

## یک الگوریتم تکراری جدید برای طراحی فیلترهای FIR فاز-خطی بر پایه روش LS

کمال محامدپور  
kmpour@kntu.ac.ir

دانشکده برق، دانشگاه صنعتی خواجه نصیرالدین طوسی

مهدی سعادت‌مند طرزجان

811301003@ee.kntu.ac.ir

شده است. الگوریتم پیشنهادی در هر مرحله، ضرایب فیلتر را با بهنگام‌سازی پاسخ فرکانسی مطلوب در روش کمترین مربعات خطا، بر اساس اختلاف بین فیلتر طراحی شده و فیلتر ایده‌آل محاسبه می‌نماید. هدف این الگوریتم، کاهش اختلاف بین فیلتر طراحی شده و فیلتر ایده‌آل به کمتر از یک مقدار مجاز از پیش تعیین شده است. فرض ما بر این است که پاسخ فرکانسی فیلتر ایده‌آل، در کل پهنای باند از قبل تعیین شده است و بنابراین در این روش نیازی به تعریف باندهای گذار که معمولاً در سایر روش‌های طراحی فیلتر مورد استفاده قرار می‌گیرند [۱-۲]، نیست. در حقیقت تعریف باندهای گذار بیشتر برای کاهش اثر گیبز (Gibbs effect) و یا برای امکان استفاده از تخمین چپی‌چف (Chebyshev approximation) در طراحی است. به عنوان مثال، یک فیلتر پایین‌گذر معمولاً برای جداسازی یک سیگنال مطلوب از یک سیگنال نامطلوب یا نویز طراحی می‌شود و طیف فرکانسی سیگنال تعیین کننده باندهای عبور و قطع مطلوب برای فیلتر مذکور است. تقریباً در همه کاربردهای عملی، نمی‌توان باند عبور و قطع را چنان از یکدیگر تفکیک نمود که بین آنها یک باند گذار در نظر گرفته شود. معمولاً طیف سیگنال‌های مطلوب و نامطلوب همپوشانی دارند و تعیین حتی یک نقطه برای تفکیک باندهای عبور و قطع کاری مشکل و دشوار است. چه رسد به تعیین یک باند گذار که تقریباً امری غیرممکن می‌باشد. در اکثر موارد، باندهای

**چکیده:** در این مقاله روشی جدید برای طراحی فیلترهای FIR-فاز خطی با شکل دلخواه ارائه شده است. الگوریتم پیشنهادی، در حقیقت شکل توسعه یافته روش کمترین مربعات خطا (LS) است. در این روش می‌توان علاوه بر تعیین پاسخ فرکانسی فیلتر، دامنه اعوجاجات آن را نیز در تمام (یا بخشی) از باند فرکانسی محدود نمود. برای این منظور، الگوریتم پیشنهادی در آغاز، بر اساس پاسخ فرکانسی ایده‌آل و محدوده اعوجاجات مجاز فیلتر، تابع هزینه‌ای را تعریف می‌کند. سپس طی یک فرآیند تکراری، در هر مرحله، پاسخ فرکانسی مطلوب در روش LS را چنان تنظیم می‌کند که تابع هزینه مذکور کمینه گردد. همچنین در این روش می‌توان همانند سایر روش‌های کلاسیک طراحی فیلترها، بخشی از باند فرکانسی را به عنوان باند عبور تعریف نمود. شبیه‌سازی‌ها بیانگر قابلیت چشم‌گیر الگوریتم پیشنهادی، در تنظیم پاسخ فرکانسی و دامنه اعوجاجات فیلتر مطابق خواسته‌های طراح است.

**واژه‌های کلیدی:** فیلتر FIR با فاز خطی، روش کمترین مربعات خطا (LS)، روش بیشترین شیب

### ۱- مقدمه

در این مقاله یک الگوریتم تکراری برای طراحی فیلترهای FIR-فاز خطی با شکل و محدوده اعوجاجات دلخواه ارائه

فرکانس‌ها، فرکانس‌های طراحی می‌گوییم). LS ضرایب  $k(n)$  را چنان تعیین می‌کند که اختلاف بین  $A(\omega_i)$  و  $H(\omega_i)$  کمینه شود. برای انجام اینکار کافی است مطابق معادلات (۴) و (۵) مشتقات تابع هزینه نشان داده شده در معادله (۱) نسبت به ضرایب  $k[n]$   $n = -L, \dots, -1, 0, 1, \dots, L$  صفر گردد [۱۰].

$$E = \frac{1}{2} \sum_{i=0}^{N-1} \left[ A(\omega_i) - \sum_{n=0}^L k[n] \cos(\omega_i n) \right]^2 \quad (۳)$$

$$\frac{\partial E}{\partial k[m]} = - \sum_{i=0}^{N-1} \left[ A(\omega_i) - \sum_{n=0}^L k[n] \cos(\omega_i n) \right] \cos(\omega_i m) = 0 \quad (۴)$$

$m = 0, 1, \dots, L$

$$\begin{bmatrix} k[0] \\ k[1] \\ \vdots \\ k[L] \end{bmatrix} = P^{-1} \begin{bmatrix} \sum_{i=0}^{N-1} A(\omega_i) \\ \sum_{i=0}^{N-1} A(\omega_i) \cos(\omega_i) \\ \vdots \\ \sum_{i=0}^{N-1} A(\omega_i) \cos(L\omega_i) \end{bmatrix} \quad (۵)$$

که در آن،

$$P = \begin{bmatrix} N & \sum_{i=0}^{N-1} \cos(\omega_i) & \dots & \sum_{i=0}^{N-1} \cos(L\omega_i) \\ \sum_{i=0}^{N-1} \cos(\omega_i) & \sum_{i=0}^{N-1} \cos^2(\omega_i) & \dots & \sum_{i=0}^{N-1} \cos(\omega_i) \cos(L\omega_i) \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=0}^{N-1} \cos(L\omega_i) & \sum_{i=0}^{N-1} \cos(L\omega_i) \cos(\omega_i) & \dots & \sum_{i=0}^{N-1} \cos^2(L\omega_i) \end{bmatrix}$$

اکنون فرض کنید  $N=2T$  زوج باشد و فرکانس‌های طراحی از محدوده  $[-\pi, \pi]$ ، بصورت ذیل انتخاب شده باشند.

$$\omega_i = \pi(i-T)/T, \quad i = 0, 1, \dots, 2T-1 \quad (۶)$$

بنابراین داریم،

$$P = \begin{bmatrix} N & 0 & \dots & 0 \\ 0 & N/2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & N/2 \end{bmatrix} \quad (۷)$$

از معادلات (۲) و (۵)-(۷) می‌توان نتیجه گرفت،

$$h[n] = \sum_{k=0}^T a_k A(\omega_k) \cos(n\omega_k), \quad n = 1, \dots, L \quad (۸)$$

$$a_k = \begin{cases} 1/N & k = 0, T \\ 2/N & \text{Otherwise} \end{cases}$$

معادله (۸) بیانگر معادلات طراحی فیلترهای FIR-فاز خطی بروش LS است به شرط آنکه فرکانس‌های طراحی مطابق معادله (۶) انتخاب شود. ما اصطلاحاً به تابع  $A(\omega)$  تابع مطلوب LS یا بطور خلاصه تابع مطلوب می‌گوییم. در بخش بعدی، با شرح الگوریتم پیشنهادی خواهید دید که می‌توان در یک الگوریتم تکراری با تصحیح  $A(\omega)$  بر اساس یک تابع

گذار برای کاهش و یا حذف نوسانات ناشی از اثر گیبز در نظر گرفته می‌شوند و منشا فیزیکی ندارند. از آنجا که الگوریتم‌های طراحی به پاسخ فرکانسی فیلتر در باند گذار نظارتی ندارند، در نتیجه ممکن است در بعضی مواقع در پاسخ فرکانس فیلتر در این نواحی، نوسانات بزرگی رخ دهد و درست در این موقع است که طراحان تصمیم می‌گیرند، الگوریتم طراحی را به شکلی اصلاح نمایند که چنین نوساناتی دیگر رخ ندهد [۳]. بعلاوه، ممکن است بخواهیم فیلتری با شکل دلخواه مثلاً گوسی طراحی نماییم. اکثر الگوریتم‌های موجود قادر به برآورده نمودن این خواسته به شکل بهینه نیستند [۴]. برای استفاده از این الگوریتم‌ها باید آنها را با معیارهایی چون محدود نمودن اعوجاجات سازگار نمود [۵-۷]. یک راه حل دیگر استفاده از الگوریتم‌های ژنتیکی است [۸] [۹] اما این الگوریتم‌ها از نظر محاسباتی بسیار سنگین هستند. ما با ارائه الگوریتم پیشنهادی در این مقاله سعی در غلبه بر مشکلات فوق داشته‌ایم. شبیه‌سازی‌ها نشان می‌دهد که الگوریتم پیشنهادی از قابلیت انعطاف فراوانی برخوردار است و می‌تواند تقریباً تمام نیازهای ما را در طراحی فیلترهای FIR برآورده سازد. در ادامه در بخش بعدی روش کمترین مربعات خطا در طراحی فیلترهای FIR-فاز خطی معرفی شده است. بخش سوم به شرح الگوریتم پیشنهادی می‌پردازد. در بخش چهارم به بررسی تمهیداتی که برای کاهش حجم محاسباتی الگوریتم پیشنهادی در نظر گرفته شده است، می‌پردازیم و در نهایت بخش‌های پنجم و ششم به شبیه‌سازی‌ها و نتیجه‌گیری اختصاص یافته است.

## ۲- طراحی فیلتر بروش LS

فرض کنید  $h(n)$  یک فیلتر با فاز خطی از درجه  $2L$  باشد بنابراین  $h(n) = h(-n)$  و پاسخ فرکانسی آن مطابق معادله ذیل محاسبه می‌شود.

$$H(\omega) = H(e^{j\omega}) = \sum_{n=0}^L k[n] \cos(\omega n) \quad (۱)$$

$$k[n] = \begin{cases} h[0] & n = 0 \\ 2h[n] & \text{Otherwise} \end{cases} \quad (۲)$$

$H(\omega)$ ، تابعی حقیقی، زوج و پرودییک است. فرض کنید مقدار  $H(\omega)$  در فرکانس‌های  $0 \leq \omega_i \leq 2\pi$   $(i=0, 2, \dots, N-1)$  معلوم و برابر با  $A(\omega_i)$   $(i=0, 2, \dots, N-1)$  باشد (به این

از مقدار مجاز  $\delta(\omega)$  بیشتر شود (این نقاط را انحرافات غیرمجاز می‌نامیم)، تابع مطلوب  $A(\omega)$  در جهت عکس انحراف، تغییر می‌کند. بدیهی است که تغییر مناسب تابع مطلوب در همسایگی‌های انحرافات غیرمجاز می‌تواند سبب بهبود عملکرد الگوریتم بهینه‌سازی شود. در این مقاله الگوریتم جدیدی ارائه شده است که این خواسته را برآورده می‌سازد. این الگوریتم براساس بهینه‌سازی تابع هزینه (۱۰) طراحی شده است.

$$J = \frac{1}{2} \sum_{\omega \in C} (|I(\omega) - H(\omega)| - \delta(\omega))^2 \quad (10)$$

$$C = \{\omega_k \mid k = 0, \dots, T; |I(\omega_k) - H(\omega_k)| > \delta(\omega_k)\} \quad (11)$$

که در آن، مجموعه  $C$  شامل فرکانس‌های مربوط به انحرافات غیرمجاز است. در این معادله، منظور از  $\omega_k$  فرکانس‌های طراحی است که مطابق معادله (۶) تعریف می‌شوند. برای تنظیم بهینه  $A(\omega)$  از روش بیشترین شیب استفاده شده است. به عبارت دیگر، در هر مرحله  $A(\omega)$  مطابق معادله ذیل بهنگام می‌شود.

$$A^k(\omega_l) = \begin{cases} I(\omega_l) & k = 0 \\ A^{k-1}(\omega_l) - \eta^k \frac{\partial J}{\partial A(\omega_l)} & \text{Otherwise} \end{cases} \quad (12)$$

که در آن  $\eta$  سرعت همگرایی الگوریتم است. انتخاب  $\eta$  از اهمیت خاصی برخوردار است. چنانچه  $\eta$  کوچک باشد، زمان محاسباتی الگوریتم افزایش یافته و الگوریتم مستعد به بدام افتادن در مینیمم‌های محلی می‌شود و اگر  $\eta$  خیلی بزرگ باشد، می‌تواند سبب ناپایداری الگوریتم گردد. برای محاسبه  $\frac{\partial J}{\partial A(\omega_l)}$  باید ابتدا  $H(\omega)$  را بدست آورد. از معادله‌های (۱)، (۲) و (۸) نتیجه می‌شود:

$$H(\omega_l) = \sum_{k=0}^T \alpha_k A(\omega_k) \left( 1 + 2 \sum_{n=1}^L \cos(n\omega_k) \cos(n\omega_l) \right) \quad (13)$$

بنابراین داریم،

$$\frac{\partial H(\omega_l)}{\partial A(\omega_k)} = \alpha_k \left[ 1 + 2 \sum_{n=1}^L \cos(n\omega_k) \cos(n\omega_l) \right] \quad (14)$$

و از آنجا داریم،

$$\frac{\partial J}{\partial A(\omega_k)} = \alpha_k \sum_{\omega \in C} S(\omega) \Psi(\omega) \left[ 1 + 2 \sum_{n=1}^L \cos(n\omega_k) \cos(n\omega) \right] \quad (15)$$

برای استفاده از الگوریتم فوق، کاربر باید در ابتدا شکل فیلتر (تابع ایده‌آل،  $I(\omega)$ )، دامنه انحرافات مجاز ( $\delta(\omega)$ )، درجه فیلتر ( $L$ )، تعداد فرکانس‌های طراحی ( $N=2T$ ) و سرعت همگرایی  $\eta$  را مشخص کند. همانطور که در معادله (۱۲) نشان داده شده است، الگوریتم در مرحله اول تابع مطلوب ( $A(\omega)$ )

هزینه، به خصوصیات مورد نظر در طراحی یک فیلتر دست یافت. LS روشی ساده و در عین حال مؤثر است. تاکنون محققین روش‌های متعددی برای توسعه LS و اضافه نمودن ضابطه‌های دیگری به آن ارائه داده‌اند [۵-۷]. اما معمولاً اینکار به قیمت از دست رفتن قابلیت انعطاف‌پذیری LS، در طراحی فیلترهایی با شکل دلخواه تمام شده است. بطورکلی LS را می‌توان به سه روش توسعه داد: (۱) بهبود تابع هزینه، (۲) تنظیم بهینه فرکانس‌های طراحی ( $\omega_i, i=0, 2, \dots, N-1$ ) و تعداد آنها ( $N$ )، و (۳) تنظیم بهینه  $A(\omega)$ . الگوریتم‌های موجود معمولاً براساس دو روش اول طراحی گردیده‌اند، درحالی که انتخاب و تنظیم بهینه  $A(\omega)$  می‌تواند، بسیار کارگشا و مؤثر باشد. الگوریتم پیشنهادی ما در این مقاله بر این اساس طراحی شده است.

### ۳- ELSE: الگوریتم پیشنهادی برای طراحی فیلتر

همانطور که قبلاً اشاره شد، الگوریتم پیشنهادی درحقیقت یک الگوریتم تکراری است که در هر مرحله، با تغییر پاسخ فرکانسی مطلوب در روش LS، سعی در کاهش خطای فیلترهای طراحی شده دارد. فرض کنید،  $I(\omega)$  پاسخ فرکانسی ایده‌آل (بطور خلاصه **تابع ایده‌آل**)،  $A(\omega)$  تابع مطلوب،  $H(\omega)$  پاسخ فرکانسی فیلتر طراحی شده به روش LS با استفاده از تابع مطلوب  $A(\omega)$  (**تابع طراحی شده**) و  $\delta(\omega)$  بیانگر بیشترین انحراف مجاز  $H(\omega)$  از  $I(\omega)$  (**تابع انحرافات مجاز**) باشد. در اینصورت، هدف ما تعیین بهینه  $A(\omega)$  به نحوی است که اختلاف بین  $H(\omega)$  و  $I(\omega)$  حتی‌الامکان در تمام باند فرکانسی کمتر از  $\delta(\omega)$  باشد.

ما در [۱۱] یک الگوریتم تکراری جدید برای برآورده نمودن این هدف ارائه نموده‌ایم. الگوریتم مذکور مطابق معادله (۹)، در مرحله  $k$ ، تابع مطلوب را در جهت کاهش انحرافات غیرمجاز  $H(\omega)$  نسبت به  $I(\omega)$  تغییر می‌دهد.

$$A^k(\omega) = \begin{cases} I(\omega) & k = 0 \\ A^{k-1}(\omega) [1 + \eta \cdot S(\omega) \cdot \max(\Psi(\omega), 0)] & \text{Otherwise} \end{cases} \quad (9)$$

$$S(\omega) = \text{sgn}(I(\omega) - H(\omega))$$

$$\Psi(\omega) = |I(\omega) - H(\omega)| - \delta(\omega)$$

همانطور که در معادله بالا مشاهده می‌کنید در هر نقطه که انحراف تابع طراحی شده  $H(\omega)$  نسبت به تابع ایده‌آل  $I(\omega)$

مختلف در نظر گرفت: ۱) اگر مقدار همبستگی نزدیک یک باشد، دو بردار هم جهت بوده اند. بنابراین می توان با افزایش  $\eta$  بر سرعت همگرایی افزود؛ ۲) اگر مقدار همبستگی نزدیک منهای یک باشد، دو بردار در خلاف جهت یکدیگرند. بنابراین برای کاهش نوسانات، باید  $\eta$  را کاهش داد؛ و ۳) اگر بین دو بردار همبستگی وجود نداشته باشد، مقدار  $\eta$  مناسب است و نیازی به تغییر آن نیست. در نتیجه، مقدار  $\eta$  در هر مرحله مطابق ذیل بهنگام می گردد:

$$\eta^k = \eta^{k-1}(1+c \cdot r^k), \quad c = \begin{cases} 0.25 & r^k \geq 0 \\ 0.50 & \text{Otherwise} \end{cases} \quad (17)$$

$$r^k = \begin{cases} 0 & k = 1 \\ (\Delta \bar{A}^k \cdot \Delta \bar{A}^{k-1}) / (\|\Delta \bar{A}^k\| \|\Delta \bar{A}^{k-1}\|) & \text{Otherwise} \end{cases} \quad (18)$$

که در آن  $r^k$  زاویه بین بردار جابجایی فعلی و بردار جابجایی مرحله قبل و  $\eta^0$  سرعت همگرایی اولیه است. برای  $\eta^0$  می توان مقداری چون ۰/۱ انتخاب نمود (در بخش بعدی در اینباره بیشتر صحبت می کنیم).

#### ۵- تخمین مرتبه فیلتر

همانطور که قبلا ذکر گردید، مرتبه فیلتر به شکل تابع ایده آل و تابع انحراف مجاز بستگی دارد. کاربرد برای تخمین مرتبه فیلتر ناچار است از روش سعی و خطا استفاده نماید که چندان قابل قبول نیست. ما برای تخمین مرتبه فیلتر از یک الگوریتم افزایشی-کاهشی استفاده نموده ایم. در فاز افزایشی، مرتبه فیلتر مرحله به مرحله با گام  $\Delta_i$  افزایش می یابد تا یک شرط خاتمه برآورده شود. فرض کنید  $L_i$  مرتبه نهایی فیلتر در این فاز باشد. سپس در فاز کاهشی، مرتبه فیلتر ( $L_d$ ) از  $L_i$  یک واحد، یک واحد، کاهش می یابد تا اینکه شرط خاتمه مذکور مجدداً نقص شود. مرتبه فیلتر قبل از نقص شرط خاتمه، تخمینی مناسب از مرتبه بهینه می باشد. در آغاز هر مرحله از فاز کاهشی، پارامترهای سرعت همگرایی و تابع مطلوب، با مقادیر نهایی این پارامترها در مرحله قبل، مقدار دهی اولیه می شوند. شرط خاتمه مذکور چنین تعریف می شود که باید انحرافات تابع طراحی شده نسبت به تابع ایده آل در تمام فرکانس های طراحی از یک حد آستانه (که آن را آستانه بیشترین خطا نامیده با  $Tr_{MxErr}$  نمایش می دهیم) کمتر باشد. طی شبیه سازی های انجام شده برای  $\Delta_i$  و  $Tr_{MxErr}$  (به صورت تجربی) به ترتیب مقادیر ۵۰ و ۰/۰۱ بدست آمد.

را با تابع ایده آل مقداردهی اولیه می کند و سپس در هر مرحله با استفاده از معادلات (۱۲) و (۱۵)،  $A(\omega)$  را (در فرکانس های طراحی) بهنگام می کند. این فرآیند تا آنجا ادامه می یابد که دیگر تغییر قابل توجهی در  $J$  رخ ندهد یا اینکه تعداد تکرارهای الگوریتم از یک تعداد معین فراتر رود. تعداد فرکانس های طراحی عاملی مهم در کیفیت نهایی فیلتر طراحی شده است. مشاهدات تجربی نشان داده که معمولاً مقادیری بین ۱۰۰ تا ۵۰۰ برای  $N$  تمام خواسته های ما را برآورده می سازند.

همانطور که مشاهده می کنید، در الگوریتم پیشنهادی نیازی به تعریف باندهای عبور نیست، اما چنانچه بخواهیم در طراحی باندهای عبور را نیز در نظر بگیریم، کافی است مقدار تابع انحراف مجاز را در باندهای عبور به اندازه کافی بزرگ انتخاب کنیم. بدیهی است که این تعریف به مراتب از تعریف باند عبور در سایر روش های طراحی متداول قابل اعتمادتر است، زیرا علیرغم تعریف باند عبور، بازهم بر دامنه انحرافات فیلتر در باند عبور بدقت نظارت می شود. اگر برای تابع انحراف مجاز در کل پهنای باند مقادیر بزرگی انتخاب شود، الگوریتم در همان مرحله اول همگرا شده و خاتمه می یابد. به عبارت دیگر، الگوریتم پیشنهادی که Extended LS (ELSE) نامیده می شود، در این حالت به روش LS ساده شده است. مقدار پارامترهای  $\eta$  و  $L$  کاملاً به شکل فیلتر و دامنه انحرافات مجاز آن وابسته هستند. تعیین مقدار این دو پارامتر توسط کاربر چندان قابل قبول و بهینه نیست. در ادامه مقاله روشی برای تعیین خودکار و بهینه مقادیر این دو پارامتر، شرح داده شده است.

#### ۴- تنظیم سرعت همگرایی

برای تنظیم سرعت همگرایی الگوریتم پیشنهادی، از روشی که در [۱۲] و [۱۳] برای تنظیم خودکار سرعت آموزش شبکه های پیش رو ارائه گردیده، استفاده شده است. در این شیوه به تغییرات متوالی بردارهای تصحیح توجه می شود. از معادله (۱۲) نتیجه می شود:

$$\Delta A^k(\omega_t) = A^k(\omega_t) - A^{k-1}(\omega_t) = -\eta^k \frac{\partial J}{\partial A(\omega_t)} \quad (16)$$

$$k > 0, t = 0, 1, \dots, T$$

که  $\Delta \bar{A}^k = [\Delta A^k(\omega_1), \Delta A^k(\omega_2), \dots, \Delta A^k(\omega_N)]$  بردار تصحیحات تابع مطلوب در مرحله  $k$  است. اکنون بسته به مقدار همبستگی دو بردار  $\Delta \bar{A}^k$  و  $\Delta \bar{A}^{k-1}$  می توان سه حالت

## ۶- شبیه‌سازی

ما الگوریتم پیشنهادیمان را شبیه‌سازی نموده و نتایج بدست آمده از آن را با LS مقایسه نموده‌ایم. برای  $T$  و  $N$  به ترتیب مقدار ۱۰۰ و ۲۰۰ انتخاب شده است. نتایج شبیه‌سازی برای دو فیلتر متفاوت در شکل‌های (۱) و (۲) نشان داده شده است. هر شکل از هشت بخش ذیل تشکیل شده است: (a) مشخصات طراحی شامل تابع مطلوب (که با خط توپر) و تابع انحراف مجاز (که با خط چین مشخص شده است)؛ (b) تابع طراحی شده با استفاده از ELSE؛ (c) تابع مطلوب نهایی در الگوریتم ELSE؛ (d) سیگنال خطای فیلتر طراحی شده با ELSE (که مطابق معادله ۱۹ محاسبه شده است)؛ (e) تابع طراحی شده با استفاده از LS؛ (f) سیگنال خطای فیلتر طراحی شده با LS (که مطابق معادله ۱۹ محاسبه شده است)؛ (g) تغییرات مقدار تابع هزینه در طول همگرایی ELSE در آخرین مرحله از فاز کاهش  $h$  تغییرات سرعت همگرایی در آخرین مرحله از فاز کاهش. تابع خطا که شامل انحرافات غیرمجاز پاسخ فرکانسی فیلتر است، مطابق معادله (۱۹) تعریف می‌شود.

$$e(\omega) = S(\omega) \cdot \max(\Psi(\omega), 0) \quad (19)$$

در شکل‌های (۱) و (۲)، از مقایسه بخش‌های  $d$  و  $f$  می‌توان بوضوح کارایی الگوریتم پیشنهادی را در کاهش انحرافات غیرمجاز در مقایسه با LS مشاهده نمود. همانطور که در این شکل‌ها نشان داده شده است، خطای فیلترهای طراحی شده توسط ELSE بسیار کمتر از فیلتر معادل طراحی شده توسط LS است. در حقیقت می‌توان گفت که ELSE تقریباً تمام خواسته‌های ما در طراحی را برآورده نموده است. بعلاوه از مقایسه بخش‌های  $b$  و  $e$  در شکل‌های مذکور می‌توان نتیجه گرفت که ELSE برای حذف انحرافات غیرمجاز در یک ناحیه، عملاً نوسانات پاسخ فرکانسی فیلتر را در سایر نواحی تا حد امکان (بطوری که انحرافات از مجزوده مجاز فراتر نرود) افزایش داده است. به عبارت دیگر، الگوریتم پیشنهادی با افزایش اندازه نوسانات در محدوده مجاز، سعی دارد نوسانات غیرمجاز در سایر نواحی را حذف نماید. در بخش  $c$  هر شکل، تابع مطلوب نهایی الگوریتم ELSE نشان داده شده است. همانطور که مشاهده می‌کنید، ELSE با افزایش مقدار تابع مطلوب در جهت عکس نوسانات غیرمجاز خصوصاً در محل لب‌ها، نوسانات غیرمجاز مذکور را تضعیف و حذف نموده

است. در بخش‌های  $g$  و  $h$  هر شکل نیز به ترتیب تغییرات مقدار تابع هزینه و سرعت همگرایی ELSE در آخرین مرحله از فاز کاهش نشان داده شده است. همانطور که در شکل‌های مذکور مشاهده می‌کنید، الگوریتم پیشنهادی از یکنواختی و سرعت مناسبی در همگرایی برخوردار است، بطوری که در کمتر از ۲۰ مرحله تکرار، همگرا گردیده است. دلیل این امر را باید در الگوریتم تنظیم خودکار سرعت همگرایی جستجو نمود. همانطور که در شکل‌های  $(g-1)$  و  $(h-1)$  بخوبی نشان داده شده است، ایجاد نوسانات در مقدار تابع هزینه با کاهش سرعت همگرایی همراه است و برعکس، تغییرات یکنواخت در مقدار تابع هزینه سبب افزایش سرعت همگرایی خواهد شد. این مکانیزم به الگوریتم پیشنهادی این امکان را می‌دهد تا حد امکان سرعت همگرایی را بدون ایجاد نوسانات ناخواسته، افزایش دهد. همچنین شبیه‌سازی‌ها نشان داد که دقت الگوریتم افزایشی-کاهشی در تخمین مرتبه فیلتر در حد بسیار مطلوبی است، بطوری که بندرت در این تخمین خطاهایی به اندازه بیش از یک درجه رخ می‌دهد.

## ۷- نتیجه‌گیری

در این مقاله یک الگوریتم تکراری جدید برای طراحی فیلترهای FIR با شکل دلخواه تحت عنوان ELSE ارائه شده است. شبیه‌سازی‌ها حاکی از عملکرد مناسب الگوریتم در مقایسه با الگوریتم‌های دیگر مانند LS می‌باشد، هر چند که پیچیدگی محاسباتی الگوریتم پیشنهادی از روش‌های کلاسیک طراحی فیلترها بیشتر است. الگوریتم پیشنهادی با تعیین دقیق درجه و شکل بهینه فیلتر، ابزاری کاملاً خودکار و ایده‌آل برای طراحی فیلترهای با فاز خطی و شکل دلخواه است. در مجموع می‌توان گفت، ELSE یک میزکار (framework) پر قدرت و کاملاً خودکار برای طراحی تمام انواع فیلترها است.

## مراجع

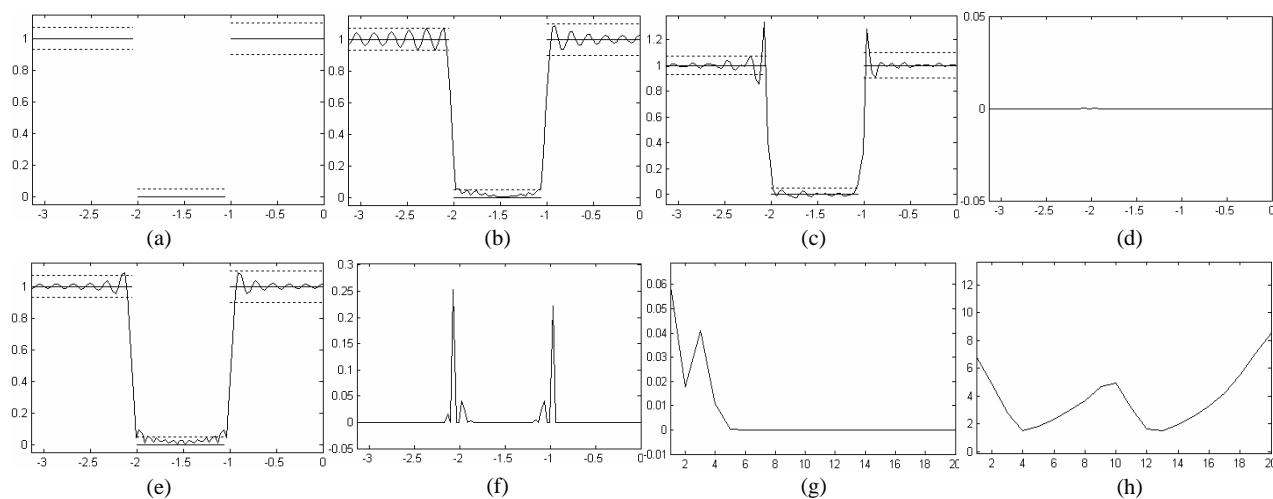
- [1] S.-P. Wu, S. Boyd, L. Vandenberghe, "FIR filter design via semi-definite programming and spectral factorization," *Proceedings of the 35th IEEE Conf. Decision and Control*, 1996.
- [2] A. V. Oppenheim, R. W. Schaffer, *Discrete-time signal processing*. Prentice-Hall, 1989.
- [3] W. Selenick, M. Lang, C. S. Burrus, "Constrained least square design of FIR filters without specified

*Proceedings of the Fifth Int'l Sym. Signal Processing and Its Applications*, 2:1005-1008, 1999.

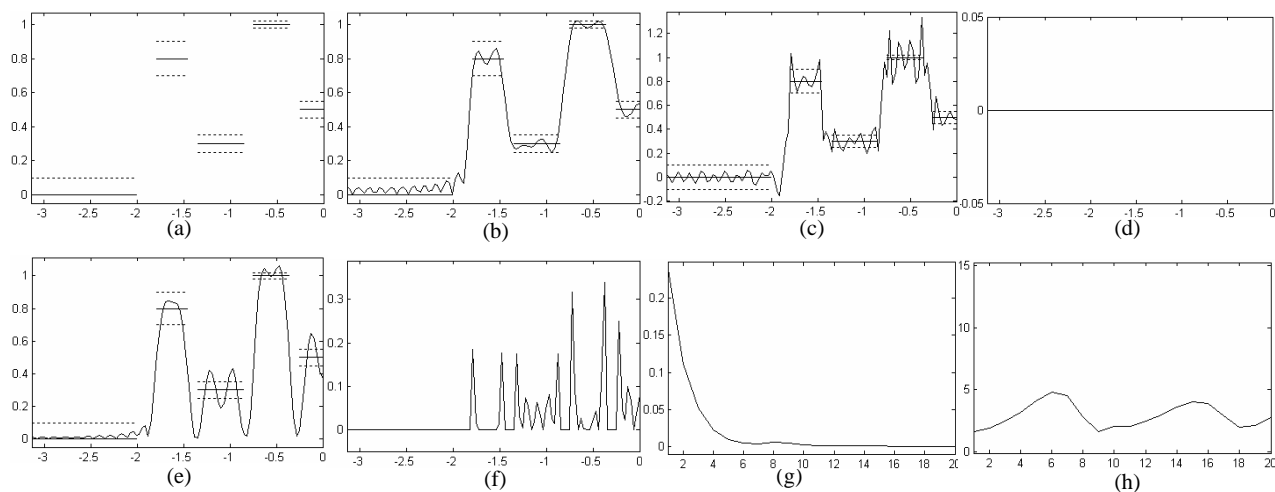
- [9] M. Oner, "A genetic algorithm for optimization of linear phase FIR filter coefficients," *Conf. Record of the Thirty-Second Asilomar Conf. Signals, Systems, and Computers*, 2:1397-1400, 1998.
- [10] S. Haykin, *Adaptive filter theory*. Prentice-Hall, 3<sup>rd</sup> ed., 1996.
- [11] K. Mohamed-pour, M. Saadatmand-T., "A new method for linear-phase FIR filter designing with desired shape," *Int'l Sym. Communication Systems, Networks, and Digital Signal Processing*, 2004.
- [12] J. Y. F. Yam, T. W. S. Chow, "Extended least squares based algorithm for training feedforward networks", *IEEE Trans on Neural Networks*, 8(3):806-810, 1997.
- [13] M. Saadatmand-T., M. Khademi, "Color defect detection of ceramic tiles using clustering and a new color clustering method," *Iranian Conf. on Machine Vision and Image Processing*, 2003.

transition bands," *IEEE Proc. Int'l Conf. Acoustic, Speech, Signal Processing*, 1995.

- [4] H. H. Dam, K. L. Teo, "The dual parameterization approach to optimal least square FIR filter design subject to maximum error constraints," *IEEE Trans. Signal Processing*, 48(8):2314-2320, 2000.
- [5] C. S. Burrus, "Multiband least squares FIR filter design," *IEEE Trans. Signal Processing*, 43(2):412-421, 1995.
- [6] S.-C. Pei, C.-C. Tseng, "FIR filter design based on total least squares error criterion," *Proceedings of IEEE Int'l Sym. Circuits and Systems*, 3:283-286, 1999.
- [7] M. Z. Komodromos, S. F. Russell, P. Tak, P. Tang, "Design of FIR filters with complex desired frequency response using a generalized Remez algorithm," *IEEE Trans. Acoustics, Speech, and Signal Processing*, 43(2):412-421, 1995.
- [8] M. Oner, M. Askar, "Incremental design of high complexity FIR filters by genetic algorithms,"



شکل ۱: طراحی یک فیلتر میان نگذر: (a) مشخصات طراحی، (b) تابع طراحی شده با استفاده از ELSE، (c) تابع مطلوب نهایی در الگوریتم ELSE، (d) سیگنال خطای فیلتر طراحی شده با ELSE، (e) تابع طراحی شده با استفاده از LS، (f) سیگنال خطای فیلتر طراحی شده با LS، (g) تغییرات مقدار تابع هزینه در طول همگرایی ELSE در آخرین مرحله از فاز کاهشی و (h) تغییرات سرعت همگرایی در آخرین مرحله از فاز کاهشی.



شکل ۲: طراحی یک فیلتر با شکل دلخواه: (a) مشخصات طراحی، (b) تابع طراحی شده با استفاده از ELSE، (c) تابع مطلوب نهایی در الگوریتم ELSE، (d) سیگنال خطای فیلتر طراحی شده با ELSE، (e) تابع طراحی شده با استفاده از LS، (f) سیگنال خطای فیلتر طراحی شده با LS، (g) تغییرات مقدار تابع هزینه در طول همگرایی ELSE در آخرین مرحله از فاز کاهشی و (h) تغییرات سرعت همگرایی در آخرین مرحله از فاز کاهشی.