

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



دانشکده مهندسی برق و رباتیک

گروه مخابرات

پایان نامه کارشناسی ارشد

حذف تطبیقی پژواک غیرخطی توسط روش های پردازش نرم

علیرضا نظام دوست

استاد راهنما :

دکتر امیدرضا معروضی

بهمن ۱۳۹۴

تقدیم به

ستون‌های استوار هدایت، ائمه بزرگوار اسلام

به ویژه امام رضا (ع) و حضرت ولی عصر (عج)

پدر و مادر عزیز و مهربانم

پروردگارا!

نه می‌توانم مویشانشان را که در راه عزت من سفید شد، سیاه کنم و نه برای دست‌های پینه‌بسته‌شان که

شمره‌ی تلاش برای پیشرفت من است، مرهمی دارم. پس توفیقم ده تا قدر تک تک لحظه‌های

بودن در کنارشان را بدانم و بتوانم گوشه‌ای از فداکاری‌هایشان را جبران نمایم...

تقدیر و تشکر:

حمایت‌های بی‌دریغ و راهنمایی‌های ارزنده استاد گران‌قدر و دلسوز جناب آقای دکتر امیدرضا معروضی، همواره شامل حال این‌جانب بوده است. لذا بر خود لازم می‌دانم سپاسگزاری صمیمانه خود را خدمت این استاد عزیز نثار کنم و سلامتی، طول عمر، موفقیت و سرفرازی ایشان را از خداوند مسئلت نمایم.

همچنین از همه دوستان و عزیزانی که در طول دوره کارشناسی ارشد به شیوه‌های گوناگون محبت، راهنمایی‌ها و حمایت‌های خود را از بنده دریغ نکردند، لحظاتی خوش و ماندگار برای من ایجاد کردند، به‌صورت ویژه تشکر می‌کنم و بهترین آرزوها را در تمامی مراحل زندگی برای آن‌ها خواستارم.

انعکاس‌پذیری باش که می‌خواهی در دیگران ببینی...

اگر عشق می‌خواهی عشق بورز، اگر صداقت می‌خواهی، راست‌گو باش

دنیا‌پذیری جز پشواک نیست...

تعهد نامه

اینجانب **علیرضا نظام دوست** دانشجوی دوره کارشناسی ارشد رشته مهندسی برق / مخابرات دانشکده مهندسی برق و رباتیک دانشگاه صنعتی شاهرود نویسنده پایان نامه **حذف تطبیقی پژواک غیرخطی توسط روش های پردازش نرم** تحت راهنمایی دکتر امیدرضا معروضی متعهد می شوم .

- تحقیقات در این پایان نامه توسط اینجانب انجام شده است و از صحت و اصالت برخوردار است .
- در استفاده از نتایج پژوهش های محققان دیگر به مرجع مورد استفاده استناد شده است .
- مطالب مندرج در پایان نامه تاکنون توسط خود یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ جا ارائه نشده است .
- کلیه حقوق معنوی این اثر متعلق به دانشگاه صنعتی شاهرود می باشد و مقالات مستخرج با نام « دانشگاه صنعتی شاهرود » و یا « **Shahrood University of Technology** » به چاپ خواهد رسید .
- حقوق معنوی تمام افرادی که در به دست آمدن نتایج اصلی پایان نامه تأثیرگذار بوده اند در مقالات مستخرج از پایان نامه رعایت می گردد.
- در کلیه مراحل انجام این پایان نامه ، در مواردی که از موجود زنده (یا بافت های آن ها) استفاده شده است ضوابط و اصول اخلاقی رعایت شده است .
- در کلیه مراحل انجام این پایان نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته یا استفاده شده است اصل رازداری ، ضوابط و اصول اخلاق انسانی رعایت شده است .

تاریخ

امضای دانشجو

مالکیت نتایج و حق نشر

- کلیه حقوق معنوی این اثر و محصولات آن (مقالات مستخرج ، کتاب ، برنامه های رایانه ای ، نرم افزار ها و تجهیزات ساخته شده است) متعلق به دانشگاه صنعتی شاهرود می باشد . این مطلب باید به نحو مقتضی در تولیدات علمی مربوطه ذکر شود .
- استفاده از اطلاعات و نتایج موجود در پایان نامه بدون ذکر مرجع مجاز نمی باشد.

چکیده:

حضور عناصر مداری غیرخطی روی عملکرد حذف‌کننده پژواک صوتی رایج اثر می‌گذارد و روی کیفیت مخابرات صوتی اثر منفی دارد. برای حل این مشکل، حذف‌کننده‌های پژواک صوتی غیرخطی به کار گرفته می‌شوند، که از مدل‌سازی مسیرهای غیرخطی و بهبود صوت به دست می‌آیند. در این پایان‌نامه کلاس جدیدی از الگوریتم‌های افقی غیرخطی بر پایه فیلترهای افقی پیوند تابعی برای مدل‌های غیرخطی کانال صوتی معرفی می‌گردد. با توجه به معماری انعطاف‌پذیر، این الگوریتم‌ها یک راه حل مؤثر برای مدل کردن غیرخطی‌های اثرگذار بر روی سیگنال صوت هستند. توابع ریاضی چبیشف، مثلثاتی و لژاندر به منظور بسط دادن ورودی استفاده می‌شوند. جداسازی بین اجزای خطی و غیرخطی در فیلترهای افقی پیوند تابعی مجزا و اشتراکی منجر به دو فیلتر افقی مختلف می‌شود. نوع خطی با هدف تخمین پاسخ ضربه صوت و نوع غیرخطی تنها وظیفه آن مدل‌سازی غیرخطی‌ها است و در مقابل درجات مختلف غیرخطی مقاوم است. با این روش عملکرد همگرایی به صورت چشم‌گیری بهبود می‌یابد. مدل اشتراکی با جلوگیری از نویزهای گرادیانی غیرخطی، زمانی که سیستم خطی است، در هر دو حالت پژواک خطی و غیرخطی دارای عملکرد خوبی در میزان حذف پژواک است. بنابراین معماری‌هایی مبتنی بر این پایه راه حلی مؤثری برای مدل‌سازی کانال صوتی در نظر گرفته می‌شوند. نتایج به دست آمده در این پایان‌نامه نشان می‌دهد الگوریتم‌های افقی بر پایه پیوند تابعی چبیشف و مثلثاتی دارای عملکرد خوبی برای حذف پژواک غیرخطی هستند و تا 50 dB پژواک را اصلاح می‌کنند.

کلید-واژه: پژواک صوتی غیرخطی، حذف‌کننده، فیلتر افقی، پیوند تابعی چبیشف، پیوند تابعی مثلثاتی، پیوند تابعی لژاندر.

فهرست مطالب

- ۱- فصل اول: پیشگفتار..... ۱
- ۱-۱ انگیزه تحقیق..... ۲
- ۲-۱ هدف تحقیق..... ۴
- ۳-۱ مروری بر فصل‌های پایان‌نامه..... ۶
- ۲- فصل دوم: مروری بر کارهای گذشته..... ۷
- ۱-۲ مقدمه..... ۸
- ۲-۲ حذف پژواک صوتی..... ۱۰
- ۳-۲ انواع مختلف پژواک در سیستم‌های مخابراتی و مسئله تأخیر..... ۱۲
- ۱-۳-۲ پژواک الکتریکی..... ۱۲
- ۲-۳-۲ پژواک صوتی..... ۱۳
- ۴-۲ مسئله تأخیر..... ۱۳
- ۵-۲ روش‌های مختلف حذف پژواک..... ۱۴
- ۱-۵-۲ سرکوب کننده‌های پژواک..... ۱۴
- ۲-۵-۲ حذف کننده‌های پژواک وفقی..... ۱۴
- ۶-۲ آشکارسازی کانال به صورت فعال..... ۱۵
- ۷-۲ کاربردهای کنترل پژواک..... ۱۵
- ۸-۲ سنجش عملکرد..... ۱۸
- ۱-۸-۲ افزایش تلفات بازگشتی پژواک..... ۱۸

۱۹	۲-۸-۲ توان لحظه‌ای سیگنال پژواک
۱۹	۹-۲ جمع بندی و نتیجه گیری
۲۱	۳- فصل سوم: فیلترهای وفقی برای سیستم غیرخطی
۲۲	۱-۳ مقدمه
۲۳	۲-۳ فیلترهای وفقی پیوند تابعی
۲۵	۳-۳ FLAF با بسط مثلثاتی
۲۶	۴-۳ FLAF باحافظه و بی حافظه
۳۰	۵-۳ تطبیق ضرایب FLAF
۳۱	۶-۳ FLAF های مقاوم برای NAEC
۳۲	۷-۳ فیلتر وفقی پیوند تابعی مجزا
۳۵	۸-۳ فیلتر وفقی پیوند تابعی اشتراکی
۳۸	۹-۳ FLAF با بسط‌های مختلف
۳۹	۱-۹-۳ FLAF با بسط چیشف
۴۰	۲-۹-۳ FLAF با بسط لژاندر
۴۱	۳-۹-۳ فیلتر وفقی پیوند تابعی مجزای موازی
۴۲	۱۰-۳ طراحی پیوند تابعی باحافظه مرتبه یک
۴۴	۱۱-۳ جمع بندی و نتیجه گیری
۴۵	۴- فصل چهارم: شبیه سازی حذف پژواک غیرخطی
۴۶	۱-۴ مقدمه

۴۶	۲-۴ نتایج تجربی
۴۹	۳-۴ ارزیابی اول
۵۵	۴-۴ ارزیابی دوم
۶۲	۵-۴ ارزیابی سوم
۶۷	۶-۴ ارزیابی چهارم
۷۲	۷-۴ ارزیابی پنجم
۷۳	۵- فصل پنجم: نتیجه‌گیری و پیشنهادها
۷۴	۱-۵ مروری بر کارهای انجام‌شده
۷۴	۲-۵ نتایج
۷۶	۳-۵ پیشنهادها
۷۷	۶- مراجع

فهرست جدول‌ها

- جدول ۴-۱: الگوریتم بر پایه FLAF ۵۰
- جدول ۴-۲: الگوریتم CFLAF ۵۷
- جدول ۴-۳: الگوریتم بر پایه FLAF موازی ۶۳
- جدول ۴-۴: زمان اجرا بر حسب ثانیه ۷۲

فهرست شکل‌ها

- شکل ۱-۱: مدل مسیر صوت ۲
- شکل ۱-۲: مدل قرارگیری بلندگو و میکروفون ۱۰
- شکل ۲-۲: مدل پردازش حذف کننده پژواک ۱۵
- شکل ۱-۳: فیلتر وفقی پیوند تابعی ۲۳
- شکل ۲-۳: مدل گسترش پیوندهای تابعی ۲۴
- شکل ۳-۳: بسط تابعی بدون حافظه ۲۸
- شکل ۴-۳: بسط تابعی با حافظه دو ۲۹
- شکل ۵-۳: فیلتر وفقی پیوند تابعی مجزا ۳۴
- شکل ۶-۳: فیلتر وفقی پیوند تابعی اشتراکی ۳۶
- شکل ۷-۳: فیلتر وفقی پیوند تابعی مجزای موازی ۴۲
- شکل ۸-۳: بسط تابعی با حافظه یک ۴۳
- شکل ۱-۴: منحنی ERLE در حالت $Q=1$ بی حافظه برای پیوند تابعی مثلثاتی، چبیشف، لژاندر و یک فیلتر خطی در مورد ورودی صوت یک خانم ۵۱
- شکل ۲-۴: منحنی MSE در حالت $Q=1$ بی حافظه برای پیوند تابعی مثلثاتی، چبیشف، لژاندر و یک فیلتر خطی در مورد ورودی صوت یک خانم ۵۱
- شکل ۳-۴: منحنی ERLE در حالت $Q=1$ بی حافظه برای پیوند تابعی مثلثاتی، چبیشف، لژاندر و یک فیلتر خطی در مورد ورودی رنگی ۵۲
- شکل ۴-۴: منحنی MSE در حالت $Q=1$ بی حافظه برای پیوند تابعی مثلثاتی، چبیشف، لژاندر و یک فیلتر خطی در مورد ورودی رنگی ۵۲
- شکل ۵-۴: منحنی ERLE در حالت $Q=10$ بی حافظه برای پیوند تابعی مثلثاتی، چبیشف، لژاندر و یک

۵۳ فیلتر خطی در مورد ورودی صوت یک خانم.

شکل ۴-۶: منحنی MSE در حالت $Q=10$ بی حافظه برای پیوند تابعی مثلثاتی، چبیشف، لژاندر و یک

۵۳ فیلتر خطی در مورد ورودی صوت یک خانم.

شکل ۴-۷: منحنی ERLE در حالت $Q=10$ بی حافظه برای پیوند تابعی مثلثاتی، چبیشف، لژاندر و یک

۵۴ فیلتر خطی در مورد ورودی رنگی.

شکل ۴-۸: منحنی MSE در حالت $Q=10$ بی حافظه برای پیوند تابعی مثلثاتی، چبیشف، لژاندر و یک

۵۴ فیلتر خطی در مورد ورودی رنگی.

شکل ۴-۹: منحنی ERLE در حالت $Q=5$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف و یک فیلتر

۵۸ خطی در مورد ورودی صوت یک خانم.

شکل ۴-۱۰: منحنی MSE در حالت $Q=5$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف و یک فیلتر

۵۸ خطی در مورد ورودی صوت یک خانم.

شکل ۴-۱۱: منحنی ERLE در حالت $Q=5$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف و یک فیلتر

۵۹ خطی در مورد ورودی رنگی.

شکل ۴-۱۲: منحنی MSE در حالت $Q=5$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف و یک فیلتر

۵۹ خطی در مورد ورودی رنگی.

شکل ۴-۱۳: منحنی ERLE در حالت $Q=20$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف و یک

۶۰ فیلتر خطی در مورد ورودی صوت یک خانم.

شکل ۴-۱۴: منحنی MSE در حالت $Q=20$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف و یک فیلتر

۶۰ خطی در مورد ورودی صوت یک خانم.

شکل ۴-۱۵: منحنی ERLE در حالت $Q=20$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف و یک

۶۱ فیلتر خطی در مورد ورودی رنگی.

شکل ۴-۱۶: منحنی MSE در حالت $Q=20$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف و یک فیلتر

- خطی در مورد ورودی رنگی ۶۱
- شکل ۴-۱۷: منحنی ERLE در حالت $Q=5$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف ،حالت موازی در مورد ورودی صوت یک خانم ۶۴
- شکل ۴-۱۸: منحنی MSE در حالت $Q=5$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف ،حالت موازی در مورد ورودی صوت یک خانم ۶۴
- شکل ۴-۱۹: منحنی ERLE در حالت $Q=5$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف ،حالت موازی در مورد ورودی رنگی ۶۵
- شکل ۴-۲۰: منحنی MSE در حالت $Q=5$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف ،حالت موازی در مورد ورودی رنگی ۶۵
- شکل ۴-۲۱: منحنی ERLE در حالت $Q=15$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف ،حالت موازی در مورد ورودی صوت یک خانم ۶۶
- شکل ۴-۲۲: منحنی MSE در حالت $Q=15$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف ،حالت موازی در مورد ورودی صوت یک خانم ۶۶
- شکل ۴-۲۳: منحنی ERLE در حالت $Q=10$ باحافظه برای پیوند تابعی مثلثاتی در مورد ورودی صوت یک خانم ۶۹
- شکل ۴-۲۴: منحنی MSE در حالت $Q=10$ باحافظه برای پیوند تابعی مثلثاتی در مورد ورودی صوت یک خانم ۶۹
- شکل ۴-۲۵: منحنی ERLE در حالت $Q=10$ باحافظه برای پیوند تابعی چبیشف در مورد ورودی صوت یک خانم ۷۰
- شکل ۴-۲۶: منحنی MSE در حالت $Q=10$ باحافظه برای پیوند تابعی چبیشف در مورد ورودی صوت یک خانم ۷۰
- شکل ۴-۲۷: منحنی ERLE در حالت $Q=10$ باحافظه برای مقایسه پیوند تابعی چبیشف و مثلثاتی در

مورد ورودی صوت یک خانم ۷۱

شکل ۴-۲۸: منحنی MSE در حالت $Q=10$ باحافظه برای مقایسه پیوند تابعی چیشف و مثلثاتی در

مورد ورودی صوت یک خانم ۷۱

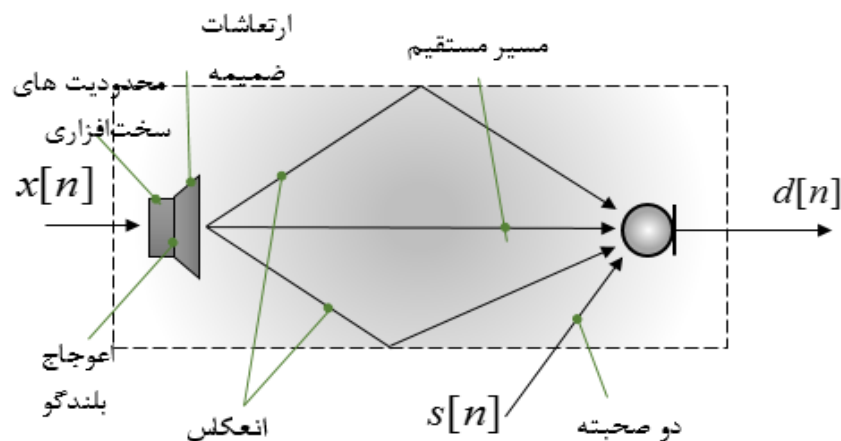
فصل اول

پیشگفتار

۱-۱ انگیزه تحقیق

استفاده از الگوریتم‌های نرم و هوشمند در ارتباطات گفتاری برای مدل‌سازی مسیرهای صوتی و درک صداهای پیچیده امری متداول است. در توسعه چنین سیستم‌های ارتباطی بسیاری از حوزه‌های پژوهشی دارای زمینه‌های مشترک می‌باشند که عبارتند از: کاهش نویز، بهبود صوت، حذف پژواک صوتی، مدل‌سازی کانال غیرخطی، ردیابی منبع و جداسازی منبع کور. در چنین زمینه‌ی پژوهشی مسئله مهم و پایه‌ای الگوریتم‌های یادگیری و ساختار فیلترهای تطبیقی می‌باشد [۱]. قابلیت‌های ساختاری این فیلترها انطباق پذیری با محیط‌های صوتی است که یک رابط صوتی هوشمند را می‌سازد. علاوه بر این عملکرد فیلتر تطبیقی^۱ به کیفیت سیگنال‌های صوتی پردازش‌شده مربوط می‌باشد.

مهم‌ترین کاربردهای صوتی که فیلتر تطبیقی نقش راهبردی در آن دارد مدل‌سازی کانال صوتی مانند حذف پژواک صوتی می‌باشد. پدیده پژواک صوتی زمانی رخ می‌دهد که یک نسخه تاخیردار از سیگنال صحبت به وسیله بلندگویی که در مجاورت میکروفون است باز تولید شده و به سمت کاربر مقصد منعکس می‌شود.



شکل ۱-۱: مدل مسیر صوت [۵]

¹ Adaptive filter

یکی از اهداف حذف‌کننده پژواک صوتی، برآورد پاسخ ضربه صوتی^۱ یعنی مدل‌سازی مسیر صوتی به‌منظور کم کردن سیگنال پژواک از سیگنال میکروفون است. بنابراین مدل‌سازی کانال صوتی یک مسئله فراگیر در ارتباطات صوتی می‌باشد زیرا یکی از مشکلات رایج در تمام تحلیل‌های صوتی است. همان‌گونه که در شکل ۱-۱ مشاهده می‌شود پاسخ ضربه به حضور عناصر نا ایستا در محیط، حضور سیگنال‌های تداخلی ناخواسته و حضور عوامل غیرخطی وابسته است. چنین پدیده‌ای، به شدت کیفیت سیگنال صحبت دریافت شده را کاهش می‌دهد حذف پژواک با استفاده از تکنیک‌های پردازش سیگنال در بازسازی سیگنال مخابره‌ای صوت نقش اساسی دارد [۲]. این دلیلی است که تحقیق ارائه شده به‌صورت اساسی با کاربردهای مدل‌سازی کانال صوتی، و به‌طور خاص حذف پژواک صوتی در ارتباط است تا تکنیک‌های فیلترهای تطبیقی جدیدی را گسترش دهد و در سایر برنامه‌های گفتگوی از راه دور مورد استفاده قرار بگیرد. با بررسی تحقیقات گذشته متوجه شدیم که، در حذف پژواک صوتی خطی پیشرفت‌های قابل توجهی به دست آمده است که در آن قابلیت‌های فیلترهای تطبیقی به‌منظور مدل کردن پاسخ ضربه صوتی به بهترین نحو مورد استفاده قرار گرفته است. اخیراً نتایج درخشانی برای برنامه‌های ارتباطی صحبت دست‌آزاد^۲ در محیط‌های پیرانعکاس و در حضور سیگنال‌های تداخلی به‌دست آمده است. عامل‌هایی که نمی‌توانند در ارتباطات همه‌جانبه نادیده گرفته شوند با این حال روش‌های متداول حذف پژواک در مورد حذف پژواک‌های غیرخطی به نتایج چشم‌گیری دست نیافته‌اند [۳].

پژواک غیرخطی با حضور اعوجاج در مسیرهای صوتی ایجاد می‌شوند و موجب کاهش عملکرد و حتی کاهش شدید کیفیت اطلاعات دریافتی می‌شود. عوامل غیرخطی اغلب در کاربردهای صوتی رخ می‌دهند چرا که آن‌ها به وسیله بلندگوها یا با ارتعاشات دستگاه‌های صوتی تولید می‌شوند [۴]. به

¹ Acoustic impulse response (AIR)

² hands-free

همین دلیل امروزه نمی‌توان به حذف‌کننده‌های پژواک که عوامل غیرخطی را در نظر می‌گیرند بی‌اعتنا بود، همچنین با توجه به گسترش دستگاه‌های صوتی کم‌هزینه که مواد و قطعات الکترونیکی کم‌کیفیتی دارند اعوجاج‌های بیش‌تری تولید می‌کنند.

از میان بهترین حذف‌کننده‌های پژواک صوتی غیرخطی، در سال‌های اخیر آن‌هایی که بر اساس فیلترهای تطبیقی ولترا بوده‌اند باقی مانده‌اند [۵]. این حذف‌کننده‌های پژواک صوتی غیرخطی دارای هزینه‌های محاسباتی بیشتری از حذف‌کننده‌های پژواک مرسوم می‌باشند. علاوه بر این حذف‌کننده‌های پژواک صوتی غیرخطی ولترا ممکن است وقتی ماهیت پژواک خطی باشد عملکرد بدتری نسبت به حذف‌کننده‌های خطی داشته باشند که این بر روی راهبردهای بسیاری از شرکت‌هایی که خدمات ارتباط از راه دور ارائه می‌دهند، تأثیر می‌گذارد و اغلب باعث می‌شود از حذف‌کننده‌های پژواک غیرخطی حتی با وجود پایین آمدن کیفیت ارتباطات استفاده نکنند. که این موضوع نیز از انگیزه‌های اصلی این پروژه تحقیقاتی است.

۱-۲ هدف تحقیق

توسعه الگوریتم‌های تطبیقی برای روابط صوتی هوشمند بر پایه سناریوهای بسیار پیچیده می‌باشد و ممکن است چندین پدیده، که موجب کاهش فهم صحبت شود را در یک ارتباط صوتی خنثی کند. ما از تجزیه و تحلیل چنین پدیده‌ی تداخلی که از آن به عنوان پژواک‌های خطی یا غیرخطی یاد می‌شود شروع می‌کنیم. چنین تقسیمی اجازه می‌دهد تا الگوریتم‌های تطبیقی را طراحی کنیم و رابط‌های صوتی هوشمندی بسازیم که در برابر تداخل سیگنال‌ها مقاوم باشند.

با در نظر گرفتن مدل‌سازی کانال صوتی دارای پژواک خطی کافی است مدل‌های تطبیقی‌ای را بررسی کنیم که از نظر آماری مقاوم هستند. با این حال از آنجا که معمولاً پژواک خطی همراه صوت وجود دارد، به منظور بازسازی بهتر صوت، آزاد از هرگونه تداخل، نویز و سیگنال‌های ناخواسته بهتر

است یک روش پردازش غیرخطی از اطلاعات به دست آمده اجرا شود که قادر به یادگیری از محیط با روش نظارت شده یا بدون نظارت باشد. در پژواک خطی و غیرخطی یادگیری خودکار و تداوم انطباق‌پذیری عناصر، برای بالا بردن کیفیت ارتباطات صوتی می‌باشد [۲].

به منظور مقابله با سیگنال تداخلی خطی ما با یک تکنیک فیلتری که اخیراً بر پایه فیلترهای تطبیقی مناسب ارائه شده است کار می‌کنیم [۳]. این الگوریتم‌ها از محدودیت‌های پراکندگی که در AIRهای معمولی می‌باشند بهره می‌برد بنابراین موجب بهبود عملکرد می‌شود و محدودیت‌های اعمال شده توسط محیط‌های صوتی را کاهش می‌دهد. بررسی این خانواده از الگوریتم‌ها با فرمول‌بندی فیلترهای تطبیقی و توسعه الگوریتم‌های تطبیقی، برای ارتباطات صوتی با کیفیت بهتر است.

در طرف دیگر به منظور مقابله با عوامل غیرخطی در مدل‌سازی کانال غیرخطی یک مدل فیلتر غیرخطی جدید بر پایه پیوندهای تابعی^۱ ارائه می‌کنیم. با استفاده از چنین مدل غیرخطی الگوریتم‌ها و ساختارهایی را برای حذف پژواک صوتی غیرخطی توسعه می‌دهیم. ایده اصلی که پیوندهای تابعی فیلترهای تطبیقی را پی‌ریزی می‌کند، تخمین و مدل‌سازی عوامل غیرخطی موجود در مسیر پژواک و حذف آن‌ها می‌باشد، بنابراین کیفیت اطلاعات صوتی دریافتی بهبود می‌یابد.

علاوه بر این در هر دو مورد پژواک خطی و غیرخطی الگوریتم‌های تطبیقی ارائه شده استفاده می‌شوند تا یک معماری فیلتری پیچیده مبتنی بر ترکیب تطبیقی فیلترها ارائه دهند. چنین ساختاری در برابر انواع مختلفی از شرایط نامطلوب محیطی در مقایسه با تکنیک فیلترهای معمولی مقاومت بیشتری نشان می‌دهد.

¹ Functional link

۳-۱ مروری بر فصل‌های پایان‌نامه

فصل‌های بعدی این پایان‌نامه بدین شرح است. در فصل دوم به بررسی سابقه موضوع حذف پژواک پرداخته می‌شود. ابتدا کارهایی که تاکنون در این زمینه انجام شده مورد بررسی قرار می‌گیرد. در ادامه اهمیت حذف پژواک صوتی و انواع مختلف آن، نحوه حذف و کنترل پژواک مرور می‌شود. در انتها نحوه سنجش پژواک را تشریح می‌گردد. در فصل سوم، الگوریتم وفقی به کار رفته برای حذف پژواک غیرخطی و همچنین الگوریتم‌های پیشنهادی بیان می‌شود. در فصل چهارم، مدل‌های ارائه شده در فصل سوم شبیه سازی و نتایج به دست آمده تحلیل می‌شوند. در فصل پنجم، به طور مختصر، مروری بر کارهای انجام شده در این پایان‌نامه صورت می‌گیرد، نتایج جمع‌بندی می‌شوند و پیشنهادهایی برای ادامه تحقیق ارائه می‌گردد.

فصل دوم

مروری بر کارهای گذشته و

مفاهیم پایه

وجود خاصیت غیرخطی در مسیر پژواک صوتی بر روی عملکرد حذف‌کننده پژواک صوتی مرسوم تأثیر می‌گذارد. در سال‌های اخیر، این موضوع به جهت رشد و گسترش سیستم‌های مخابراتی کم‌هزینه است که اغلب از المان‌ها با کیفیت کم تشکیل شده‌اند، اهمیت بیشتری یافته است. بیش‌تر المان‌های الکترونیکی نظیر تقویت‌کننده و بلندگو و مواد پوششی نظیر محفظه پلاستیکی ممکن است باعث خاصیت غیرخطی مهمی در پاسخ ضربه صوتی^۱ شود که موجب کاهش محسوس کیفیت صوت می‌شود [۶, ۷]. برای حل این مشکل، حذف‌کننده‌های پژواک صوتی غیرخطی به کار گرفته شده‌اند که از مدل‌سازی مسیرهای غیرخطی و بهبود صوت حاصل می‌شوند.

در سال‌های اخیر، ساختارهای مختلفی مورد بررسی قرار گرفته‌اند تا مسیرهای پژواک صوتی غیرخطی را مدل کنند. یک تکنیک متداول بر پایه استفاده از تبدیلات غیرخطی^۲ است که انواع مختلف اعوجاج‌ها را مدل‌سازی می‌کند [۴, ۸].

یک تابع کسینوسی^۳ در [۸] استفاده شده است تا هر غیرخطی‌ای را مدل کند. در [۹] یک تابع سیگموئید^۴ دو پارامتری ارائه شده که شیب و دامنه آن می‌تواند در طی فرآیند یادگیری به‌روزرسانی شود. تابع سیگموئید وفقی دیگری در [۱۰] استفاده شده است تا عملکرد حذف‌کننده پژواک غیرخطی^۵ را به عنوان تغییرات زمانی^۶ ارزیابی کند. یک‌راه حل انعطاف‌پذیرتر در [۴] با استفاده از توابع اسپلاین^۷ که منحنی پارامترهای هموار هستند و به وسیله درونیابی^۸ از نقاط کنترل شده

¹ Acoustic

² Nonlinear transformations

³ cosine

⁴ Sigmoid function

⁵ Nonlinear Acoustic Echo Cancellation (NAEC)

⁶ time changes

⁷ spline

⁸ interpolation

مناسب که در یک جدول جمع شده‌اند، ارائه شده است. مدل‌هایی بر پایه بلوک وینر-همراشتاین^۱ با استفاده از توابع غیرخطی نیز در [۱۱] و [۱۲] بررسی شده‌اند.

علاوه بر اینکه حذف کننده‌های پژواک غیرخطی با استفاده از توابع غیرخطی عملکرد خوبی را ارائه می‌کنند بهترین مدل غیرخطی برای کاربردهای حذف پژواک بر پایه فیلترهای^۲ ولفی^۳ وولترا [۱۳] هستند. ساختار کلی فیلترهای وولترا از سری‌های معروف تیلور گرفته می‌شود و می‌تواند به صورت یک تعمیم کلی از فیلترهای ولفی خطی در نظر گرفته شود. فیلترهای وولترا می‌توانند بازه وسیعی از غیرخطی‌های باحافظه و بدون حافظه را مدل کنند. با این حال، حذف پژواک صوتی برای ارتباطات صوتی نیازمند فیلترهای ولفی بزرگ برای مدل‌سازی AIR است [۱۴]. چون هزینه محاسباتی متناسب با طول پاسخ ضربه فیلتر است، تطبیق فیلترهای وولترا می‌تواند گران باشد و مشکلی برای پیاده‌سازی بلادرنگ^۴ ایجاد می‌کند. لذا در سال‌های اخیر، فیلترهای وولترا با کاهش پیچیدگی محاسباتی بررسی شده تا پیاده‌سازی بلادرنگ امکان‌پذیر باشد [۱۵]. دیگر تکنیک مدل‌سازی غیرخطی بر پایه پیوندهای تابعی^۵ است. پیوند تابعی یک عملگر تابعی است و اجازه ارائه یک الگو از ورودی در یک فضای ویژگی^۶ را می‌دهد و حجم پردازش افزایش می‌یابد تا به نتیجه مطلوب برسد. پیوندهای تابعی در ابتدا توسط Pao [۱۶] پیشنهاد شد و هدف گسترش کلاسی از شبکه‌های عصبی مسیر مستقیم تک لایه^۷ تحت عنوان شبکه‌های مصنوعی پیوندهای تابعی^۸ بود. Pao نشان داد که FLANN چگونه برای تقریب تابع و شناسایی الگو^۸ با نرخ همگرایی بیشتر و بار محاسباتی کمتر نسبت به

¹ Wiener-Hammerstein

² VFS

³ Real-time

⁴ Functional link

⁵ Feature space

⁶ Single-layer feedforward neural networks

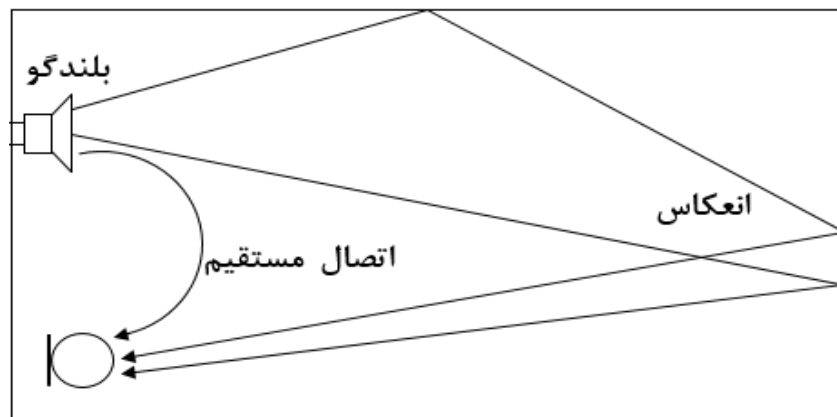
⁷ Functional link artificial neural networks (FLANNs)

⁸ Pattern recognition

شبکه‌های مصنوعی پرسپترون چندلایه^۱، به کار می‌روند.

۲-۲ حذف پژواک صوتی

یک کاربرد معمولی مدل‌سازی کانال صوتی حذف پژواک صوتی^۲ است. پژواک صوتی در سیستم‌های مخابراتی توسط اتصال^۳ صوتی بین یک بلندگو و یک میکروفون، همان‌طور که در شکل ۱-۲ نشان داده شده است تولید می‌شود. حذف پژواک نه تنها نیازمند دانستن سطح آن است بلکه باید تأخیر آن را نیز بدانیم [۱۷]. در انتقال با مسافت بالا پژواک یک زمان تأخیر زیاد را نشان می‌دهد و تا حد زیادی کیفیت مخابره صوت را کاهش می‌دهد. هنگامی که تأخیر به یک چهارم ثانیه می‌رسد، بیشتر مردم متوجه می‌شوند که داشتن یک مکالمه عادی دشوار است.



شکل ۱-۲: مدل قرارگیری بلندگو و میکروفون

پیش از اینکه نظریه حذف پژواک در دهه ۱۹۶۰ توسط محققان آزمایشگاه بل توسعه پیدا کند، ارتباطات صوتی دوطرفه غیرقابل قبول بود، البته اگر غیرممکن نبود [۱۸]. برای یک سیستم شنیداری با چندین میکروفون و بلندگو لازم است که چندین مسیر شناسایی شود. پس بدون توجه به تعداد میکروفون‌ها، AEC همیشه با توجه به هر یک از میکروفون‌ها به صورت انفرادی انجام می‌گیرد. اما تعداد بلندگوهای موجود در سیستم، یک اختلاف تئوری بین حذف‌کننده پژواک‌های یک کاناله (یک

¹ Multi-layer perceptron artificial neural networks

² Acoustic Echo Cancellation (AEC)

³ coupling

بلندگو) و چندکاناله (چندین بلندگو) و مشکلات مربوط به ردیابی مسیرهای پژواک ایجاد می‌کند.

در حذف پژواک سیگنال‌های منبع (بلندگو) باید شناخته شوند. بنابراین کنترل پژواک از نظر تئوری یک مسئله ساده به نظر می‌آید [۱۷] ولی کاربردهای عملی آن نسبتاً مشکل‌تر از کنترل انواع نویز مانند نویز جمع شونده، ارتعاشات صوتی و صدای ناخواسته بوده است.

از نظر تاریخی، مطالعه حذف پژواک صوتی با شناسایی سیستم و فیلترهای فوقی آغاز گردید. در حقیقت، یک فیلتر فوقی نقشی اساسی، در یک سیستم حذف پژواک ایفا می‌کند. تا زمانی که پاسخ ضربه کانال مسیر پژواک بتواند به سرعت و به دقت تعیین شود، فیلتر فوقی به صورت آسان به تولید یک تخمین خوب از پژواک می‌پردازد و آن را از سیگنال میکروفون کم می‌کند. از آن جایی که سیگنال بلندگو به عنوان مرجع پژواک در دسترس است فیلترهای فوقی متعددی برای شناسایی سیستم جهت حل مشکل بکار برده شده است [۲، ۱۹].

فیلترهای فوقی یک زمینه گسترده از پردازش سیگنال‌های دیجیتال می‌باشد [۲۰] و تکنیک‌های فوقی دارای کاربردهای فراوانی در زمینه‌های مختلف مانند شناسایی سیستم^۱، جبران سازی^۲ و ... می‌باشد. که اجرای الگوریتم‌های پیچیده با قیمت مناسب را ممکن ساخته است. بنابراین وجود پردازنده‌های سیگنال دیجیتال ارزان قیمت باعث استفاده آسان و وسیع از الگوریتم فیلترهای فوقی می‌شود.

حذف‌کننده‌های پژواک^۳ با استفاده از فناوری پردازش سیگنال‌های دیجیتال^۴ در اواخر دهه ۱۹۷۰ عرضه شدند. حذف‌کننده پژواک، سیگنال پژواک را از مکالمه تلفنی حذف می‌کند و تضعیفی به مدار

¹ system identification

² equalization

³ Echo Cancellor

⁴ Digital signal processing (DSP)

اعمال نمی‌کند، در نتیجه هر دو طرف مکالمه می‌توانند همزمان مکالمه کنند، بدون اینکه دچار مشکلی شوند.

مسئله کنترل پژواک صوتی برای طراحی ارتباطات صوتی که به روش دست‌آزاد کار می‌کنند، ضروری است. مخابرات دست‌آزاد یعنی مکالمه از طریق یک دستگاه مخابراتی بدون استفاده از دست‌ها^۱ که سال‌ها قبل مطرح شده است. مثلاً در یک اتومبیل، تلفن دست‌آزاد برای ایمنی بیشتر استفاده می‌شود که در نتیجه راننده می‌تواند در حین صحبت کردن آزادانه رانندگی کند. کنترل پژواک صوتی در این‌گونه سیستم‌ها و همچنین در سیستم‌های کنفرانس از راه دور^۲ و دستگاه‌های کنفرانس ویدئویی^۳ به روش‌های مختلفی انجام می‌شود. علی‌رغم سادگی ظاهر مسئله، کنترل پژواک صوتی یکی از پرچالشترین کارهای پردازش سیگنال دیجیتال است. کارهای تحقیقاتی زیادی در حال حاضر با این عنوان در سراسر جهان انجام می‌شود.

۲-۳ انواع مختلف پژواک در سیستم‌های مخابراتی و مسئله تأخیر

دو نوع پژواک در سیستم‌های مخابراتی وجود دارد:

۲-۳-۱ پژواک الکتریکی

هرگونه عدم تطابق امپدانس^۴ در حلقه محلی^۵ باعث ایجاد پژواک می‌شود. عدم تطابق امپدانس زمانی رخ می‌دهد که یک خط تلفن به حلقه محلی اضافه شود. پژواک الکتریکی باعث دوتایی شدن صدای اصلی می‌شود اثراتی که توسط پژواک ایجاد می‌شود می‌تواند توسط یک حذف‌کننده پژواک کاهش یابد و دلیل اصلی آن ترانسفورماتور هیبریدی^۶ است که قسمت چهارسیمه تلفن را به دوسیمه

¹ hands-set

² teleconference

³ video conferencing

⁴ Impedance mismatch

⁵ Local Loop

⁶ hybrid

تبدیل می‌کند.

۲-۳-۲ پژواک صوتی

این نوع پژواک در انواع سیستم‌های مخابراتی ایجاد می‌شود که این سیستم‌ها شامل مدارهای رفع پژواک صوتی می‌باشند. مسئله پژواک صوتی، زمانی اتفاق می‌افتد که بلندگو، انعکاس‌های صدا و صداهای محیط و اشیا اطراف را دریافت کند، که در نتیجه مدار صوتی ممکن است ناپایدار شده و صداهای ناهنجاری^۱ ایجاد شود. در یک سیستم مخابراتی عموماً بلندگوها و میکروفون‌ها به گونه‌ای قرار می‌گیرند که روی هم اثر نگذارند.

۲-۴ مسئله تأخیر

در یک تلفن آنالوگ، احتمالاً پژواک صوتی بیشتری نسبت به یک تلفن دیجیتال به وجود می‌آید چون تلفن آنالوگ برای کار در شبکه با تأخیرهای خیلی کم طراحی شده است. در یک مکالمه وقتی تأخیر از حدی بیشتر شود، پژواک شنیده می‌شود. اگر در مدار بیشتر از $30ms$ تأخیر رفت و برگشت وجود داشته باشد، پژواک شنیده می‌شود اگر این تأخیر به $50ms$ برسد، در صورتی که پژواک کنترل نشود، بیشتر مشترکین از مشکل پژواک معترض خواهند شد. اصولاً تأخیر در شبکه‌های مخابراتی به وسیله تجهیزات و وسایل انتقال ایجاد می‌شود مثلاً سوئیچ دیجیتال، باعث تأخیر ناچیزی در شبکه مخابراتی می‌شوند بسته به توپولوژی شبکه و نوع تجهیزات انتقال استفاده شده در شبکه، وقتی تأخیر از $30ms$ بیشتر شود، لازم است که از حذف‌کننده پژواک استفاده شود.

^۱ howling

۲-۵ روش‌های مختلف حذف پژواک

۲-۵-۱ سرکوب کننده‌های پژواک

ساده‌ترین روش، استفاده از سرکوب کننده‌های پژواک^۱ است که یک تلفات متغیر را روی مسیرهای ارسال و دریافت ایجاد می‌کنند (در هنگام ارسال روی مسیر دریافت تلفات ایجاد می‌کنند و برعکس). در صورت حذف پژواک به این روش، یک مخابره دوطرفه واقعی ممکن نمی‌باشد. همچنین این حذف کننده‌ها در محیط‌های نویزی کارکرد مناسبی ندارند.

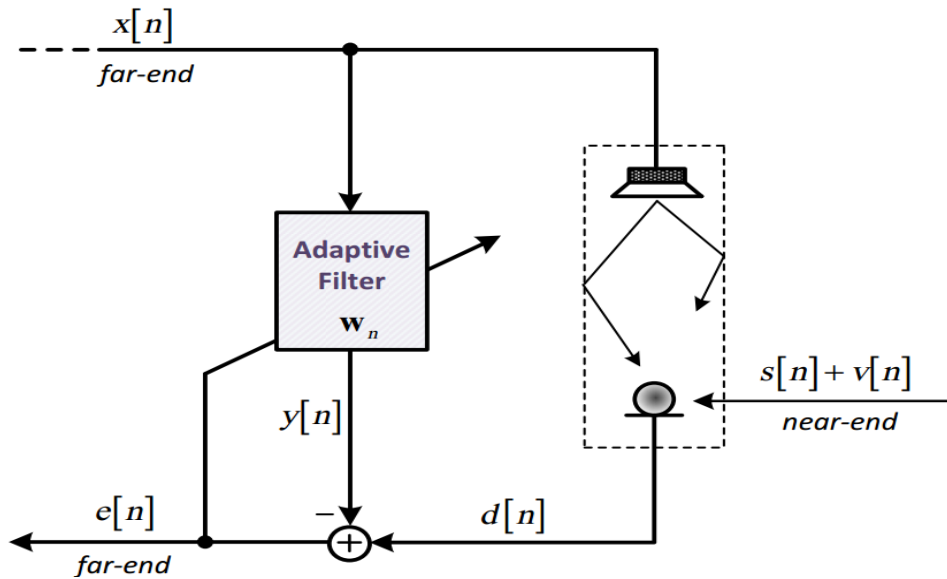
۲-۵-۲ حذف کننده‌های پژواک وقتی

روش دیگر که کاملاً بهتر از روش قبلی است، استفاده از حذف کننده‌های پژواک وقتی^۲ می‌باشد، که به وسیله یک روش فیلتر وقتی، مسیر پژواک صوتی ساخته می‌شود. همان‌طور که در شکل ۲-۱ نشان داده شده است، پژواک صوتی از انتشار صدا از طریق مسیر صوتی بین بلندگو و میکروفون ایجاد می‌شود [۲۱].

چگونگی حذف پژواک صوتی در یک سیستم مخابرات راه دور یک کاناله، در شکل ۲-۲ نشان داده شده است. این نوع ترکیب در یک سیستم کنفرانس از راه دور استفاده می‌شود. مسیر صوتی از بلندگو تا میکروفون باعث می‌شود در شنونده دور، صدای صحبت کننده مانند پژواک تکرار شود [۲۲].

¹ echo suppressors

² Adaptive Echo Canceller



شکل ۲-۲: مدل پردازش حذف کننده پژواک [۷]

۶-۲ آشکارسازی کانال به صورت فعال

یک مشخصه اصلی الگوریتم AEC، آشکارسازی کانال فعال^۱ می باشد. وقتی اپراتور دور^۲ ساکت است و اپراتور نزدیک^۳ صحبت می کند، فیلتر نباید تنظیم شود، چون اپراتور نزدیک، دیگر یک پژواک نیست. آشکارسازی کانال فعال، با محاسبه انرژی سیگنال و سپس مقایسه آن با یک سطح آستانه به دست می آید.

وقتی کانال دریافت غیرفعال است، نرم افزار AEC دیگر پردازش نمی شود یعنی در مدت سکوت، ضرایب فیلتر تنظیم نمی شود همچنین هنگامی که کانال ارسال غیرفعال است و پژواک خیلی کم است، تنظیم فیلتر متوقف می شود که این کار به منظور کاهش محاسبات انجام می شود.

۷-۲ کاربردهای کنترل پژواک

چون با بلندگو و میکروفون های باکیفیت خوب، کل سیستم تقریباً یکنواخت است، مسیر پژواک

¹ active channel detection

² far-end

³ near-end

صوتی از بلندگو تا میکروفون به طور کامل با پاسخ ضربه‌اش مشخص می‌شود. این پاسخ ضربه‌ها از تغییر مشخصات صوتی محیط (باز بودن پنجره، حرکت اشیا، تغییر موقعیت نسبی مکالمه کننده و سیستم، تغییر درجه حرارت و ...) تغییر می‌کنند. مشخصات مسیر پژواک صوتی بستگی به فاکتورهای مختلفی مانند جهت^۱ بلندگو و میکروفون، فاصله بین این دو مبدل، مشخصات صوتی اتاق (زمان انعکاس) دارد. پاسخ ضربه عموماً به عنوان یک مدل اساسی برای اجرای حذف کننده‌های پژواک استفاده می‌شود [۲۳].

در مورد ساختار بهینه فیلترهای حذف پژواک به خاطر پاسخ ضربه طولانی محیط ممکن است یک فیلتر IIR مزایای بیشتری نسبت به فیلتر FIR نشان ندهد [۲۴].

سیستم‌های کنفرانس راه دور و سیستم‌های کنفرانس صوتی، تجهیزاتی هستند که باید کیفیت صدا در آن‌ها به اندازه کافی خوب باشد تا گفتگوها و جلسات به خوبی انجام شوند. بنابراین کنترل پژواک در این تجهیزات باید به دقت صورت گیرد. همچنین سیستم‌های دست‌آزاد در مخابرات در بعضی مواقع مثلاً به دلایل ایمنی در کاربردهایی مانند تلفن‌های موبایل رادیویی، در اتومبیل‌ها کاربرد دارند که در اینجا نیز پژواک صوتی مشکل ساز است.

در ارتباطات ویدئویی برای همزمانی بین حرکت لب‌ها و صحبت، ایجاد یک تأخیر برای جبران تأخیر کدکننده تصویر لازم است، که معمولاً هم بزرگ می‌باشد. بنابراین کنترل پژواک صوتی به خاطر تأخیر در سیستم ضروری است.

چنانچه ذکر شد به علت عدم تطبیق امپدانس در حلقه محلی، پژواک الکتریکی ایجاد شده است باید در نظر گرفت دلیل اصلی ایجاد این نوع پژواک، هیبرید می‌باشد که برای ارتباط بین حلقه محلی

¹ directivity

دوسیمه و ترانک^۱ چهارسیمه^۲ استفاده می‌شود. حلقه‌های محلی دوسیمه برای اتصال خانه‌ها و اماکن مختلف به مراکز تلفن محلی استفاده می‌شوند درحالی‌که ترانک‌های چهارسیمه برای اتصال مراکز تلفن محلی به شبکه‌های راه دور، شبکه‌های تلفن موبایل و شبکه‌های بسته‌ای^۳ استفاده می‌شوند.

هیبرید حلقه محلی دوسیمه را به دو جفت سیم جدا تقسیم می‌کند، یکی برای مسیر ارسال و دیگری برای مسیر دریافت. چون هیبرید نمی‌تواند حلقه دوسیمه را کاملاً جدا کند کمی از سیگنال دریافتی به مسیر ارسال نشت می‌کند و باعث پژواک می‌شود.

برای درک بهتر کاربرد AEC اجازه دهید شرح کوتاهی از پردازشی که توسط یک حذف‌کننده‌ی پژواک صوتی، در مفهوم ارتباط کنفرانس از راه دور بین دو کاربر که در محیط‌های مختلف قرار دارند بگوییم. همان‌طور که از شکل ۲-۲ می‌توان مشاهده کرد در n امین لحظه، سیگنال سخن از کاربر دور می‌آید و همچنین به‌عنوان سیگنال دور شناخته می‌شود و توسط $x[n]$ نشان داده می‌شود به سمت دیگر ارتباط می‌رسد و مجدداً توسط بلندگو تولید می‌شود. در طول باز تولید سیگنال دور، ممکن است توسط غیرخطی‌های بلندگو خراب شود. علاوه بر این، به دلیل شناور بودن ارتباطات گفتاری سیگنال دور باز تولید شده توسط بلندگو، توسط میکروفون‌های واسط کاربر نزدیک تعیین می‌گردند. اتصال صوتی بین میکروفون و بلندگو توسط یک مسیر صوتی مشخص می‌شود که شامل اطلاعاتی درباره ارتعاشات محیطی است.

سیگنالی که توسط بلندگو منتشر می‌شود و توسط میکروفون دریافت می‌شود سیگنال پژواک را نمایش می‌دهد که ممکن است به سهم سیگنال کاربر نزدیک که همان اطلاعات مطلوب برای کاربر دور است اضافه شود. سیگنال نزدیک ترکیب سیگنال صوت نزدیک $S[n]$ به‌اضافه نویز پس‌زمینه

¹ trunk

² 4 wire local loop

³ packet

$v[n]$ است. سیگنال کلی میکروفون معمولاً به عنوان سیگنال مطلوب نامیده می شود و با $d[n]$ نشان داده می شود. در یک زمان یکسان، سیگنال دور $x[n]$ توسط حذف کننده پژواک صوتی پردازش می شود تا AIR بین میکروفون و بلندگو را تخمین بزند. سیگنال خروجی این فرآیند از فیلتر $y[n]$ سیگنال پژواک تخمین زده شده ای را نشان می دهد. و بعداً توسط سیگنال میکروفون $d[n]$ که اطلاعات کاربر نزدیک را نگه می دارد، کم می شود تا تولید سیگنال خطای $e[n]$ که به کاربر دور فرستاده شده پایان یابد.

حذف پژواک صوتی در سیستم های مخابراتی در همه بخش ها اثر گذار است و شامل تخمین پاسخ ضربه، تخمین و حذف اثرات عناصر غیر ایستان در محیط، تخمین و حذف سیگنال های تداخل ناخواسته، تخمین و حذف اثرات پژواک غیر خطی می باشد [۲].

۲-۸ سنجش عملکرد

برای ارزیابی عملکرد الگوریتم های فیلتر وفقی در کاربردهای AEC معمولاً از معیار افزایش تلفات بازگشتی پژواک^۱ یا توان لحظه ای سیگنال پژواک، که منحنی مینیمم مربعات خطا^۲ نامیده می شود، استفاده می کنیم.

۲-۸-۱ افزایش تلفات بازگشتی پژواک

افزایش تلفات بازگشتی پژواک توسط G.168 به صورت "تضعیف سیگنال پژواکی که از مسیر حذف پژواک عبور می کند" تعریف می شود. ERLE از نسبت بر حسب dB توان لحظه ای سیگنال مطلوب $d[n]$ یعنی سیگنال میکروفون بر توان لحظه ای سیگنال پژواک باقیمانده $e[n]$ نتیجه می شود [۲۵]:

¹ Echo Return Loss Enhancement (ERLE)

² Mean Square Error (MSE)

$$ERLE[n] = 10 \log \frac{E\{d^2[n]\}}{E\{e^2[n]\}} \quad (1-2)$$

یک مقدار بزرگ ERLE عملکرد خوب حذف‌کننده پژواک صوتی را نشان می‌دهد، در حالی که یک مقدار کوچک ERLE حضور حائز اهمیت سیگنال پژواک در سیگنال پردازش‌شده را نشان می‌دهد.

۲-۸-۲ توان لحظه‌ای سیگنال پژواک

توان لحظه‌ای سیگنال پژواک که از توان لحظه‌ای سیگنال پژواک باقیمانده $e[n]$ نتیجه می‌شود و برحسب dB است. هرچه مقدار آن کوچک‌تر باشد، شاهد عملکرد بهتر سیستم و حذف پژواک صوتی بیشتری هستید.

$$MSE[n] = 10 \log E\{e^2[n]\} \quad (2-2)$$

۲-۹ جمع‌بندی و نتیجه‌گیری

در این فصل انواع مختلف پژواک و روش‌های مختلف حذف پژواک صوتی و نحوه‌ی مدل ایجاد پژواک در یک کانال و محیط و چگونگی حذف آن توسط فیلترهای فوقی تشریح شد، دو روش موجود برای سنجش میزان حذف پژواک معرفی شد، در این پایان‌نامه از همین معیارها برای سنجش میزان پژواک استفاده شده است.

فصل سوم

فیلترهای افقی برای سیستم

غیرخطی

در [۲۶] یک مدل فیلتر وفقی غیرخطی، بر پایه پیوندهای تابعی معرفی می‌کند که فیلتر وفقی پیوندهای تابعی^۱ نامیده می‌شود و قابلیت مدل کردن غیرخطی‌ها را بر اساس پیوندهای تابعی و خواص الگوریتم فیلترهای وفقی خطی را، به کار می‌گیرد. این مدل، از فرمول بندی اصلی FLANNها متفاوت است اما از ایده اصلی FLANNهایی که اخیراً پیشنهاد شده استفاده می‌کند. (برای مثال [۲۷] و [۲۸] را ببینید) برای جلوگیری از هرگونه ابهامی مدل مجدداً به عنوان FLAF نام‌گذاری شده است. FLAFها از نظر محاسباتی کم هزینه تر از FLANNها و فیلترهای ولترا هستند، لذا در کاربردهای صوتی مانند حذف پژواک غیرخطی، ابزار مؤثری برای مدل‌سازی غیرخطی‌هایی که بر سیگنال صحبت اثر می‌گذارند، به شمار می‌روند.

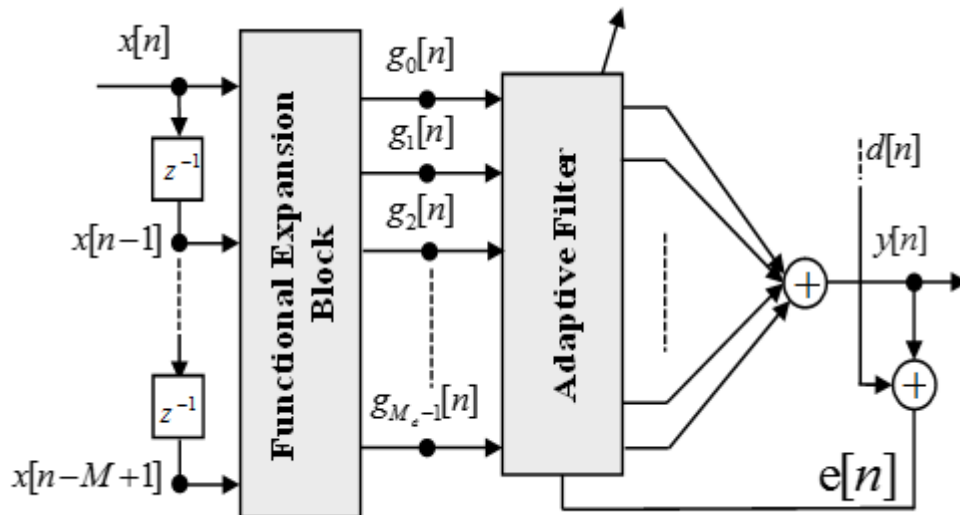
در [۲۶] دو معماری جدید بر پایه FLAF را ارائه می‌کند و برای رسیدگی به برخی از خواص ذاتی AEC پیشنهاد شده است، در ابتدا در [۲۹] ارائه شده‌اند. اولین معماری با نام FLAF مجزا^۲ ارائه شده که در آن مؤلفه‌های خطی و غیرخطی به صورت مجزا تطبیق می‌یابند تا عملکرد همگرایی را بهبود بخشند. دومین طرح که بر پایه FLAF است کمبود نیازهای پیشین در مورد حضور اعوجاج غیرخطی یا نسبت بین توان اعوجاج‌های خطی و غیرخطی را بررسی می‌کند. این قبیل اطلاعات مربوط به انتخاب یک AEC هستند. زیرا هنگامی که یک مسیر پژواک خطی است، یا شامل غیرخطی‌های قابل اغماض است، یک NAEC به خاطر نویز گرادیان وارد شده توسط فیلتر غیرخطی می‌تواند بدتر از یک AEC مرسوم عمل کند [۳۰]. راه‌حل پیشنهادی به این مسئله با نام FLAF اشتراکی^۳ ارائه شده، که الهام گرفته از کارهای اخیر برای فیلترهای وفقی ترکیبی است [۳۱، ۳۲]. فیلترهای ترکیبی، در کاربردهای AEC و نیز دیگر زمینه‌های پردازش سیگنال وفقی استفاده شده‌اند و به گونه‌ای متفاوت از

¹ Functional Link adaptive filter (FLAF)

² SFLAF

³ Collaborative FLAF (CFLAF)

سیستم‌های ترکیبی و معماری‌های اشتراکی ارائه شده اجازه تبادل اطلاعات بین فیلترهای موجود را می‌دهد.



شکل ۳-۱: فیلتر وفقی پیوند تابعی [۲۶]

۲-۳ فیلترهای وفقی پیوند تابعی

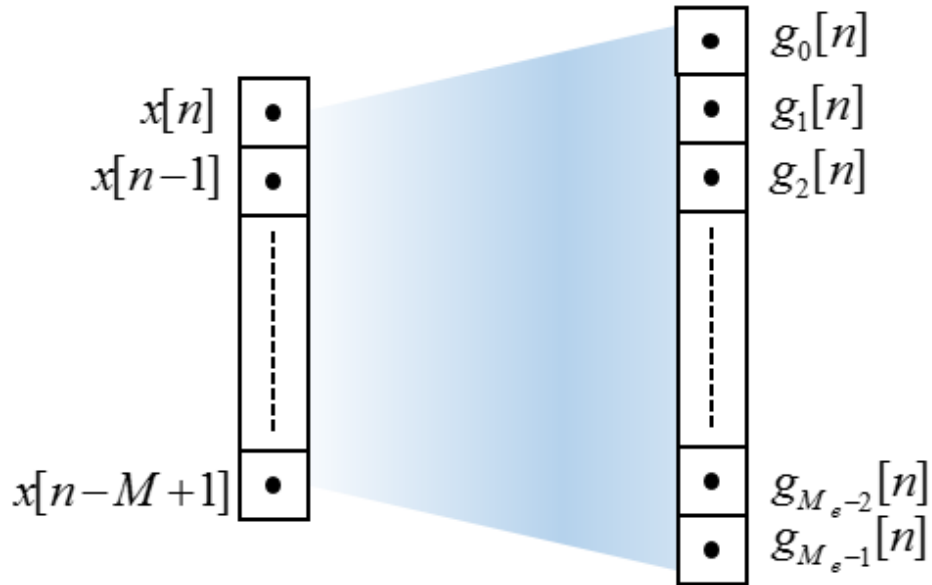
ایده اصلی که رهیافت FLAF را پشتیبانی می‌کند این است که طول سیگنال ورودی اصلی را افزایش می‌دهد و به یک فضا در ابعاد بالاتر می‌برد. این فرآیند به صورت مستقیم از تئوری یادگیری ماشین^۱ برداشت شده است و به صورت دقیق‌تر روی جدایی پذیری الگوها انجام می‌شود [۳۳]. فیلتر وفقی پیوند تابعی در دو مرحله انجام می‌شود، بسط تابعی غیرخطی از ورودی و به دنبال آن، فیلتر کردن خطی که در شکل ۳-۱ ترسیم شده است.

در لحظه n ام، FLAF یک بافر ورودی $\mathbf{x}_n \in \mathbb{R}^M = [x[n] \ x[n-1] \ \dots \ x[n-M]]^T$ را دریافت می‌کند که M طول بافر ورودی است. سپس FLAF، بافر ورودی را به وسیله بلوک بسط تابعی^۲ همانگونه که شکل ۲-۳ می‌بینید پردازش می‌کند. FEB شامل مجموعه‌ای از توابع است که

^۱ Machine learning

^۲ Functional Expantion Block (FEB)

ممکن است زیر مجموعه‌ی کاملی از توابع پایه متعامد^۱ باشد که محدودیت تقریب کلی را برآورده می‌کند [۳۴].



شکل ۳-۲: مدل گسترش پیوندهای تابعی

عبارت پیوندهای تابعی در واقع به توابعی شامل مجموعه انتخابی $\Phi = \{\varphi_0(\cdot), \varphi_1(\cdot), \dots, \varphi_{Q-1}(\cdot)\}$ برمی‌گردد که Q تعداد پیوندهای تابعی است. FEB بافر ورودی را با گذر هر عنصر از بافر \mathbf{x}_n به عنوان تابع ورودی انتخابی پردازش می‌کند و هر یک، زیر بردار $\bar{\mathbf{g}}_{i,n} \in \mathbb{R}^Q$ را نتیجه می‌دهد:

$$\bar{\mathbf{g}}_{i,n} = \begin{bmatrix} \varphi_0(x[n-i]) \\ \varphi_1(x[n-i]) \\ \vdots \\ \varphi_{Q-1}(x[n-i]) \end{bmatrix} \quad (۱-۳)$$

فرآیند تشریح شده برای $i = 0, \dots, M-1$ منجر به یک بافر توسعه‌یافته \mathbf{G}_N می‌شود:

^۱ orthonormal

$$G_N = [\bar{G}_{0,n}^T \bar{G}_{1,n}^T \dots \bar{G}_{M-1,n}^T]^T = [g_0[n] g_1[n] \dots g_{M_e}[n]]^T \quad (2-3)$$

که یک زیر بردار است که اجزای ورودی آن‌ها به‌عنوان $g_m[n]$ با $m = 0, \dots, M_e - 1$ نشانه‌گذاری شده‌اند که در آن $M_e \geq M$ طول بافر توسعه یافته است.

گرچه هیچ اطلاعات جدیدی جایگزین نشده است، ولی فرآیند توسعه، مدل‌سازی غیرخطی را ارتقا می‌دهد. با دستیابی به بافر توسعه‌یافته، فرآیند فیلترهای و فقی پیوند تابعی توسط فیلترهای خطی ساده بافر توسعه یافته، کامل شده است. استفاده از فیلتر و فقی بعد از FEB که در کارهای قبلی [۲۷، ۲۸] پیاده‌سازی شده است یک تفاوت مهم از FLANNها دارد و آن در فرمول بندی اصلی و بعضی استفاده‌های اخیر آن‌ها است [۳۵] و چندین مزیت دیگر FLAF که در بخش‌های بعد شرح داده خواهد شد.

۳-۳ FLAF با بسط مثلثاتی

یک انتخاب مهم در طراحی FEB مربوط به نوع بسط یعنی توابع پایه یا زیر مجموعه‌ای از آن، برای مقداردهی به هر پیوند تابعی است. چون FEB ورودی را با توجه به این تابع بسط می‌دهد، این انتخاب بیشتر بستگی به کاربرد و مخصوصاً بستگی به سیگنال‌های درگیر در پردازش دارد [۲۶].

توابع پایه که محدودیت‌های تخمین کلی را برآورده می‌کند [۳۴] ممکن است یک زیر مجموعه از چند جمله‌ای‌های متعامد^۱ مانند و چند جمله‌ای‌های مثلثاتی^۲ یا فقط توابع تخمین از قبیل توابع گاوسی یا سیگموئید^۳ باشد. در این بین یکی از معروف‌ترین بسط‌های تابعی شامل توابع چند جمله‌ای

¹ Orthogonal polynomials

² Trigonometric Polynomials

³ Sigmoid and Gaussian functions

مثلثاتی است [۳۵, ۲۸] که بهترین نمایش فشرده از هر تابع غیرخطی در مفهوم میانگین مربعات را فراهم می‌کند [۳۶]. همچنین از نظر محاسباتی توابع مثلثاتی ارزان‌تر از چند جمله‌ای‌ها بر پایه سری‌های توانی هستند. به خاطر این خواص ما بسط مثلثاتی را برای مدل کردن سیگنال‌های صوت غیرخطی انتخاب می‌کنیم.

آمین نمونه بافر ورودی را در نظر بگیرید، یک مجموعه کلی از پیوندهای تابعی برای FLAF که با استفاده از بسط پایه مثلثاتی می‌تواند به صورت زیر نوشته شود:

$$\varphi_j(x[n-i]) = \begin{cases} x[n-i], & j = 0 \\ \sin(p\pi x[n-i]), & j = 2p - 1 \\ \cos(p\pi x[n-i]), & j = 2p \end{cases} \quad (3-3)$$

که $p = 1, \dots, P$ نمایه بسط است. و P مرتبه بسط است و $j = 0, \dots, Q - 1$ نمایه لینک تابعی است. که به شاخص لینک تابعی و دنباله نمونه خروجی اشاره دارد که از بسط غیرخطی هر نمونه ورودی، جدای از نوع تبدیلات پیاده‌سازی شده توسط هر پیوند کاربردی، نتیجه شده است. مجموعه پیوند تابعی Φ در این مورد متشکل از $Q = 2P + 1$ پیوند تابعی است که 1 به عنصر خطی $\varphi_0(x[n-i])$ برمی‌گردد که یک کپی از آمین نمونه ورودی فعلی است. در این روش، بافر بسط یافته g_n شامل جفت عناصر خطی و غیرخطی می‌باشد.

۳-۴ FLAF با حافظه و بی حافظه

انتخاب مهم دیگر در طراحی FLAF، مربوط به حافظه بافر ورودی است که روی تعداد نمونه‌های بافر ورودی و نمونه‌های بافر توسعه یافته تأثیر دارد. انتخاب نوع حافظه، به طور جدی به ماهیت سیستم برای شناسایی بستگی دارد و در حقیقت به نوع غیرخطی که سیگنال ورودی را خراب می‌کند، وابسته است. و دقیق‌تر انتخاب نوع حافظه، بستگی به این دارد که غیرخطی لحظه‌ای^۱ است

¹ Instantaneous

(یعنی مستقل از لحظه زمانی) یا اینکه دینامیک^۱ است. (یعنی به لحظه زمانی نیز بستگی دارد).

نوع غیرخطی‌هایی که معمولاً در سیستم‌های صوتی با آن مواجه می‌شوید بی‌حافظه^۲ یا لحظه‌ای است [۱۴]. مدل‌سازی غیرخطی‌های بی‌حافظه بسیار مهم است زیرا بسیاری از سیستم‌های غیرخطی پیچیده می‌توانند به یک سیستم خطی و یک سیستم غیرخطی لحظه‌ای تقسیم شوند. غیرخطی‌های بی‌حافظه نیازمند یک FLAF بی‌حافظه است که یک رابطه مستقیم بین بافر ورودی و بافر توسعه یافته همان‌طور که در شکل ۳-۳ ترسیم شده است، ایجاد کند. مجموعه پیوندهای تابعی که توسط رابطه (۳-۳) شرح داده شده است یک مثال مرسوم از بسط بی‌حافظه است، چون هر پیوند تابعی تنها به \hat{A} آمین نمونه ورودی فعلی بستگی دارد.

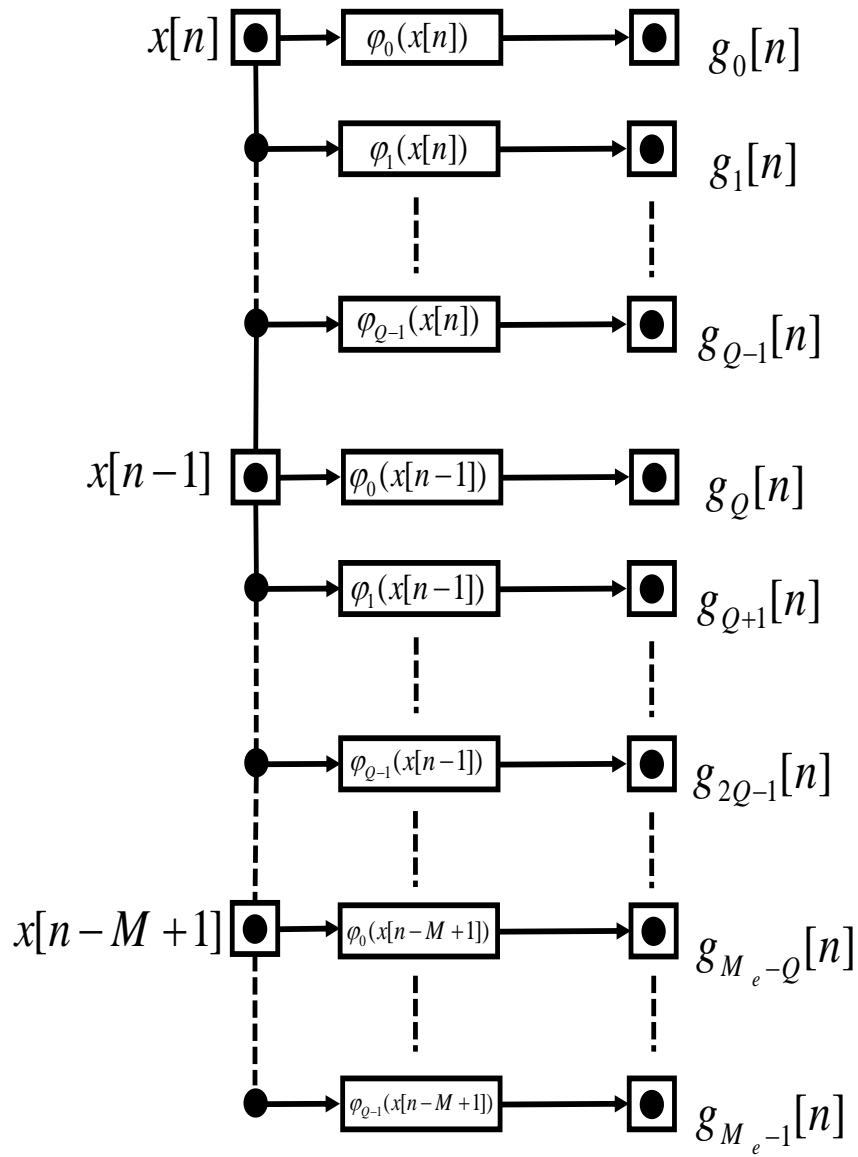
یک مجموعه از پیوندهای تابعی بی‌حافظه، حتی اگر غیرخطی‌ها دینامیک یا لحظه‌ای باشد، یک تقریب قابل قبول از توابع چند متغیر پیوسته فراهم می‌کند. هرچند در مورد سیستم‌های دینامیک غیرخطی ممکن است با وجود انعطاف‌پذیری FEB نتایج بهتری حاصل شود. به‌طور خاص، این امکان‌پذیر است که پیوندهای تابعی بیشتری به ساختار بی‌حافظه اضافه شود تا حافظه یک غیرخطی دینامیک خاص، در نظر گرفته شود.

به این‌گونه مجموعه‌ها $\Phi_k = \{\varphi_0(\cdot), \varphi_1(\cdot), \dots, \varphi_{Q_k-1}(\cdot)\}$ به عنوان پیوندهای تابعی باحافظه^۳ ارجاع می‌دهیم. که K مرتبه حافظه است و $Q_k > Q$ یک عدد کلی از پیوندهای تابعی با حافظه است. یک روش در نظر گرفتن حافظه غیرخطی، در نظر گرفتن ضرب خارجی \hat{A} آمین نمونه ورودی با پیوندهای تابعی از k نمونه ورودی قبلی است. همان‌طور که در شکل ۳-۴ ترسیم شده است.

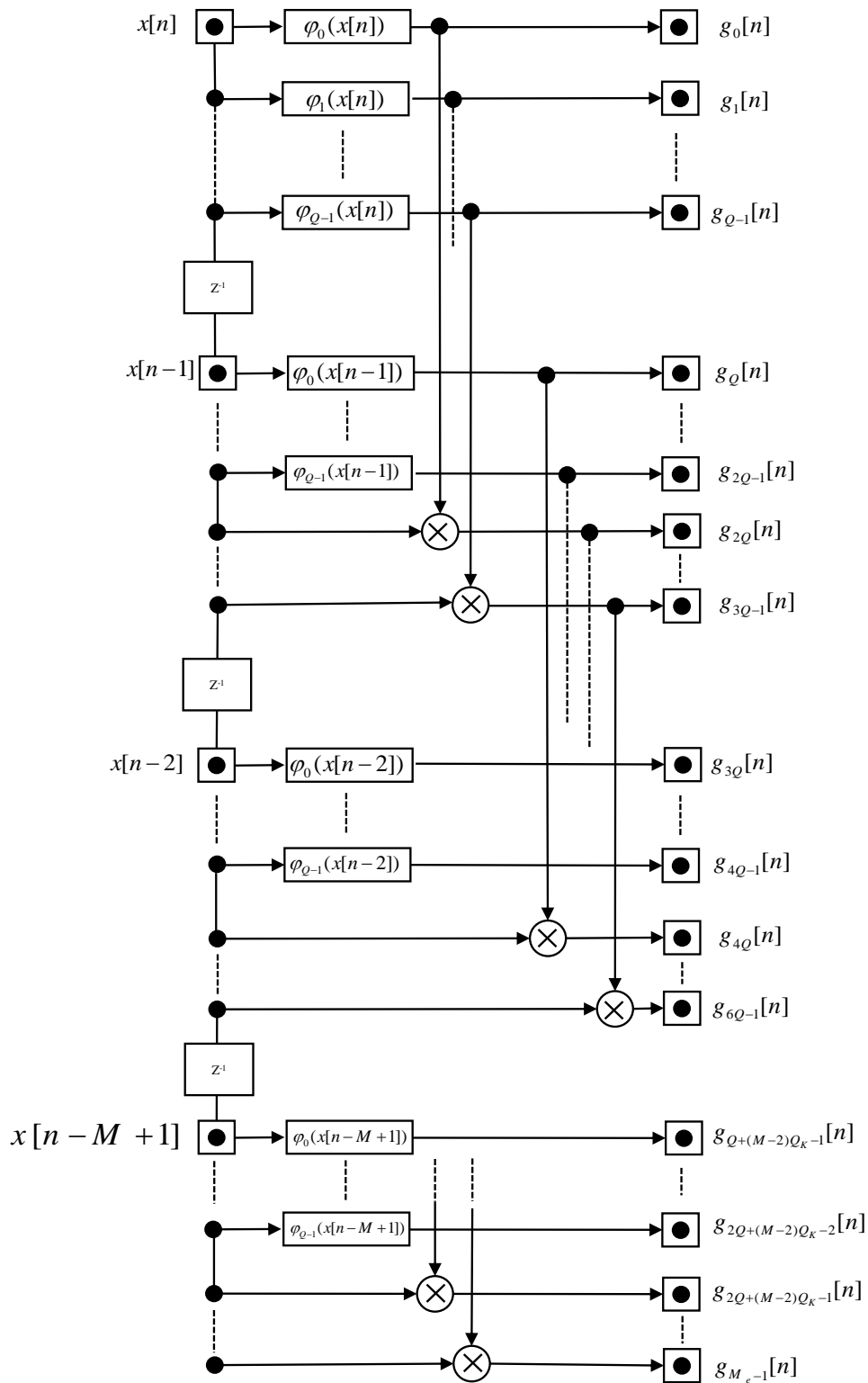
¹ Dynamic

² Memoryless

³ Functiona Link with memory



شکل ۳-۳: بسط تابعی بدون حافظه در FLAF [۲۶]



شکل ۳-۴: بسط تابعی باحافظه دو در FLAF [۲۶]

در یک FLAF باحافظه، مرتبه حافظه k طول پیوندهای تابعی اضافه را تعیین می‌کند. یعنی عمق ضرب بیرونی بین آامین نمونه ورودی و پیوندهای تابعی مرتبط با نمونه‌های ورودی قبلی است. شکل ۳-۴ یک بسط با مرتبه حافظه $k = 2$ را نشان می‌دهد.

۳-۵ تطبیق ضرایب FLAF

زمانی که مجموعه‌ای از توابع پایه انتخاب شدند، مسئله روی پیدا کردن ضرایب بردار وزن FLAF. $\mathbf{w}_n \in \mathbb{R}^{M_e}$ متمرکز می‌شود که به صورت زیر تعریف می‌شود:

$$\mathbf{w}_n = [w_0[n] \ w_1[n] \ \dots \ w_{M_e-1}[n]]^T \quad (۴-۳)$$

برای حصول بهترین تخمین ممکن، از مدل غیرخطی با یک مقدار خطای کوچک.

نمایش صریح سیگنال خطای FLAF، $e[n]$ بدین صورت است:

$$e[n] = d[n] - y[n] = d[n] - \mathbf{G}_n^T \mathbf{w}_n \quad (۵-۳)$$

که کمینه‌سازی آن، به یک تخمین مناسب از وزن‌های فیلتر بستگی دارد. به منظور تنظیم بردار ضریب \mathbf{w}_n ممکن است از هر الگوریتم وفقی بر پایه قاعده گرادیان نزولی^۱ استفاده شود. در [۲۵] از الگوریتم وفقی بر پایه گرادیان تصادفی^۲ برای تطبیق ضرایب فیلتر استفاده می‌شود. با این حال هدف کار انجام‌شده، ارائه معماری بر پایه FLAF جدید است. و رنج وسیعی از الگوریتم‌های وفقی می‌تواند به راحتی برای بدست آمدن ضرایب FLAF استفاده شوند.

استفاده مشترک از پیوندهای تابعی و تطبیق گرادیان تصادفی چند مزیت دارد. اول این‌که، FLAF یک معماری انعطاف‌پذیر بزرگ دارد و این به خاطر بسط غیرخطی مقیاس‌پذیر^۳ و نیز پیچیدگی

¹ Gradient descent

² Stochastic Gradient

³ scalable

ساختاری مقیاس‌پذیر آن است. خاصیت اول اجازه انتخاب یک مجموعه مناسب از پیوندهای تابعی را طبق کاربرد مورد علاقه می‌دهد. همچنین ارائه توابع مرتبه بالا مستلزم توانایی تعمیم‌سازی قوی است [۳۷]. از طرف دیگر، پیچیدگی با قابلیت مقیاس‌پذیر، اجازه می‌دهد تا با سیگنال ورودی، با ابعاد بالا سر و کار داشته باشیم و ساختار FEB را مدل کنیم تا رابطه صحیح بین عملکرد و پیچیدگی محاسباتی را پیدا کنیم که آن هم بر طبق نیازهای کاربردی و منابع محاسباتی موجود است. انعطاف‌پذیری معماری FLAF اجازه یک یکپارچگی ساده از یک سیستم غیرخطی مشخص را می‌دهد. FLAF ممکن است مقداری اشکال هم داشته باشد، که معمولاً به دلیل انعطاف‌پذیری زیاد معماری، که ممکن است باعث یک همگرایی بایاس^۱ شود، است. این قبیل مسائل می‌تواند با معرفی معماری‌های FLAF مقاوم‌تر آدرس‌دهی شود [۳۸].

۳-۶ FLAF های مقاوم برای NAEC

عملکرد FLAF ممکن است در یک مسئله بایاس شده به خاطر این حقیقت که FEB کل بافر ورودی را بسط می‌دهد، موجب خسارت شود. یک کنترل روی بافر بسط یافته گیج‌کننده به نظر می‌آید، چون انتخاب طول بافر ورودی، مقید به تخمین دقیق پاسخ ضربه صوتی است. علاوه بر این، تنظیم پارامترهای بهینه فیلترهای افقی، از قبیل اندازه گام^۲ ممکن است بحرانی باشد. این امر ممکن است در حضور غیرخطی‌های مسیر پژواک که در زمان تغییر می‌کند رخ بدهد، چون معمولاً این یک قانون در AEC است از این جهت، پیشرفت‌هایی می‌تواند حاصل شود که ساختار FLAF را توصیف می‌کند تا به معماری فیلترهای مقاومتری برسد که صراحتاً مناسب کاربردهای NAEC هستند. بر طبق این ما دو معماری بر پایه FLAF را ارائه می‌کنیم:

¹ biased convergence

² step size

✓ FLAF مجزا^۱، انطباق سرعت برای جداسازی عناصر خطی و غیرخطی مسیر پژواک را دارد تا مشکل همگرایی حل شود.

✓ FLAF اشتراکی^۲ مدل سازی کانال صوتی را به صورت قابل تطبیقی بایاس می کند و برایش فرقی ندارد که مسیر از غیرخطی ها تأثیر بگیرد یا نه.

۳-۷ فیلتر وفقی پیوند تابعی مجزا

در مدل سازی کانال صوتی، اولین بهبود مهم در عملکرد FLAF می تواند با جداسازی تطبیق عناصر خطی و غیرخطی در بافر بسط یافته حاصل شود به طور خاص ممکن است که دو فیلتر وفقی مختلف را به صورت موازی در نظر بگیریم، یکی کاملاً خطی و دیگری غیرخطی خالص. در مرجع ها سایر ساختارهای موازی که با پیوندهای تابعی در ارتباطند، ارائه شده اند (برای مثال [۲۹, ۳۵]) اما هیچ کدام از آن ها عناصر خطی و غیرخطی را جدا نکرده اند. در SFLAF فیلتر خطی کل بافر ورودی را دریافت می کند. و منحصرأ قصد دارد که مسیر پژواک را تخمین بزند. از طرف دیگر فیلتر غیرخطی یک بافر بسط یافته تهی از عناصر غیرخطی را دریافت می کند که توسط یک مجموعه غیرخطی خالص از پیوندهای تابعی تولید شده اند:

$$\varphi_j(x[n-i]) = \begin{cases} \sin(p\pi x[n-i]), & j = 2p - 2 \\ \cos(p\pi x[n-i]), & j = 2p - 1 \end{cases} \quad (۶-۳)$$

که در آن، دوباره $j = 0, \dots, Q - 1$ و $p = 1, \dots, P$ است. در این مورد، مجموعه پیوند تابعی متشکل از $Q = 2P$ پیوندهای تابعی است. معادله (۶-۳) یک بسط بی حافظه را توصیف می کند. یک بسط باحافظه می تواند به راحتی با جمع ضرب خارجی عبارات همان طور که در شکل ۳-۴ نشان داده شده است، به دست آید. بنابراین، FLAF غیرخطی تنها روی مدل سازی غیرخطی هایی که روی سیگنال های پژواک تأثیر می گذارند، متمرکز می شود. در این روش ممکن است که بین دو فیلتر

¹ Split FLAF

² Collaborative FLAF

متفاوت، با تنظیم مختلف پارامترها تمایز قائل شد به گونه‌ای که هر فیلتر می‌تواند وظیفه‌اش به بهترین نحو انجام شود.

با استفاده از این ساختار، FEB می‌تواند کل بافر ورودی یا تنها بخشی از آن را دریافت کند. این دلیلی است که چرا طول بافر ورودی غیرخطی با $M_i \leq M$ در نظر گرفته شده است. امکان انتخاب طول بافر بسط یافته برای SFLAF همچنین ممکن است از یک نقطه نظر محاسباتی در مقایسه با FLAF شرح داده شده در بخش‌های قبل FLANN [۲۷، ۳۵، ۳۶، ۳۹] مفید باشد. این معماری که در آن ما به فیلتر وفقی پیوند تابعی مجزا^۱ نسبت می‌دهیم، همان‌طور که در شکل‌های ترسیم شده می‌توان مشاهده کرد، سیگنال خروجی SFLAF از مجموع خروجی‌های فیلتر خطی و خروجی FLAF غیرخطی نتیجه می‌شود:

$$y[n] = y_L[n] + y_{FL}[n] \quad (۷-۳)$$

که در آن

$$y_L[n] = \mathbf{x}_n^T \mathbf{w}_{L,n-1} \quad (۸-۳)$$

$$y_{FL}[n] = \mathbf{g}_n^T \mathbf{w}_{FL,n-1}$$

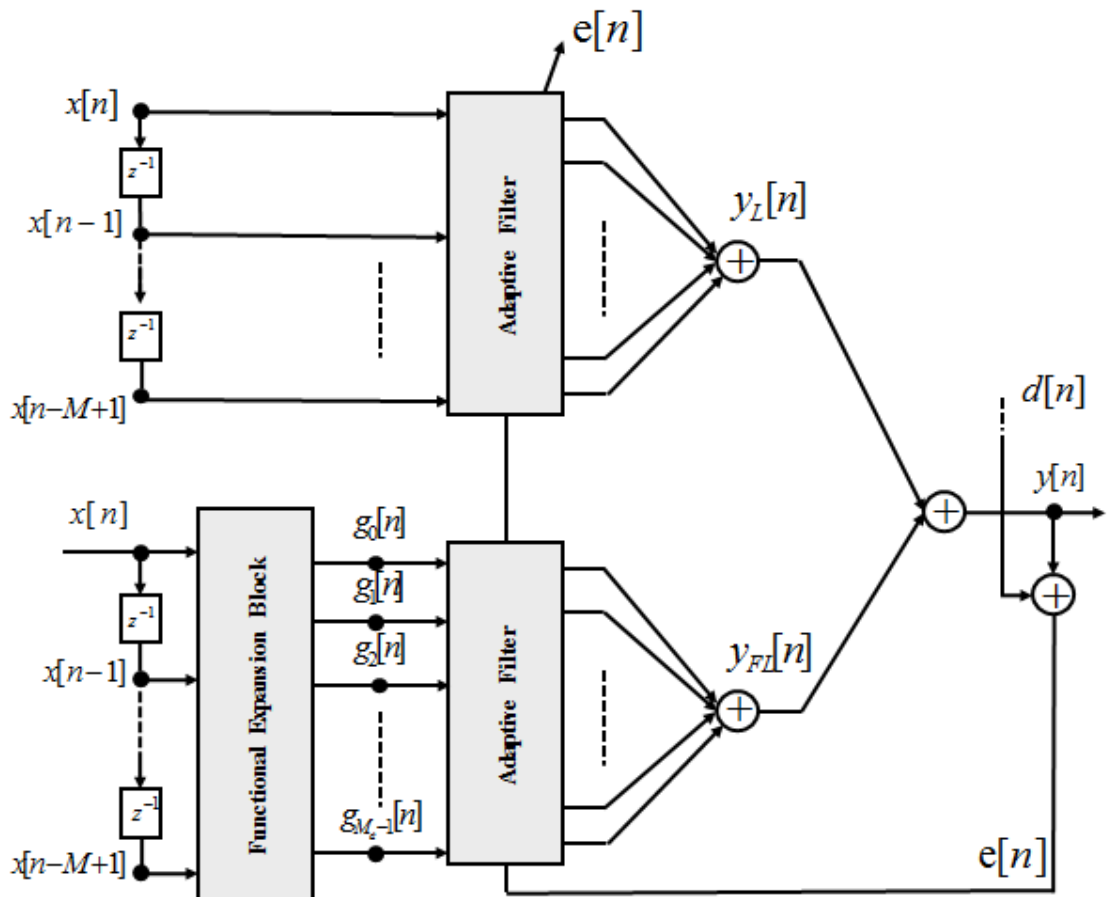
که $\mathbf{w}_{L,n-1} \in \mathbb{R}^M = [w_0[n] \ w_1[n] \ \dots \ w_{M-1}[n]]^T$ بردار ضرایب فیلتر خطی است

و $\mathbf{w}_{FL,n-1} \in \mathbb{R}^M = [w_0[n] \ w_1[n] \ \dots \ w_{M_e-1}[n]]^T$ بردار ضرایب FLAF غیرخطی را نشان

می‌دهد، که هر دو فیلتر خطی و غیرخطی با استفاده از سیگنال خطای کلی $e[n] = d[n] - y[n]$ تطبیق داده شوند. با این حال هر فیلتر می‌تواند با استفاده از یک قانون تطبیق متفاوت و تنظیم پارامترهای مختلف تطبیق داده شود، که ما اینجا با در نظر گرفتن یک NLMS برای تطبیق، هر دو بخش خطی و غیرخطی را تطبیق می‌دهیم و این معماری یک بهبود مهم در مورد عملکرد همگرایی و

^۱ SFLAF

انعطاف پذیری زیاد مهیا می کند که اجازه رسیدن به یک رابطه خوب بین بار محاسباتی و بهبود سیگنال را می دهد.



شکل ۳-۵: فیلتر وفقی پیوند تابعی مجزا [۲۶]

پیچیدگی محاسباتی SFLAF در معرض چندین پارامتر، جدای از الگوریتم وفقی انتخاب شده است. با در نظر گرفتن یک NLMS برای تطبیق هر دو بخش خطی و غیرخطی، الگوریتم SFLAF به $2M + 2$ ضرب و $2M$ جمع برای مسیر خطی احتیاج دارد، درحالی که برای مسیر غیرخطی بدون حافظه $3M_e + PM_i + P + 1$ ضرب، $3M_e + 1$ جمع و $2P$ ارزیابی تابع (یعنی سینوس و کسینوس) مورد نیاز است. این توابع ارزیابی می تواند به صورت مفیدی با جداول بازبینی^۱ پیاده سازی شود. یک مسیر غیرخطی با حافظه نیاز به هزینه اضافی $\sum_{k=1}^{K-1} kP + (M_i - K)KP$ ضرب دارد. همان طور که

^۱ Lookup table

می‌توان مشاهده کرد، پیچیدگی به صورت قابل توجهی بر اساس طول‌های بافر M_i و M_e ، مرتبه بسط P و مرتبه حافظه K تغییر می‌کند.

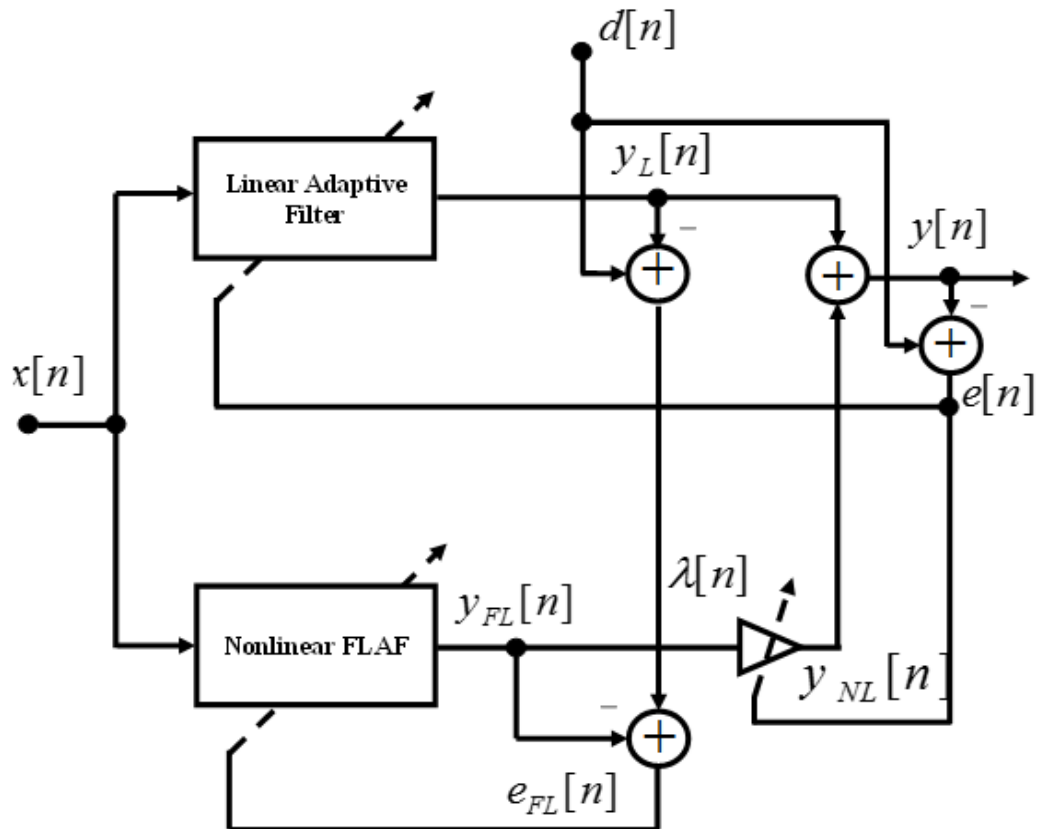
۳-۸ فیلتر وفقی پیوند تابعی اشتراکی

تغییرات پیشنهاد شده در SFLAF به دلیل امکان انتخاب درست برای پارامترهای اصلی فیلتر، به NAEC مقاومت می‌دهد، بعضی معایب ممکن است هنگامی که درجه‌ی غیرخطی در زمان تغییر کند، باقی بماند به‌طور خاص، یک فیلتر غیر بهینه ممکن است زمانی رخ دهد که نسبت توان خطی به غیرخطی^۱ در زمان تغییر کند. هنگامی که مسیر پژواک خطی باشد چون تحت تأثیر غیرخطی نیست یا سطح غیرخطی‌ها قابل اغماض است، عملکرد NAEC ممکن است بدتر از AEC خطی مرسوم باشد. در چنین مواردی، فیلتر غیرخطی تنها مقداری نویز گرادیان در فرآیند فیلتر دارد و بنابراین کاهش عملکرد NAEC را به ارمغان می‌آورد. علاوه بر این، LNLR یک پارامتر ناشناخته و تغییرپذیر با زمان است و برای سیگنال‌های غیر ایستاد، مانند صوت است. به موجب آن یک پارامتر برای دانستن اینکه NAEC حذف کردن پژواک را بهبود می‌بخشد یا تقویت می‌کند ممکن نیست این مسئله، باعث بهای محاسباتی سنگین NAEC می‌شود و روی استراتژی‌های بسیاری از شرکت‌ها که خدمات ویدئو کنفرانس را فراهم می‌کنند، اثر می‌گذارد. و حتی به بهای کیفیت مکالمات معمولاً انتخاب می‌کنند که استفاده از حذف‌کننده‌های پژواک غیرخطی را از قلم بیندازند. باهدف طراحی NAEC مقاوم برای تغییرات LNLR، یک معماری مشارکتی بر پایه ترکیب همگرایی فیلترهای وفقی ارائه شده است [۳۲]. این FLAF های مشارکتی (CFLAF) قابلیت فعال یا غیرفعال کردن خودکار مسیر فیلترهای غیرخطی را دارند. بنابراین تعیین مشخصه‌های عملکردی برای فیلتر وفقی، مدل کردن کانال صوتی بر اساس توان لحظه‌ای باقیمانده مسیر پژواک $e[n]$ است CFLAF در شکل ۳-۶ ترسیم شده است که

^۱ Linear-to-nonlinear power ratio(LNLR)

در آن ممکن است مشاهده شود که سیگنال خروجی کلی $y[n]$ از رابطه (۷-۳) متفاوت است و به صورت زیر بدست می‌آید:

$$y[n] = y_L[n] + \lambda[n]y_{FL}[n] \quad (9-3)$$



شکل ۶-۳: فیلتر وفقی پیوند تابعی اشتراکی [۲۶]

که $\lambda[n]$ یک پارامتر افت^۱ است که اجازه می‌دهد که خروجی FLAF غیرخطی را به صورتی که نیاز است، حفظ کنیم. این بدین معنی است که $\lambda[n]$ در مسیر غیرخطی CFLAF مانند یک سوئیچ هوشمند وفقی است و می‌تواند به صورت یک ترکیب همگرا بین FLAF غیرخطی با یک فیلتر مجازی خنثی تفسیر شود و چون ضرایب آن ایستاد هستند روی صفر تنظیم شده‌اند. در حقیقت بر طبق تابع هزینه انتخابی برای تطبیق پارامترها، $\lambda[n]$ می‌تواند به صورت اتوماتیک فیلتر وفقی را فعال یا غیر

¹ Shrinkage parameters

فعال کند [۵]. به خاطر اینکه فیلترهای خطی و غیرخطی وظایف مختلفی دارند، هر فیلتر در CFLAF با استفاده از سیگنال‌های خطای مختلف بهنگام سازی می‌شود تا از ساختار اشتراکی به طور کامل بهره‌برد. به‌طور خاص، چون سهم خروجی فیلتر خطی همیشه حاضر است وزن‌های فیلتر خطی $w_{L,n}$ کمینه‌سازی سیگنال خطای کلی را $e[n] = d[n] - y[n]$ دنبال می‌کند. علاوه بر این، تخمین AIR هنگامی که سیگنال خطای کلی را به جای سیگنال خطای محلی در نظر می‌گیریم دقیق‌تر نتیجه می‌دهد. به طریقی دیگر، وزن‌های FLAF غیرخطی، $w_{FL,n}$ با استفاده از خطای محلی $e_{FL}[n]$ بهنگام سازی می‌شود که از خروجی خطی $y_L[n]$ کم می‌شود چون همیشه توسط فیلتر خطی در نظر گرفته می‌شود:

$$e_{FL}[n] = d[n] + (y_L[n] + y_{FL}[n]) \quad (۱۰-۳)$$

افت (یا به هم ریختگی) پارامتر $\lambda[n]$ می‌تواند با در نظر گرفتن محدودیت‌های همگرایی یعنی $0 \leq \lambda[n] < 1$ باشد. گرچه انطباق پارامتر کمکی $a[n]$ به وسیله یک تابع سیگموئید، به $\lambda[n]$ مرتبط است [۲۱].

$$\lambda[n] = \text{sgm}(a[n]) = \frac{1}{(1 + e^{-a[n]})} \quad (۱۱-۳)$$

$\lambda[n]$ با تطبیق $a[n]$ طبق قانون گرادیان نزولی به صورت $a[n+1] = a[n] + \Delta a[n]$ محاسبه می‌شود که $\Delta a[n]$ از نرمالیزه کردن انطباق حداقل میانگین مربعات نتیجه می‌شود:

$$\begin{aligned} \Delta a[n] &= -\frac{1}{2} \mu_a \frac{\partial e^2[n]}{\partial a[n]} \\ &= -\frac{\mu_a}{r[n]} e[n] \frac{\partial (d[n] - y_L[n] - \lambda[n] y_{FL}[n])}{\partial \lambda[n]} \frac{\partial \lambda[n]}{\partial a[n]} \quad (۱۲-۳) \\ &= \frac{\mu_a}{r[n]} e[n] y_{FL}[n] \lambda[n] (1 - \lambda[n]) \end{aligned}$$

که در آن

$$r[n] = \beta r[n] + (1 - \beta)y_{FL}^2[n] \quad (13-3)$$

یک تخمین فیلتر شده پایین گذر دشوار از توان سیگنال مورد علاقه است [۴۰]. پارامتر β یک فاکتور هموارسازی است که تضمین می‌کند $r[n]$ از هر جز فیلتر سریع‌تر است مقدار $a[n]$ به دلایل عملی بین $[-۴,۴]$ نگه داشته می‌شود [۳۲].

به طور کل می‌توان نتیجه گرفت که، معماری بر پایه FLAF اشتراکی، چون جدای از ساختارهای مرکب است [۳۱، ۴۱]، اجازه تبادل اطلاعات بین فیلترهایی درگیر را می‌دهد. همان‌طور که در شکل ۳-۶ مشاهده می‌شود، سیگنال خطایی که با آن فیلتر خطی منطبق می‌شود، ممکن است به خاطر رابطه (۳-۹) شامل سهم خروجی فیلتر غیرخطی باشد. از طرف دیگر خطایی که با آن FLAF غیرخطی منطبق می‌شود همیشه شامل سهم خروجی خطی است همان‌طور که در رابطه (۳-۱۰) نشان داده شده است. CFLAF ارائه شده برای هر سطح غیرخطی مقاوم است چون هنگامی که مسیر پژواک فقط خطی است $\lambda[n]$ به صفر همگرا می‌شود و طرح کلی مانند یک فیلتر خطی خالص رفتار می‌کند بنابراین از هر نویز گرادیانی از FLAF غیرخطی جلوگیری می‌شود. از طرف دیگر، هنگامی که مسیر پژواک غیرخطی‌ها را منتقل می‌کند پارامتر افت بر طبق سطح غیرخطی در مسیر پژواک به یک می‌رسد. توجه کنید که هنگامی که $\lambda[n] = 1$ باشد معماری CFLAF مانند SFLAF کار می‌کند. با توجه به پیچیدگی محاسباتی، CFLAF یک اضافه‌بار با توجه به SFLAF فقط ده تا ضرب، پنج تا جمع و یک تابع تخمین اعمال می‌کند.

۳-۹ FLAF با بسط‌های پیشنهادی

همان‌گونه که گفته شد یک انتخاب مهم در طراحی پیوندهای تابعی مربوط به نوع بسط، یعنی توابع پایه یا زیرمجموعه‌ای از آن، برای مقداردهی به هر پیوند تابعی است. این انتخاب بیشتر بستگی به کاربرد و به‌ویژه بستگی به سیگنال‌های درگیر در پردازش دارد.

ما در اینجا برای توابع پایه که محدودیت‌های تخمین کلی را برآورده می‌کند چبیشف^۱ و لژاندر^۲ در نظر گرفته‌ایم.

۳-۹-۱ FLAF با بسط چبیشف

توابع چند جمله‌ای چبیشف به عنوان ابزاری قدرتمند با قابلیت برآورد غیرخطی شناخته شده‌اند [۴۲]. به همین دلیل در زمینه‌های مختلف کاربرد گسترده‌ای دارند. به‌طور خاص، چند جمله‌ای چبیشف به‌طور گسترده در طبقه بندی الگو^۳ [۴۳] و نیز در مسائل تقریب توابع^۴ [۴۲] استفاده می‌شود. برای مثال یک شبکه عصبی مصنوعی با بسط چند جمله‌ای چبیشف دارای قابلیت تقریب عمومی و همگرایی سریع‌تری نسبت به یک شبکه چندلایه^۵ (MLP) است. علاوه بر این، چند جمله‌ای چبیشف در ساختار FLANN برای مسئله شناسایی سیستم‌های دینامیکی^۶ غیرخطی، با وجود نویز در ورودی سیستم مورد استفاده قرار گرفته شده، که یک اثر مقاوم را نشان می‌دهد [۴۴]. همچنین، FLANN با استفاده از بسط چبیشف در برابر سازی کانال [۴۵, ۴۶] استفاده می‌شود.

اثر چند جمله‌ای چبیشف عمدتاً به این دلیل است که بسط چبیشف شامل ورودی‌های توابعی از توابع قبلی است. علاوه بر این، بسط چبیشف بر اساس بسط سری توانی است، که ممکن است یک تابع غیرخطی را با یک خطای بسیار کوچک در نزدیکی نقطه بسط^۷ تقریب بزند. با این حال، جدای از نقطه بسط، خطا به سرعت افزایش می‌یابد [۴۷]. با توجه به دیگر سری‌های توانی با همان درجه، چند جمله‌ای‌های چبیشف کاملاً از لحاظ محاسباتی، ارزان و کارآمدتر هستند [۴۲]، و به همین دلیل است که غالباً برای تقریب تابع استفاده می‌شوند. اگرچه، هنگامی که سری توانی به آرامی همگرا شود هزینه

¹ Chebyshev

² Legendre

³ Pattern classification

⁴ functional approximation

⁵ multi layer perceptrons

⁶ dynamic

⁷ point of expansion

محاسباتی به طور چشمگیری افزایش پیدا می کند. ولی توابع چبیشف نسبت به توابع چند جمله‌ای مثلثاتی برای محاسبه آسان تر هستند.

با در نظر گرفتن آامین ورودی $x[n - i]$ ، بسط چند جمله‌ای چبیشف را می توان به صورت زیر

نوشت:

$$g_j = \begin{cases} 1, & j = 0 \\ x[n - i] & j = 1 \\ 2x^2[n - i] - 1, & j = 2 \\ 2x[n - i]g_{j-1}[n] - g_{j-2}[n], & j = 3, \dots, Q - 1 \end{cases} \quad (۱۴-۳)$$

۳-۹-۲ FLAF با بسط لژاندر

توابع چند جمله‌ای لژاندر، مشابه چند جمله‌ای چبیشف، برای عملکرد بهتر از پیوندهای تابعی برای محاسبات استفاده می کند [۳۹]. توابع چند جمله‌ای لژاندر به طور گسترده‌ای برای تقریب تابع با استفاده از شبکه عصبی مصنوعی متعامد [۴۸] و همچنین پیوندهای تابعی بر اساس شبکه عصبی مصنوعی [۴۹] استفاده می شود. اکولایزر مدولاسیون دامنه متعامد^۱ بر اساس لژاندر از اکولایزر تابع پایه‌ای شعاعی (RBF) خطی عملکرد بهتری دارند [۳۹]. با این حال، عملکرد آن شبیه اکولایزر بر اساس چبیشف [۵۰] است.

با در نظر گرفتن آامین ورودی، $x[n - i]$ بسط چند جمله‌ای لژاندر را می توان به صورت زیر

نوشت:

$$g_j = \begin{cases} 1, & j = 0 \\ x[n - i] & j = 1 \\ (3x^2[n - i] - 1)/2, & j = 2 \\ \frac{\{(2j - 1)x[n - i]g_{j-1}[n] - (j - 1)g_{j-2}[n]\}}{j}, & j = 3, \dots, Q - 1 \end{cases} \quad (۱۵-۳)$$

^۱ Quadrature Amplitude Modulation (QAM)

۳-۹-۳ فیلتر وقتی پیوند تابعی مجزای موازی

در کارهای انجام شده تا اینجا ما از یک فیلتر خطی NLMS و یک فیلتر غیرخطی از مدل SFLAF استفاده کردیم حال ما از دو فیلتر غیرخطی، به همزمان برای ورودی غیرخطی استفاده می‌کنیم. در این مدل همانگونه که در شکل ۳-۷ مشخص است، ما به جای فیلتر خطی یک فیلتر غیر خطی قرار می‌دهیم یعنی ما ورودی به ازای سیستم خطی را هم، توسط پیوند های تابعی بسط می‌دهیم.

این معماری که به آن فیلتر وقتی پیوند تابعی مجزا موازی^۱ نام می‌دهیم، همان‌طور که در شکل ۳-۷ می‌توان مشاهده کرد که سیگنال خروجی PSFLAF از مجموع خروجی‌های فیلترهای غیرخطی نتیجه می‌شود:

$$y[n] = y_{FL1}[n] + y_{FL2}[n] \quad (۱۶-۳)$$

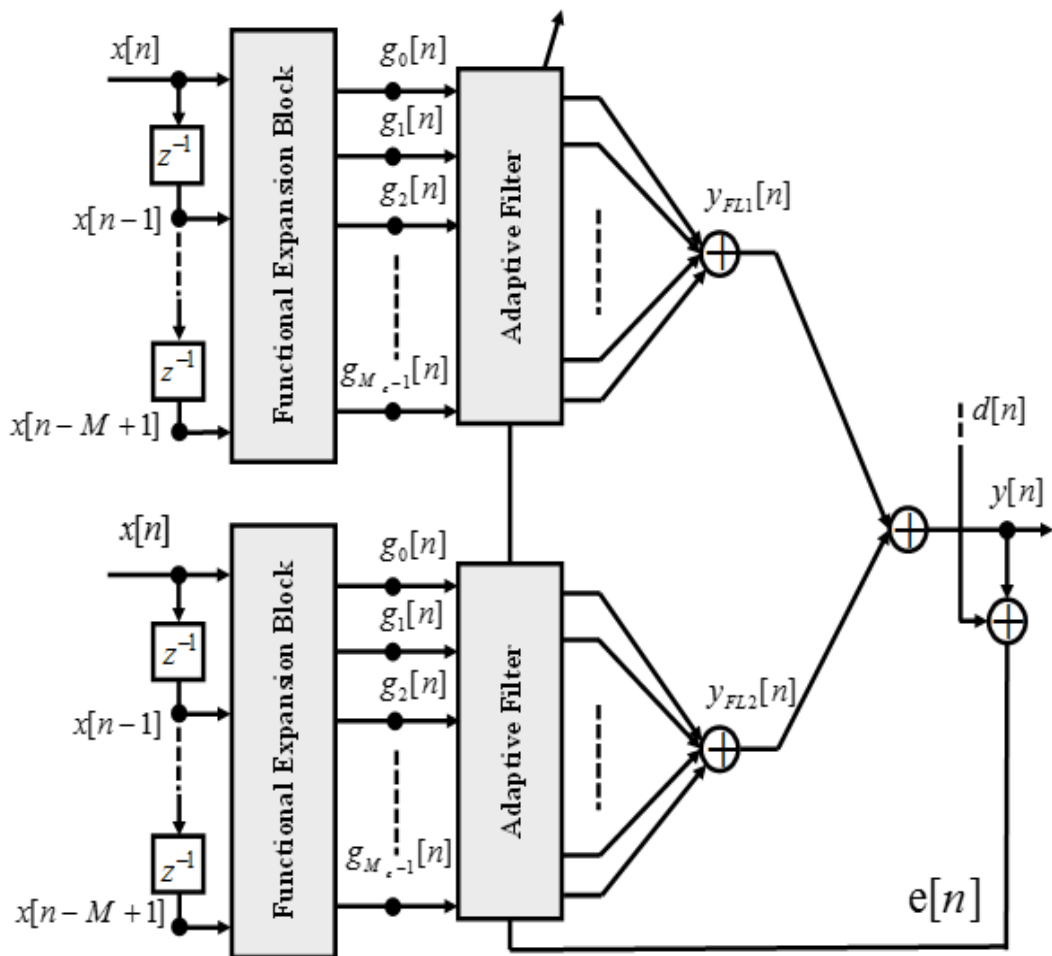
که در آن

$$y_{FL1}[n] = \mathbf{g}_n^T \mathbf{w}_{FL1,n-1} \quad (۱۷-۳)$$

$$y_{FL2}[n] = \mathbf{g}_n^T \mathbf{w}_{FL2,n-1}$$

که در اینجا هم مانند بخش‌های قبل از یک NLMS برای تطبیق هر دو بخش غیرخطی استفاده می‌کنیم و عملکرد این معماری یک بهبود مهم در مورد حذف پژواک غیرخطی است که آزمایش‌های انجام شده در فصل بعد نحوه عملکرد را نشان می‌دهد.

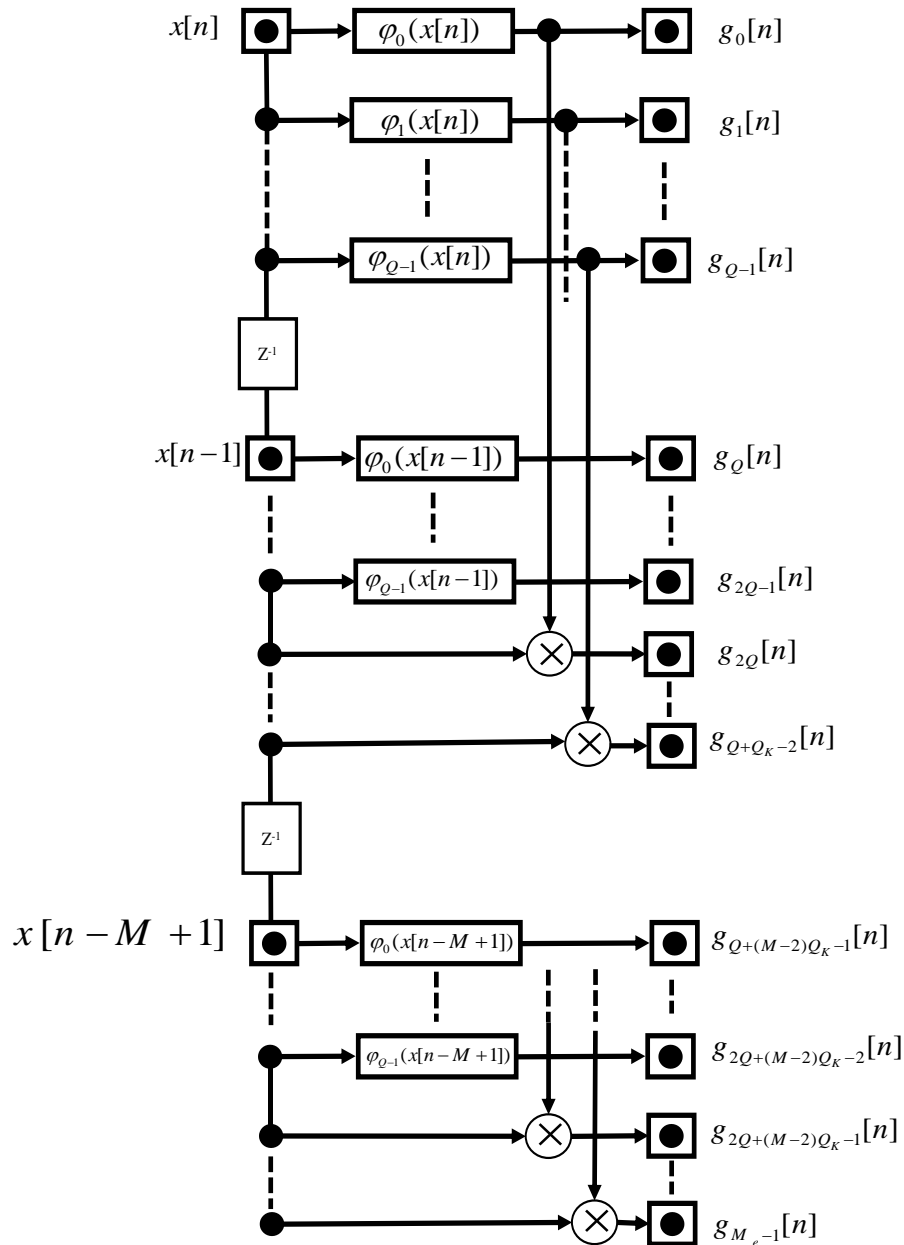
^۱ Parallel SFLAF (PSFLAF)



شکل ۳-۷: فیلتر وقفی پیوند تابعی مجزای موازی

۳-۱۰ طراحی پیوند تابعی با حافظه مرتبه یک

انعطاف‌پذیری، یکی از مهم‌ترین خصیصه‌های معماری بر پایه FLAF است. حقیقت امر این است که طراحی FEB ممکن است در بهترین غیرخطی‌ها در مسیر پژواک مدل شود، همان‌طور که در بخش ۲-۳ توضیح داده شده است. در حقیقت هنگامی که غیرخطی‌ها در مسیر پژواک کاملاً دینامیک هستند ممکن است که یک بسط تابعی با حافظه انتخاب کنیم. برای داشتن یک ارزیابی واضح از مزایا و معایب، با استفاده از بسط پیوند تابعی با حافظه نسبت به نوع بی حافظه ما مدل با حافظه مرتبه یک را طراحی و بررسی نمودیم.



شکل ۸-۳: بسط تابعی با حافظه یک

شکل ۸-۳ نشان دهنده یک FEB مرتبه یک است. با توجه به عملکرد بسط پیوند تابعی با حافظه نسبت به نوع بی حافظه آن ما یک سناریو دینامیک که توسط رابطه (۴-۱) و (۴-۴) شرح داده شده را در نظر می‌گیریم. ما انواع FEB های موجود و همچنین مجموعه قبلی از آزمایش‌ها را با استفاده از همین تنظیمات محاسبه می‌کنیم تا یک مقایسه منصفانه داشته باشیم.

۳-۱۱ جمع‌بندی و نتیجه‌گیری

در این فصل به مدل جدیدی از حذف کننده‌های پژواک غیرخطی که آن را فیلتر وفقی پیوند تابعی می‌نامند، پرداخته شد. ابتدا مدل با پیوند مثلثاتی و مدل‌های مختلف باحافظه و بی‌حافظه پیوندهای تابعی را معرفی کردیم، در ادامه چگونگی عملکرد فیلتر وفقی پیوند تابعی مجزا و پیوند تابعی اشتراکی را تشریح کردیم که برای سیستم‌های غیرخطی اهمیت زیادی دارند. همچنین مدل حذف پژواک باحافظه یک و مدل‌های پیوند تابعی چبیشف و لژاندر را پیشنهاد دادیم.

فصل چهارم

شبیه‌سازی حذف پژواک غیرخطی

۱-۴ مقدمه

در این فصل شبیه‌سازی عملکرد الگوریتم‌های حذف پژواک غیرخطی ارائه شده است و برای حالت‌های مختلف ورودی صدای صوت واقعی و ورودی رنگی مورد مقایسه قرار می‌گیرد همچنین نشان می‌دهیم که طول پیوندهای تابعی چه اثری بر حذف پژواک دارد و به بررسی مدل باحافظه و بی‌حافظه الگوریتم‌ها می‌پردازیم سپس نتایج به دست آمده از شبیه‌سازی‌ها را تحلیل می‌کنیم.

۲-۴ نتایج تجربی

در این بخش عملکرد معماری‌های مختلف بر پایه پیوندهای تابعی را در سناریوی NAEC ارزیابی می‌کنیم. مجموعه آزمایش‌ها با توجه به ارتباط در محیط تله‌کنفرانس شبیه‌سازی شده و AIR بین بلندگو و میکروفون به وسیله جعبه ابزار Roomsimsim متلب با نرخ نمونه برداری ۸ kHz شبیه‌سازی شده است [۵۱]. پاسخ ضربه صوتی بعد از نمونه $M=300$ بریده می‌شود که برابر با طول فیلترهای خطی و غیرخطی نیز هست. برای شبیه‌سازی یک اعوجاج بلندگوی نامتقارن، سیگموئید غیرخطی بی‌حافظه زیر را به سیگنال دور، پیش از کانولوشن شدن با AIR اعمال می‌کنیم:

$$\bar{y}[n] = \gamma \left(\frac{1}{1 + e^{(-\rho q[n])}} - \frac{1}{2} \right) \quad (1-4)$$

که در آن

$$q[n] = \frac{3}{2}x[n] - \frac{3}{10}x^2[n] \quad (2-4)$$

در رابطه (۱-۴) پارامتر γ بهره سیگموئید است و $\gamma = 2$ تنظیم شده است درحالی‌که ρ شیب

سیگموئید را نشان می‌دهد و به صورت زیر انتخاب می‌شود:

$$\rho = \begin{cases} 4, & q[n] > 0 \\ \frac{1}{2}, & q[n] \leq 0 \end{cases} \quad (3-4)$$

با هدف شبیه‌سازی سیستم‌های پژواک غیرخطی دینامیک، $q[n]$ در رابطه (۴-۱) را با یک رابطه غیرخطی مرتبه پنجم پراکنده جایگزین کردیم [۲۶].

$$\begin{aligned} q_d[n] = & \frac{3}{2}x[n] - \frac{3}{10}x^2[n] + \frac{9}{5}x[n]x[n-1] \\ & + \frac{1}{2}x[n]x[n-2] - \frac{2}{5}x[n]x[n-3] \\ & - \frac{3}{2}x[n-1]x[n-2] + \frac{9}{10}x[n]x[n-1]x[n-3] \\ & + \frac{1}{2}x[n-1]x[n-2]x[n-3] - \frac{1}{10}x^2[n-1] \\ & + \frac{1}{5}x^2[n-2] - \frac{1}{10}x^2[n-3] + \frac{3}{10}x[n-3]x[n-4] \\ & - \frac{6}{5}x^2[n-4] + \frac{1}{5}x[n-1]x[n-5] \\ & + \frac{3}{10}x[n-3]x[n-5] + \frac{6}{5}x^2[n-5] \end{aligned} \quad (4-4)$$

دو نوع سیگنال ورودی دور استفاده شده است یک نویز رنگی و یک ورودی گفتار خانم، سیگنال رنگی به وسیله مدل برگشتی خودکار مرتبه اول که تابع انتقال آن رابطه (۴-۵) و با $\theta = 0.8$ است تولید می‌شود و توسط یک فرآیند تصادفی گاوسی با توزیع مستقل و متقارن (i.i.d) تغذیه می‌شود، در هر دو مورد، نویز سفید گاوسی در خروجی مسیر پژواک که تا ۲۰ dB از نسبت سیگنال به نویز^۱ است جمع می‌شود و طول مدت آزمایش ۱۰ ثانیه است عملکرد همه معماری‌ها از طریق تلفات بازگشتی پژواک (ERLE) و توان لحظه‌ای سیگنال پژواک (MSE) ارزیابی می‌شود.

¹ Signal-to-Noise Ratio (SNR)

$$\frac{\sqrt{1 - \theta^2}}{1 - \theta z^{-1}} \quad (5-4)$$

از یک فیلتر میانگین لغزشی^۱ برای مسطح سازی خمیدگی‌ها استفاده می‌کنیم به دلیل وجود زیاد تعداد نمونه‌ها در سیگنال صوتی منحنی حذف پژواک را تا مرتبه ۳۰۰۰ فیلتر نرم می‌کنیم که دارای وضوح دید برای مشاهده حذف پژواک باشد و سپس با توجه به آن منحنی مورد نظر را ترسیم می‌کنیم.

^۱ Moving Average

۴-۳ ارزیابی اول

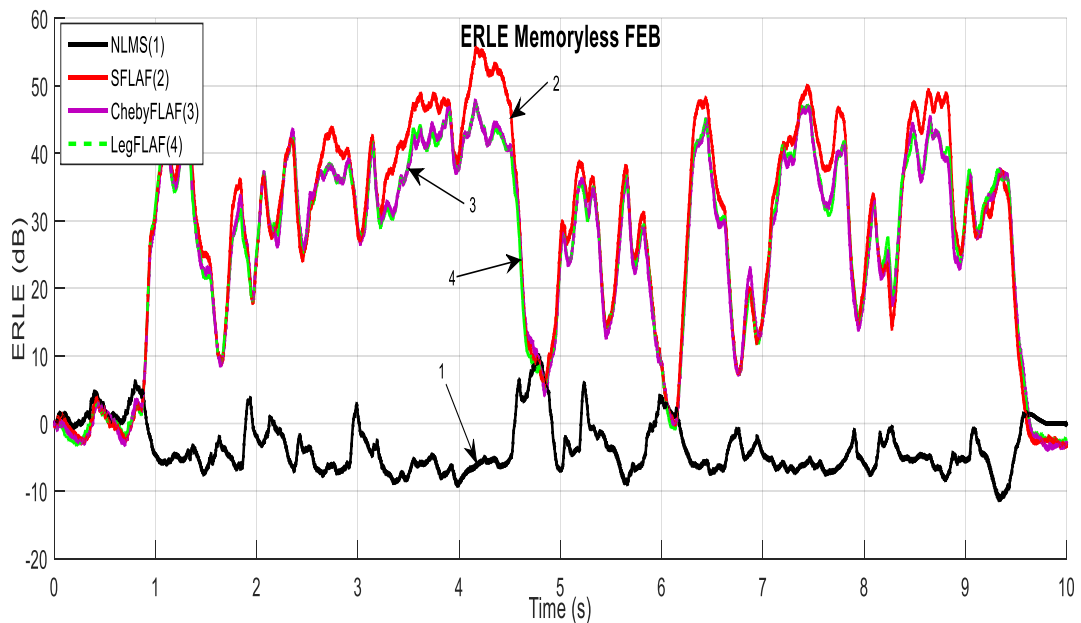
در این بخش ما عملکرد الگوریتم‌های مثلثاتی، چبیشف و لژاندر را برای حالتی که طول پیوند تابعی در دو حالت $Q = 1$ و $Q = 10$ می‌باشد را بررسی می‌کنیم. همچنین نشان می‌دهیم این الگوریتم‌ها نسبت به فیلتر وفقی حالت خطی در حالت پژواک غیرخطی عملکرد بسیار بهتری دارد در این آزمایش ما از تنظیم پارامترهای یکسانی برای تمام این ارزیابی استفاده می‌کنیم و طول بافر ورودی برای همه الگوریتم‌ها $M = 300$ در نظر گرفته شده است اندازه گام فیلتر خطی $\mu_L = 0.2$ برای بهره‌برداری از مزیت داشتن دو فیلتر جداگانه SFLAF ما یک پارامتر اندازه گام متفاوت برای فیلتر غیرخطی SFLAF مساوی با $\mu_{FL} = 0.5$ را در نظر می‌گیریم. برای داشتن یک مقایسه منصفانه ما از یک سناریوی بی حافظه استفاده می‌کنیم که رابطه (۴-۱) و رابطه (۴-۲) را درگیر می‌کند. ما معماری بی حافظه را برای حالت‌های مختلف FEB در FLAF به دست می‌آوریم. طول سیگنال مورد آزمایش ۱۰ ثانیه است، عملکرد معماری‌های ذکر شده بالا را در عبارت ERLE و MSE هم برای ورودی رنگی و هم ورودی صوت یک خانم مقایسه می‌کنیم.

نتایجی که در شکل ۴-۱، شکل ۴-۲، شکل ۴-۳ و شکل ۴-۴ ترسیم شده است که در آن $Q=1$ در نظر گرفته شده، بهبود عملکرد، توسط SFLAF در مقایسه با معماری فیلترهای دیگر آشکار است هرچند در مورد ورودی صوت بهبود کمتر واضح است چون غیرخطی‌ها روی سیگنال در یک راه یکنواخت اثر نمی‌گذارد و این امر به خاطر طبیعت تغییرپذیر سیگنال صوت است. همان‌گونه که در منحنی‌ها معلوم است پیوند تابعی چبیشف و لژاندر، در این حالت دارای مقادیر ثابتی هستند و منحنی آن‌ها کامل بر روی هم قرار گرفته است. نوسانات منحنی‌های ورودی رنگی بسیار کم است و به دلیل انتخاب شدن اعداد به صورت تصادفی ممکن است در منحنی ERLE دارای مقداری تداخل باشد که موجب خطای دید شود با رسم منحنی MSE به صورت واضح‌تر می‌توان تفاوت الگوریتم‌ها را دید.

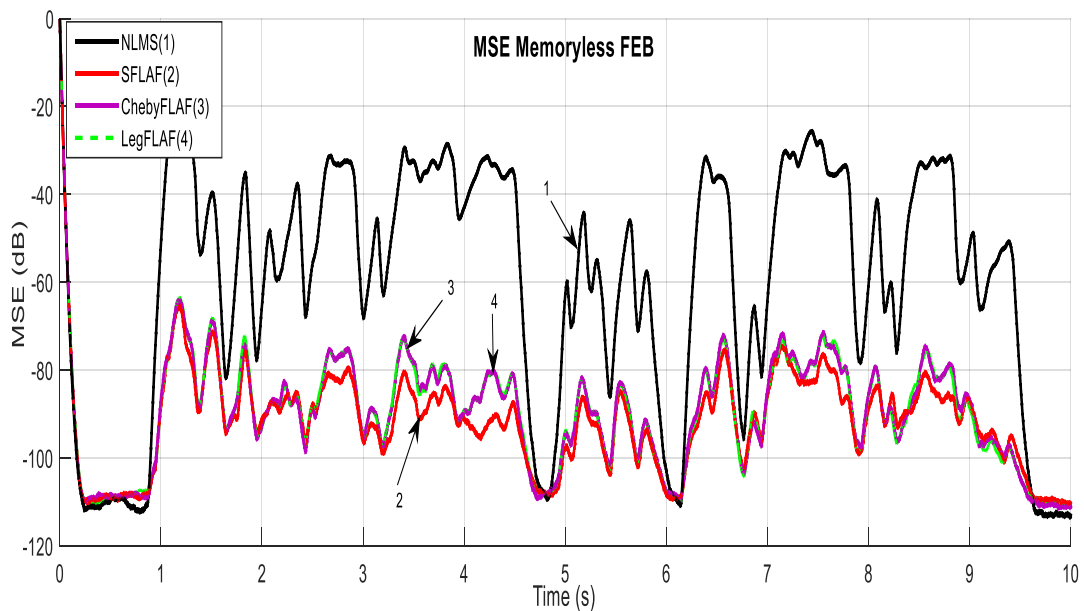
نتایجی که در شکل ۴-۵، شکل ۴-۶، شکل ۴-۷ و شکل ۴-۸ ترسیم شده است، در آن $Q=10$ نظر گرفته شده، بهبود عملکرد الگوریتم چبیشف را در مقایسه با معماری فیلترهای دیگر می بینیم این به این معنا است که با بالا رفتن تعداد پیوندهای تابعی چبیشف بهبود قابل ملاحظه‌ای دارد این امر یعنی افزایش تعداد پیوند تابعی در تابع لژاندر هیچ بهبودی را موجب نشد و تابع مثلثاتی با توجه به آزمایش دارای اندکی بهبود شد همان گونه که مشخص است بهبود عملکرد الگوریتم‌های پیشنهادی در بعضی نواحی منحنی ERLE بین ۵۰dB تا ۶۰dB دیده می شود و میزان حذف پژواک در حالت خطی بسیار کمتر از الگوریتم‌های پیشنهادی است.

جدول ۴-۱: الگوریتم بر پایه FLAF

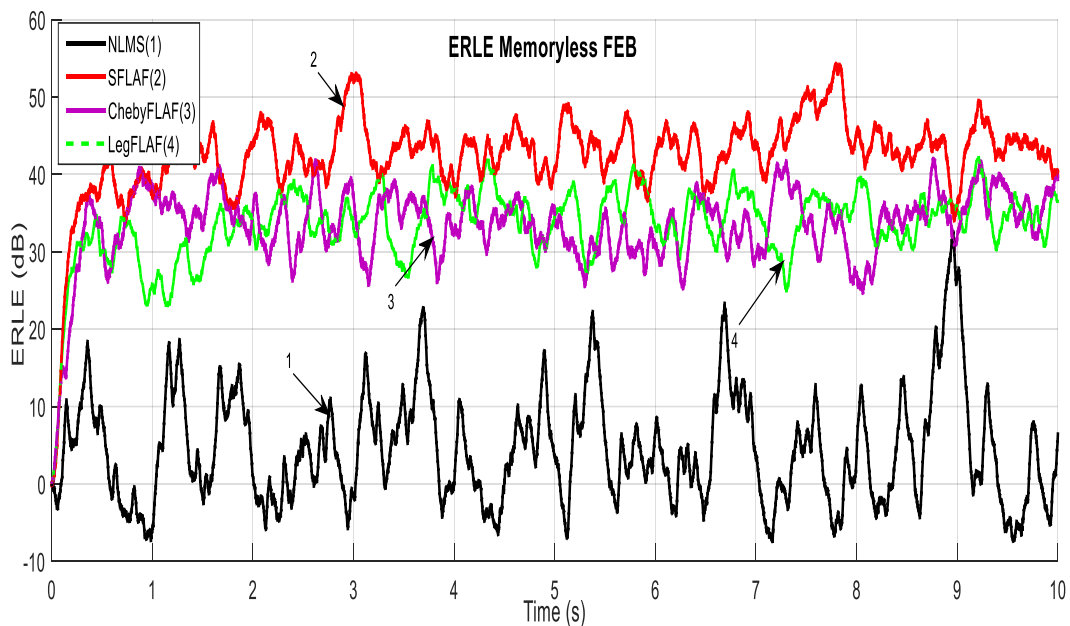
<p>INITIALIZATION: $\mathbf{W}_{L,0} = \mathbf{W}_{FL,0} = 0$</p> <p>VECTOR DEFINITION: $\mathbf{u}_k = [u(k) \ u(k-1) \ \dots \ u(k-M+1)]^T$</p> <p>GENERATE EXPANDED BUFFER: $\mathbf{g}_k = FEB(\mathbf{u}_k)$</p> $e_K = d_k - (\mathbf{u}_k^T \mathbf{W}_{L,k-1} + \mathbf{g}_k^T \mathbf{W}_{FL,k-1})$ $\mathbf{W}_{L,k} = \mathbf{W}_{L,k-1} + \mu \frac{\mathbf{u}_k}{\mathbf{u}_k^T \mathbf{u}_k + \delta} e_K$ $\mathbf{W}_{FL,k} = \mathbf{W}_{FL,k-1} + \mu \frac{\hat{\mathbf{g}}_k}{\mathbf{g}_k^T \mathbf{g}_k + \delta} e_K$
--



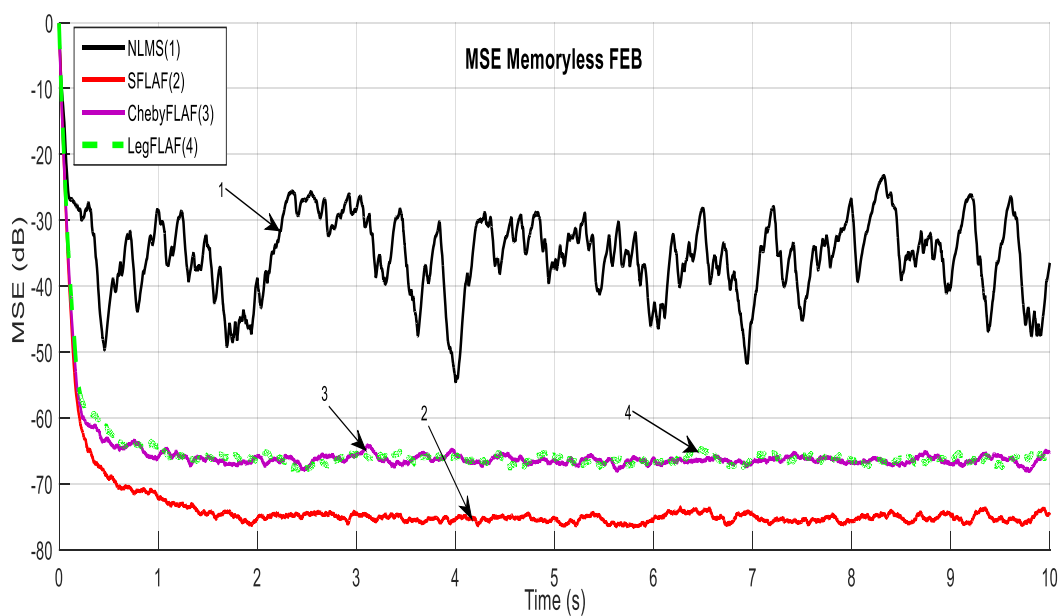
شکل ۴-۱: منحنی ERLE در حالت $Q=1$ بی حافظه برای پیوند تابعی مثلثاتی، چبیشف، لژاندر و یک فیلتر خطی در مورد ورودی صوت یک خانم



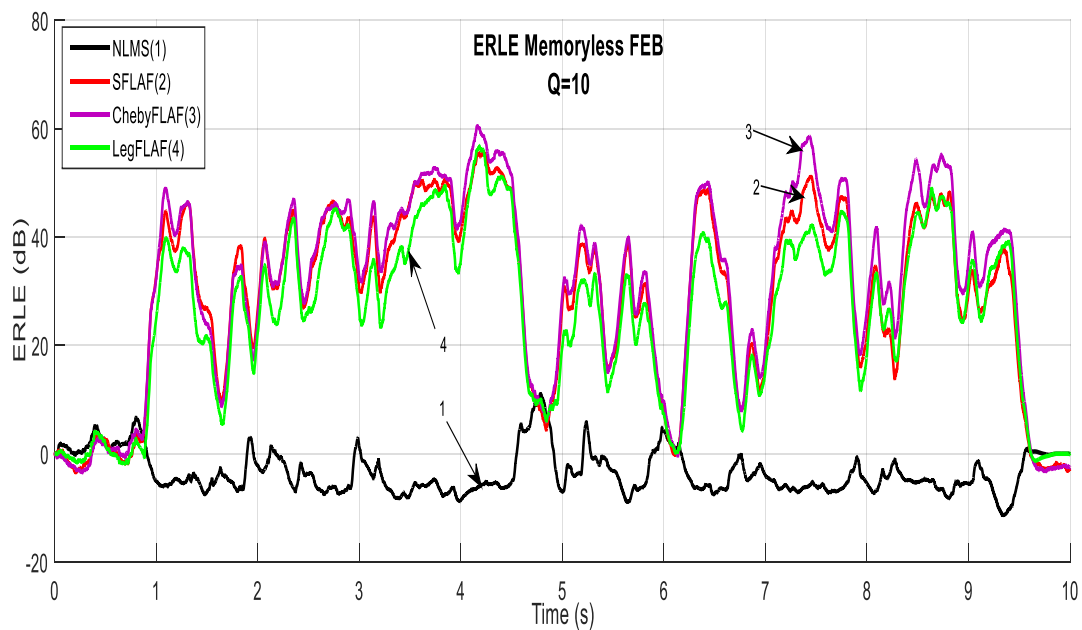
شکل ۴-۲: منحنی MSE در حالت $Q=1$ بی حافظه برای پیوند تابعی مثلثاتی، چبیشف، لژاندر و یک فیلتر خطی در مورد ورودی صوت یک خانم



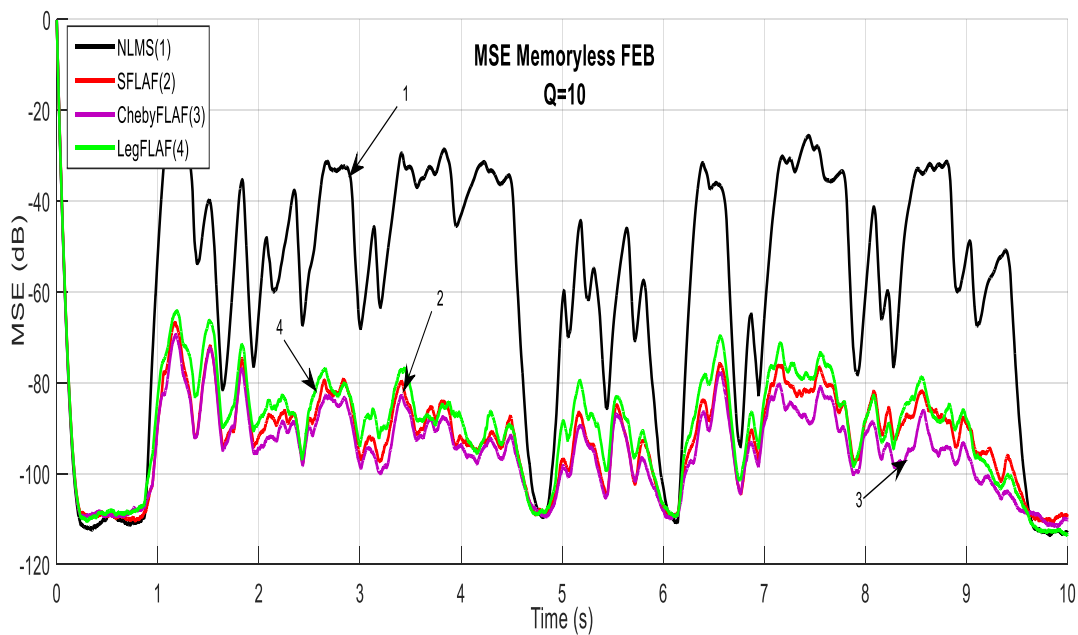
شکل ۳-۴: منحنی ERLE در حالت $Q=1$ بی حافظه برای پیوند تابعی مثلثاتی، چبیشف، لژاندر و یک فیلتر خطی در مورد ورودی رنگی



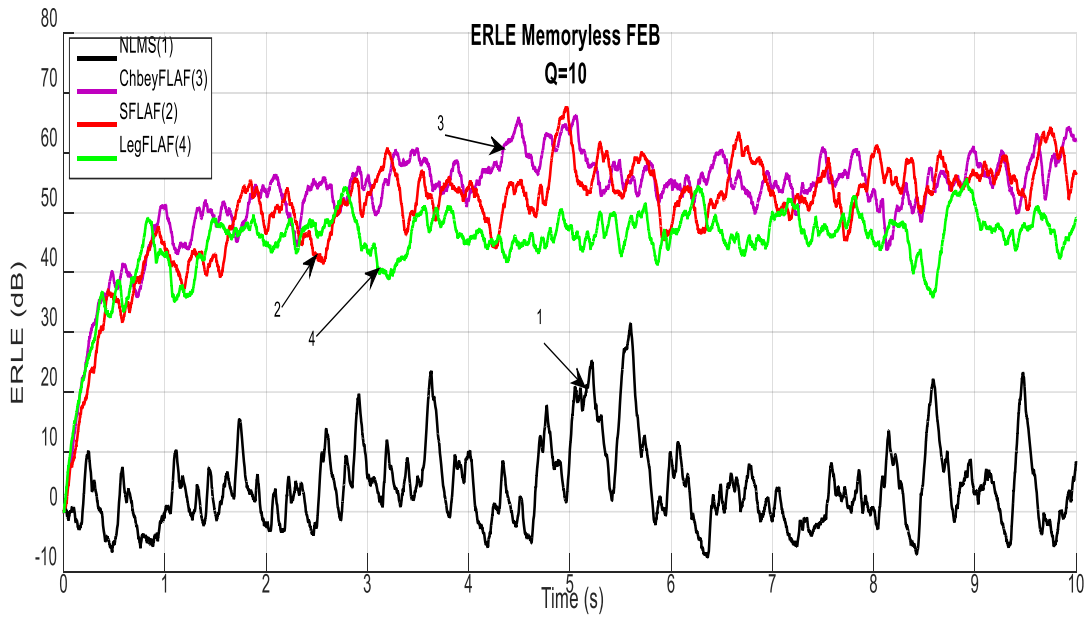
شکل ۴-۴: منحنی MSE در حالت $Q=1$ بی حافظه برای پیوند تابعی مثلثاتی، چبیشف، لژاندر و یک فیلتر خطی در مورد ورودی رنگی



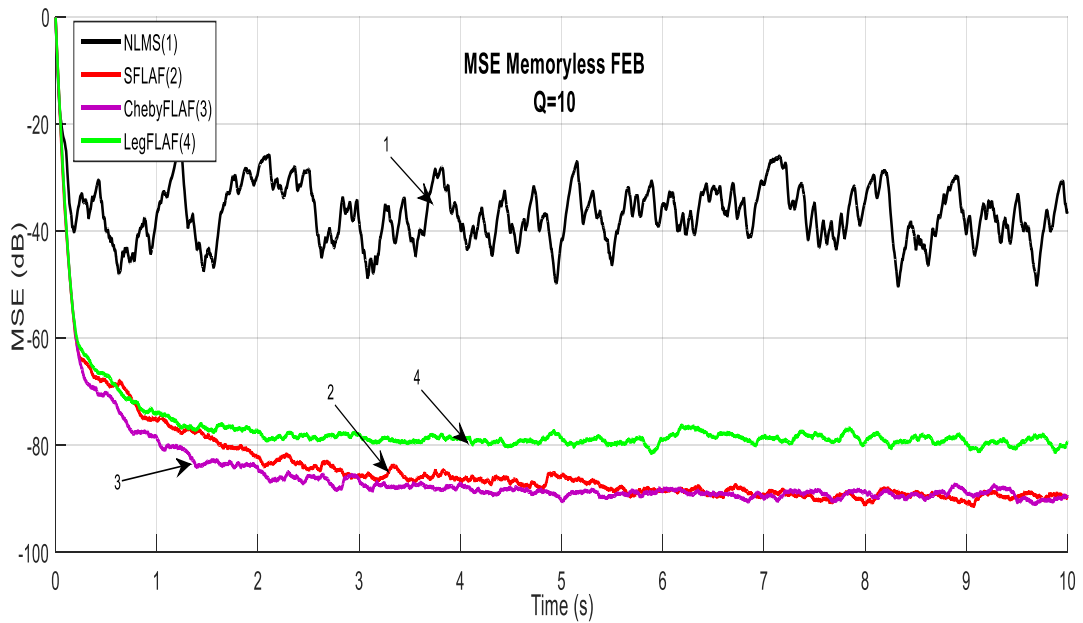
شکل ۴-۵: منحنی ERLE در حالت $Q=10$ بی حافظه برای پیوند تابعی مثلثاتی، چبیشف، لزاندر و یک فیلتر خطی در مورد ورودی صوت یک خانم



شکل ۴-۶: منحنی MSE در حالت $Q=10$ بی حافظه برای پیوند تابعی مثلثاتی، چبیشف، لزاندر و یک فیلتر خطی در مورد ورودی صوت یک خانم



شکل ۷-۴: منحنی ERLE در حالت $Q=10$ بی حافظه برای پیوند تابعی مثلثاتی، چبیشف، لژاندر و یک فیلتر خطی در مورد ورودی رنگی



شکل ۸-۴: منحنی MSE در حالت $Q=10$ بی حافظه برای پیوند تابعی مثلثاتی، چبیشف، لژاندر و یک فیلتر خطی در مورد ورودی رنگی

۴-۴ ارزیابی دوم

در این بخش ما عملکرد الگوریتم‌های مثلثاتی، چبیشف را برای حالتی که ۵ ثانیه اول پژواک خطی و ۵ ثانیه دوم پژواک غیرخطی اعمال شده باشد در دو معماری SFLAF و CFLAF که طول پیوند تابعی در دو حالت $Q = 5$ و $Q = 20$ می‌باشد را بررسی می‌کنیم. در این آزمایش بسط پیوند تابعی چبیشف را برای حالت CFLAF و بسط تابعی مثلثاتی را برای هر دو حالت CFLAF و SFLAF اعمال کردیم. همچنین نشان می‌دهیم الگوریتم‌های پیشنهادی در حالت پژواک خطی چگونه عمل می‌کنند و فیلتر وقتی خطی در حالت پژواک خطی و غیرخطی آزموده می‌شود، در این آزمایش ما از تنظیم پارامترهای یکسانی برای تمام این ارزیابی استفاده می‌کنیم و طول بافر ورودی برای همه الگوریتم‌ها در حالت SFLAF و CFLAF، $M = 300$ در نظر گرفته شده است اندازه گام فیلتر خطی $\mu_L = 0.2$ برای بهره‌برداری از مزیت داشتن دو فیلتر جداگانه در CFLAF و SFLAF، یک پارامتر اندازه گام متفاوت برای فیلتر غیرخطی که مساوی با $\mu_{FL} = 0.5$ را در نظر می‌گیریم. برای داشتن یک مقایسه منصفانه ما از یک سناریوی بی حافظه استفاده می‌کنیم که رابطه (۴-۱) و رابطه (۴-۲) را درگیر می‌کند. ما معماری بی حافظه را برای حالت‌های مختلف FEB در FLAF به دست می‌آوریم. طول سیگنال مورد آزمایش ۱۰ ثانیه است، که ۵ ثانیه اول پژواک خطی و ۵ ثانیه دوم پژواک غیرخطی اعمال شده است. عملکرد معماری‌های ذکر شده بالا را در عبارت ERLE و MSE هم برای ورودی رنگی و هم ورودی صوت یک خانم مقایسه می‌کنیم.

نتایجی که در شکل ۴-۹، شکل ۴-۱۰، شکل ۴-۱۱، شکل ۴-۱۲ ترسیم شده است که در آن $Q = 5$ در نظر گرفته شده، الگوریتم CFLAF با بسط تابعی مثلثاتی نسبت به SFLAF در ۵ ثانیه اول مانند فیلتر خطی عمل کرده و این برتری این الگوریتم را در حالت‌هایی که پژواک خطی باشد را نشان می‌دهد در ۵ ثانیه بعدی که سیستم غیرخطی شده است الگوریتم CFLAF و SFLAF عملکرد تقریباً یکسانی دارند در الگوریتم CFLAF با بسط تابعی چبیشف همان گونه که در منحنی‌ها مشخص

می‌شود در حالت خطی از SFLAF بهتر عمل کرده و از CFLAF با بسط تابعی مثلثاتی با کمی اختلاف بهبود کمتری دیده می‌شود و در ۵ ثانیه بعدی که سیستم غیرخطی شده است این الگوریتم عملکرد ضعیف‌تری نسبت به حالت‌های قبل داشته فیلتر وقتی خطی در حالت پژواک خطی دارای حذف پژواک بالا و به محض ورود غیر خطی‌ها افت قابل مشاهده‌ای دارد.

نتایجی که در شکل ۴-۱۳، شکل ۴-۱۴، شکل ۴-۱۵، شکل ۴-۱۶ ترسیم شده است که در آن $Q = 20$ نظر گرفته شده، الگوریتم CFLAF با هر دو بسط تابعی مثلثاتی و چبیشف در ۵ ثانیه اول که دارای پژواک خطی است مانند فیلتر خطی عمل کرده و برتری این الگوریتم‌ها را نسبت به SFLAF در حالت‌هایی که پژواک خطی باشد، نشان می‌دهد در ۵ ثانیه بعدی که سیستم غیرخطی شده است الگوریتم CFLAF با بسط تابعی مثلثاتی با الگوریتم SFLAF عملکرد تقریباً یکسانی دارند در الگوریتم CFLAF با بسط تابعی چبیشف همان‌گونه که در منحنی‌ها مشخص می‌شود در این حالت نسبت به الگوریتم‌های SFLAF و CFLAF با بسط تابعی مثلثاتی با اختلاف کاملاً قابل مشاهده بهبود دیده می‌شود در مجموع از رسم نمودارهای ERLE و MSE هر دو الگوریتم می‌توان این‌گونه برداشت کرد که الگوریتم CFLAF نسبت به SFLAF عملکرد بهتری دارد. این بدان معناست که الگوریتم CFLAF برای سیستم خطی عملکرد بهتری نسبت به SFLAF دارد. بهبود عملکرد، توسط CFLAF در مقایسه با معماری فیلترهای دیگر آشکار است با توجه به تعداد Q در حالت پژواک خطی بهبود عملکرد الگوریتم‌ها در بعضی نواحی منحنی ERLE تا ۴۵dB و در نواحی پژواک غیرخطی تا ۵۵dB دیده می‌شود.

INITIALIZATION:

$$\mathbf{W}_{L,0} = \mathbf{W}_{FL,0} = 0$$

VECTOR DEFINITION:

$$\mathbf{u}_k = [u(k) \ u(k-1) \ \dots \ u(k-M+1)]^T$$

GENERATE EXPANDED BUFFER:

$$\mathbf{g}_k = FEB(\mathbf{u}_k)$$

$$e_{FL,k} = d_k - (\mathbf{u}_k^T \mathbf{W}_{L,k-1} + \mathbf{g}_k^T \mathbf{W}_{FL,k-1})$$

Limit the adaptation parameter:

$$\text{If } abs(a[k]) > 4$$

$$a[k] = sign(a[k]) \times 4;$$

end

$$\lambda[k] = sgm(a[k]) = \frac{1}{(1 + e^{-a[k]})}$$

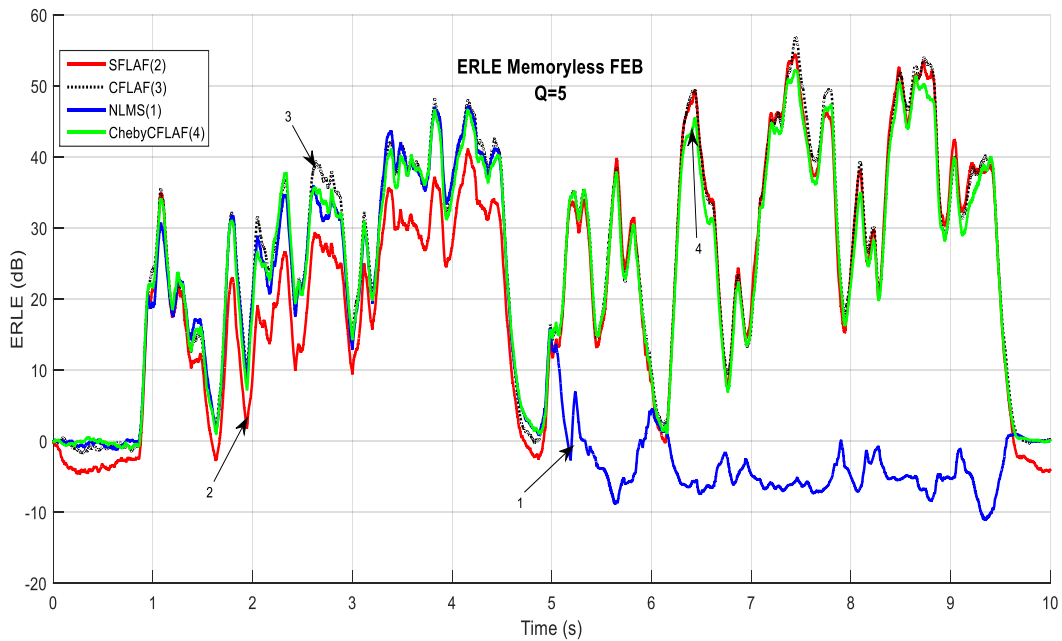
$$r[k] = \beta r[k-1] + (1 - \beta) (\mathbf{g}_k^T \mathbf{W}_{FL,k-1})^2$$

$$e_k = d_k - \mathbf{u}_k^T \mathbf{W}_{L,k-1} - \lambda[k] \mathbf{g}_k^T \mathbf{W}_{FL,k-1}$$

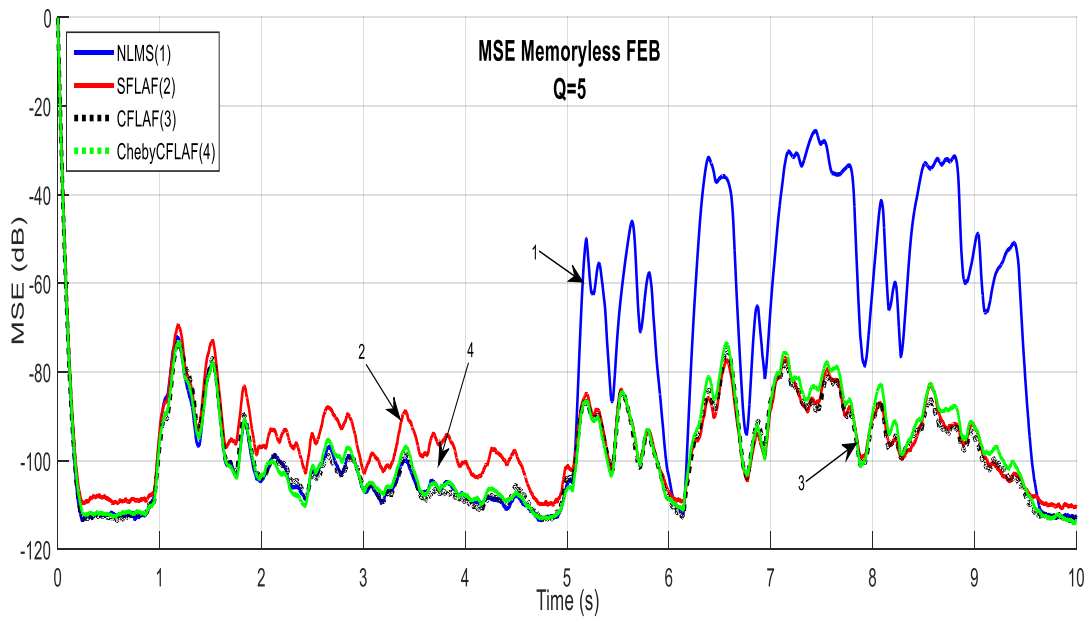
$$\mathbf{W}_{L,k} = \mathbf{W}_{L,k-1} + \mu_L \frac{\mathbf{u}_k}{\mathbf{u}_k^T \mathbf{u}_k + \delta} e_k$$

$$\mathbf{W}_{FL,k} = \mathbf{W}_{FL,k-1} + \mu_{FL} \frac{\mathbf{g}_k}{\mathbf{g}_k^T \mathbf{g}_k + \delta} e_{FL,k}$$

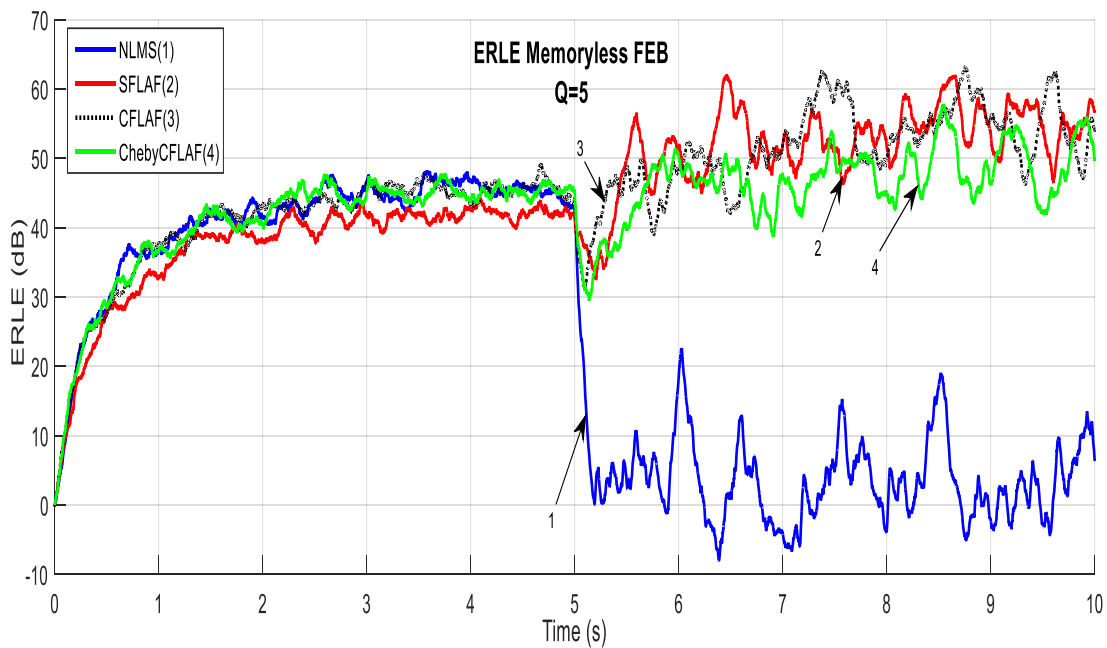
$$a[k+1] = a[k] + \frac{\mu_a}{r[k]} e[k] (\mathbf{g}_k^T \mathbf{W}_{FL,k-1}) \lambda[k] (1 - \lambda[k])$$



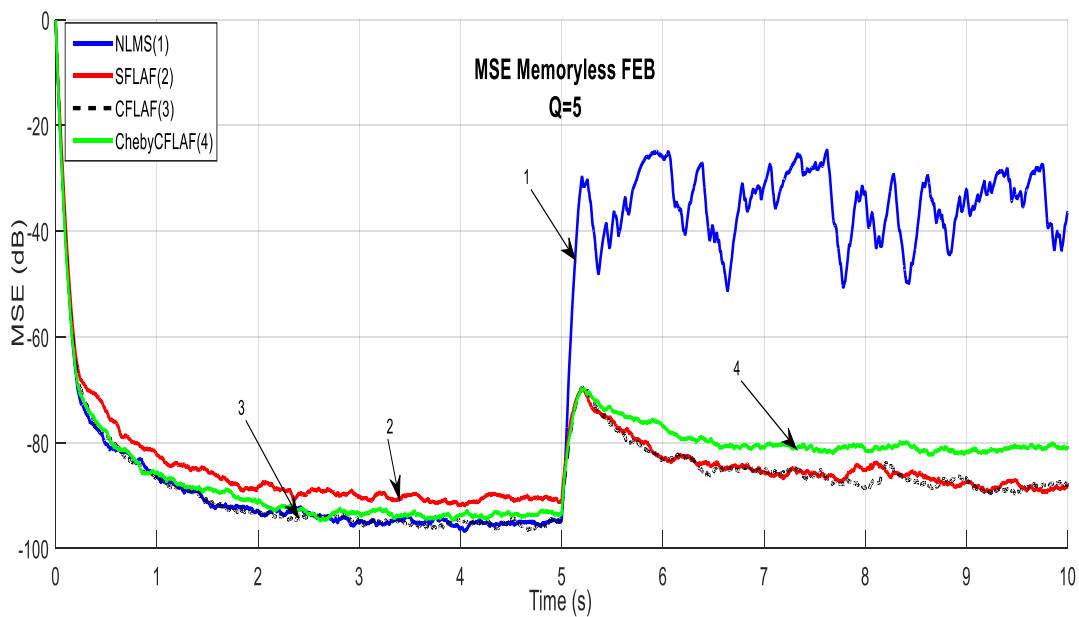
شکل ۴-۹: منحنی ERLE در حالت $Q=5$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف و یک فیلتر خطی در مورد ورودی صوت یک خانم



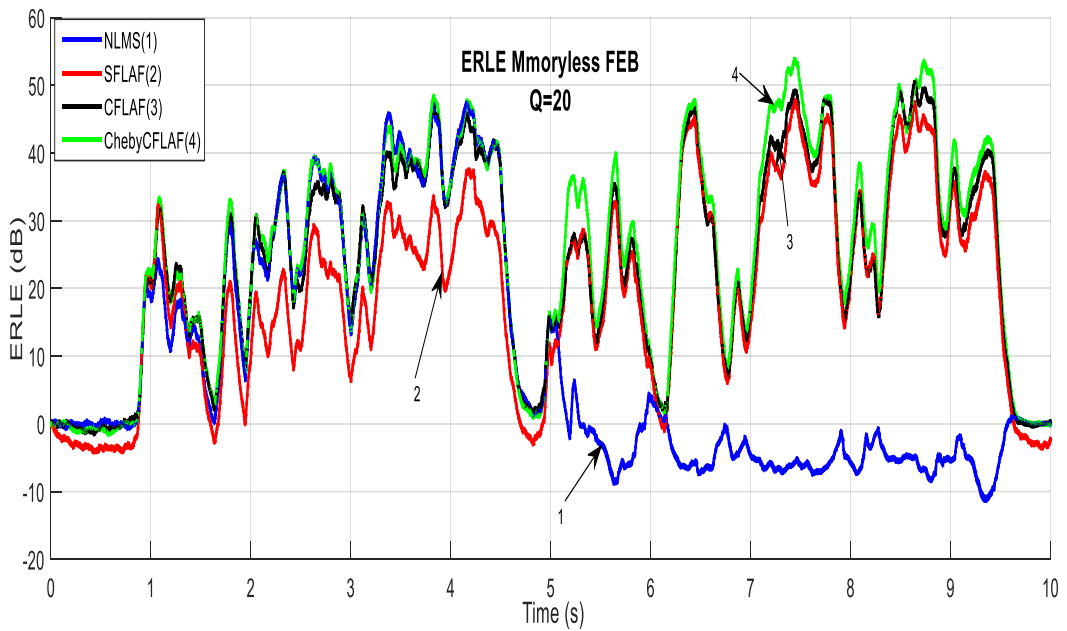
شکل ۴-۱۰: منحنی MSE در حالت $Q=5$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف و یک فیلتر خطی در مورد ورودی صوت یک خانم



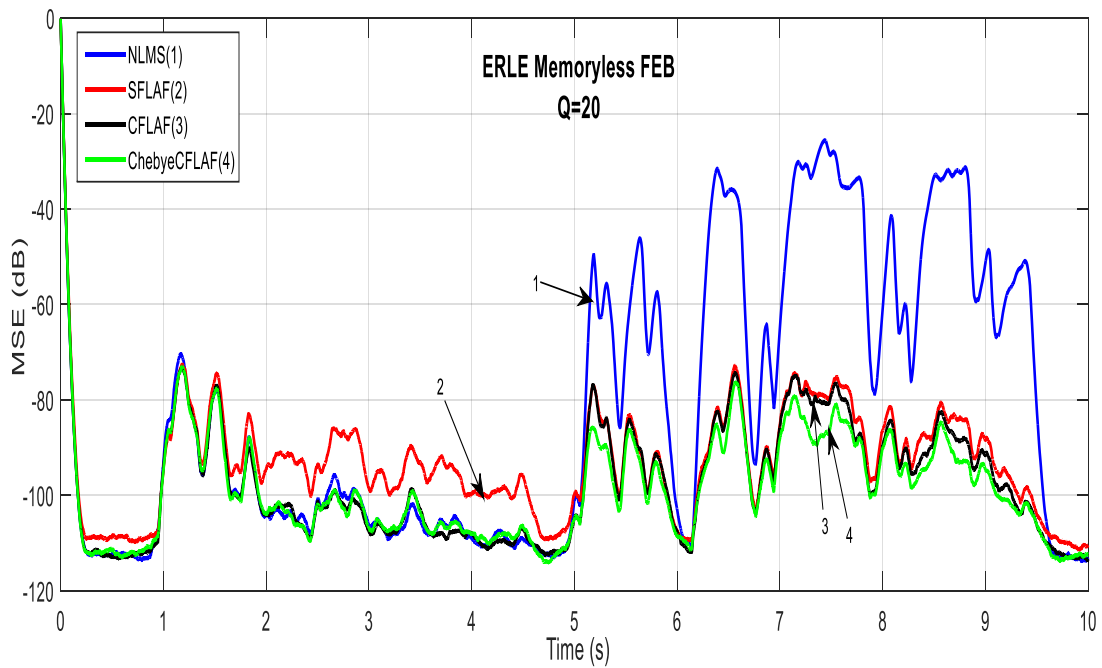
شکل ۴-۱۱: منحنی ERLE در حالت $Q=5$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف و یک فیلتر خطی در مورد ورودی رنگی



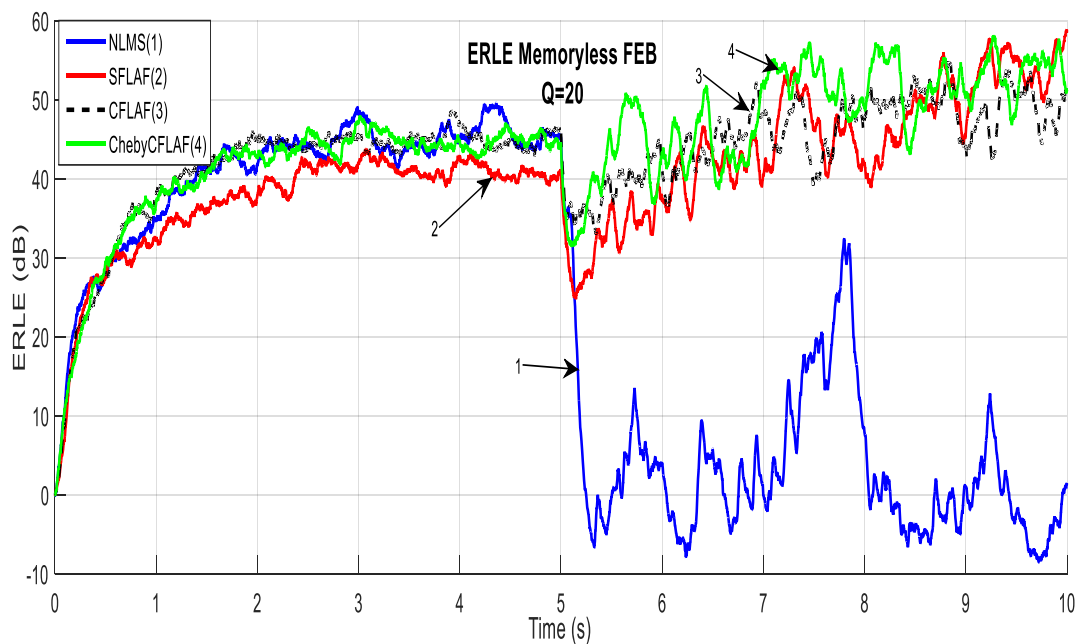
شکل ۴-۱۲: منحنی MSE در حالت $Q=5$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف و یک فیلتر خطی در مورد ورودی رنگی



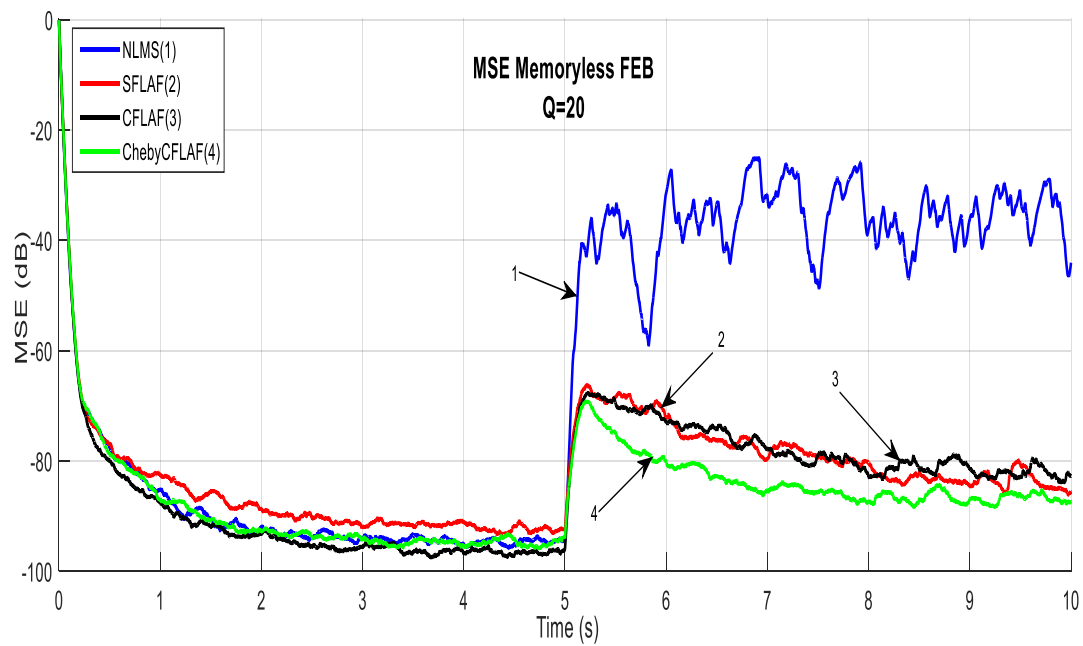
شکل ۴-۱۳: منحنی ERLE در حالت $Q=20$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف و یک فیلتر خطی در مورد ورودی صوت یک خانم



شکل ۴-۱۴: منحنی MSE در حالت $Q=20$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف و یک فیلتر خطی در مورد ورودی صوت یک خانم



شکل ۴-۱۵: منحنی ERLE در حالت $Q=20$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف و یک فیلتر خطی در مورد ورودی رنگی



شکل ۴-۱۶: منحنی MSE در حالت $Q=20$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف و یک فیلتر خطی در مورد ورودی رنگی

۴-۵ ارزیابی سوم

در این بخش ما عملکرد الگوریتم‌های مثلثاتی، چبیشف و لژاندر را برای حالت دو پیوند تابعی موازی^۱ مانند شکل ۳-۷ ارزیابی می‌کنیم و عملکرد این حالت‌ها را نسبت به حالت یک پیوند تابعی نشان می‌دهیم. که طول پیوند تابعی در دو حالت $Q = 5$ و $Q = 15$ می‌باشد. همچنین حالت موازی را با حالت SFLAF پیوند تابعی مثلثاتی مقایسه می‌کنیم. در این آزمایش ما از تنظیم پارامترهای یکسانی برای تمام این ارزیابی استفاده می‌کنیم و طول بافر ورودی برای همه الگوریتم‌ها در حالت، $M = 300$ در نظر گرفته شده است اندازه گام فیلتر خطی $\mu_L = 0.2$ و برای غیرخطی برابر با $\mu_{FL} = 0.5$ را در نظر می‌گیریم. برای داشتن یک مقایسه منصفانه ما از یک سناریوی بی حافظه استفاده می‌کنیم که رابطه (۴-۱) و رابطه (۴-۲) را درگیر می‌کند. طول سیگنال مورد آزمایش ۱۰ ثانیه است، عملکرد معماری‌های ذکر شده بالا را در عبارت ERLE و MSE هم برای ورودی رنگی و هم ورودی صوت یک خانم مقایسه می‌کنیم.

نتایجی که در شکل ۴-۱۷، شکل ۴-۱۸، شکل ۴-۱۹ و شکل ۴-۲۰ ترسیم شده است که در آن $Q = 5$ در نظر گرفته شده، همان‌گونه که مشخص می‌شود الگوریتم پیوند تابعی مثلثاتی موازی^۲ و پیوند تابعی چبیشف موازی^۳ نسبت به SFLAF و پیوند تابعی لژاندر موازی^۴ عملکرد بهتری دارند و این نشان دهنده این است که حذف پژواک غیرخطی با دو فیلتر موازی غیرخطی بهتر از حالت‌های قبل با همین مشخصات است. ناگفته نماند سرعت همگرایی سیستم در حالت موازی ضعیف‌تر از حالت عادی عمل می‌کند که در شکل ۴-۲۰ می‌توانید مشاهده کنید.

نتایجی که در شکل ۴-۲۱ و شکل ۴-۲۲ ترسیم شده است که در آن $Q = 15$ نظر گرفته شده،

¹ Parallel

² PSFLAF

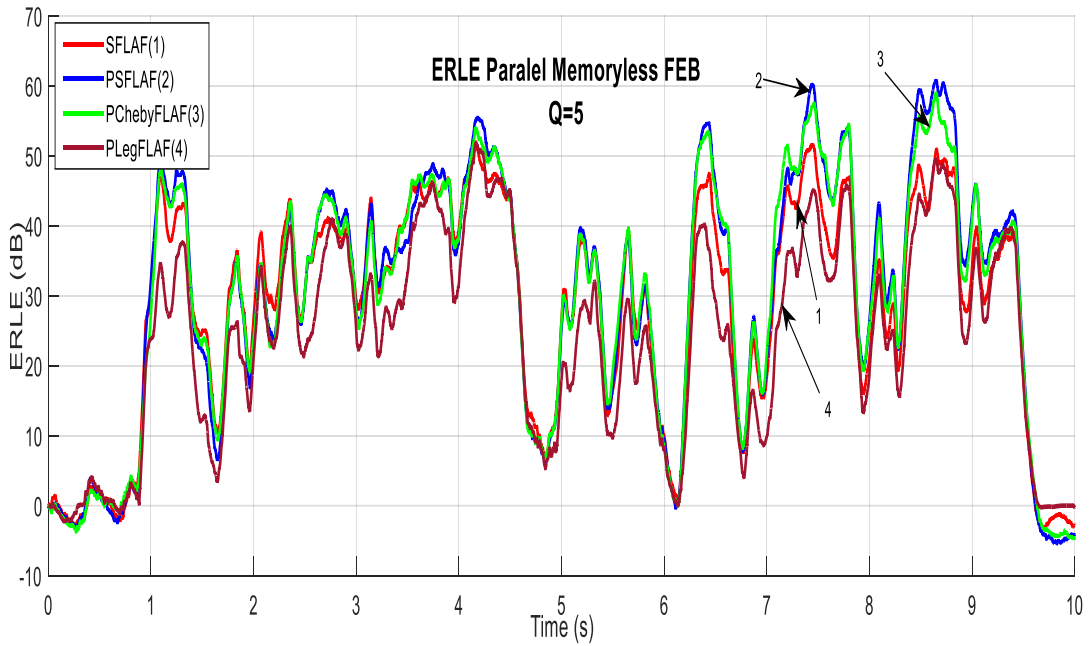
³ PChebyFLAF

⁴ PLegFLAF

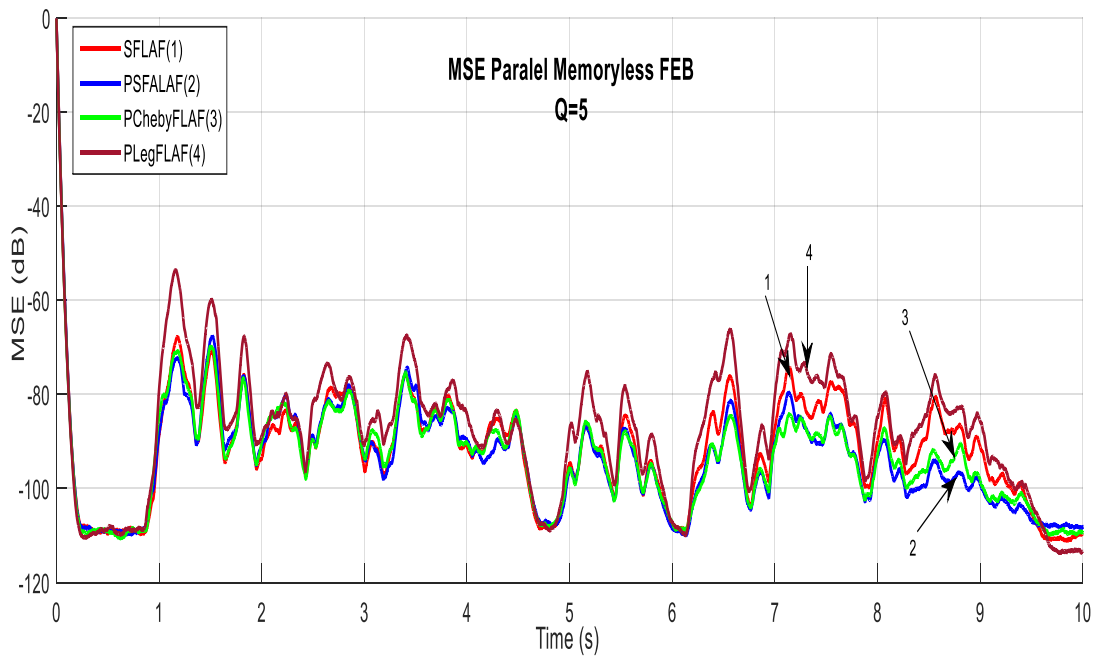
ضعف همگرایی سیستم موازی کاملاً نمایان است و در ابتدا سیستم SFLAF عملکرد بهتری داشته و در ادامه الگوریتم‌های PChebyFLAF و PSFLAF سطح حذف پژواک بیشتری از SFLAF را دارند و عملکرد بهتر حالت موازی را اثبات می‌کند. همان‌گونه که معلوم است پس از همگرا شدن الگوریتم‌های موازی، الگوریتم پیشنهادی PChebyFLAF به دلیل اینکه در سیستم‌های غیرخطی با پیوند تابعی بیشتر بهتر نمایان می‌شود همان‌گونه که مشخص است عملکرد بهتری نسبت به همه حالت‌ها دارد بهبود عملکرد الگوریتم‌ها در این حالت در منحنی ERLE بین ۵۰dB تا ۶۰dB قابل مشاهده است.

جدول ۴-۳: الگوریتم بر پایه FLAF موازی

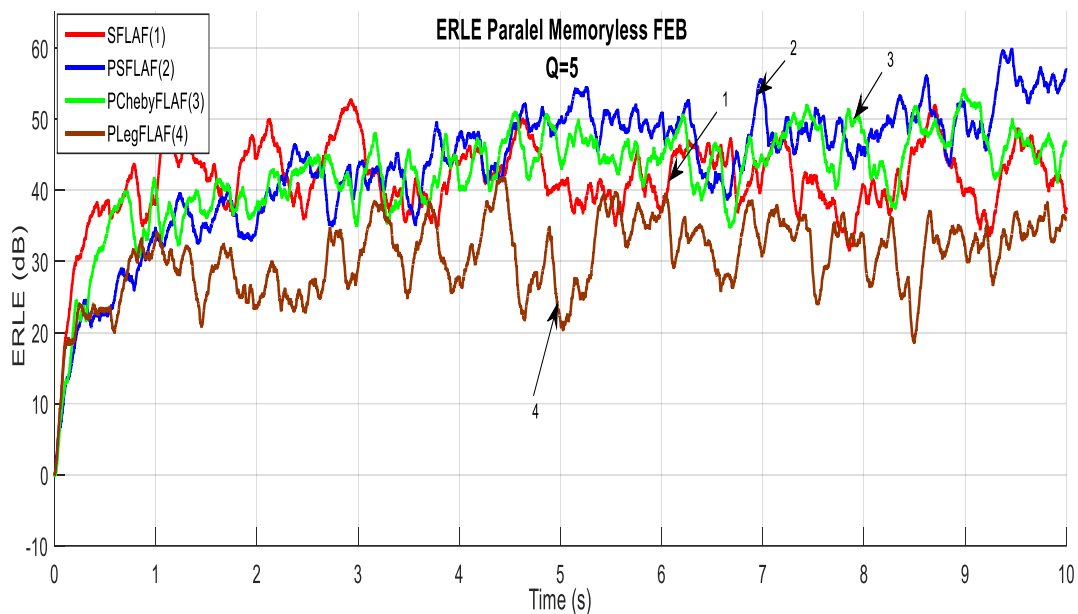
<p>INITIALIZATION: $\mathbf{W}_{FL1,0} = \mathbf{W}_{FL2,0} = 0$</p> <p>VECTOR DEFENITION: $\mathbf{u}_k = [u(k) \ u(k-1) \ \dots \ u(k-M+1)]^T$</p> <p>GENERATE EXPANDED BUFFER: $\mathbf{g}_{1k} = FEB(\mathbf{u}_{1k})$ $\mathbf{g}_{2k} = FEB(\mathbf{u}_{2k})$ $e_k = d_k - (\mathbf{g}_{1k}^T \mathbf{W}_{FL1,k-1} + \mathbf{g}_{2k}^T \mathbf{W}_{FL2,k-1})$ $\mathbf{W}_{FL1,k} = \mathbf{W}_{FL1,k-1} + \mu_{FL1} \frac{\hat{\mathbf{g}}_{1k}}{\mathbf{g}_{1k}^T \mathbf{g}_{1k} + \delta} e_k$ $\mathbf{W}_{FL2,k} = \mathbf{W}_{FL2,k-1} + \mu_{FL2} \frac{\hat{\mathbf{g}}_{2k}}{\mathbf{g}_{2k}^T \mathbf{g}_{2k} + \delta} e_k$</p>



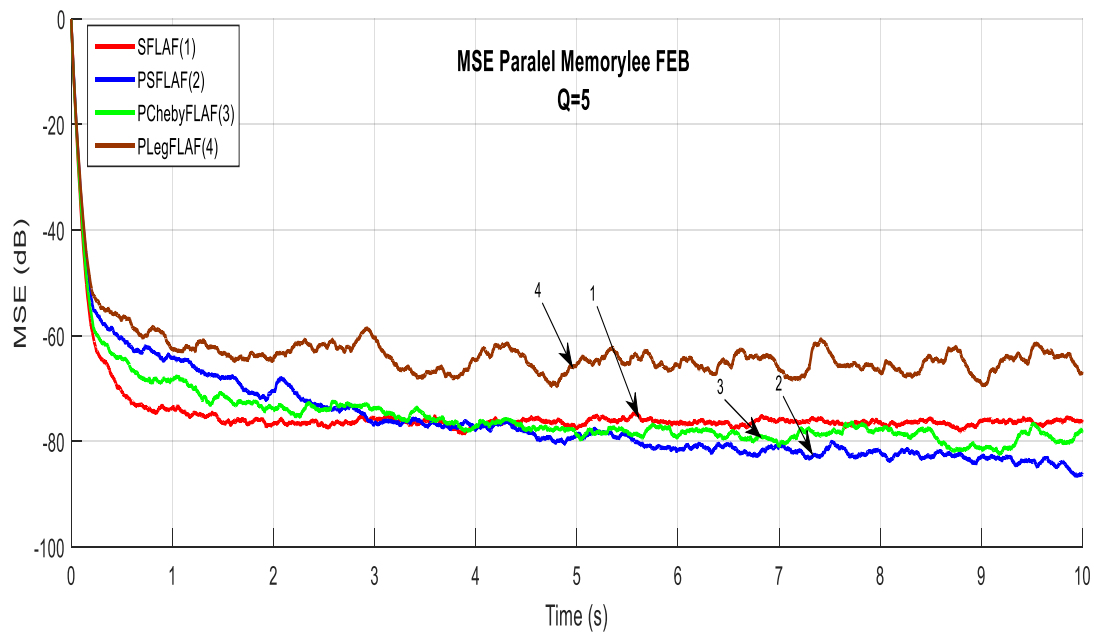
شکل ۴-۱۷: منحنی ERLE در حالت $Q=5$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف، حالت موازی در مورد ورودی صوت یک خانم



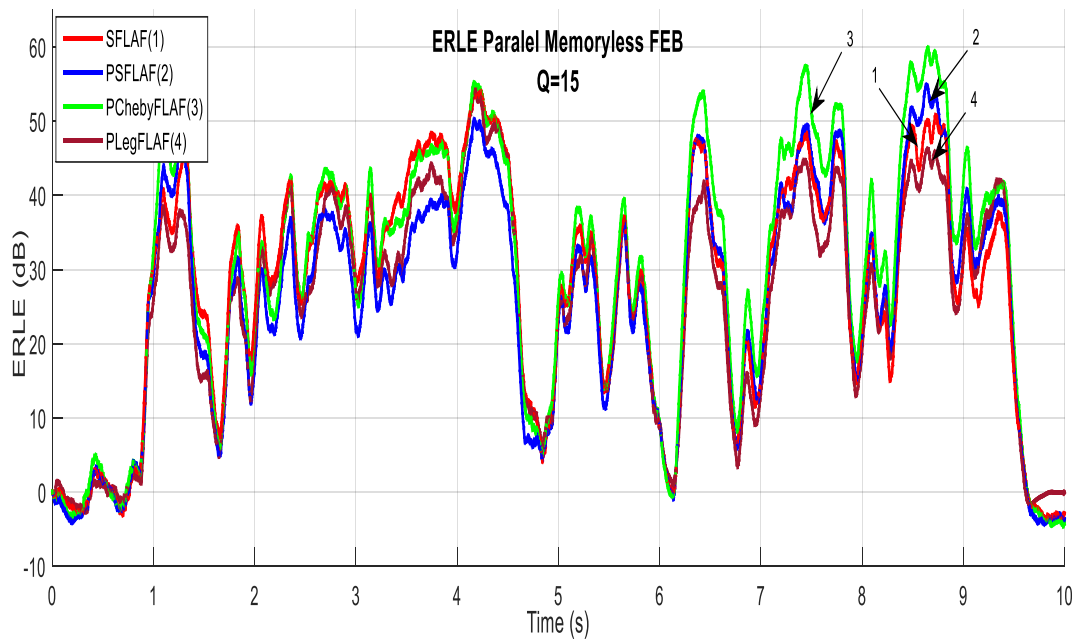
شکل ۴-۱۸: منحنی MSE در حالت $Q=5$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف، حالت موازی در مورد ورودی صوت یک خانم



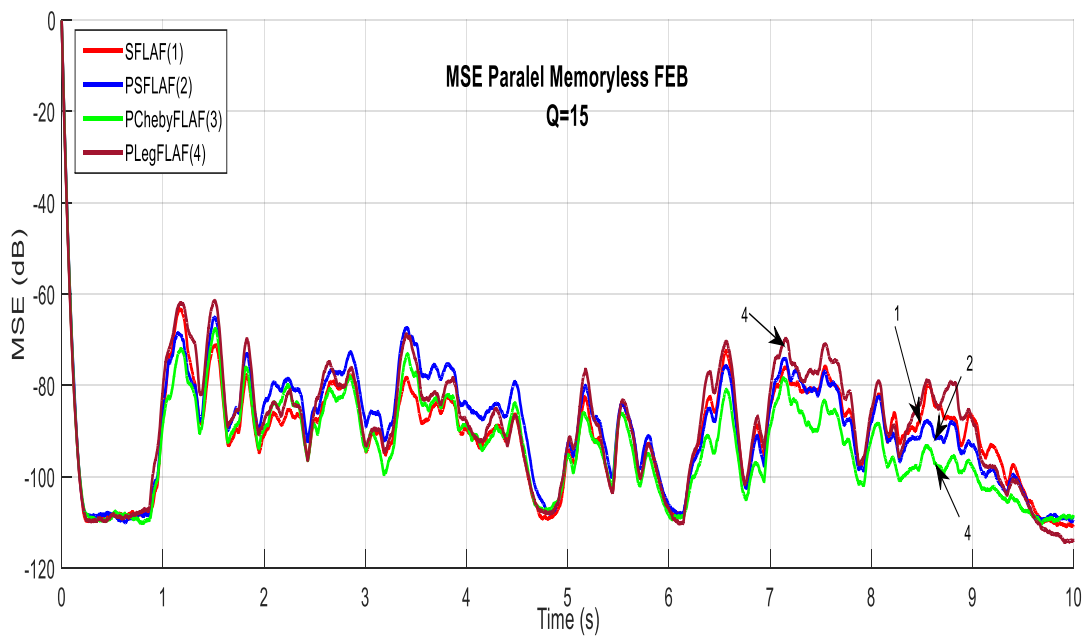
شکل ۴-۱۹: منحنی ERLE در حالت $Q=5$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف، حالت موازی در مورد ورودی رنگی



شکل ۴-۲۰: منحنی MSE در حالت $Q=5$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف، حالت موازی در مورد ورودی رنگی



شکل ۴-۲۱: منحنی ERLE در حالت $Q=15$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف، حالت موازی در مورد ورودی صوت یک خانم



شکل ۴-۲۲: منحنی MSE در حالت $Q=15$ بی حافظه برای پیوند تابعی مثلثاتی و چبیشف، حالت موازی در مورد ورودی صوت یک خانم

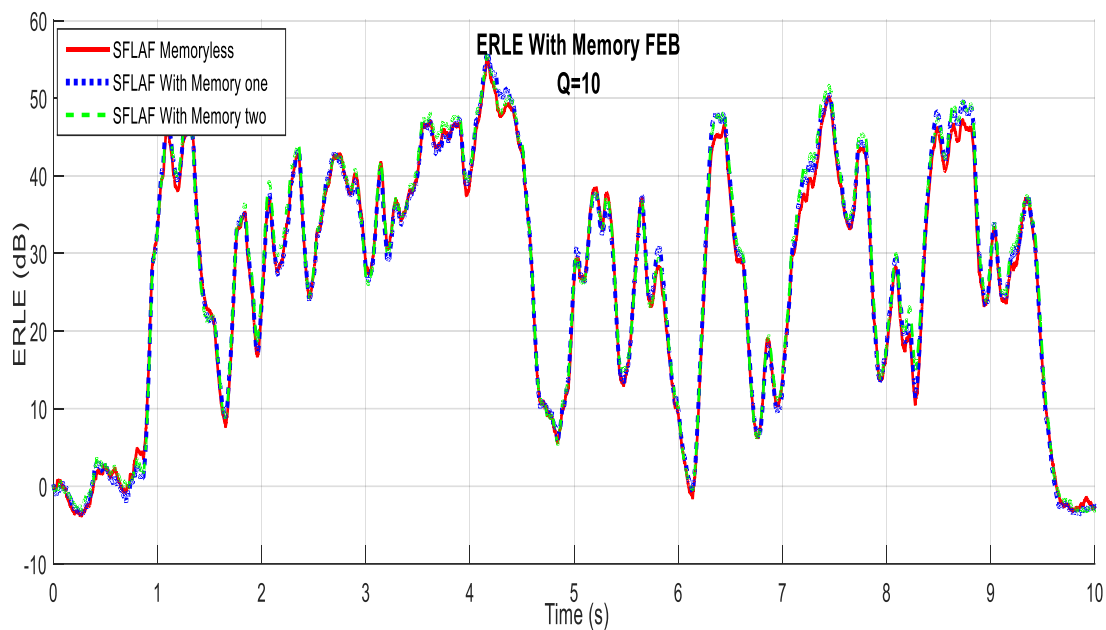
۶-۴ ارزیابی چهارم

در این بخش ما عملکرد الگوریتم‌های مثلثاتی، چبیشف را در حالت باحافظه و بی‌حافظه برای حالتی که طول پیوند تابعی در $Q = 10$ می‌باشد را بررسی می‌کنیم. همچنین نشان می‌دهیم. در این آزمایش ما از تنظیم پارامترهای یکسانی برای تمام این ارزیابی استفاده می‌کنیم و طول بافر ورودی برای همه الگوریتم‌ها $M = 300$ در نظر گرفته شده است اندازه گام فیلتر خطی $\mu_L = 0.2$ برای بهره‌برداری از مزیت داشتن دو فیلتر جداگانه SFLAF ما یک پارامتر اندازه گام متفاوت برای فیلتر غیرخطی SFLAF مساوی با $\mu_{FL} = 0.5$ را در نظر می‌گیریم. برای داشتن یک مقایسه منصفانه ما از یک سناریوی باحافظه استفاده می‌کنیم که رابطه (۴-۱) و رابطه (۴-۴) را درگیر می‌کند و پژواک مرتبه ۵ را تولید می‌کند. ما معماری باحافظه یک و دو را با حالت بی‌حافظه در FLAF مورد مقایسه قرار می‌دهیم. طول سیگنال مورد آزمایش ۱۰ ثانیه است، عملکرد معماری‌های ذکر شده بالا را در عبارت ERLE و MSE هم برای ورودی رنگی و هم ورودی صوت یک خانم مقایسه می‌کنیم.

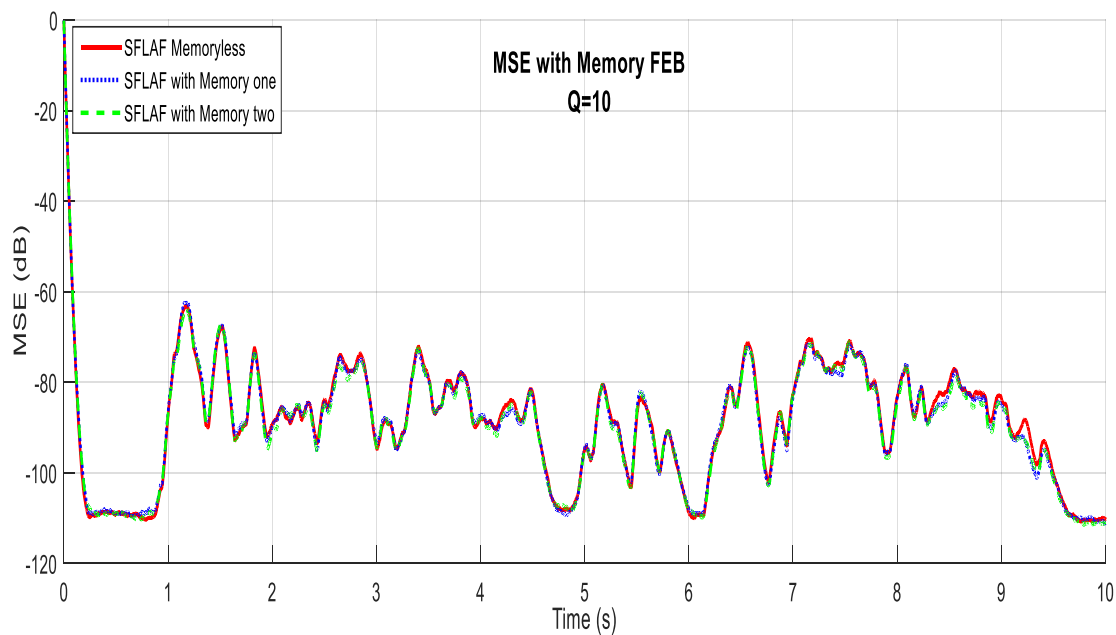
نتایجی که در شکل ۴-۱، شکل ۴-۲، شکل ۴-۳ و شکل ۴-۴ ترسیم شده است که در آن $Q=1$ در نظر گرفته شده، بهبود عملکرد، توسط SFLAF در مقایسه با معماری فیلترهای دیگر آشکار است هرچند در مورد ورودی صوت بهبود کمتر واضح است چون غیرخطی‌ها روی سیگنال در یک راه یکنواخت اثر نمی‌گذارد و این امر به خاطر طبیعت تغییرپذیر سیگنال صوت است. همان‌گونه که در منحنی‌ها معلوم است پیوند تابعی چبیشف و لژاندر، در این حالت دارای مقادیر ثابتی هستند و منحنی آن‌ها کامل بر روی هم قرار گرفته است. نوسانات منحنی‌های ورودی رنگی بسیار کم است و به دلیل انتخاب شدن اعداد به صورت تصادفی ممکن است در منحنی ERLE دارای مقداری تداخل باشد که موجب خطای دید شود با رسم منحنی MSE به صورت واضح‌تر می‌توان تفاوت الگوریتم‌ها را دید.

نتایجی که در شکل ۴-۲۳، شکل ۴-۲۴، شکل ۴-۲۵ و شکل ۴-۲۶ ترسیم شده است که در آن

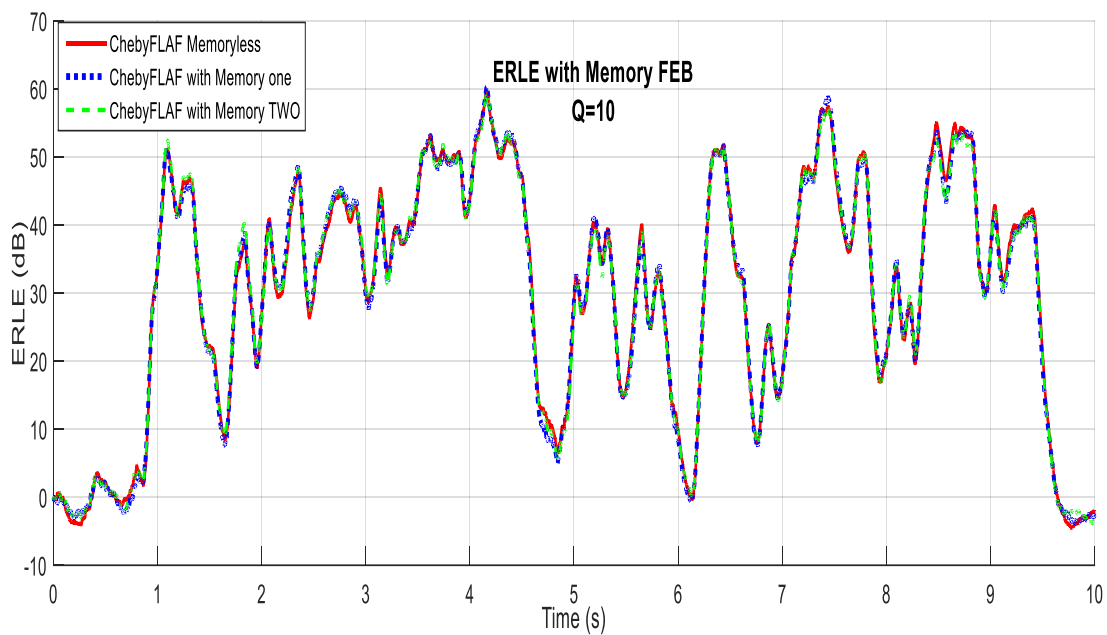
$Q=10$ نظر گرفته شده، تفاوت بین الگوریتم‌های باحافظه و بی حافظه نسبتاً کم است. فقط بهبود جزئی در نزدیکی قله سیگنال رخ داده است. می‌توان مشاهده کرد که تفاوت بین معماری بر پایه FLAF در مقایسه با آزمایش‌های قبلی تغییر زیادی نداشته و این به خاطر این حقیقت است که FLAF انعطاف‌پذیر برای هر مرتبه پژواک عمل می‌کند. با این وجود، حتی در این مورد هم پیوند تابعی چبیشف عملکرد بهتری نسبت به پیوند تابعی مثلثاتی در شکل ۴-۲۷ و شکل ۴-۲۸ نشان می‌دهد. بنابراین استفاده از بسط باحافظه ممکن است در افزایش کیفیت صوت مهم باشد گرچه این درگیر بار محاسباتی بزرگ‌تری در مقایسه با بسط بی‌حافظه است. این دلیل این است که چرا یک انتخاب کلی بهتر وجود ندارد با این حال یک طراحی FLAF مناسب می‌تواند روی منابع موجود و نیاز کیفیت، پایه‌گذاری شود. بهبود عملکرد الگوریتم‌ها در این حالت در منحنی ERLE بین ۵۰dB تا ۶۰dB قابل مشاهده است.



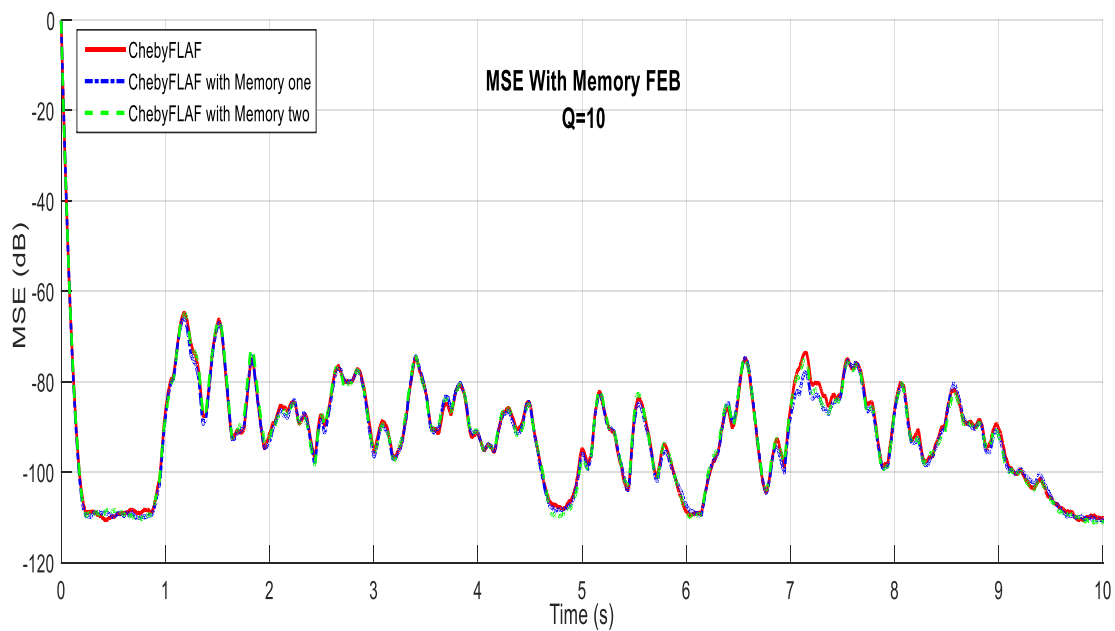
شکل ۴-۲۳: منحنی ERLE در حالت $Q=10$ باحافظه برای پیوند تابعی مثلثاتی در مورد ورودی صوت یک خانم



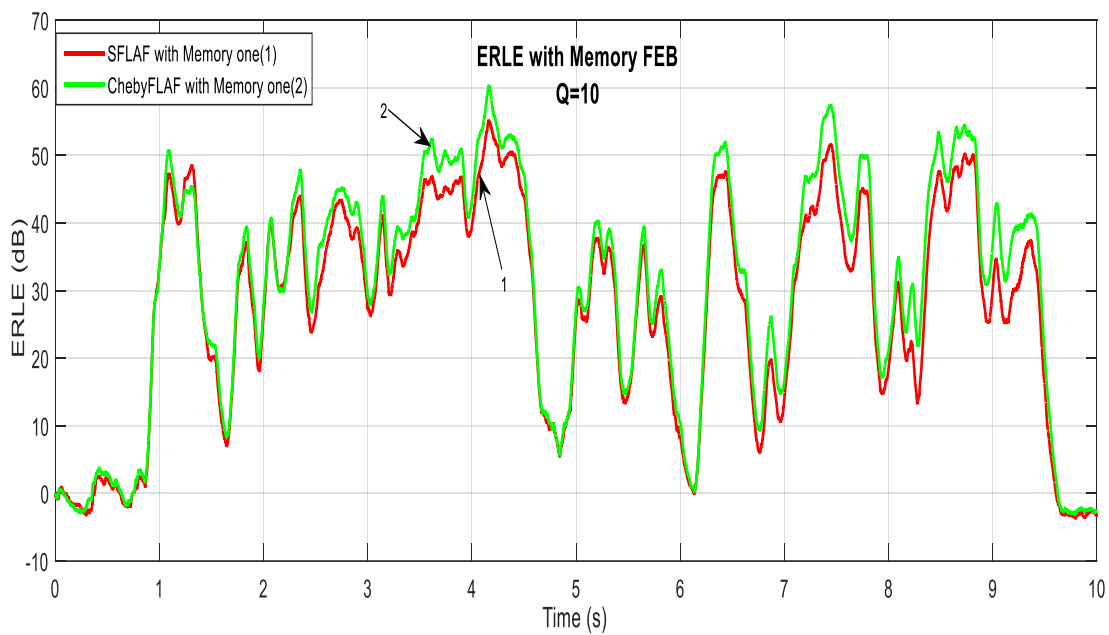
شکل ۴-۲۴: منحنی MSE در حالت $Q=10$ باحافظه برای پیوند تابعی مثلثاتی در مورد ورودی صوت یک خانم



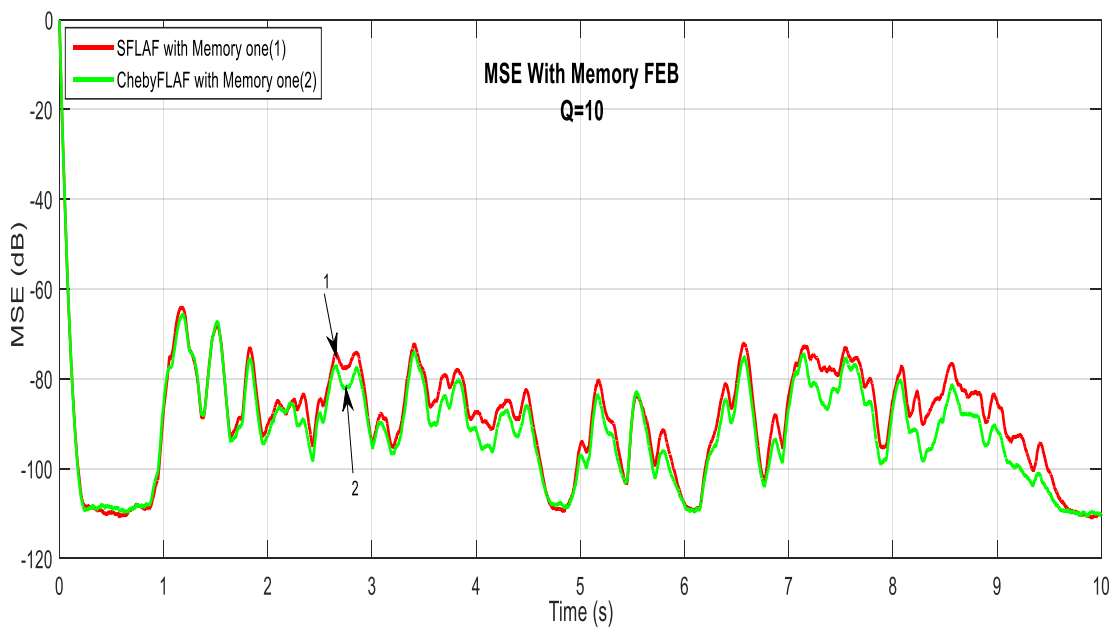
شکل ۴-۲۵: منحنی ERLE در حالت $Q=10$ باحافظه برای پیوند تابعی چبیشف در مورد ورودی صوت یک خانم



شکل ۴-۲۶: منحنی MSE در حالت $Q=10$ باحافظه برای پیوند تابعی چبیشف در مورد ورودی صوت یک خانم



شکل ۴-۲۷: منحنی ERLE در حالت $Q=10$ باحافظه برای مقایسه پیوند تابعی چبیشف و مثلثاتی در مورد ورودی صوت یک خانم



شکل ۴-۲۸: منحنی MSE در حالت $Q=10$ باحافظه برای مقایسه پیوند تابعی چبیشف و مثلثاتی در مورد ورودی صوت یک خانم

۷-۴ ارزیابی پنجم

در این بخش پیوندهای تابعی مثلثاتی، چبیشف و لژاندر، برای الگوریتم‌های مختلف در حذف پژواک غیرخطی مرتبه دو آورده شده است و زمان اجرای برنامه‌ها را برای تعداد مختلفی از پیوندهای تابعی در جدول ۴-۴ نشان می‌دهیم. برای داشتن مقایسه سرعت اجرای برنامه‌ها ما از یک سناریوی استفاده می‌کنیم که رابطه (۱-۴) و رابطه (۲-۴) را درگیر می‌کند و پژواک مرتبه ۲ را تولید می‌کند. همان‌گونه که در جدول ۴-۴ مشاهده می‌شود الگوریتم چبیشف در حالت بدون حافظه دارای سرعت بالایی است و همان‌گونه که در ارزیابی‌های قبل مشخص شد این الگوریتم دارای حذف پژواک و سرعت همگرایی بالایی می‌باشد.

جدول ۴-۴: زمان اجرا برحسب ثانیه

نام الگوریتم	Q=1	Q=5	Q=10	Q=15
SFLAF	۸/۲	۲۶/۳۴	۴۶/۳	۶۴/۸
ChebyFLAF	۵/۲۶	۱۶/۱۲	۲۶/۱۵	۳۶/۵۱
LegFLAF	۵/۴۶	۱۶/۵۷	۲۶/۶۴	۳۶/۷
CFLAF	۱۱/۲۵	۲۹/۵۲	۵۱/۸۷	۶۹/۲
PSFLAF	۱۲/۴۴	۴۶/۷۸	۹۴/۲۸	۱۱۹/۱۳
PChebyFLAF	۸/۱۸	۲۸/۸۵	۴۷	۶۵/۶۸
PLegFLAF	۸/۵	۳۰/۵	۴۹/۲۵	۷۱/۱۲
SFLAF with Memory1	۱۸۶/۱	۲۵۳/۳۵	۳۵۳/۵۰	۳۴۲/۴۶
ChebyFLAF with Memory1	۲۰۷/۴۶	۲۷۲/۱۲	۲۷۴/۷۹	۳۲۰/۴۴

فصل پنجم

نتیجه‌گیری و پیشنهادها

۵-۱ مروری بر کارهای انجام شده

در این پایان نامه حذف پژواک غیرخطی در سیستم‌های ارتباطی با در نظر گرفتن انواع مختلف پژواک مطالعه و بررسی گردید.

در فصل دوم، به بررسی سابقه موضوع حذف پژواک پرداخته شد. با توجه به گستردگی موضوع و ارتباط آن با مفاهیم و فناوری‌های گوناگون انواع مدل پژواک، انواع حذف کننده‌های پژواک، چگونگی فعال بودن سیستم حذف پژواک و نحوه‌ی سنجش میزان پژواک حذف شده، سابقه آن و کارهای انجام شده در این حوزه مورد بحث قرار گرفت.

در فصل سوم، نسل جدیدی از حذف کننده‌های پژواک خطی به نام پیوندهای تابعی تشریح شد که در آن دو مدل فیلتر وفقی پیوند تابعی مجزا و فیلتر وفقی تابعی پیوند اشتراکی بررسی شد و بر اساس آن مدل‌های پیشنهادی خود را که شامل فیلتر وفقی پیوند تابعی چبیشف، فیلتر پیوند تابعی لژاندر و مدل موازی الگوریتم‌های پیشنهادی و همچنین مدل با حافظه یک را ارائه شد.

در فصل چهارم، در ابتدا به مدل‌های ایجاد پژواک مرتبه ۲ و مرتبه ۵ و تابع تبدیل ساخت سیگنال رنگی به کار گرفته شده اشاره شد. الگوریتم‌های ارائه شده برای حذف پژواک بر اساس سناریوهای شبیه سازی مدل تله کنفرانس در ۵ ارزیابی با شرایط یکسان و ایجاد مدل پژواک غیر خطی مورد ارزیابی قرار گرفتند. تعداد پیوندهای تابعی برابر برای هر آزمایش در نظر گرفته می‌شود. مقایسه و نتیجه‌گیری ارزیابی‌ها توسط ارائه منحنی‌های شاخص‌های بهبود حذف پژواک ERLE و میانگین مربع خطا در فیلتر وفقی MSE برای هر آزمایش عملکرد الگوریتم‌های را نشان می‌دهد.

۵-۲ نتایج

کارهای انجام شده و نتایج به دست آمده در این پایان نامه را می‌توان به طور خلاصه به صورت زیر

مطرح نمود:

- در این پایان نامه کلاس جدیدی از الگوریتم‌های فیلتر وفقی غیرخطی بر پایه مدل فیلترهای وفقی پیوند تابعی برای مدل‌سازی غیرخطی کانال‌های صوتی معرفی شده است. به خاطر معماری انعطاف‌پذیر مبتنی بر این پایه، یک راه حل مؤثر برای مدل کردن پژواک‌های غیرخطی که روی سیگنال صوت اثر می‌گذارند ارائه شده است.
- از الگوریتم‌هایی که بر اساس پیوندهای تابعی چبیشف بوده‌اند می‌توان نتیجه گرفت که الگوریتم‌های ChebyFLAF, ChebyCFLAF, PChebyFLAF در هر دو حالت مجزا و اشتراکی با بالا رفتن تعداد پیوندهای تابعی بهترین عملکرد در حذف پژواک را دارد، همچنین مشاهده شد که در حالت پیوند اشتراکی با وجود پژواک خطی در لحظه اول، پیوند تابعی چبیشف نسبت به مثلثاتی عملکرد بهتری هم در حذف پژواک خطی و هم در حذف پژواک غیرخطی داشته است که می‌توان نتیجه گرفت معماری انعطاف‌پذیری دارد.
- از الگوریتم‌هایی که بر اساس پیوند تابعی مثلثاتی به دست آمده می‌توان نتیجه گرفت الگوریتم‌های SFLAF, CFLAF, PSFLAF در تعداد کم پیوندهای تابعی مثلاً ($Q = 1$ تا $Q = 5$) در حالت‌های مجزا و اشتراکی بهبود عملکرد قابل چشم‌گیری دارد می‌توان مشاهده کرد این الگوریتم در بسیاری از منحنی‌های آورده شده عملکرد بهتری را نشان داده و همچنین قابلیت تخمین زدن و حذف پژواک در هر دو حالت خطی و غیرخطی را دارد همچنین این الگوریتم‌ها در تعداد پیوندهای بالاتر هم بهبود قابل توجهی دارند ولی در مقایسه با الگوریتم‌های چبیشف سرعت همگرایی کمتری دارند.
- از الگوریتم‌هایی که بر اساس پیوند تابعی لژاندر بدست آمده می‌توان نتیجه گرفت الگوریتم‌های LegFLAF, PLegFLAF در حذف پژواک عملکرد نسبتاً خوبی دارند و از هر دو حالت مثلثاتی و چبیشف ضعیف‌تر عمل می‌کنند با این حال سرعت همگرایی بالایی دارد.
- تمامی الگوریتم‌های ذکر شده در پژواک غیرخطی برتری خودشان را نسبت به فیلتر وفقی

خطی در منحنی‌ها نشان دادند و بسته به کاربرد می‌توان از هر کدام استفاده کرد. الگوریتم‌هایی که دارای دو فیلتر موازی غیرخطی هستند بهتر است برای سیستم‌هایی استفاده شود که در آن‌ها حذف پژواک غیرخطی مد نظر است.

۳-۵ پیشنهادها

- برای تطبیق فیلترها از دیگر الگوریتم‌های افقی مانند حداقل مربعات بازگشتی^۱ یا الگوریتم تناسبی نرمال حداقل میانگین مربعات^۲ می‌توان استفاده نمود.
- سیستم را برای پژواک مرتبه بالاتر از ۵ بررسی نموده و بهتر است برای سنجش سرعت از اندازه گام‌های مختلف استفاده شود.
- برای بالا رفتن سرعت همگرایی می‌توان از یک اندازه گام متغیر^۳ برای حالت تطبیق استفاده کرد.
- می‌توان بسط‌های دیگر را برای پیوندهای تابعی استفاده کرد در این صورت می‌توان تغییرات عملکردی را بررسی کرد.

¹ Recursive least squares

² proportionate normalized least-mean-square

³ Variable Step-Size

مراجع

- [1] A. H. Sayed, *Adaptive filters*: John Wiley & Sons, 2011.
- [2] J. Benesty, T. Gänslér, D. R. Morgan, M. M. Sondhi, and S. L. Gay, *Advances in network and acoustic echo cancellation*: Springer, 2001.
- [3] C. Paleologu, J. Benesty, and S. Ciochina, "Sparse adaptive filters for echo cancellation," *Synthesis Lectures on Speech and Audio Processing*, vol. 6, pp. 1-124, 2010.
- [4] A. Uncini, A. Nalin, and R. Parisi, "Acoustic echo cancellation in the presence of distorting loudspeakers," in *Signal Processing Conference, 2002 11th European*, pp. 1-4, 2002.
- [5] T. G. Burton, R. Goubran, and F. Beaucoup, "Nonlinear system identification using a subband adaptive Volterra filter," *Instrumentation and Measurement, IEEE Transactions on*, vol. 58, pp. 1389-1397, 2009.
- [6] L .A. Azpicueta-Ruiz, M. Zeller, A. R. Figueiras-Vidal, J. Arenas-García, and W. Kellermann, "Adaptive combination of Volterra kernels and its application to nonlinear acoustic echo cancellation," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 19, pp. 97-110, 2011.
- [7] A. Birkett and R. Goubran, "Limitations of handsfree acoustic echo cancellers due to nonlinear loudspeaker distortion and enclosure vibration effects," in *Applications of Signal Processing to Audio and Acoustics, 1995., IEEE ASSP Workshop on*, pp. 103-106, 1995.
- [8] H. Dai and W.-P. Zhu, "Compensation of loudspeaker nonlinearity in acoustic echo cancellation using raised-cosine function," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 53, pp. 1190-1194 ,2006.
- [9] J. Fu and W.-P. Zhu, "A nonlinear acoustic echo canceller using sigmoid transform in conjunction with RLS algorithm," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 55, pp. 1056-1060, 2008.
- [10] R. Parisi, R. Russo, M. Scarpiniti, and A. Uncini, "Performance of acoustic nonlinear echo cancellation in the presence of reverberation," in *Proc. Int. Symp. Front. Res. Speech Music (FRSM'09)*, pp. 106-111, 2009.

- [11] J. Costa, A. Lagrange, and A. Arliaud, "Acoustic echo cancellation using nonlinear cascade filters," in *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, pp. V-389-92, 2003.
- [12] M. Scarpiniti, D. Comminiello, R. Parisi, and A. Uncini, "Comparison of Hammerstein and Wiener systems for nonlinear acoustic echo cancelers in reverberant environments," in *Digital Signal Processing (DSP), 2011 17th International Conference on*, pp. 1-6, 2011.
- [13] A. Guérin, G. Faucon, and L. Bouquin-Jeannés, "Nonlinear acoustic echo cancellation based on Volterra filters," *Speech and Audio Processing, IEEE Transactions on*, vol. 11, pp. 672-683, 2003.
- [14] A. Stenger, L. Trautmann, and R. Rabenstein, "Nonlinear acoustic echo cancellation with 2nd order adaptive Volterra filters," in *Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on*, pp. 877-880, 1999.
- [15] A. Fermo, A. Carini, and G. L. Sicuranza, "Simplified Volterra filters for acoustic echo cancellation in GSM receivers ",in *Signal Processing Conference, 2000 10th European*, 2000, pp. 1-4.
- [16] Y. Pao, "Adaptive pattern recognition and neural networks," 1989.
- [17] Y. Huang, J. Chen, and J. Benesty, "Immersive audio schemes," *Signal Processing Magazine, IEEE*, vol. 28, pp. 20-32,2011.
- [18] M. Sondhi and A. Presti, "A Self-Adaptive Echo Canceller," *Bell System Technical Journal*, vol. 45, pp. 1851-1854, 1966.
- [19] M. M. Sondhi, "Adaptive echo cancelation for voice signals," in *Springer Handbook of Speech Processing*, ed: Springer, pp. 903-928, 2008.
- [20] D. D. Falconer, "Adaptive filter theory and applications," in *Analysis and Optimization of Systems*, ed: Springer, pp. 161-188, 1980.
- [21] K. Sheno, *Digital signal processing in Telecommunications*: Prentice-Hall, Inc., 1995.

- [22] M. Reed and M. Hawksford, "Acoustic echo cancellation with the fast affine projection," in *Audio and Music Technology: The Challenge of Creative DSP (Ref. No. 1998/470)*, IEE Colloquium on, pp. 16/1-16/8, 1998.
- [23] C. Breining, P. Dreiscitel, E. Hänsler, A. Mader, B. Nitsch, H. Puder, *et al.*, "Acoustic echo control. An application of very-high-order adaptive filters," *Signal Processing Magazine, IEEE*, vol. 16, pp. 42-69, 1999.
- [24] A. P. Liavas and P. Regalia, "Acoustic echo cancellation: Do IIR models offer better modeling capabilities than their FIR counterparts?," *Signal Processing, IEEE Transactions on*, vol. 46, pp. 2499-2504, 1998.
- [25] S. S. Haykin, *Adaptive filter theory*: Pearson Education India, 2008.
- [26] D. Comminiello, M. Scarpiniti, L. Azpicueta-Ruiz, J. Arenas-Garcia, and A. Uncini, "Functional link adaptive filters for nonlinear acoustic echo cancellation," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 21, pp. 1502-1512, 2013.
- [27] D. P. Das and G. Panda, "Active mitigation of nonlinear noise processes using a novel filtered-s LMS algorithm," *Speech and Audio Processing, IEEE Transactions on*, vol. 12, pp. 313-322, 2004.
- [28] G. L. Sicuranza and A. Carini, "A generalized FLANN filter for nonlinear active noise control," *IEEE transactions on audio, speech, and language processing*, vol. 19, pp. 2412-2417, 2011.
- [29] D. Comminiello, L. Azpicueta-Ruiz, M. Scarpiniti, A. Uncini, and J. Arenas-Garcia, "Functional link based architectures for nonlinear acoustic echo cancellation," in *Hands-free Speech Communication and Microphone Arrays (HSCMA), 2011 Joint Workshop on*, pp. 180-184, 2011.
- [30] D. Comminiello, M. Scarpiniti, R. Parisi, and A. Uncini, "A functional link based nonlinear echo canceller exploiting sparsity," in *Proc. Int. Workshop Acoust. Echo, Noise Control (IWAENC'10)*, 2010.
- [31] J. Arenas-García, M. Martínez-Ramón, Á. Navia-Vázquez, and A. R.

- Figueiras-Vidal, "Plant identification via adaptive combination of transversal filters," *Signal Processing*, vol. 86, pp. 2430-2438, 2006.
- [32] J. Arenas-García, A. R. Figueiras-Vidal, and A. H. Sayed, "Mean-square performance of a convex combination of two adaptive filters," *Signal Processing, IEEE Transactions on*, vol. 54, pp. 1078-1090, 2006.
- [33] S. S. Haykin, S. S. Haykin, S. S. Haykin, and S. S. Haykin, *Neural networks and learning machines* vol. 3: Pearson Education Upper Saddle River, 2009.
- [34] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of control, signals and systems*, vol. 2, pp. 303-314, 1989.
- [35] H. Zhao and J. Zhang, "Adaptively combined FIR and functional link artificial neural network equalizer for nonlinear communication channel," *Neural Networks, IEEE Transactions on*, vol. 20, pp. 665-674, 2009.
- [36] J. C. Patra, R. N. Pal, B. Chatterji, and G. Panda, "Identification of nonlinear dynamic systems using functional link artificial neural networks," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 29, pp. 254-262, 1999.
- [37] A. Namatame and Y. Kimata, "Improving the generalising capabilities of a backpropagation network," *The International Journal of Neural Networks Research & Applications*, vol. 1, pp. 86-94, 1989.
- [38] L. Stanković, "Performance analysis of the adaptive algorithm for bias-to-variance tradeoff," *Signal Processing, IEEE Transactions on*, vol. 52, pp. 1228-1234, 2004.
- [39] J. C. Patra, W. C. Chin, P. K. Meher, and G. Chakraborty, "Legendre-FLANN-based nonlinear channel equalization in wireless communication system," in *Systems, Man and Cybernetics, 2008. SMC 2008. IEEE International Conference on*, pp. 1826-1831, 2008.
- [40] L. Azpicueta-Ruiz, A. R. Figueiras-Vidal, and J. Arenas-García, "A normalized adaptation scheme for the convex combination of two

- adaptive filters," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pp. 3301-3304, 2008.
- [41] S. S. Kozat and A. C. Singer, "Multi-stage adaptive signal processing algorithms," in *Sensor Array and Multichannel Signal Processing Workshop. 2000. Proceedings of the 2000 IEEE*, pp. 380-384, 2000.
- [42] T.-T. Lee and J.-T. Jeng, "The Chebyshev-polynomials-based unified model neural networks for function approximation," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 28, pp. 925-935, 1998.
- [43] A. Namatame and N. Ueda, "Pattern classification with Chebychev neural networks," *International Journal of Neural Networks*, vol. 3, pp. 23-23, 1992.
- [44] J. C. Patra and A. C. Kot, "Nonlinear dynamic system identification using Chebyshev functional link artificial neural networks," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 32, pp. 505-511, 2002.
- [45] W.-D. Weng, C.-S. Yang, and R.-C. Lin, "A channel equalizer using reduced decision feedback Chebyshev functional link artificial neural networks," *Information Sciences*, vol. 177, pp. 2642-2654, 2007.
- [46] H. Zhao and J. Zhang, "Functional link neural network cascaded with Chebyshev orthogonal polynomial for nonlinear channel equalization ",*Signal Processing*, vol. 88, pp. 1946-1957, 2008.
- [47] S. Dehuri and S.-B. Cho, "A comprehensive survey on functional link neural networks and an adaptive PSO–BP learning for CFLNN," *Neural Computing and Applications*, vol. 19, pp. 187-205, 2010.
- [48] S.-S. Yang and C.-S. Tseng, "An orthogonal neural network for function approximation," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 26, pp. 779-785, 1996.
- [49] J. C. Patra and A. Van den Bos, "Modeling of an intelligent pressure sensor using functional link artificial neural networks," *ISA transactions*, vol. 39, pp. 15-27, 2000.
- [50] J. C. Patra, W. B. Poh, N. S. Chaudhari, and A. Das, "Nonlinear

channel equalization with QAM signal using Chebyshev artificial neural network ",in *Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on*, pp. 3214-3219, 2005.

- [51] D. Campbell, "Roomsim: a MATLAB simulation shoebox room acoustics," ed, 2007 .

Abstract

Presence of nonlinear component of circuit affects the performance of conventional acoustic echo canceller and has negative impact on the quality of voice telecommunication. To solve this problem, nonlinear acoustic echo cancellers are used which are obtained from modeling of nonlinear paths and sound improvement. In this thesis, a new class of nonlinear adaptive algorithms based on functional link adaptive filters is proposed for nonlinear models of acoustic channel. Due to the flexible architecture, these algorithms are an efficient solutions to model effective nonlinearities on acoustic signal. Chebyshev, trigonometric and Legendre mathematical functions are used for input expansion. The separation between the linear and nonlinear components in collaborative and split adaptive filters, leading to two different adaptive filters. The linear one with the purpose of estimating the acoustic impulse response, and the nonlinear one just with duty of nonlinearities modeling which is robust against various degrees of nonlinearity. With this method, performance of convergence is improved dramatically. The Collaborative model with prevention of gradient nonlinear noises when the system is linear, has a good performance in any echo conditions. Thus, architectures based on this foundation can be considered as an effective solution for acoustic channels model. The results obtained in this thesis show that adaptive algorithms based on Chebyshev and trigonometric functional links have a good performance for nonlinear echo cancellation purpose and can correct the echo up to 50dB.

Keywords: Nonlinear acoustic echo, echo canceller, adaptive filter, Chebyshev functional link, Chebyshev and trigonometric functional link, Legendre functional link.



Shahrood University of Technology

Electrical and Robotics Engineering school

Adaptive Non liner Echo Cancellation by Soft Computing Algorithms

Alireza Nezamdoost

Supervisor:

Dr. Omid Reza Maarouzi

February 2016