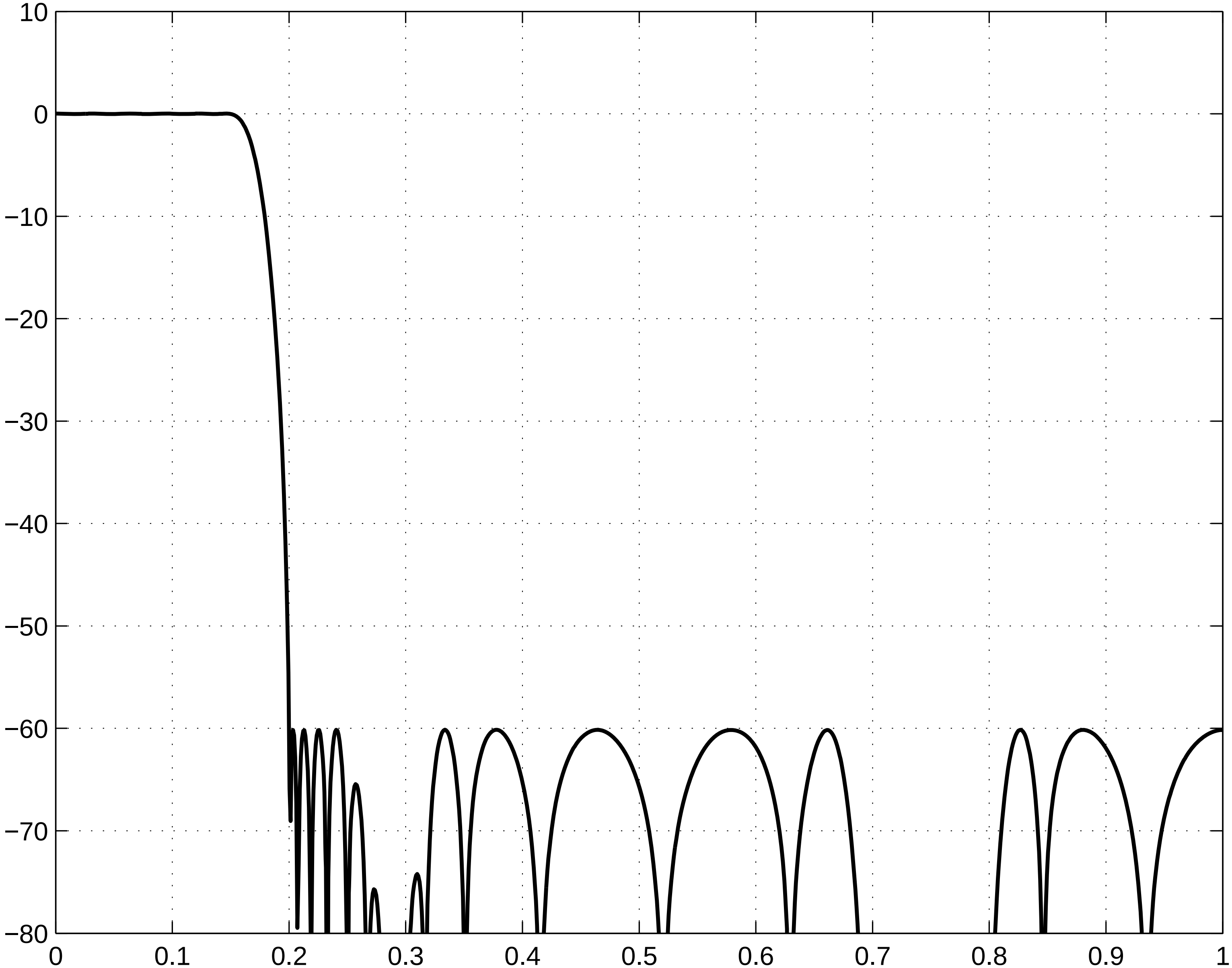
Example 1 The first algorithm was applied to design a lowpass IFIR filter with normalized passband edge $\omega_p = 0.15\pi$, stopband edge $\omega_a = 0.2\pi$. The sparsity factor was set to $L = 4$, and orders of $F(z)$ and $M(z)$ are 31 and 17, respectively. The frequency weight $w(\omega)$ was set to $w(\omega) \equiv 1$ for ω in the passband and $w(\omega) \equiv 2$ for ω in the stopband. An initial was generated by the standard technique proposed in [3]. A total of $K = 1400$ frequency grids were uniformly placed in $[0, \omega_p] \cup [\omega_a, \pi]$ to form the discrete set Ω_d for problem. It took the algorithm 91 iterations to converge to an IFIR filter with $A_p = 0.03171$ dB and $A_a = 60.84$ dB.

The same design problem was addressed as Example 10.29 in [4] using the method described in [Saramäki, 1993]. The method was implemented as function ifir in Signal Processing Toolbox of MATLAB. With

```
[F,M] = ifir(4,'low', [0.15, 0.2], [0.002, 0.001],'advanced')
```

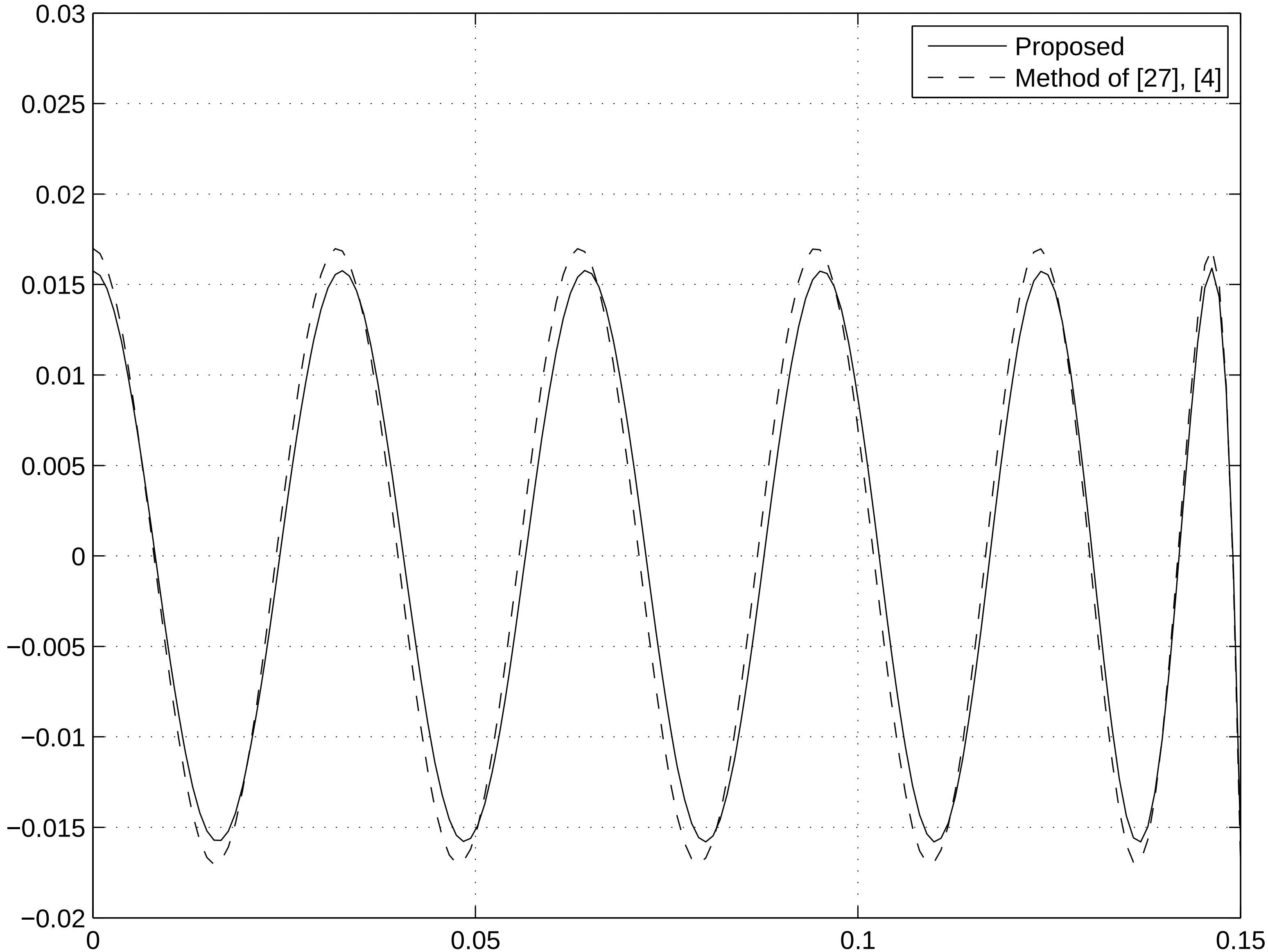
the function returns with optimized impulses of filter $F(z)$ of order 31 and $M(z)$ of order 17 (in Example 10.29 of [4], the order of $M(z)$ was said to be 16, however the order of $M(z)$ produced by the above MATLAB code was actually 17), with $A_p = 0.0340$ dB and $A_a = 60.18$ dB.

Amplitude Response in dB



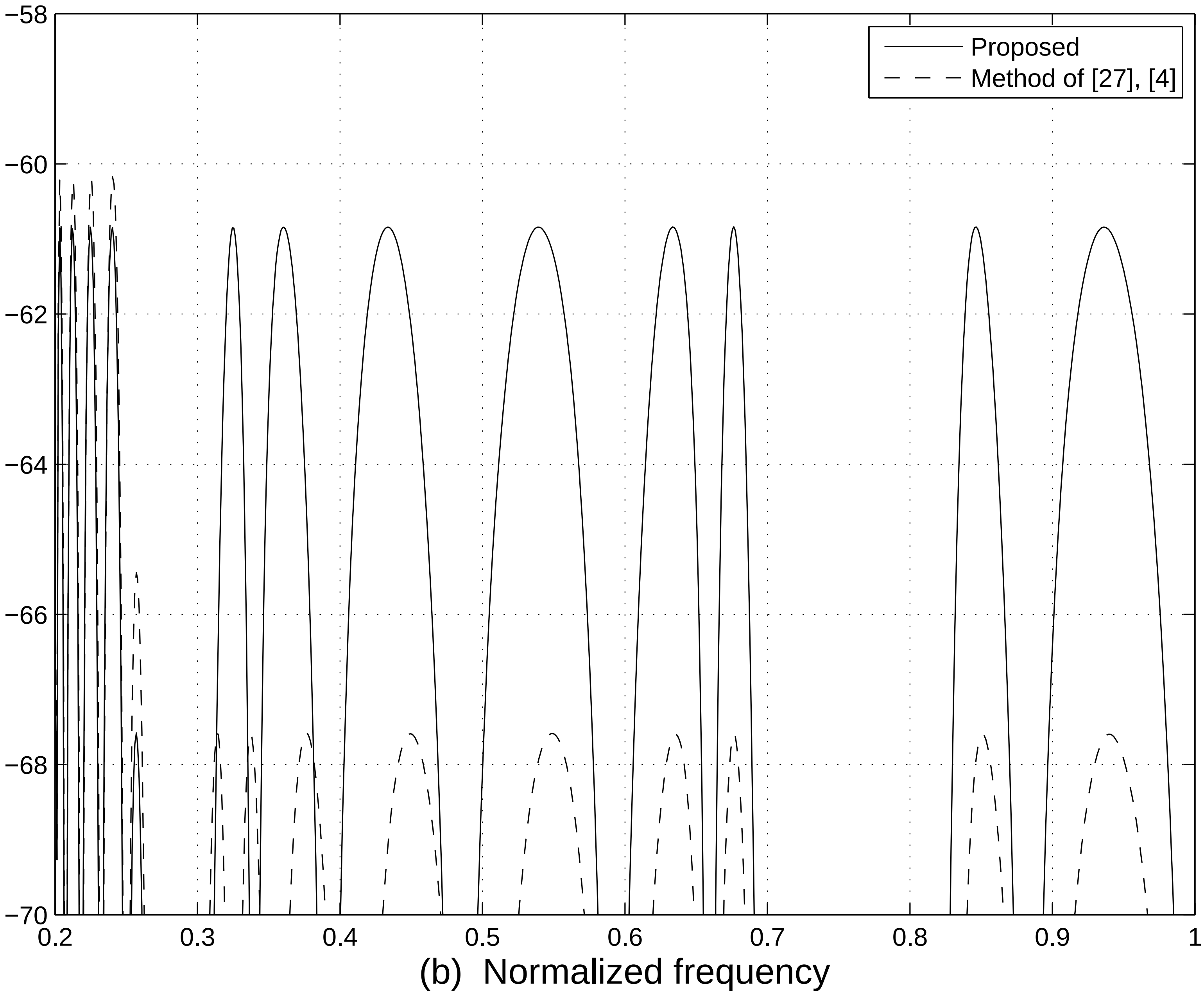
(a)

Passband Ripple (0.03171 dB vs 0.0340 dB)



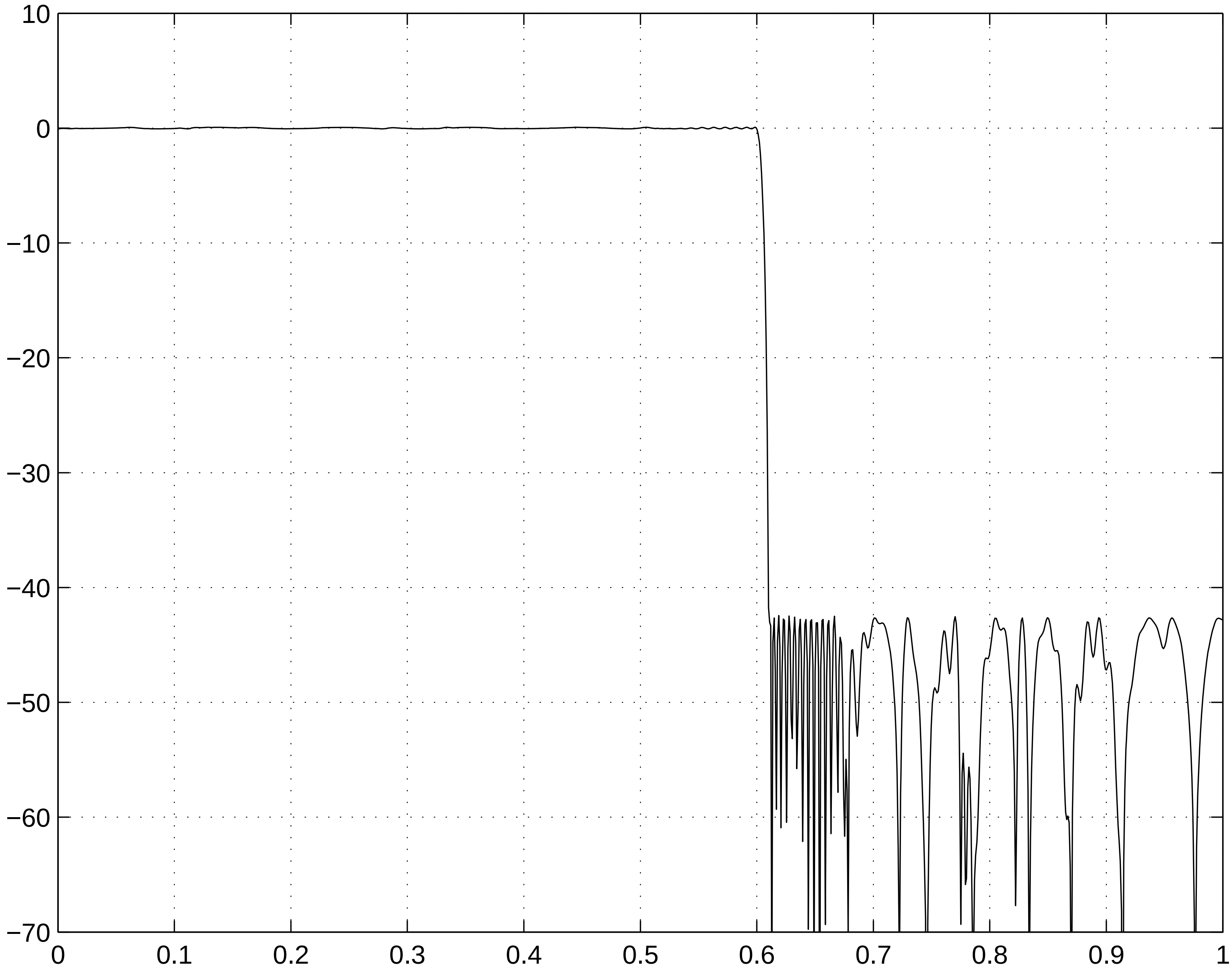
(a) Normalized frequency

Stopband Attenuation (60.84 dB vs 60.18 dB)

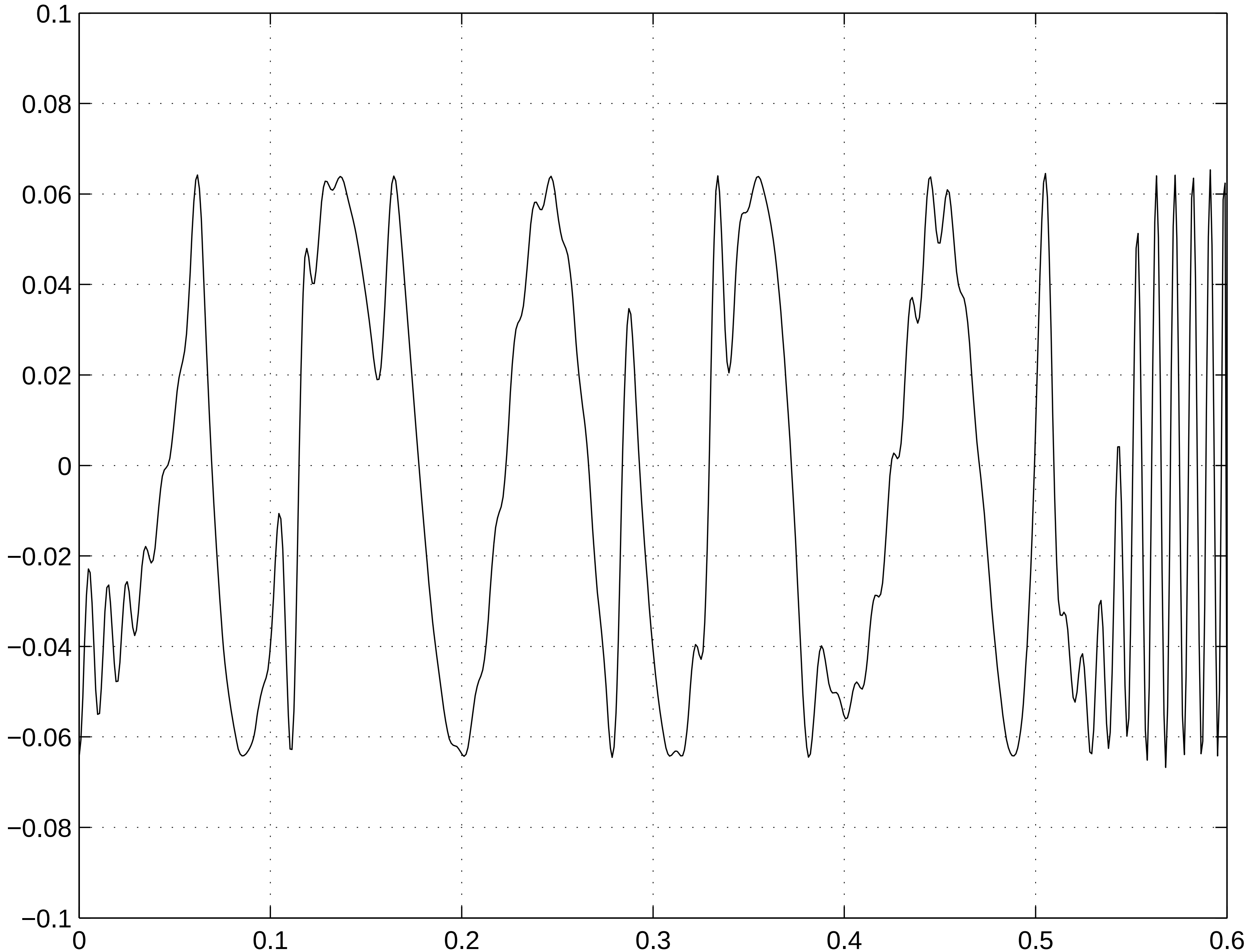


Example 2 The second algorithm was applied to design a lowpass FRM filter with the same design specifications as in the first example in [3] and [10]. The normalized passband and stopband edges were $\omega_p = 0.6\pi$ and $\omega_a = 0.61\pi$. The sparsity factor was set to $L = 9$, and orders of $F(z)$, $M_a(z)$, and $M_c(z)$ were 44, 40, and 32, respectively. A trivial weight $w(\omega) \equiv 1$ was utilized. With $K = 1100$, it took the algorithm 60 iterations to converge to an FRM filter with $A_p = 0.1321$ dB and $A_a = 42.44$ dB, which are favorably compared with those achieved in [3] ($A_p = 0.1792$ dB and $A_a = 40.96$ dB), which has been a benchmark for FRM filters, and those reported in [10] ($A_p = 0.1348$ dB and $A_a = 42.25$ dB).

Amplitude Response in dB



Passband Ripple in dB



Thank you.

Q & A