



دانشگاه سندھ

دانشکده برق و رباتیک

پایان نامه کارشناسی ارشد برق - کنترل

عنوان

طراحی کنترلر حذف نویز به روش فعال، به کمک الگوریتم
مبتنی بر جمعیت اجتماع ذرات

اساتید راهنما :

دکتر حداد ظریف

دکتر احمدی فرد

دانشجو :

حمیدرضا مدرس

تابستان ۸۵

تقدیم به پدر و مادر بسیار عزیزم

و

روح گرامی دوست ارجمندم، محسن رادمرد

بر خود واجب می دانم زحمات اساتید بزرگوار، دکتر حداد
ظریف و دکتر احمدی فرد را ارج نهاده و با خلوص نیت از
تلاشهای این اساتید در مدت انجام پایان نامه قدردانی و
تشکر نمایم. موفقیت روز افزون همه اساتید ارجمند در
مسیر تعلیم و تربیت و پیشبرد علم و دانش را از خداوند
متعال خواستارم

	الگوریتم بهترین محلی	۳-۳-۳
۵۰	ارزش گذاری	۴-۳
۵۰	همگرایی	۵-۳
۵۱	پارامترهای PSO	۶-۳
۵۵	تغییر ساختار استاندارد PSO	۷-۳
۵۹	غلبه بر همگرایی زودرس	۸-۳
۵۹	افزایش سرعت همگرایی	۹-۳
۶۰	مواجه شدن با مسائل دینامیکی	۱۰-۳
۶۲	سیستم ANC با الگوریتم وفقی مبتنی جمعیت PSO	فصل چهارم
۶۳	مقدمه	-
۶۴	سیستم ANC ، با بکارگیری الگوریتم وفقی مبتنی بر جمعیت PSO	۱-۴
۷۵	نتایج شبیه سازی	فصل پنجم
۷۶	شبیه سازی سیستم ANC خطی	۱-۵
۷۶	نتایج حاصل از روش filtered-x LMS برای سیستم ANC خطی با کنترلر خطی FIR	۱-۱-۵
۸۱	نتایج حاصل از روش PSO برای سیستم ANC خطی با کنترلر خطی FIR	۲-۱-۵
۸۷	شبیه سازی سیستم ANC غیرخطی	۲-۵
۸۹	نتایج حاصل از الگوریتم مبتنی بر کاهش گرادیان ۱ و بر پایه filtered-x برای سیستم ANC غیرخطی با کنترلر شبکه عصبی	۱-۲-۵
۹۲	نتایج حاصل از الگوریتم PSO برای سیستم ANC غیرخطی با کنترلر شبکه عصبی	۲-۲-۵
۹۶	شبیه سازی روش پیشنهادی برای حذف نویز واقعی حاصل از ماشین تراش و سشوار	۳-۵
۹۷	مقایسه روشهای filtered-x LMS و PSO پیشنهادی از لحاظ پیچیدگی محاسباتی	۴-۵
۹۷	مقایسه دو روش، در یک سیستم ANC با مدل خطی	۱-۴-۵
۱۰۰	مقایسه پیچیدگی محاسباتی دو روش، در سیستم ANC غیرخطی	۲-۴-۵
۱۰۱	نتیجه گیری و پیشنهادات	فصل ششم

¹. steepest descent algorithms

۱۰۱

۱۰۴

۱۰۶

نتیجه گیری

پیشنهادات

۱-۶

۲-۶

مراجع

فهرست شکل ها

شماره صفحه	عنوان شکل	شکل
۳	حذف نویز به روش غیرفعال	شکل ۱-۱
۴	حذف نویز به روش فعال	شکل ۲-۱
۶	شماتیک سیستم ANC تک کاناله	شکل ۳-۱
۷	شماتیک حذف نویز به روش پیشخور برای نویزهای باند وسیع	شکل ۴-۱
۷	بلوک دیاگرام سیستم ANC تک کاناله به روش پیشخور	شکل ۵-۱
۹	وجود فیدبک داخلی در سیستم حذف نویز	شکل ۶-۱
۱۰	از بین بردن اثر فیدبک داخلی	شکل ۷-۱
۱۱	حذف نویز برای سیگنالهای شبه پرودیگ	شکل ۸-۱
۱۳	شماتیک حذف نویز به روش پسخور	شکل ۹-۱
۱۳	بلوک دیاگرام حذف نویز به روش پسخور	شکل ۱۰-۱
۲۰	مدل ریاضی یک نورو	شکل ۱-۲
۲۰	توپولوژی پرسپترون چند لایه (MLP)	شکل ۲-۲
۲۱	مهمترین کاربرد شبکه عصبی	شکل ۳-۲
۲۴	فیلتر FIR ساخته شده از شبکه عصبی	شکل ۴-۲
۲۵	ساختار الگوریتم وقتی LMS	شکل ۵-۲
۲۸	ساختار الگوریتم وقتی LMS با در نظر گرفتن مسیر ثانویه	شکل ۶-۲
۳۰	($\hat{\theta}$) الگوریتم استاندارد LMS در مقابل (b) الگوریتم filtered-x LMS	شکل ۷-۲
۳۲	بلوک دیاگرام سیستم ANC با کنترلر غیر خطی	شکل ۸-۲
۳۴	بلوک دیاگرام (a) نورو لایه مخفی (b) لایه ورودی	شکل ۹-۲
۴۴	توپولوژیهای مطرح در PSO	شکل ۱-۳
۶۷	بلوک دیاگرام سیستم ANC تک کاناله با استفاده از الگوریتم PSO	شکل ۱-۴
۷۷	سیگنال خطای باقیمانده در حوزه فرکانس، برای نویز اولیه ۱۰۰Hz، قبل و بعد از اعمال کنترلر به روش filtered-x LMS	شکل ۱-۵(a)
۷۷	سیگنال خطای باقیمانده در حوزه زمان برای نویز اولیه ۱۰۰Hz با توان ۰.۵، با بکارگیری روش filtered-x LMS	شکل ۱-۵(b)
۷۸	سیگنال خطای باقیمانده در حوزه فرکانس برای نویز اولیه ۲۰۰Hz، قبل و بعد از اعمال کنترلر به روش filtered-x LMS	شکل ۲-۵(a)
۷۸	سیگنال خطای باقیمانده برای نویز اولیه ۲۰۰Hz، با بکارگیری روش filtered-x LMS	شکل ۲-۵(b)
۷۹	سیگنال خطای باقیمانده با تغییر ناگهانی فرکانس نویز از ۱۰۰Hz به ۲۰۰Hz با بکارگیری روش filtered-x LMS	شکل ۳-۵

۹۷	سیگنال خطای باقیمانده با تغییر ناگهانی فرکانس نویز از ۲۰۰Hz به ۱۰۰Hz با بکارگیری روش filtered-x LMS	شکل ۴-۵
۸۰	سیگنال خطای باقیمانده با افزایش توان نویز (از ۰.۵ به ۲) با بکارگیری روش filtered-x LMS	شکل ۵-۵
۸۰	سیگنال خطای باقیمانده با افزایش توان نویز (از ۰.۵ به ۴.۵) با بکارگیری روش filtered-x LMS	شکل ۶-۵
۸۰	سیگنال خطای باقیمانده با افزایش توان نویز (از ۰.۵ به ۸) و با بکارگیری روش filtered-x LMS	شکل ۷-۵
۸۰	سیگنال خطای باقیمانده با افزایش توان نویز (از ۰.۵ به ۲۰) و با بکارگیری روش filtered-x LMS	شکل ۸-۵
۸۲	سیگنال خطای باقیمانده در حوزه فرکانس برای نویز اولیه ۱۰۰Hz قبل و بعد از اعمال کنترلر به روش PSO	شکل ۹-۵ (a)
۸۲	سیگنال خطای باقیمانده در حوزه زمان، برای نویز اولیه ۱۰۰Hz با توان ۰.۵، با بکارگیری روش PSO	شکل ۹-۵ (b)
۸۲	سیگنال خطای باقیمانده در حوزه فرکانس برای نویز اولیه ۲۰۰Hz قبل و بعد از اعمال کنترلر به روش PSO	شکل ۱۰-۵ (a)
۸۲	سیگنال خطای باقیمانده در حوزه زمان، برای نویز اولیه ۲۰۰Hz، با توان ۰.۵، با بکارگیری روش PSO	شکل ۱۰-۵ (b)
۸۳	تغییرات ضریب وزن اینرسی به ازای تغییر ناگهانی فرکانس نویز، با بکارگیری روش PSO	شکل ۱۱-۵ (a)
۸۴	سیگنال منحنی تغییرات یکی از ضرایب فیلتر با تغییر ناگهانی در فرکانس نویز و با بکارگیری روش PSO	شکل ۱۱-۵ (b)
۸۴	منحنی تغییرات یکی از ضرایب فیلتر با تغییر ناگهانی در فرکانس نویز و با بکارگیری روش PSO	شکل ۱۱-۵ (c)
۸۴	منحنی تغییرات یکی از ضرایب فیلتر با تغییر ناگهانی در فرکانس نویز و با بکارگیری روش PSO	شکل ۱۱-۵ (d)
۸۴	تغییرات ضریب وزن اینرسی به ازای تغییر ناگهانی فرکانس نویز، با بکارگیری روش PSO	شکل ۱۲-۵ (a)
۸۵	سیگنال خطای باقیمانده با تغییر ناگهانی فرکانس نویز با بکارگیری روش PSO	شکل ۱۲-۵ (b)
۸۵	منحنی تغییرات یکی از ضرایب فیلتر با تغییر ناگهانی در فرکانس نویز و با بکارگیری روش PSO	شکل ۱۲-۵ (c)
۸۵	منحنی تغییرات یکی از ضرایب فیلتر با تغییر ناگهانی در فرکانس نویز و با بکارگیری روش PSO	شکل ۱۲-۵ (d)
۸۶	سیگنال خطای باقیمانده با افزایش ناگهانی در توان آن (از ۰.۵ به ۱۲.۵) و با بکارگیری روش PSO	شکل ۱۳-۵ (a)
۸۷	تغییرات ضریب وزن اینرسی به ازای تغییر توان نویز (از ۰.۵ به ۱۲.۵) و با بکارگیری روش PSO	شکل ۱۳-۵ (b)
۸۷	تغییرات یکی از ضرایب فیلتر با تغییر ناگهانی توان نویز (از ۰.۵ به ۱۲.۵) و با بکارگیری روش PSO	شکل ۱۳-۵ (c)
۸۷	تغییرات ضریب وزن اینرسی به ازای تغییر ناگهانی توان نویز (از ۰.۵ به ۵۰) با بکارگیری روش PSO	شکل ۱۴-۵ (a)
۸۷	سیگنال خطای باقیمانده با تغییر ناگهانی توان نویز (از ۰.۵ به ۵۰)، با بکارگیری روش PSO	شکل ۱۴-۵ (b)
۸۹	سیگنال خطای باقیمانده در حوزه فرکانس برای سیستم ANC غیر خطی و نویز اولیه ۱۰۰Hz، قبل و بعد از اعمال روش filtered-x LMS برای کنترلر غیر خطی	شکل ۱۵-۵ (a)
۹۰	سیگنال خطای باقیمانده برای سیستم ANC غیر خطی و نویز اولیه ۱۰۰Hz، با بکارگیری روش filtered-x LMS برای کنترلر غیر خطی	شکل ۱۵-۵ (b)
۹۰	سیگنال خطای باقیمانده در حوزه فرکانس برای سیستم ANC غیر خطی و نویز اولیه ۲۰۰Hz، قبل و بعد از اعمال روش filtered-x LMS برای کنترلر غیر خطی	شکل ۱۶-۵ (a)
۹۰	سیگنال خطای باقیمانده برای سیستم ANC غیر خطی و نویز اولیه ۲۰۰Hz، با بکارگیری روش filtered-x LMS برای کنترلر غیر خطی	شکل ۱۶-۵ (b)
۹۱	سیگنال خطای باقیمانده در حوزه فرکانس برای سیستم ANC غیر خطی و نویز اولیه ۲۰۰Hz، قبل و بعد از اعمال روش filtered-x LMS برای کنترلر خطی FIR	شکل ۱۷-۵ (a)

- شکل ۵-۱۷ (b) سیگنال خطای باقیمانده برای سیستم ANC غیر خطی و نویز اولیه ۲۰۰Hz، با بکارگیری روش filtered-x LMS برای کنترلر خطی FIR
- شکل ۵-۱۸ (a) سیگنال خطای باقیمانده در حوزه فرکانس، برای سیستم ANC غیر خطی و نویز اولیه ۱۰۰Hz، قبل و بعد از اعمال روش PSO برای کنترلر غیر خطی
- شکل ۵-۱۸ (b) سیگنال خطای باقیمانده برای سیستم ANC غیر خطی و نویز اولیه ۱۰۰Hz، با بکارگیری روش PSO برای کنترلر غیر خطی
- شکل ۵-۱۹ (a) سیگنال خطای باقیمانده در حوزه فرکانس برای سیستم ANC غیر خطی و نویز اولیه ۲۰۰Hz، قبل و بعد از اعمال روش PSO برای کنترلر غیر خطی
- شکل ۵-۱۹ (b) سیگنال خطای باقیمانده برای سیستم ANC غیر خطی و نویز اولیه ۲۰۰Hz، با بکارگیری روش PSO برای کنترلر غیر خطی
- شکل ۵-۲۰ (a) سیگنال خطای باقیمانده در حوزه فرکانس برای سیستم ANC غیر خطی و نویز اولیه ۲۰۰Hz، قبل و بعد از اعمال روش PSO برای کنترلر خطی FIR
- شکل ۵-۲۰ (b) سیگنال خطای باقیمانده برای سیستم ANC غیر خطی و نویز اولیه ۲۰۰Hz، با بکارگیری روش PSO برای کنترلر خطی FIR
- شکل ۵-۲۱ سیگنال خطای باقیمانده در حوزه فرکانس برای نویز واقعی حاصل از یک ماشین تراش، قبل و بعد از اعمال روش PSO برای کنترلر خطی FIR
- شکل ۵-۲۲ سیگنال خطای باقیمانده در حوزه فرکانس برای نویز واقعی حاصل از یک سشوار، قبل و بعد از اعمال روش PSO برای کنترلر خطی FIR

فصل اول :

مقدمه ای بر روشهای حذف
نویز

مقدمه

نویز صوتی یک تهدید بزرگ برای سلامت روانی انسان از طرف سازمان سلامت جهانی گزارش شده است. تحقیقات نشان داده که قرار گرفتن در معرض صدای شدید می تواند باعث کاهش همیشگی شنوایی انسان شود. همچنین تحقیقات اخیر روی میمونها نشان داده که اگر در معرض صدای شدید واقع شوند، فشار خون آنها با اثری که ماهها بعد از آن باقی می ماند، افزایش می یابد [1]. دیگر مطالعات نیز رابطه بین در معرض نویز قرار گرفتن و رفتارهای ضد اجتماعی از جمله عصبانیت شدید را نشان می دهد [2]. نویز شدید نه تنها برای سلامتی انسان مضر است، بلکه در واحدهای صنعتی می تواند باعث کاهش کیفیت تولید شود. به عنوان مثال، کارخانه های نیمه هادی برای اینکه هوای تمیز در محیطهای سرپسته کوچک داشته باشند از پنکه ها و بادبزنهای بزرگی استفاده می کنند. با این وجود، نویز تولید شده توسط آنها باعث ارتعاش در تجهیزات پروسه ساخت قطعات می شود که کیفیت تولید محصول را می تواند پائین بیاورد [3].

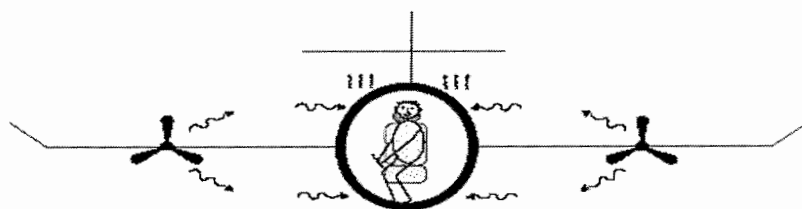
تعدادی از زمینه هایی که کاهش نویز در آنها برای انسان حائز اهمیت است عبارتند از:

- بزرگراهها و ریلهای قطار نزدیک به حومه شهرها
- بلند شدن و نشستن هواپیماها
- هواپیماهای نظامی و وسایل نقلیه
- ماشینهای صنعتی (برش چوب و فلز و...)

- وسایل نقلیه عمومی

به طور کلی دو نوع نویز در طبیعت وجود دارد که عبارتند از نویز باند وسیع^۱، که انرژی آن در تمام بازه فرکانسی پخش شده است و دیگری نویز باند باریک^۲، که بیشتر انرژی آن اطراف فرکانسهای خاصی متمرکز شده است.

روشهای موجود برای کاهش نویز نیز به دو کلاس فعال^۳ و غیر فعال^۴ تقسیم می شوند. در روش غیرفعال، برای کاهش نویز از مواد ایزوله و دفع کننده نویز در اطراف محیطی که قرار است از سروصدا مصون باشد استفاده می شود (شکل ۱-۱). از آنجائی که در این روش هیچ انرژی اضافی به سیستم تزریق نمی شود و کاهش نویز با استفاده از انعکاس و یا پخش کردن انرژی نویز توسط مواد ایزوله و دفع کننده صورت می گیرد، این روش را غیر فعال می نامند.



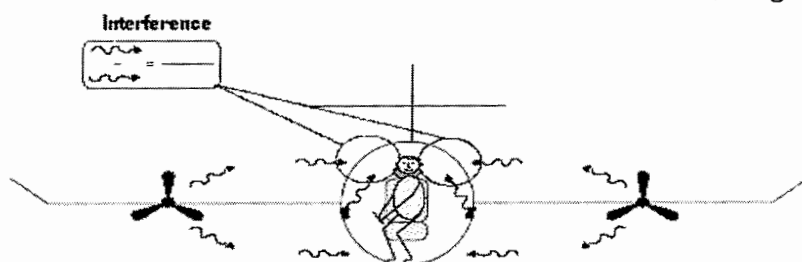
شکل ۱-۱- حذف نویز به روش غیرفعال

مزیت این روش این است که در فرکانسهای بالا (بیشتر از 1KHZ) می تواند به طور موثری نویز را کاهش دهد. با وجود این برای کاهش نویزهای با فرکانس کم از آنجائی که ضخامت مواد ایزوله کننده با

¹. broadband noise
². narrowband noise
³. active
⁴. passive

طول موج امواج مزاحم در ارتباط است، ضخامت آنها باید افزایش یابد. به عبارتی اگر نویزهای فرکانس پائین را ایزوله کنیم، اضافه کردن مواد ایزوله کننده سنگین و بزرگ را می طلبد که خود باعث اشغال فضای زیاد و همچنین صرف هزینه بالا خواهد شد. علاوه بر این، بکارگیری این مواد می تواند باعث کاهش بازده یک سیستم شود. به عنوان مثال، در صورت عدم استفاده آنها در کاهش صدای موتور اتومبیل، ۵ تا ۶ درصد در مصرف سوخت صرفه جوئی می شود [4].

برای کاهش نویزهای فرکانس پائین از روش دیگری به نام حذف نویز به روش فعال^۱ استفاده می-کنیم. اساس این روش، که به اختصار ANC نامیده می شود براین است که با یک منبع نویز ساختگی (منبع ثانویه^۲)، نویزی هم اندازه و هم فرکانس نویزاولیه (ناخواسته) ولی با فاز مخالف آن ایجاد کنیم که اثر متقابل این دو منبع نویز در منطقه سکوت^۳ (که قرار است حذف نویز در آن صورت گیرد)، می بایست حداقل گردد. (شکل ۱-۲)



شکل ۱-۲ - حذف نویز به روش فعال

¹ .Active Noise Cancellation (ANC)

² . Secondary source

³ . Silence zone

دلیل نامگذاری به روش فعال این است که در این روش برخلاف روش غیر فعال، برای ایجاد نویز ثانویه به سیستم انرژی تزریق می شود.

ایده این روش اولین بار توسط paul lueg در سال ۱۹۳۶ ارائه شد [5] ولی پیاده سازی واقعی در آن زمان به علت نبودن پردازنده های دیجیتالی سرعت بالا مقدور نبود. در دو دهه اخیر پیدایش پردازنده های دیجیتالی سرعت بالا باعث شده که ANC به طور موفقیت آمیزی در سیستم های واقعی پیاده سازی شود

۱-۱- مقدمه ای بر تئوری حذف نویز به روش فعال، برای سیستم تک کاناله

به طور کلی یک سیستم ANC شامل سه قسمت اساسی زیر می باشد:

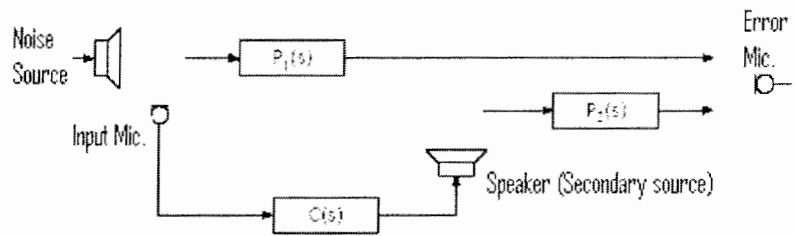
- سنسورها^۱: برای اندازه گیری نویز در نقطه (نقاطی) از فضا که می خواهیم نویز را حذف نماییم بکار می روند. در واقع خطای حاصل از جمع نویز ساختگی و اولیه توسط سنسورها اندازه گیری می شود .
 - کنترلر^۲: سیگنال کنترلی مناسب را برای راه انداز (بلندگو) فراهم می کند
 - راه اندازها^۳: ایجاد نویز ساختگی را بعهده دارند
- در سیستم تک کاناله، یک سنسور اندازه گیری نویز و یک راه انداز خواهیم داشت.

¹.sensors
².controller
³.actuators

۱-۲-۱- جبران سازی به روش پیشخور برای حذف نویز باند وسیع

شماتیک مسئله حذف نویز به روش پیشخور برای سیستم تک کاناله (سیستم با یک سنسور برای

اندازه گیری خطا و یک سنسور برای اندازه گیری نویز) در شکل زیر نشان داده شده است:



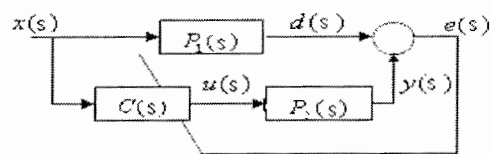
شکل ۱-۴- شماتیک حذف نویز به روش پیشخور برای نویزهای باند وسیع

اصولاً سیگنال نویز را نمی توان مستقیماً در محل منبع نویز به طور مؤثری کاهش داد. اغلب، نویز

را در محل های خاصی که می توانیم از آنها به عنوان منطقه سکوت^۱ یاد کنیم کاهش دهیم. به عنوان

مثال لازم است که نویز حاصل از موتور هواپیما و جریان هوای اطراف هواپیما (منابع نویز) را در محل

سرنشینان در کابین هواپیما کاهش دهیم.



شکل ۱-۵- بلوک دیاگرام سیستم ANC تک کاناله به روش پیشخور

^۱ .silence zone

بلوک دیاگرام سیستم ANC تک کاناله به روش پیشخور در شکل ۱-۵ نشان داده شده است. از روی شکل مشخص است که برای اینکه سیگنال نویز موجود در محل منبع نویز، یعنی x به منطقه سکوت برسد، از مسیری که آن را مسیر اولیه^۱ می نامیم باید بگذرد. تابع تبدیل این مسیر را $P_1(s)$ را می نامیم. سیگنالی که از منبع نویز و پس از عبور از این تابع تبدیل به منطقه سکوت می رسد را d می نامیم.

در روش پیشخور به یک تخمین از سیگنال نویز به عنوان سیگنال مرجع و ورودی کنترلر نیاز داریم. برای ساختن سیگنال مرجع، معمولاً از یک سنسور و یا میکروفون در مسیر نویز اولیه استفاده می شود.

سیگنال کنترلی u که توسط کنترلر ساخته می شود، منبع نویز ساختگی (یک بلندگو) را تحریک می کند. بدین وسیله نویز ثانویه تولید می گردد. این سیگنال نیز برای رسیدن به منطقه سکوت از یک مسیر ثانویه^۲ با تابع تبدیل $P_2(s)$ می گذرد. سیگنالی که پس از عبور از این تابع تبدیل به منطقه سکوت می رسد را y می نامیم.

جمع دو سیگنال y و d خطای باقیمانده (E) در منطقه سکوت را بدست می دهد و داریم:

$$E = x(s)P_1(s) + x(s)C(s)P_2(s) \quad (1-1)$$

¹ .primary path
² .secondary path

هدف سیستم کنترلی حداقل کردن این خطا می باشد. از صفر قرار دادن رابطه بالا تابع تبدیل

کنترلر به صورت زیر بدست می آید:

$$C(s) = -P_1(s)P_2(s)^{-1} \quad (2-1)$$

در سیستمهای حذف نویز صوتی، سرعت انتشار پائین نویز صوتی، تاخیرهای انتقالی را برای مسیرهای سیگنال که شامل مسیر اولیه و مسیر ثانویه می شود، نتیجه می دهد. در چنین مواردی توابع تبدیل $P_1(s)$ و $P_2(s)$ عنصر تاخیر فاز خواهند داشت. بنابراین هیچ تضمینی وجود ندارد که جبران ساز $C(s)$ که برای صفر کردن خطا طراحی شده است علی باشد.

به عنوان مثال، اگر $P_1(s) = e^{-sT_1}$ و $P_2(s) = e^{-sT_2}$ باشد، آنگاه طبق رابطه 2-1 تابع تبدیل

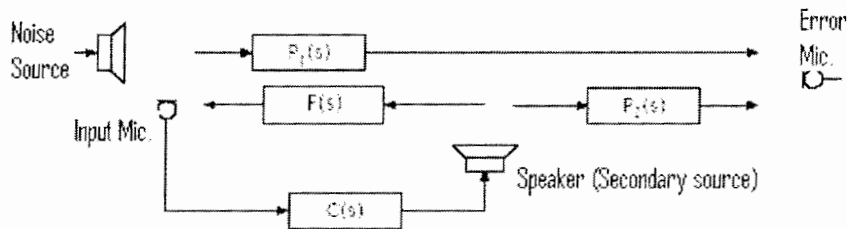
جبران ساز به صورت $C(s) = e^{-s(T_1-T_2)}$ خواهد بود که تنها در صورتی سیستم علی خواهد بود که

$T_1 > T_2$. یعنی باید سیستم طوری باشد که تاخیر ناشی از مسیر اولیه بیشتر از مسیر ثانویه باشد.

در حذف نویز صوتی به روش پیشخور نکته ای که باید در نظر گرفت این است که یک مسیر

فیدبک ($F(s)$) بین منبع نویز ثانویه (بلندگو) و سنسور اندازه گیری نویز اولیه (میکروفون) وجود دارد

(شکل 1-6) که این مسئله بصورت دو مشکل ظاهر می شود:



شکل 1-6- وجود فیدبک داخلی در سیستم حذف نویز

۱- تخمین سیگنال نویز را دچار مشکل می کند چون منبع نویز ثانویه نیز روی سنسور

تخمین سیگنال تاثیر می گذارد.

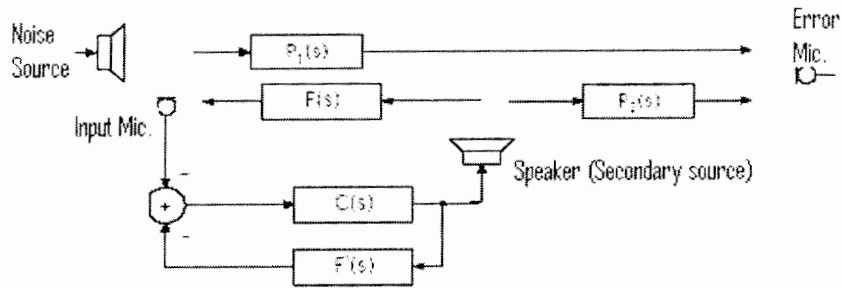
۲- اثر مضرتر این که فیدبک داخلی، پتانسیل اینکه حلقه بسته تشکیل شده از دو تابع

تبدیل $F(s), C(s)$ ناپایدار باشد را بوجود می آورد.

این مشکلات را می توان با انتخاب مناسب محل بلندگو (منبع نویز ثانویه) و میکروفون تشخیص

خطا کاهش داد. روش متفاوت دیگر برای کاهش تاثیر $F(s)$ این است که مطابق شکل ۱-۷، با تخمین

$F(s)$ (که $F(s)$ می نامیم)، مطابق سیستم زیر، اثر مخرب $F(s)$ را کاهش دهیم.

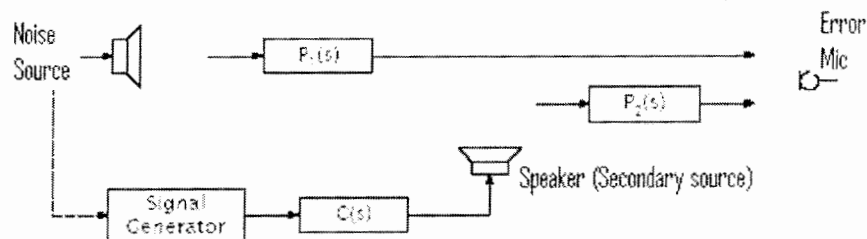


شکل ۱-۷ - از بین بردن اثر فیدبک داخلی

۱-۲-۲- جبرانسازی به روش پیشخور برای سیگنالهای نویز باندباریک

تعداد زیادی از منابع نویز، مانند موتورها، پنکه ها، کمپرسورها و انواع وسایل گردان، طبیعتاً نویز باند باریک تولید می کنند. این خصوصیت باعث ایجاد مزایای زیادی نسبت به نویزهای باند وسیع، در طراحی سیستم کنترلی ANC برای حذف نویز می شود.

در حقیقت هنگامی که نویز، یک سیگنال شبه پریودیک است می توان از یک سیگنال ژنراتور به جای بکارگیری میکروفون برای تخمین مستقیم نویز استفاده کرد (شکل ۱-۷). به عنوان مثال برای حذف نویز شبه پریودیک تولید شده توسط یک قطعه ماشینی گردان، می توان از یک سنسور غیر صوتی (به عنوان مثال یک تاکومتر) برای تولید یک موج سینوسی با فرکانس چرخش آن قطعه استفاده کرد که اطلاعاتی از نویز را هم در خود دارد.



شکل ۱-۸ - حذف نویز برای سیگنالهای شبه پریودیک

از مزایای استفاده از یک سنسور غیر صوتی^۱ می توان موارد زیر را نام برد:

- سنسورهای غیر صوتی اغلب قابل اطمینان تر می باشند و مانند میکروفونها نسبت به

تغییرات محیطی حساس نیستند.

^۱ non-acoustic sensors

- استفاده از یک سنسور غیر صوتی امکان بوجود آمدن مسیر فیدبکی که بین منبع نویز ثانویه (بلندگو) و میکروفون را حذف می کند.

- امکان حذف یک نویز مزاحم پریودیک بدون تداخل با سایر صداهای موجود در محیط میسر می گردد.

- محدودیت علی بودن که راجع به آن صحبت شد و لزوم تاخیر بیشتر در مسیر اولیه از مسیر ثانویه ($T1 > T2$) را حذف می کند. برای شرح این مطلب فرض کنید که سیگنال نویز، شبه پریودیک با دوره تناوب T باشد. خروجی جبران ساز می تواند با ضربی از دوره تناوب بدون تاثیر روی سیگنال خروجی تاخیر داده شود بنابراین تابع تبدیل جبران ساز را می توان به صورت زیر نوشت:

$$C(s) = e^{-s(T1-T2)} e^{-skT} = e^{-s(T1+kT-T2)} \quad (3-1)$$

که در آن، $k \in \{0,1,2,\dots\}$. با انتخاب مناسب k ، می توان از علی بودن جبران ساز مطمئن شد.

- برای سیگنالهای نویز با پهنای باند باریک، درجه جبران ساز می تواند به طور قابل ملاحظه ای پائین بیاید.

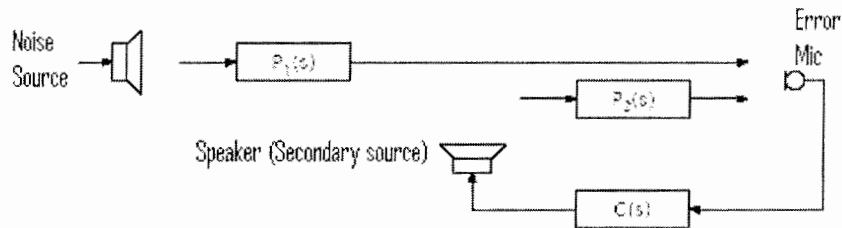
۱-۲-۳- حذف نویز با استفاده از حلقه پسخور (جبران سازی به روش پسخور)

همانطور که قبلا هم گفته شد در روش پیشخور نیاز به یک تخمین از نویز لازم می باشد. بر خلاف

روش پیشخور، در روش پسخور هیچ دانشی از نویز لازم نداریم.

از کاربردهای این روش می توان مواردی که امکان اندازه گیری و تخمین سیگنال نویز حاصل از منبع نویز وجود ندارد را نام برد. به عنوان مثال اگر بخواهیم نویز ناشی از جریان هوای اطراف کابین هواپیما را در داخل کابین هواپیما حذف کنیم، منبع نویز در تمام سطح هواپیما پخش شده و ممکن نیست که یک سیگنال مرجع مناسب را فراهم کنیم.

نمای شماتیک روش پسخور در شکل ۹-۱ نشان داده شده است.



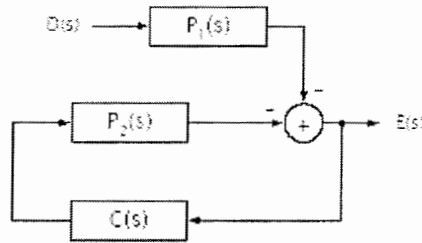
شکل ۹-۱ - شماتیک حذف نویز به روش پسخور

یک محدودیت اساسی این روش به خاطر تاخیر موجود در مسیر ثانویه حاصل می شود. در حقیقت تاخیر بین سنسور تشخیص خطا و منبع نویز ثانویه را نمی توان با یک فیلتر علی جبران کرد و این روش برای نویزهای باند وسیع در هر سیستمی که تاخیر مسیر ثانویه قابل ملاحظه است، مناسب نیست. ولی اگر نویز شبه پریودیک باشد این ساختار می تواند حتی برای حالت تاخیر زیاد در مسیر ثانویه مناسب باشد [4].

بلوک دیاگرام روش پسخور در شکل ۱۰-۱ نشان داده شده است. تابع تبدیل حلقه بسته از

سیگنال نویز $D(s)$ به سیگنال خطا $E(s)$ به صورت زیر خواهد بود:

$$H(s) = \frac{E(s)}{D(s)} = \frac{P_1(s)}{1 - C(s)P_2(s)} \quad (4-1)$$



شکل ۱-۱۰- بلوک دیاگرام حذف نویز به روش پسخور

حذف نویز، با طراحی کنترلر به نحوی که اندازه تابع تبدیل رابطه ۴-۱ را حداقل کند محقق می گردد. این هدف با حداکثر کردن اندازه گین حلقه $-C(s)P_2(s)$ حاصل می شود. با وجود این، حذف کامل نویز ممکن نیست زیرا به یک گین حلقه بی نهایت نیازمندیم که در موارد عملی، افزایش گین به خاطر اطمینان از پایداری با محدودیت مواجه می شود.

طبق معیار نایکوئیست، به منظور اطمینان از پایداری حلقه بسته پسخور، گین حلقه باز سیستم می بایست در نقطه با فاز 360° درجه کمتر از یک باشد. این موضوع را می توان برای یک تحریک سینوسی با فرکانس ω به صورت رابطه زیر نشان داد:

$$|C(j\omega)P_2(j\omega)| < 1 \quad \forall \omega \ni \angle C(j\omega)P_2(j\omega) \geq 360 \text{ degrees} \quad (5-1)$$

از آنجائی که جبرانساز از یک فیدبک منفی استفاده می کند، بهره در فاز 180° درجه سیستم می بایست از یک کمتر باشد و از طرفی چون حذف نویز با مقادیر بیشتر گین جبرانساز، بهتر تحقق می یابد، روش حذف نویز اشاره شده در بالاترین رنج فرکانسی مفید محدودیت ایجاد می کند.

۳-۱ طراحی کنترلر وقتی

در اکثر مسائل حذف نویز به روش فعال، دینامیک سیستم به خاطر تغییرات در دما، رطوبت و ساختار فیزیکی محیطی که حذف نویز صورت می گیرد با زمان در حال تغییر است. از طرفی ماهیت منبع نویز متغیر با زمان می باشد. به خاطر این مسئله، کنترلر وقتی در سیستم حذف نویز عملکرد مناسبی را بوجود می آورد.

در طراحی کنترلر برای سیستم ANC کنترلر و همچنین الگوریتم وقتی مورد استفاده در عملکرد سیستم اهمیت بسزائی دارد. استفاده از از فیلترهای خطی مانند فیلترهای FIR [6] و یا IIR [7,8] و همچنین غیر خطی [9,10,11] (به کمک شبکه های عصبی) در کنترلر وقتی ANC گزارش شده است. فیلترهای FIR کاربرد بیشتری از سایر فیلترها در سیستمهای ANC یافته اند [6].

مزیت فیلترهای IIR به FIR امکان همگرایی با تعداد ضرایب کمتری می باشد. ولی متأسفانه رفتار فیلتر IIR از لحاظ همگرایی قابل پیش بینی نیست و ممکن است واگرا شود، در حالی که ما همیشه از پایداری فیلترهای FIR مطمئن هستیم.

در مقایسه فیلترهای خطی و غیر خطی می توان گفت چون در مسیر ثانویه سیستم ANC مدل غیر خطی وجود دارد که مهمترین منبع غیر خطی در این مسیر راه انداز (بلندگو) می باشد، استفاده از فیلترهای غیر خطی به جای فیلترهای خطی می تواند در کاهش نویز موثر باشد. البته فیلترهای غیر خطی نوعاً دارای سرعت همگرایی کمتری نسبت به فیلترهای خطی می باشند [11].

الگوریتمهای وفقی زیادی برای کنترلرهای مورد بحث پیشنهاد شده اند که انتخاب آنها بر اساس

سه فاکتور زیر می باشد [12]:

- عملکرد الگوریتم^۱
- مقاوم بودن الگوریتم^۲
- پیچیدگی محاسباتی^۳

عملکرد سیستم شامل موارد زیر می باشد:

- سرعت همگرایی
- قدرت کاهش نویز
- ردیابی نویزهای غیر ایستاد

سه پارامتر یاد شده که عملکرد سیستم را نشان می دهند، از هم مستقل نیستند و امکان اینکه

یک الگوریتم هر سه را با هم بهبود بخشد وجود ندارد و باید بسته به نوع کاربرد یکی را از درجه

اهمیت بیشتری قرار دهیم.

منظور از مقاوم بودن الگوریتم نیز این است که عملکرد الگوریتم باید نسبت به تغییرات محیط و

تغییر مشخصات منبع نویز، مثلاً تغییرات فرکانس و یا توان نویز، مقاوم باشد

¹. Performance

². Robustness

³. Computational Complexity

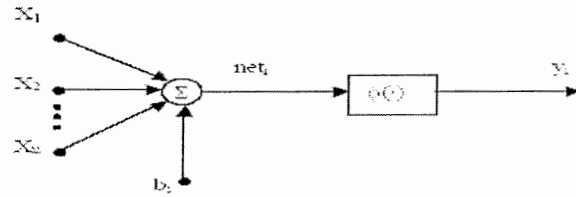
پیچیدگی محاسباتی، مربوط به تعداد جمع کننده ها و ضرب کننده ها و میزان حافظه لازم برای پیاده سازی الگوریتم می باشد. یک الگوریتم پیچیده تر، به یک DSP سریعتر و با حافظه بیشتر که قیمت سیستم ANC را بالا می برد نیازمند می باشد.

رایج ترین الگوریتمی که برای تغییر ضرایب فیلتر وفقی سیستم ANC وجود دارد، الگوریتم $filtered-x LMS^1$ می باشد [۶]. الگوریتمهای تغییر یافته آن نیز برای بهبود عملکرد سیستم ارائه شده اند [13]. در فصل دوم چگونگی بکارگیری این الگوریتم در سیستمهای ANC خطی و غیر خطی شرح داده می شود.

¹.Least Mean Square

فصل دوم :

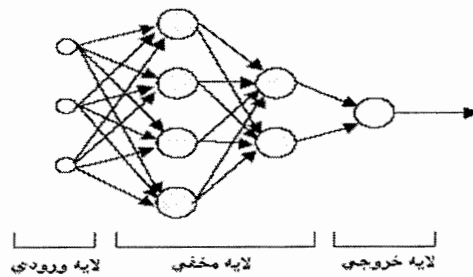
سیستم ANC با الگوریتم وفقی
مبتنی بر کاهش گرادیان خطا



شکل ۲-۱-مدل ریاضی یک نورون

مطابق شکل ۲-۱، ورودیهای وزن داده شده که به نورون می رسند، به همراه ورودی وزن داده شده بایاس، توسط یک جمع کننده خطی با هم جمع می شوند و در نهایت حاصل جمع آنها از یک تابع غیر خطی که تابع فعال ساز^۱ شبکه می باشد عبور می کند و خروجی نورون را می سازد.

از بهم پیوستن تعدادی از این نورونها، مدل شبکه عصبی به وجود می آید. وضعیت نسبی سلولها در شبکه (تعداد و گروهبندی و نوع اتصالات) را توپولوژی شبکه می گویند. توپولوژی در واقع سیستم اتصالات سخت افزار نورونها به یکدیگر است، که توام با نرم افزار مربوطه (یعنی روش ریاضی جریان اطلاعات و محاسبه وزنها) نوع عملکرد شبکه عصبی را تعیین می کند. ساده ترین و معمولترین توپولوژی شبکه عصبی در شکل ۲-۲ نمایش داده شده است.

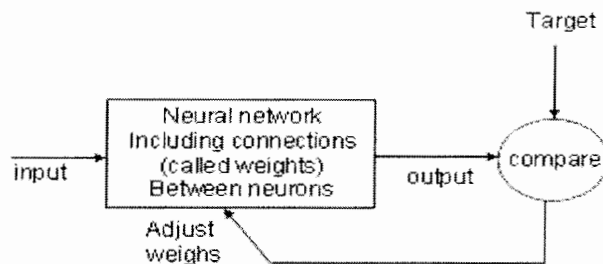


شکل ۲-۲- توپولوژی پرسپترون چند لایه (MLP)

^۱.Activation function

در این توپولوژی که پرسپترون چند لایه^۱ نام دارد، یک لایه ورودی وجود دارد که اطلاعات را دریافت می کند. تعدادی لایه مخفی وجود دارد که اطلاعات را از لایه های قبلی می گیرند و در نهایت یک لایه خروجی وجود دارد که نتیجه محاسبات به آنها می رود و جوابها در آن قرار می گیرند. هر سلول در هر لایه به کلیه سلولهای لایه مجاور بعدی متصل می شود. از آنجائی که در این نوع توپولوژی جریان اطلاعات همیشه از ورودی به خروجی است و هیچ مسیر پیشخوری در اتصالات آن وجود ندارد، به آن توپولوژی پیشخور نیز می گویند.

رایجترین کاربرد شبکه های عصبی، آموزش آنها به طریقی است که ورودیهای خاص منجر به خروجیهای خاص شود (شکل ۲-۳). در این حالت، سیستم شبکه عصبی در فرایند یادگیری طوری وزنهای شبکه را تغییر می دهد که بتواند با هر سری تحریکات ورودی (داده های هر نمونه) جریان خروجی مناسب را ایجاد کند.



شکل ۲-۳ - مهمترین کاربرد شبکه عصبی

^۱.Multi Layer Perceptron(MLP)

چگونگی ریاضی این تغییر وزنها ظریفترین بخش مکانیسم عملکرد شبکه است. در ابتدا وزندهای سیناپسی، مقادیری اتفاقی هستند. علت این مسئله به روش محاسبه وزنها در سیستم شبکه عصبی برمی گردد که روش مبتنی بر تکرار است. در شروع مسئله یادگیری، اطلاعات مربوط به نمونه ها یکی یکی به شبکه داده می شود. اطلاعات ورودی در شبکه جریان پیدا می کند به این معنی که در وزندهای شبکه ضرب شده و نتیجه فعالیت هر نورون بصورت سیگنالی، خود ورودی نوروتهای لایه بعدی خواهد بود. در نهایت، در پایان جریان اطلاعات هر نمونه، شبکه پاسخی در لایه خروجی خواهد داشت. مطابق شکل ۲-۳، با مقایسه خروجی (Output) با پاسخ مطلوب (Target)، خطای شبکه به ازای نمونه ورودی حاصل می شود. الگوریتمهای مختلفی که برای یادگیری شبکه وجود دارند، به نحوی ضرایب شبکه را تغییر می دهند که در نهایت این خطا به صفر برسد. معمولترین روال برای کاهش این خطا روش توزیع معکوس خطا^۱ می باشد در این روش، پس از محاسبه خطای هر نمونه، وزندهای سیناپسی از آخرین لایه به سوی نخستین لایه بتدریج طوری تغییر می کند که خطای حاصل کمتر شود. پس از این، اطلاعات نمونه دوم به شبکه خورانده می شود. مسلماً با همان وزندهای سیناپسی، نمونه جدید مجدداً خطا خواهد داشت. بنابراین روش توزیع معکوس مجدداً دست به کار شده و وزنها را طوری تغییر می دهد که کمترین خطا را هم برای این نمونه و هم برای نمونه پیشین ایجاد کند. بدین ترتیب پس از خوراندن تعداد نمونه کافی به شبکه، منحنی خطای شبکه به می نیمم ترین نقطه خود می رسد و اصطلاحاً شبکه همگرا می شود.

^۱.Back propagation of error

لازم است نکاتی چند در مورد توپولوژی شبکه بدانیم. یکی از پارامترهایی که توسط کاربر در کلیه برنامه های شبیه سازی شبکه های عصبی قابل کنترل است، تعداد نورونهای هر لایه و تعداد لایه ها است. هر نورون را یک گره^۱ نیز می گویند. تعداد گره های لایه ورودی به تعداد ورودی ها است. تعداد گره های لایه خروجی نیز بسته به تعداد خروجیهای مورد نیاز می باشد. تعداد لایه ها و تعداد گره ها در هر لایه مخفی از پارامترهایی است که توسط کاربر قابل تنظیم است. هر چه تعداد لایه ها بیشتر باشد، سیستم قادر به درک پیچیدگیهای بیشتری می باشد، از طرفی زیاد بودن بیش از حد لایه ها دقت عدد پیشگویی را کم می کند و ممکن است مانع همگرایی شود. بیشتر مسائلی که محققین با آن سر و کار دارند با ۲ تا ۳ لایه مخفی قابل حل هستند.

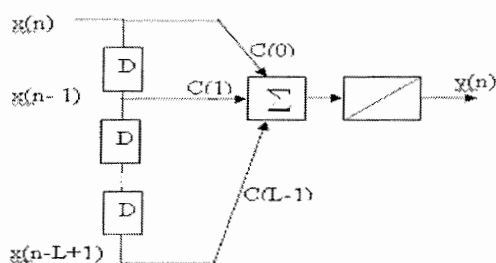
نکته دیگر تعداد گره ها در لایه مخفی می باشد. بطور تخمینی تعداد مناسب گره های لایه مخفی بین نصف تا یک و نیم برابر تعداد گره های لایه ورودی است. منابع مختلف مقیاسهای تجربی متفاوتی بعنوان حدس اولیه برای این تعداد پیشنهاد کرده اند. کم بودن مفرط گره ها قدرت تجزیه و تحلیل و بدنبال آن دقت پیشگویی را کاهش می دهند. از سوئی زیاد بودن مفرط این تعداد منجر می شود که سیستم بجای تجزیه و تحلیل داده ها، آنها را حفظ کند.

با استفاده از این توپولوژی، یک فیلتر FIR وفقی را می توان با شبکه عصبی ساخت. رابطه ورودی و خروجی نمونه n ام فیلتر خطی FIR با توجه به ضرایب فیلتر مطابق رابطه زیر تعریف می شود:

$$y[n] = \sum_{k=0}^{m-1} C[k]x[n-k] \quad (1-2)$$

^۱. Node

m تعداد ضرایب فیلتر، $y[n]$ نمونه n ام سیگنال خروجی، $C[k]$ ضریب k ام فیلتر و $x[n-k]$ نمونه n-k ام سیگنال ورودی می باشد. از آنجائی که خروجی نمونه n ام به ورودی نمونه n ام و نمونه های تاخیر یافته آن بستگی دارد، برای ساختن فیلتر FIR با یک شبکه عصبی، ورودیهای شبکه، نمونه های تاخیر یافته سیگنال ورودی می باشند. یک گره در لایه خروجی به عنوان خروجی فیلتر وجود دارد و تابع فعال ساز آن نیز یک تابع خطی می باشد و همچنین لایه مخفی وجود ندارد. شکل ۲-۴، بلوک دیاگرام فیلتر FIR با رابطه ورودی، خروجی طبق رابطه ۱-۲، ساخته شده با یک شبکه عصبی را نشان می دهد.



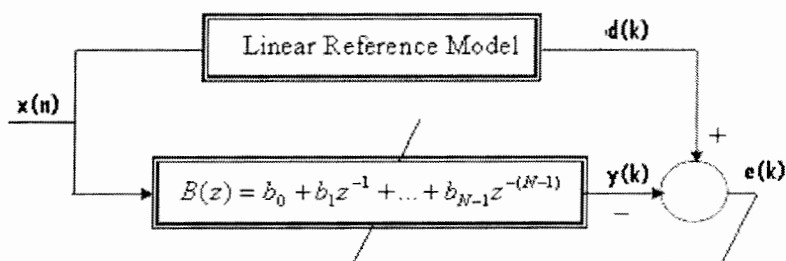
شکل ۲-۴- فیلتر FIR ساخته شده از شبکه عصبی

۲-۲- سیستم ANC خطی، با بکارگیری الگوریتم وقتی مبتنی بر کاهش گرادیان خطا

الگوریتمی که بیشترین کاربرد را در فیلترهای وقتی داشته، الگوریتم توسعه یافته LMS که

filtered-x LMS نامیده می شود، می باشد.

فیلتر خطی وقتی شکل ۲-۵ را در نظر بگیرید. فرض کنید که $B(z)$ یک فیلتر خطی FIR با ضرایب $b = b_0, b_1, \dots, b_{N-1}$ باشد که ضرایب فیلتر قرار است در جهت حداقل کردن اختلاف بین خروجی فیلتر $y[k]$ و سیگنال مطلوب $d[k]$ ، تغییر کنند.



شکل ۲-۵- ساختار الگوریتم وقتی LMS

فرض کنید که برای بازه زمانی نمونه های $k \in [0, M]$ ورودی، بخواهیم این اختلاف را می نیم کنیم. از آنجائی که تعداد ضرایب فیلتر N است، برای محاسبه خروجی فیلتر به ازای هر نمونه ورودی، به $N-1$ نمونه قبل آن نیز نیاز است. بردار X را برای بازه نمونه های ورودی داده شده به صورت زیر تعریف

می کنیم:

$$X = \begin{bmatrix} x[0] & x[-1] & \dots & x[-N] & x[-N+1] \\ x[1] & x[0] & \dots & x[-N+1] & x[-N+2] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x[M] & x[M-1] & \dots & x[M-N] & x[M-N+1] \end{bmatrix} \quad (2-2)$$

همچنین اگر تعریف کنیم:

$$b = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_{N-1} \end{bmatrix}, \quad Y = \begin{bmatrix} y[0] \\ y[1] \\ \vdots \\ y[M] \end{bmatrix}, \quad d = \begin{bmatrix} d[0] \\ d[1] \\ \vdots \\ d[M] \end{bmatrix}, \quad e = \begin{bmatrix} e[0] \\ e[1] \\ \vdots \\ e[M] \end{bmatrix} \quad (2-3)$$

بردارهای b ، Y و e به ترتیب، ضرایب فیلتر، مقادیر خروجی فیلتر و خطای بین خروجی فیلتر و ورودی را در بازه زمانی $[0, M]$ بیان می کند.

در این صورت، بردار خروجی فیلتر و بردار خطا در این بازه به صورت زیر خواهد بود:

$$Y = Xb \quad (4-2)$$

$$e = d - Xb \quad (5-2)$$

برای داشتن یک معیار مناسب از عملکرد فیلتر وقتی می توانیم مجموع مربعات خطا در تمام بازه $k \in [0, M]$ که با رابطه زیر تعریف شده است را در نظر بگیریم:

$$s = \sum_{k=0}^M e[k]^2 = e^T e = (d - Xb)^T (d - Xb) \quad (6-2)$$

برای بدست آوردن بردار b ضرایب فیلتر، به قسمی که رابطه بالا را می نیمم کند، می توانیم از رابطه بالا نسبت به b مشتق گرفته و مساوی صفر قرار دهیم که در این صورت داریم:

$$\hat{b} = (X^T X)^{-1} X^T d \quad (7-2)$$

این روش حداقل کردن مربع خطا را که همه داده های ورودی به طور همزمان برای یافتن نتیجه به کار رفته اند batch mode least square می نامند. از آنجائی که این روش نیاز به حافظه زیادی دارد و دارای پیچیدگی محاسباتی می باشد (در هر مرحله بهبود ضرایب، خطای مربوط به همه نمونه های ورودی باید محاسبه شود) مورد توجه قرار نگرفته است. روش متفاوت LMS به عنوان روشی با پیچیدگی محاسباتی کمتر از روش قبل برای تغییر وقتی ضرایب فیلتر استفاده می شود.

در واقع در روش LMS همه نمونه های ورودی به طور همزمان برای تغییر ضرایب بکار گرفته نمی شوند، بلکه با بکار گرفتن هر نمونه، ضرایب فیلتر وقتی بهبود می یابند .

LMS یک روش کاهش گرادیانی می باشد که با ورود هر نمونه سیگنال ورودی، همه ضرایب فیلتر را به طریقی که مربع خطا، $e[k]^2$ ، کاهش یابد، تغییر می دهد. نتیجه رابطه بهبود ضریب مفروض i فیلتر را در لحظه k می توان به فرم زیر نوشت [9]:

$$b_i[k+1] = b_i[k] - \eta \frac{\partial e^2[k]}{\partial b_i} \quad \forall i \in \{0,1,\dots,N-1\} \quad (8-2)$$

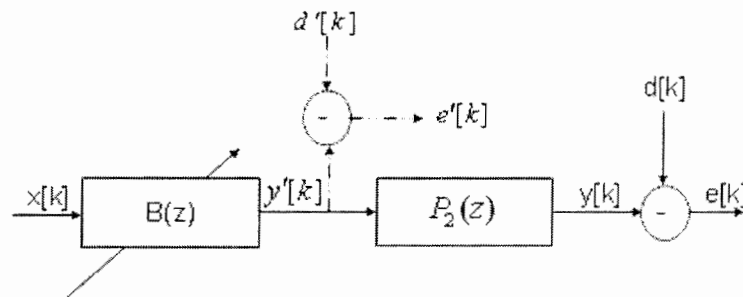
پارامتر η که ضریب همگرایی می باشد، نرخ همگرایی الگوریتم را مشخص می کند و باید به طور مناسبی انتخاب شود تا ضمن اطمینان از پایداری، سرعت همگرایی نیز مناسب باشد.

قسمت شامل مشتق رابطه بالا را می توان مطابق رابطه زیر (رابطه ۹-۲) محاسبه کرد و در نتیجه رابطه بهبود ضرایب فیلتر به صورت رابطه ۱۰-۲ خواهد بود.

$$\frac{\partial e[k]^2}{\partial b_i} = -2\{d[k] - y[k]\} \frac{\partial y[k]}{\partial b_i} = -2e[k]x[k-i] \quad (9-2)$$

$$b_i[k+1] = b_i[k] + 2\eta e[k]x[k-i] \quad (10-2)$$

ولی متأسفانه از این رابطه بهبود ضرایب نمی توانیم به طور مستقیم برای کنترلر حذف نویز صوتی استفاده کنیم، چون در سیستم ANC، مسیر ثانویه ای بین خروجی کنترلر و محل تشکیل خطا وجود دارد که در مدل بررسی شده در نظر گرفته نشده است. شکل ۱۲-۲ مسئله را با در نظر گرفتن مسیر ثانویه نشان می دهد. $B(z)$ یک جبران ساز خطی و $P_2(z)$ ارائه دهنده مسیر ثانویه می باشد.



شکل ۶-۲- ساختار الگوریتم وقفی LMS با در نظر گرفتن مسیر ثانویه

برای اعمال کردن الگوریتم LMS به سیستم حذف نویز، باید مسیر ثانویه را همانند شکل ۶-۲ در نظر گرفته و بهبود ضرایب برای حداقل کردن خطای موجود و قابل دسترسی $e[k]$ (و نه $e'[k]$) صورت گیرد. بنابراین رابطه بهبود ضرایب در لحظه k به صورت زیر خواهد بود :

$$b_i[k+1] = b_i[k] - \eta \frac{\partial e^2[k]}{\partial b_i} \quad \forall i \in \{0,1,\dots,N-1\} \quad (۱۱-۲)$$

فرض کنیم $P_2(z)$ با یک فیلتر FIR با L ضریب ثابت $n = n_0, n_1, n_2, \dots, n_{L-1}$ مدل شود. هر چند این فرض باعث ساده سازی رابطه می شود، ولی مدل FIR برای مسیر ثانویه اغلب در عمل نیز بکار برده می شود.

قسمت شامل مشتق رابطه بالا را می توان مطابق رابطه زیر محاسبه کرد:

$$\frac{\partial e[k]^2}{\partial b_i} = 2e[k] \frac{\partial}{\partial b_i[k]} e[k] \quad (۱۲-۲)$$

خروجی فیلترهای $(y'[k])B(z)$ و $(y[k])P_2(z)$ را مطابق شکل ۶-۲ می توان از روابط زیر

بدست آورد:

$$y'[k] = \sum_{i=0}^{N-1} b_i x[k-i] \quad (۱۳-۲)$$

$$y[k] = \sum_{j=0}^{L-1} n_j y'[k-j] \quad (۱۴-۲)$$

با جایگذاری معادلات ۱۳-۲ و ۱۴-۲ در رابطه ۱۲-۲ و پس از ساده سازی بدست می آوریم:

$$\begin{aligned} \frac{\partial e^2[k]}{\partial b_i[k]} &= 2e(k) \frac{\partial}{\partial b_i[k]} (d(k) - \sum_{j=0}^{L-1} n_j \sum_{l=0}^{N-1} b_l [k-j] x[k-j-l]) = \\ &- 2e[k] \sum_{j=0}^{L-1} n_j \sum_{l=0}^{N-1} x[k-j-l] \frac{\partial}{\partial b_i[k]} b_l [k-j] \end{aligned} \quad (۱۵-۲)$$

اگر فرض کنیم ضرایب جبران ساز به آرامی تغییر می کنند می توان از تخمین زیردر رابطه بالا

استفاده کرد:

$$\frac{\partial}{\partial b_i[k]} \cong \frac{\partial}{\partial b_i[k-j]} \quad \forall j \in \{0,1,\dots,L-1\} \quad (۱۶-۲)$$

در این صورت داریم:

$$\frac{\partial e^2[k]}{\partial b_i[k]} = -2e[k] \sum_{j=0}^{L-1} n_j \sum_{l=0}^{N-1} x[k-j-l] \quad (۱۷-۲)$$

اگر $V[z]$ را مطابق رابطه ۱۸-۲، یک نسخه $X[k]$ که توسط تابع تبدیل $P_2(z)$ فیلتر شده است

تعریف شود، آنگاه رابطه ۱۷-۲ را می توان به صورت رابطه ۱۹-۲ نوشت.

$$V[z] \triangleq P_2(z) X(z) \quad (۱۸-۲)$$

$$\frac{\partial e^2[k]}{\partial b_i[k]} = -2e[k] V[k] \quad (۱۹-۲)$$

و در انتها با جایگزینی رابطه ۲-۱۹ در رابطه ۲-۱۱، رابطه بهبود ضرایب در لحظه k به صورت

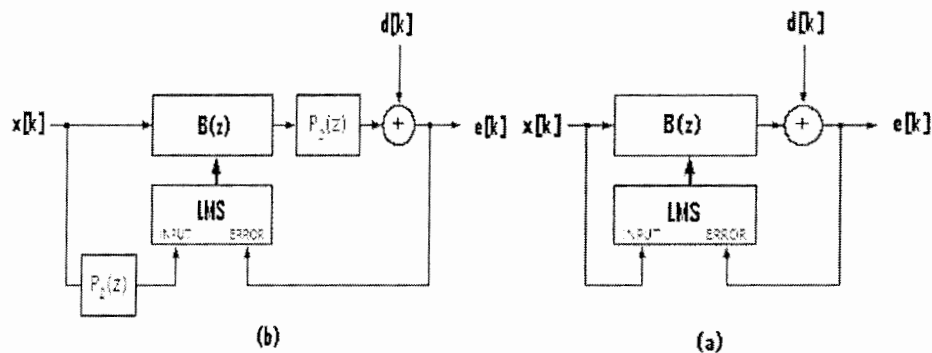
رابطه زیر در می آید:

$$b_i[k+1] = b_i[k] + 2\eta e[k]v[k-i] \quad (2-20)$$

ملاحظه می شود که رابطه بهبود ضرایب برای این حالت مشابه الگوریتم LMS برای بهبود ضرایب خواهد بود، با این تفاوت که به جای ورودی مرجع $X[k-i]$ ، مدل فیلتر شده آنها بکار گرفته شده است و

از این رو به آن، روش filtered-x LMS می گویند. شکل ۲-۷ بلوک دیاگرام کلی روش filtered-x LMS

مورد استفاده در سیستم ANC با فیلتر خطی و مقایسه آن با روش استاندارد LMS را نشان می دهد.



شکل ۲-۷- (a) الگوریتم استاندارد LMS در مقابل (b) الگوریتم filtered-x LMS

۲-۳- سیستم ANC غیر خطی، با بکارگیری الگوریتم وفقی مبتنی بر کاهش گرادیان خطا

ممکن است در سیستم ANC، عوامل غیر خطی یافت شود. مهمترین عامل غیر خطی در این

رابطه، راه انداز مولد نویز ثانویه (باندگو) می باشد. یک راه انداز به طور معمول، هنگامی که ورودی آن

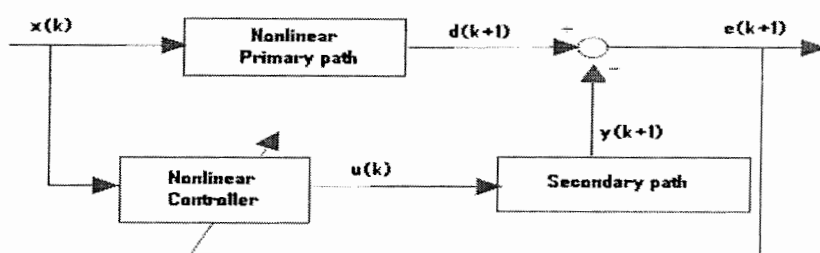
نزدیک به (یا بیشتر از) ورودی نامی آن باشد و یا اینکه در فرکانسی خارج از رنج فرکانس نامی خود کار

کند، خواص غیر خطی از خود نشان می دهد [9]. همچنین خواص غیر خطی می تواند در مسیر اولیه عبور سیگنال یافت شود. به عنوان مثال، هنگامی که نویز حاصل از یک منبع نویز با توان صوتی بالا در یک لوله انتشار می یابد، مسیر عبور سیگنال خواص غیر خطی از خود نشان می دهد [14].

در چنین مواردی که عوامل غیر خطی در سیستم وجود دارد، کنترلرهای خطی عملکرد خوبی ندارند. به عنوان مثال، در سیستم با مسیر اولیه غیر خطی، نویزی که به منطقه سکوت می رسد، هارمونیکهای فرکانس اصلی نویز اولیه را نیز خواهد داشت که ممکن است در بعضی از این هارمونیکها توان نویز قابل توجه باشد، در حالی که کنترلر خطی به هیچ وجه قادر نیست که در این هارمونیکهای فرکانسی، نویز را کاهش دهد. در این موارد می توان از یک کنترلر غیر خطی در سیستم استفاده کرد که معمولترین و پرکاربردترین آن شبکه عصبی می باشد.

وظیفه الگوریتم وفقی برای این کنترلر، تغییر وزنه‌های شبکه به نحوی است که خطای سیستم را حداقل کند. از آنجائی که الگوریتم وفقی filtered-x LMS از خواص خطی کنترلر استفاده می کرد، نمی توان آن را برای کنترلر غیر خطی استفاده کرد.

یک سیستم ANC با مسیر اولیه غیر خطی در شکل ۲-۸ نشان داده شده است. مسیر ثانویه، خطی و با یک فیلتر FIR مدل شده است.



شکل ۲-۸- بلوک دیاگرام سیستم ANC با کنترلر غیر خطی

سیستم ANC مفروض را می توان با رابطه زیر توصیف کرد.

$$e(k+1) = d(k+1) + y(k+1) = g(X(k)) + \sum_{j=0}^m I(j)u(k-j) \quad (21-2)$$

و $X(k) = [x(k)x(k-1)\dots x(k-n)]^T$ بردار سیگنال مرجع، $u(k)$ خروجی کنترلر غیر خطی، و

$I(j)$ ضرایب فیلتر FIR مربوط به مسیر ثانویه می باشد. d سیگنال نویز است که از

منبع نویز اولیه به محل میکروفن تشخیص خطا رسیده است و $g(\cdot)$ یک تابع غیر خطی و مدل مسیر

اولیه می باشد. کنترلر غیر خطی استفاده شده در سیستم، شبکه عصبی از نوع MLP می باشد. خروجی

کنترلر غیر خطی را می توان به صورت رابطه ۲۲-۲ نوشت.

$$u(k) = f(X(k), W) \quad (22-2)$$

$f(\cdot)$ یک تابع غیر خطی بوده و W بردار وزنه های شبکه می باشد.

رابطه ۲۳-۲ را به عنوان معیار خطا برای عملکرد الگوریتم وقتی در سیستم غیر خطی انتخاب می

کنیم.

$$J(k+1) = \frac{1}{2}e^2(k+1) = \frac{1}{2}[y(k+1) - d(k+1)]^2 \quad (23-2)$$

وظیفه الگوریتم وقتی، تغییر وزنه‌های شبکه برای حداقل کردن معیار خطا می باشد. برای این منظور، وزنه‌های شبکه می توانند طبق رابطه زیر که از روشهای کاهش گرادیانی بهره می برد، تغییر کنند [11].

$$W(k+1) = W(k) - \eta \frac{\partial J(k+1)}{\partial W(k)} = W(k) - \eta e(k+1) \frac{\partial y(k+1)}{\partial W(k)} \quad (24-2)$$

η که ضریب همگرایی نامیده می شود، نرخ همگرایی را مشخص می کند. با اعمال قاعده زنجیره ای به

رابطه ۲۴-۲ داریم:

$$\frac{\partial y(k+1)}{\partial W(k)} = \sum_{j=0}^m \frac{\partial y(k+1)}{\partial u(k-j)} \frac{\partial u(k-j)}{\partial W(k)} = \sum_{j=0}^m I(j) \frac{\partial u(k-j)}{\partial W(k)} \quad (25-2)$$

اگر وزنه‌های شبکه به آرامی تغییر کنند، گرادیان u در رابطه ۲۵-۲ را می توان به صورت زیر

تخمین زد:

$$\frac{\partial u(k-j)}{\partial W(k)} \approx \frac{\partial u(k-j)}{\partial W(k-j)} = \left. \frac{\partial f(X, W)}{\partial W} \right|_{X=X(k-j), W=W(k-j)} \quad (26-2)$$

با استفاده از رابطه ۲۴-۲ و گرادیانهای محاسبه شده در روابط ۲۵-۲ و ۲۶-۲، رابطه بهبود ضرایب

شبکه به صورت کلی زیر در می آید:

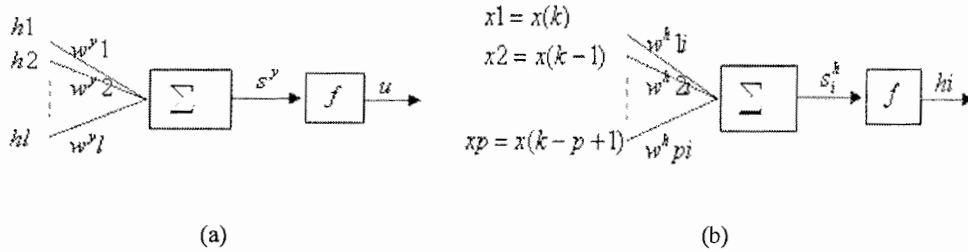
$$W(k+1) = W(k) - \eta e(k+1) \sum_{j=0}^m I(j) \left. \frac{\partial f(X, W)}{\partial W} \right|_{X=X(k-j), W=W(k-j)} \quad (27-2)$$

در این پایان نامه از یک شبکه MLP با یک لایه میانی استفاده شده است. گرادیان موجود در رابطه

۲۶-۲ را برای وزنه‌های لایه خروجی و همچنین لایه مخفی باید بدست آوریم. از آنجائی که سیستم ANC

تک کاناله موضوع بحث ماست، لایه خروجی شبکه تنها یک گره خواهد داشت. تعداد گره های لایه ورودی

را p و تعداد گره های لایه مخفی را l می گیریم. شکل ۹-۲ بلوک دیاگرام یک نورون لایه مخفی و لایه خروجی را به طور جداگانه نشان می دهد.



شکل ۹-۲- بلوک دیاگرام (a) نورون لایه مخفی (b) لایه ورودی

از بلوک دیاگرام شکل ۹-۲، مشاهده می شود که ورودی گره های لایه ورودی در لحظه k ، نمونه لحظه k ام و ورودی در لحظات ماقبل می باشد. وزن بین گره ورودی i ام و گره z ام در لایه مخفی را با w^{zi} بیان می کنیم. وزنه های لایه خروجی را w^{ji} می نامیم که اندیس i مربوط به i امین گره لایه مخفی می باشد که به گره خروجی متصل می شود. خروجی جمع کننده گره i ام لایه مخفی را s_i^h و خروجی نهائی آن، پس از عبور از تابع فعال ساز را h_i می نامیم. خروجی جمع کننده لایه خروجی را s^y و خروجی نهائی آن که خروجی شبکه نیز می باشد را u می نامیم. تابع فعال ساز شبکه را $f(\cdot)$ در نظر می گیریم.

می توان رابطه خروجی شبکه با ورودیها و وزنه های شبکه را به صورت زیر نوشت :

$$u(k) = f(X(k), W) = f(s^y(k)) \quad (2-28)$$

از طرفی با توجه به بلوک دیاگرام شکل (b) ۹-۲ داریم:

$$s^y(k) = \sum_{i=1}^{l+1} h_i(k) w_i^y(k) \quad (2-29)$$

همچنین با استفاده از شکل (a) ۹-۲ داریم :

$$h_i(k) = \sum_{j=1}^p x_j(k) w_{ji}^h(k) \quad (۳۰-۲)$$

با ترکیب روابط ۲۹-۲ و ۳۰-۲ بدست می آوریم :

$$s^y(k) = \sum_{i=1}^{l+1} \left[\sum_{j=1}^p x_j(k) w_{ji}^h(k) \right] w_i^y \quad (۳۱-۲)$$

حال می توان رابطه بهبود وزنها را برای لایه مخفی و خروجی این شبکه، با استفاده از روابط ۲۷-۲،

۲۸-۲، و ۳۱-۲ بدست آورد. گرادیان موجود در رابطه ۲۷-۲ برای یک ضریب از لایه میانی به صورت زیر

بدست می آید:

$$\frac{\partial f(X, W)}{\partial w_{mn}^h(k)} = f'(s^y(k)) \times \frac{\partial s^y(k)}{\partial w_{mn}^h(k)} = f'(s^y(k)) \times [x_m(k) w_n^y(k)] \quad (۳۲-۲)$$

و به همین ترتیب برای یک ضریب از لایه خارجی داریم:

$$\frac{\partial f(X, W)}{\partial w_m^h(k)} = f'(s^y(k)) \times \frac{\partial s^y(k)}{\partial w_m^h(k)} = f'(s^y(k)) \times \left[\sum_{j=1}^p x_j(k) w_{jm}^h(k) \right] = f'(s^y(k)) \times h_m(k) \quad (۳۳-۲)$$

و در نهایت رابطه تغییر وزن برای لایه مخفی با استفاده از روابط ۲۷-۲ و ۳۲-۲ طبق رابطه زیر بدست می

آید:

$$w_{mn}^h(k+1) = w_{mn}^h(k) - \eta \times e(k+1) \sum_{j=0}^m l(j) f'(s^y(k-j)) x_m(k-j) w_n^y(k-j) \quad (۳۴-۲)$$

و برای یک وزن از لایه خارجی این رابطه به صورت زیر خواهد بود:

$$w_m^y(k+1) = w_m^y(k) - \eta \times e(k+1) \sum_{j=0}^m l(j) f'(s^y(k-j)) h_m(k-j) \quad (۳۵-۲)$$

فصل سوم :

روش بهینه سازی اجتماع ذرات

مقدمه

بسیاری از مسائل بهینه سازی دنیای مهندسی صنعتی ذاتاً بسیار پیچیده اند و حل آنها به کمک روشهای بهینه سازی متداول، بدلیل پیچیدگی سیستم، وجود پارامترهای نامطمئن و متغیر با زمان بودن آنها بسیار دشوار بوده و در بعضی موارد عملاً غیر ممکن می باشد.

در یک مسئله بهینه سازی، هدف یافتن بهترین راه حل از بین همه راه حل های ممکن برای مسئله مورد نظر می باشد که می توان آن را به صورت زیر بیان کرد:

فرض کنید S فضای جستجو، $F \subset S$ فضائی که جوابهای موجه در آن قرار گرفته اند و f تابع معیار که ارزش هر جواب بوسیله آن سنجیده می شود، باشد. هدف یک مسئله بهینه سازی، یافتن $\vec{x} \in F$ می باشد به قسمی که:

$f(\vec{x}) \leq f(\vec{y})$ برای مسائل یافتن مینیمم تابع معیار

$f(\vec{x}) \geq f(\vec{y})$ برای مسائل یافتن ماکزیمم تابع معیار

اگر روابط بالا به ازای هر $\vec{y} \in F$ برقرار باشد، \vec{x} را نقطه بهینه کلی^۱ گویند .

اگر $N(\vec{x})$ را یک همسایگی \vec{x} تعریف کنیم و تعریف بالا را برای هر $\vec{y} \in N(\vec{x})$ داشته باشیم،

آنگاه \vec{x} را نقطه بهینه محلی^۲ می گویند.

^۱. Global optimum

^۲. Local Optimum

الگوریتم های کلاسیک متنوع و فراوانی برای جستجوی فضای جوابها و یافتن جواب بهینه طراحی شده و مورد استفاده قرار می گیرند. سئوالی که مطرح می باشد این است که چرا برای حل یک مسئله بهینه سازی خاص، راه حل های متنوع زیادی ارائه شده است.

جواب این سئوال این است که هر یک از روشهای کلاسیک و سنتی موجود برای برخی مسائل بهینه سازی جواب خوبی پیدا می کنند در حالیکه در سایر مسائل دچار مشکل می باشند. بدین ترتیب، واضح است که وقتی یک مسئله جدید مطرح می شود باید یک راه حل جدید برای آن ارائه نمود.

مشکل دیگر روشهای کلاسیک و سنتی این است که فقط در مسائلی که توسط یک مدل دقیق توصیف شده باشند می توانند قابل اعمال هستند و بنابراین در تعداد زیادی از مسائل دنیای واقعی که تعریف کامل مسئله و معادلات ریاضی آن را دست نمی باشد کارائی ندارند.

در اکثر روشهای سنتی موجود، در صورت یافتن جواب مسئله، آن جواب دقیق می باشد. و از این رو روشهای دقیق^۱ نامیده می شوند.

در مواردی که بدلیل پیچیدگی مسئله تعریف شده رسیدن به جواب با روشهای سنتی غیرممکن و یا روند همگرایی الگوریتم های یاده شده کند باشد، مجبوریم جواب را تخمین بزنیم و به عبارتی، از روشهای تقریبی^۲ استفاده کنیم. هنگام مواجه شدن با مسائل دنیای واقعی، استفاده از این روشها ارجحیت دارد.

^۱ . Exact Methods

^۲ . Approximate Methods

از جمله روشهای سنتی که جواب را به صورت تقریبی بدست می آورند، می توان به روشهای گرادیانی اشاره کرد. این روش اقدام به جستجوی محلی می نماید. در یک الگوریتم تکرار، جواب در همسایگی یک نقطه اولیه بررسی و با یافتن جواب بهتر، جایگزین نقطه قبلی می گردد. در صورت پیدا نکردن جوابی بهتر در همسایگی جواب مرحله قبلی، الگوریتم متوقف می گردد. یک خطر مهم از دست دادن جواب واقعی با بدام افتادن در نقاط بهینه محلی می باشد.

اگر شعاع همسایگی را بزرگ انتخاب کنیم امکان فرار از بهینه محلی بیشتر می شود ولی این کار باعث کند شدن الگوریتم می شود که غیرقابل قبول خواهد بود.

برای غلبه بر مشکلات روشهای سنتی، روشهای تجربی مدرن^۱ بوجود آمده اند که روشهای تقریبی می باشند و تمرکز عمده آنها در فرار از نقطه بهینه محلی و یافتن بهینه کلی می باشد. مزیت این روشها این است که این الگوریتم ها همه منظوره بوده و قابل اعمال به طیف وسیعی از مسائل می باشند.

روشهای مدرن را می توان به دو گروه کلی روشهای مبتنی بر جمعیت^۲ و روشهای مبتنی بر جواب واحد^۳ تقسیم بندی کرد. در روشهای مبتنی بر جمعیت، اعضاء جمعیت که به آنها ذرات نیز اطلاق می شود، هر یک جواب بالقوه مسئله می باشد که با همکاری همدیگر نقطه بهینه را جستجو می کنند. در حالی که در روشهای مبتنی بر جواب واحد، یک نقطه تصادفی در فضای جستجو به عنوان جواب اولیه

^۱ .Modern Heuristics Methods

^۲ .Population Based Method

^۳ .Single Solution Methods

مسئله انتخاب می شود و در یک الگوریتم تکرار، با در نظر گرفتن معیار جواب بهینه، به سمت این نقطه سوق داده می شود.

برای فهم بیشتر تفاوت این دو روش در یافتن جواب بهینه، ابتدا فرض کنید که به تنهایی در یک منطقه کوهستانی واقع شده آید و تمامی فضای اطراف شما را مه شدیدی پوشانیده و حتی امکان اینکه تا چند متر جلو تر خود را ببینید وجود ندارد. هدف شما این است که بالاترین قله را ترجیحا در حداقل زمان ممکن فتح کنید. در ضمن هیچ نقشه و قطب نمایی و هیچ ایده آئی که به کدام سمت بروید تا به بالاترین قله برسید وجود ندارد. پس شما تصمیم می گیرید که بهترین راه این است که از نقطه ای که در آن واقع شده آید شروع به بالا رفتن کنید و سرانجام به یک قله برسید. ولی به احتمال زیاد این بلندترین قله نیست، چون شانس کمی وجود دارد که در اولین بالا رفتن به صورت تصادفی و بدون هیچ ایده ای به بلندترین قله برسید.

حال از یک نقطه دیگر شروع به بالا رفتن می کنید و باز هم به یک قله می رسید که با مقایسه ارتفاع آن با قله قبلی می توانید حدس بزنید که درچه مسیر احتمالی حرکت بعدی را شروع کنید تا به یک قله بلندتر برسید. در واقع با نگاه کردن به اختلاف بین دوراه حل، مسیر احتمالی که به سمت نقطه بهینه منتهی می شود را می توان بدست آورد.

در روشی دیگر، تجسم کنید که شما عضوی از یک گروهی هستید که با هم برای یافتن بلندترین قله به کوهستان رفته آید و شرایط جوی مشابه شرایط مسئله قبل باشد. هر یک از اعضاء گروه مجهز به

یک فرستنده و گیرنده رادیویی و ارتفاع سنج هستند که می توانند با سایر اعضا تبادل اطلاعات کنند. برای رسیدن به بلندترین قله هر عضو از یک نقطه متفاوت شروع به بالا رفتن می کند و به قله ای با ارتفاع مشخص می رسد. با برقراری ارتباط رادیویی بین سایر اعضا و این عضو، مسیر بعدی عضو در رسیدن به بلندترین قله، انتخاب می گردد.

مسئله بالا را می توان به صورت یک مسئله بهینه سازی در نظر گرفت که در آن، منطقه کوهستانی به عنوان تابع ارزش و هر موقعیتی از منطقه، یک جواب مسئله می باشد. بهترین جواب، بلندترین قله در منطقه می باشد.

در اولین سناریو که شما یک جواب واحد را تغییر می دهید، یافتن بلندترین قله، مانند یافتن ماکزیمم یک تابع ارزش توسط الگوریتم کاهش گردآیانی می باشد. اما دومین سناریو جستجوی جمعی را توصیف می کند که مبنای روشهای مبتنی بر جمعیت است. واضح است که حرکت هر فرد به تنهایی هوشمند نمی باشد ولی حرکت جمعی آنها با همکاری همدیگر یک حرکت هوشمند و هدفدار خواهد بود. در ادامه به معرفی یکی از روشهای بهینه سازی مبتنی بر جمعیت که در این پایان نامه از آن استفاده شده است، می پردازیم.

روش بهینه سازی اجتماع ذرات

این روش که اجتماع ذرات نام دارد و به اختصار PSO^1 نامیده می‌شود مانند الگوریتم‌های ژنتیک از روشهای مبتنی بر جمعیت می‌باشد. PSO یکی از شاخه های هوش جمعی² می‌باشد که حوزه وسیعی از دانش هوش مصنوعی را شامل می‌شود. هوش جمعی در واقع بر پایه شبیه سازی رفتار جمعی گروهی از حشرات مانند زنبورها، مورچه‌ها و موریانه‌ها همینطور گروه‌های حیوانی مانند ماهیها شکل گرفته است. از شاخه های هوش جمعی، علاوه بر PSO ، می‌توان به روش اجتماع مورچگان³ اشاره کرد که در آن از تقلید رفتار جمعی از مورچه‌ها دریافتن غذا، برای مسائل مسیریابی بهینه استفاده می‌شود.

۳-۱- تعاریف و مقدمات

ایده‌های اولیه روش اجتماع ذرات براساس علاقه‌ای که برای شبیه سازی گرافیکی رفتار پروازی جالب و غیر قابل پیش بینی پرندگان ایجاد شد، توسط kennedy شکل گرفت [15]. پرندگان توانایی پرواز همزمان و هماهنگ با هم را دارند.

هر کدام از پرندگان در یک دسته، در عین حال که به توانایی‌های ناوبری خود تکیه دارند و مسیر مورد نظر خود را در نظر می‌گیرند، به موقعیت خود نسبت به سایر پرنده‌های جمعیت و نیز حفظ فاصله مناسب با آنها توجه دارند. بدین جهت مشاهده می‌کنیم که دسته های پرندگان ، مثلاً غازه‌های مهاجر

¹ . Particle Swarm Optimization

² . Swarm Intelligence

³ . Ant Colony

بسیار هماهنگ با یکدیگر حرکت می کنند. علاوه بر این مسئله، آنها از راهنمایی یک پرنده در جلوی دسته بهره مندند. مسیر حرکت جلوترین پرنده، تأثیر زیادی در نحوه حرکت دسته دارد. این راهنما البته در طول مسیر طولانی ثابت نیست و پرندگان مرتباً جای خود را با راهنما عوض می کنند.

همه این رفتارها در شبیه سازی ریاضی الگوریتم PSO مد نظر قرار می گیرند. در PSO هر کدام از اعضاء یک اجتماع، یک ذره نامیده می شود. این ذرات همگی در فضای چند بعدی شناور و بر اساس تبادل اطلاعات با دیگر اعضاء حرکت می نمایند.

تغییر موقعیت هر ذره بر اساس تجربه و دانش خود و همسایگانش می باشد. برای مطالعه این روش ابتدا می بایست همسایگی را در یک اجتماع تعریف کرده و همچنین نحوه ارتباط ذرات را بیان نماییم.

۳-۲- انواع توپولوژی و اصل همسایگی

توپولوژی در واقع ساختار شبکه ای را به صورت نمادین آشکار می سازد و به درک ما از نحوه ارتباط ذرات جمعیت کمک می کند. ابتدا لازم است توضیحی در مورد همسایگی آورده شود. همسایه یک ذره در اجتماع ذرات، ذره ای است که با آن ذره می تواند تبادل اطلاعات کند. همسایگی همیشه بر اساس فاصله اقلیدسی تعریف نمی شود. و نزدیکی از نظر فاصله، با اینکه بعضی از مواقع بکار می رود، همیشه معیاری برای همسایه بودن نیست. در بسیاری از موارد با تخصیص یک اندیس به ذرات، می توان ارتباط همسایگی

آنها را مشخص نمود. به عنوان مثال، کلیه ذراتی که دارای اندیس ۱ باشند با هم همسایه بوده، آنهایی که اندیس ۲ دارند با هم و الی آخر.

تخصیص این اندیس‌ها، بصورت دلخواه، تصادفی و گاهی براساس معیارهای خاص صورت می‌گیرد.

تعداد ذراتی یک همسایگی، از پارامترهای PSO می‌باشند که در بخشهای بعدی در مورد آن صحبت خواهیم کرد.



شکل ۳-۱-: توپولوژی‌های مطرح در PSO

۳-۳-۱- توپولوژی ستاره^۱

در این توپولوژی، هر ذره می‌تواند با سایر ذرات موجود در اجتماع ارتباط داشته باشند. اتصال کامل

هر گره با سایر گره‌های گراف در شکل (۳-۱ الف) مبین این مسئله است. در این توپولوژی هر ذره تمایل

به پیروی از حرکت بهترین ذره جمعیت را دارد. بهترین ذره، ذره ای است که بیشترین ارزش را با توجه به

تابع معیار دارد. توپولوژی ستاره در الگوریتم PSO با نام بهترین جهانی^۲ استفاده می‌شود که به آن

خواهیم پرداخت.

^۱. Star Topology

^۲. Global Best

۳-۲-۲- توپولوژی حلقه^۱

در این مورد، هر ذره با n همسایه مجاور خود ارتباط دارد. در حالت $n = 2$ ، یک ذره با دو همسایه نزدیک خود مرتبط است و شکل (۳-۱ ب) نشان دهنده این توپولوژی است. در این توپولوژی هر ذره تمایل حرکت به طرف بهترین ذره همسایگی خود را داراست. نسخه بهترین محلی^۲ الگوریتم مربوط به این توپولوژی می باشد. این نسخه نسبت به الگوریتم بهترین جهانی دارای این مزیت است که ناحیه بیشتری از فضا را جستجو می کند، البته این الگوریتم سرعت همگرایی پایین تری نسبت به الگوریتم بهترین جهانی دارد.

۳-۲-۳- توپولوژی چرخه^۳

در این توپولوژی، تمام ذرات جمعیت فقط با یک ذره در ارتباط هستند. این ذره که ذره کانونی^۴ نامیده می شود موقعیت خود را به سمت بهترین ذره تنظیم می کند و اگر این تنظیم، بهبودی ایجاد کند، آنرا به بقیه ذرات اطلاع می دهد. شکل (۳-۱ ج) این توپولوژی را نشان می دهد.

۳-۳- الگوریتم های PSO

1 . Ring Topology
2 . Local Best
3 . Local Best
4 . Focal Particle

در یک مسئله بهینه سازی به روش PSO هر ذره در واقع بیانگر یک جواب بالقوه برای مسئله می باشد. موقعیت ذرات با توجه به تجربه خودشان و همسایگانشان عوض می شود. از این پس از $\vec{x}_i(t)$ برای نشان دادن موقعیت ذره P_i ام در زمان t استفاده می کنیم. سرعت حرکت ذره i ام بصورت زیر تعریف می شود:

$$\vec{v}_i(t) = \vec{x}_i(t) - \vec{x}_i(t-1) \quad (1-3)$$

در ادامه به بررسی سه الگوریتم متداول در روش PSO می پردازیم.

۱-۳-۳- الگوریتم بهترین فردی^۱

در این الگوریتم، تغییر موقعیت ذرات فقط بر اساس تجارب خودشان در طول حرکت، می باشد، لذا ذرات با یکدیگر هیچگونه ارتباطی ندارند. با اینکه این نوع الگوریتم استفاده چندانی ندارد، توضیح آن به عنوان مقدمه ای بر الگوریتم های بعدی لازم می باشد.

۱- ابتدا فرض کنید که اجتماع اولیه ای بطور تصادفی شکل گیرد. در این اجتماع، P ، موقعیت ذره

i ام، \vec{x}_i ، و سرعت آن، \vec{v}_i ، در ابتدا به طور تصادفی در فضای چند بعدی انتخاب می شوند.

۲- در هر مرحله از الگوریتم، ارزش^۱ هر ذره با توجه به موقعیت حاضرش (لحظه t) محاسبه شود.

این ارزش با توجه به تابع معیار $F(\cdot)$ محاسبه می گردد. مقادیر بزرگتر $F(\vec{x}_i(t))$ را ارزشمندترند در نظر می گیریم.

^۱. Individual Best

۳- با داشتن ارزش ذرات در لحظه t ، مقایسه ای بین ارزش هر ذره در حال حاضر و بهترین تجربه گذشته‌اش $pbest_i^2$ صورت می گیرد و در شرایطی که ارزش فعلی ذره از بهترین تجربه گذشته ذره بیشتر باشد، بهترین تجربه گذشته ذره تصحیح می گردد.

- اگر $pbest_i > F(\bar{x}_i(t))$ آنگاه

$$pbest_i = F(\bar{x}_i(t)) \quad (\text{الف})$$

$$\bar{x} \text{ } pbest_i = \bar{x}_i(t) \quad (\text{ب})$$

۴- سرعت هر طبق رابطه زیر محاسبه می گردد:

$$v_i(t) = \bar{v}_i(t-1) + \rho(\bar{x} \text{ } pbest_i - \bar{x}_i(t)) \quad (2-3)$$

که در آن ρ یک عدد مثبت تصادفی است.

۵- با داشتن سرعت حرکت هر ذره، مکان جدید هر ذره از روی مکان قبلی آن به صورت زیر

محاسبه می گردد:

$$\bar{x}_i(t) = \bar{x}_i(t-1) + \bar{v}_i(t)$$

۶- الگوریتم از مرحله ۲ و تا رسیدن به همگرایی تکرار می شود.

¹ . Fitness

² . Personal best

در الگوریتم تشریح شده، حداکثر مقدار p از پارامترهای PSO است که بعداً بحث بیشتری راجع به آن خواهیم کرد. مقادیر بزرگ برای این پارامتر، موجب حرکت نوسانی ذرات شده و مقادیر کوچک آن حرکت هموارتری را میسر می‌سازند ولی سرعت همگرایی را کند می‌کند.

۳-۲-۳- الگوریتم بهترین جهانی

این الگوریتم، متناظر با توپولوژی ستاره است. در این الگوریتم حرکت هر ذره با استفاده از تجربه خودش و حرکت سایر ذرات صورت می‌گیرد. واضح است که در این الگوریتم همبستگی زیادی در اجتماع وجود دارد و حرکت ذرات کاملاً به هم مرتبط هستند.

مراحل این الگوریتم را نیز از نظر می‌گذرانیم:

۱- تشکیل جمعیت اولیه (همانند آنچه در مورد الگوریتم بهترین فردی ذکر شد، به صورت تصادفی صورت می‌پذیرد).

۲- محاسبه ارزش ذرات با استفاده از موقعیت فعلیشان.

۳- مقایسه ارزش کنونی ذرات و بهترین تجربیاتشان و جایگزینی شرطی زیر.

- اگر $pbest_i > F(P_i)$ آنگاه

$$pbest_i = F(P_i) \quad (\text{الف})$$

$$\bar{x} \quad pbest_i = \bar{x}_i(t) \quad (\text{ب})$$

۴- مقایسه ارزش کنونی هر ذره با بهترین تجربه قبلی کلیه ذرات جمعیت و جایگزینی ذیل :

- اگر $gbest < F(P_i)$ آنگاه

$$gbest = F(P_i) \quad (\text{الف})$$

$$\vec{x} \text{ gbest} = \vec{x}_i(t) \quad (\text{ب})$$

۵- تغییر سرعت هر ذره توسط رابطه زیر :

$$\vec{v}_i(t) = \vec{v}_i(t-1) + \rho_1 (\vec{x} \text{ pbest}_i - \vec{x}_i(t)) + \rho_2 (\vec{x} \text{ gbest} - \vec{x}_i(t)) \quad (3-3)$$

که در آن ρ_1, ρ_2 اعداد مثبت تصادفی می باشد.

۶- حرکت دادن ذرات به موقعیتهای جدیدشان.

$$\vec{x}_i(t) = \vec{x}_i(t-1) + \vec{v}_i(t)$$

۷- رفتن به مرحله ۲ و تکرار الگوریتم تا رسیدن به همگرایی.

۳-۳-۳ الگوریتم بهترین محلی

این الگوریتم در واقع به توپولوژی حلقه یا اصطلاحاً ساختار همسایگی دایره ای اشاره دارد. در این الگوریتم ذرات تحت تأثیر بهترین موقعیت بدست آمده در همسایگی خود قرار دارند. مراحل الگوریتم کاملاً مشابه با آنچه در مورد الگوریتم بهترین جهانی آمد می باشد تنها با این تفاوت که در مراحل ۴ و ۵، $lbest^1$ جایگزین $gbest^1$ می شود.

¹ . local best

$lbest_i$ ، نشانگر بالاترین ارزش بدست آمده در همسایگی i ام است. بر خلاف $gbest$ که در اجتماع یکتاست، به تعداد همسایگی‌های اجتماع $lbest$ وجود دارد.

با اینکه سرعت همگرایی در نسخه بهترین محلی نسبت به نسخه بهترین جهانی پایین تر است ولی در عوض نتیجه بهتری در این نسخه حاصل می‌شود. مهمترین برتری آن نسبت به نسخه بهترین جهانی، کاهش احتمال بدام افتادن در نقاط بهینه محلی است.

۳-۴- ارزش گذاری^۲

همانطور که در مرحله ۲ الگوریتم های PSO اشاره شد، در هر مرحله از تکرار الگوریتم، می بایست ارزش ذرات محاسبه گردد. ارزش گذاری در PSO همان است که در پردازش تکاملی از آن تحت عنوان محاسبه تابع تطابق یاد می‌شود.

به عنوان مثال اگر هدف یافتن ماکزیمم تابع $f(z) = \sin z - 3z^2$ باشد، تابع تطابق می‌تواند همان $f(x_i)$ انتخاب شود.

۳-۵- همگرایی^۳

¹ . global best
² . Fitness Calculation
³ . Convergence

شرط توقف الگوریتم را می توان بدلخواه برگزید. در برخی روشها، تعداد تکرار مشخص را شرط توقف الگوریتم قرار می دهند، در حالیکه برخی دیگر تغییر جزئی در بهترین ارزش ذرات و یا بردار سرعت، در تکرارهای متوالی را شرط توقف می گیرند.

۳-۶- پارامترهای PSO

اگر رابطه تغییر سرعت که در بند ۵ الگوریتم بهترین جهانی داشتیم را بصورت زیر بنویسیم:

$$\vec{v}_i(t) - \vec{v}_i(t-1) = \rho_1 (\vec{x} pbest_i - \vec{x}_i(t)) + \rho_2 (\vec{x} gbest - \vec{x}_i(t)) \quad (4-3)$$

مشخص است که سمت چپ رابطه در واقع بیانگر شتاب ذره i ام در زمان t می باشد.

$$\vec{a}_i(t) = \vec{v}_i(t) - \vec{v}_i(t-1) \quad (5-3)$$

ضرائب ρ_1, ρ_2 بصورت زیر در نظر گرفته می شوند :

$$\rho_1 = r_1 C_1 \quad (6-2)$$

$$\rho_2 = r_2 C_2 \quad (7-2)$$

که در آن r_1, r_2 اعدادی تصادفی در بازه (۰ و ۱) می باشند و C_1, C_2 ضرائب شتاب نام دارند و اعداد

ثابت مثبتی می باشند.

در مرجع [17] بررسی بر روی تأثیر C_1, C_2 در مسیر حرکت ذرات انجام شده است و شرط لازم

برای همگرایی PSO بصورت زیر آمده است.

$$C_1 + C_2 \leq 4 \quad (۸-۲)$$

در مرجع [18] با توجه به ضرائب شتاب، الگوریتم های PSO به چهار گونه تقسیم بندی شده اند:

- ($C_1, C_2 > 0$) - Full Model
- ($C_1 > 0, C_2 = 0$) - Cognition Only
- ($C_1 = 0, C_2 > 0$) - Social Only
- ($C_1 = 0, C_2 > 0, g \neq i$) - Selfless

علاوه بر پارامتر شتاب که ذکر شد، در PSO استاندارد [16] چند پارامتر دیگر وجود دارد که راجع

به هر کدام توضیحی می آوریم:

• بعد مسئله:

این پارامتر در واقع همان بعد فضایی است که ذرات در آن شناور می باشند. این بعد به تعریف

مسئله و نحوه کد گذاری وابسته است. PSO در مسائل با بعد بالا، عملکرد خوبی نشان داده است [19]

• تعداد ذرات:

تعداد ذرات در اجتماع (Swarm Size) معمولاً بطور سعی و خطا برای همگرایی بهتر انتخاب

می شود. با توجه به اینکه اساس عملکرد PSO بر تفکر جمعی است، تعداد ذرات نسبتاً زیاد به پاسخ های

بهتری منجر می‌شود. ولی در عین حال، مشخص است که ذرات زیادتر، نیازمند محاسبات بیشتر و بنابراین صرف زمان بیشتری است.

• حداکثر سرعت:

قرار دادن یک حد بالا برای سرعت (V_{max}) موجب می‌شود که ذرات با سرعت زیاد در ناحیه جستجو پرش ننمایند و در نتیجه فضا بطور دقیق‌تری برای دستیابی به مناطق مطلوب‌تر بررسی گردد و همچنین از واگرایی الگوریتم به ازای سرعت‌های زیاد خودداری می‌کند. پس از به روز کردن بردار سرعت برای هر ذره در یک تکرار مشخص الگوریتم، حد بالا برای سرعت را بصورت شرط زیر اعمال می‌کنیم:

$$\bullet \text{ اگر } V_i(t) > V_{max} \text{ آنگاه } V_i(t) = V_{max}$$

$$\bullet \text{ اگر } V_i(t) < -V_{max} \text{ آنگاه } V_i(t) = -V_{max}$$

مقدار V_{max} معمولاً با توجه به محدوده تغییرات درایه های بردار مکان ذرات انتخاب می‌شود.

در مرجع [20] نشان داده شده است که در صورت بکارگیری روابط زیر نیازی به V_{max} وجود ندارد.

$$\vec{V}_i(t) = K(\vec{V}_i(t-1) + \rho_1(\vec{x}_{pbest} - \vec{x}_i(t)) + \rho_2(\vec{x}_{gbest} - \vec{x}_i(t))) \quad (9-3)$$

که در آن:

$$K = 1 - \frac{1}{\rho} + \frac{\sqrt{|\rho^2 - 4\rho|}}{2} \quad (10-3)$$

$$\rho = \rho_1 + \rho_2 > 4 \quad (11-3)$$

در رابطه بالا K ضریب انقباض¹ نام دارد.

• اندازه همسایگی:

همانطور که ذکر شد، نسخه بهترین جهانی در واقع، حالت خاصی از بهترین محلی است که در آن فقط یک همسایگی موجود است. بدلیل اینکه در الگوریتم نسخه بهترین جهانی، بدام افتادن در یک می نیمم محلی ممکن می باشد، نسخه بهترین محلی که در آن تعداد همسایگی های جستجوی جواب، بیشتر از یکی است، ترجیح داده می شود. شعاع همسایگی کوچکتر یا بطور معادل تعداد همسایگی بیشتر، موجب می شود که اولاً ناحیه بیشتری از فضا جستجو شود و ثانیاً با جلوگیری از تحت تأثیر بودن تمامی ذرات از یک ذره، احتمال سکون در یک می نیمم محلی به مراتب کمتر می شود. البته تعداد همسایگی های بیشتر موجب همگرایی کندتر می گردد.

• وزن اینرسی

با در نظر گرفتن پارامتری به نام وزن اینرسی (φ) عملکرد بهتری در PSO مشاهده می شود [21].

این ضریب بصورت زیر اعمال می شود:

$$\vec{V}_i(t) = \varphi \vec{V}_i(t-1) + \rho_1 (\vec{x}_{pbest} - \vec{x}_i(t)) + \rho_2 (\vec{x}_{gbest} - \vec{x}_i(t)) \quad (12-3)$$

¹ . Constriction Coefficient

وزن اینرسی اثر سرعت لحظه قبل را بر سرعت حال حاضر کنترل می نماید. مقادیر بزرگ این پارامتر، موجب کاوش وسیعتر در فضای جستجو و مقادیر کوچک آن تمرکز در ناحیه ای کوچک را سبب می شود.

معمولاً، φ ابتدا با مقدار یک تنظیم شده و در طول تکرار الگوریتم، به مرور کاهش می یابد [22].

مسئله مهم بعدی در رابطه با φ ، دقت به برقراری نامساوی زیر می باشد:

$$\varphi > \frac{1}{2}(C_1 + C_2) - 1 \quad (3-13)$$

که در آن C_1 و C_2 به ترتیب ضرایب شتاب مربوط به pbest و gbest می باشد. در صورت عدم بر

قراری (3-13)، رفتار PSO نوسانی و شاید واگرا گردد [23].

3-7- تغییر ساختار استاندارد PSO

پس از تشریح الگوریتم های استاندارد PSO ، حال به چند مورد از تحقیقات اخیر در این زمینه که موجب افزایش کارایی PSO استاندارد می شوند اشاره می کنیم. از اصلاحات انجام شده، استفاده از روش انتخاب نظیر آنچه در الگوریتم ژنتیک مرسوم است، در PSO می باشد. طرحهای زیادی برای طریقه استفاده از روش های انتخاب، بررسی و چندین استراتژی ارائه شده اند [24].

کار مهم دیگر در تغییر ساختار استاندارد PSO، ایجاد الگوریتم های PSO همراه با تولید مثل می -

باشد [24]. این الگوریتمها که Breeding PSO نامیده می شوند، از عملگر برش در الگوریتم ژنتیک بهره

می‌برند. معمولاً برای برش از عملگر استاندارد ترکیب میانی استفاده می‌شود. باید توجه شود که برای انتخاب والدین در ترکیب برش نایستی از رویه انتخاب تناسبی استفاده شود زیرا به دلیل بستگی انتخاب به مقدار ارزش ذرات، ذرات بهتر به سرعت بر بقیه غلبه کرده و همگرایی زودرس را نتیجه می‌دهد [25].

از دیگر تحقیقات به عمل آمده در مورد PSO، ساختارهای همسایگی است. همانطور که دیدیم، الگوریتم بهترین محلی بر پایه اصل شکل دهی به همسایگی ذرات بنیان گذاری شده است. روشهای متعددی برای تعریف همسایه بودن دو ذره در اجتماع وجود دارد. همانطور که قبلاً آمد، تخصیص یک اندیس می‌تواند همسایه بودن دو ذره را مشخص کند. روش دیگری بر پایه فاصله فضایی میان ذرات در

سال ۱۹۹۹ ارائه شده است [26]. در این شیوه دو ذره P_a و P_b در یک همسایگی قرار دارند اگر:

$$\frac{\|\bar{x}_a - \bar{x}_b\|}{d_{man}} < \xi \quad (۱۴-۳)$$

که در آن d_{man} بیشترین فاصله موجود بین جفت های ذرات و پارامتر ξ بصورت زیر در طی

تکرارهای الگوریتم تغییر می‌کند:

$$\xi = \frac{3t + 0.6t_{man}}{t_{man}} \quad (۱۵-۳)$$

در این استراتژی شعاع همسایگی ها با افزایش شماره تکرار الگوریتم، افزایش می‌یابد. مزیت این

نوع انتخاب آن است که در ابتدا تعداد بیشتری همسایگی امکان جستجوی وسیعتر را فراهم می‌آورند و به

تدریج که حوزه تقریباً مشخص می‌شود، با سرعت بیشتر و تعداد همسایگی‌های کمتر، الگوریتم به جواب

نزدیک می‌شود.

برای غلبه بر مشکلات ذیل که ممکن است در الگوریتم PSO بروز کند، نیز روش‌هایی پیشنهاد

گردیده اند. این مشکلات که به عنوان نقاط ضعف PSO می توان از آنها نام برد عبارتند از :

- همگرایی زودرس

- کم بودن سرعت همگرایی به سمت نقطه بهینه

- مواجه شدن با مسائل دینامیکی دنیای واقعی

الگوریتم PSO طبیعتاً یک روش سریع است که دقیقاً همان چیزی است که در مسائل نه چندان پیچیده و با بُعد کم که احتمال دچار شدن در بهینه محلی کم است مطلوب می باشد. ولی درمسائل با بُعد بالاتر و پیچیده تر همگرایی سریع می تواند نشان دهنده این باشد که الگوریتم در یک نقطه بهینه محلی به دام افتاده است. وقتی نقاط بهینه محلی زیادی در فضای جوابها وجود دارد، باید برای گریز از به دام افتادن در نقطه بهینه محلی، دقت و وقت بیشتری را صرف جستجو کرد. از طرفی باید به این نکته هم توجه کرد که سرعت همگرایی به سمت نقطه بهینه نباید خیلی کم باشد.

مشکل بعدی روش PSO در مواجه شدن با مسائل دینامیکی می باشد. از آنجائی که PSO حافظه دار بوده و هر ذره با توجه به تجربه های قبلی خود و سایر اعضا، مسیر حرکت بعدی خود را مشخص می

کند، مشکلاتی در حل مسائل دینامیکی با این روش، پدید می آید.

به طور کلی دو راه ممکن برای تغییر نقطه بهینه عبارتند از:

۱. تغییر نقطه بهینه با گذشت زمان در فضای جستجو

۲. تغییر ارزش نقطه بهینه با گذشت زمان

در مورد اول اگر تغییر موقعیت کوچک باشد، الگوریتم استاندارد PSO قادر است سریعاً خود را با این تغییر تطبیق دهد و به سمت نقطه بهینه جدید حرکت کند. ولی اگر نقطه بهینه سریع حرکت کند و تغییر موقعیت زیاد باشد، PSO به آسانی نمی تواند نقطه بهینه را دنبال کند زیرا برای دنبال کردن نقطه بهینه، برادرهای حافظه $pbest$ و $gbest$ ذرات باید بهبود یابند و به منظور بهبود این برادرها، ذره باید یک موقعیت بهتری را در فضای جستجو بیابد. ولی از آنجائی که ذرات حول نقطه بهینه قبلی تجمع دارند، اگر تغییرات نقطه بهینه جدید نسبت به قبلی زیاد باشد ذرات نمی توانند موقعیت نقطه بهینه جدید را ردیابی کنند و به تبع آن نمی توانند موقعیت بهتری از موقعیتهای $pbest$ و $gbest$ که در حافظه خود دارد، بیابند.

در مورد راه حل دوم، مسئله از این هم بدتر می شود و آن وقتی است که ارزش نقطه بهینه با گذشت زمان کم شود. در این مورد، ذرات قادر نخواهد بود برادرهای حافظه خود را بهبود بخشند چون ارزش موقعیت قبلی آنها با توجه به اینکه ارزش نقطه بهینه با گذشت زمان کم می شود بهتر از حالتی بعدی خواهد بود و بنابراین جمعیت بدون حرکت می باشد.

برای این مشکلات گفته شده روشهای متعددی پیشنهاد شده است که در زیر به بررسی بعضی از

آنها می پردازیم.

۳-۸- غلبه بر همگرایی زودرس :

مشکل همگرایی زودرس در PSO، به خاطر این است که اطلاعات به سرعت بین ذرات رد و بدل می شوند و ذرات به سرعت به هم نزدیک می شوند نتیجتاً تنوع ذرات در فضای جستجو سریعاً کاهش می یابد و فرار از می نیمم های محلی را دشوار می کند. روشهای ارائه شده برای حل این مشکل سعی بر آن دارند که تنوع ذرات را در فضای جستجو کنترل کنند [27,28].

۳-۹- افزایش سرعت همگرایی :

اگرچه سرعت همگرایی در PSO استاندارد، خوب می باشد ولی همان طور که قبلاً ذکر شد، بکارگیری PSO استاندارد، خطر همگرایی زودرس را دارد. از طرفی بکارگیری الگوریتهائی که برای غلبه بر این مشکل ارائه شده باعث کاهش سرعت همگرایی خواهد شد.

افزایش سرعت همگرایی در بسیاری از مقالات با تنظیم پارامترهای وزن اینرسی (ϕ) و ضریب انقباض K به تنهایی صورت گرفته است [29]. ولی مدلهای تغییر یافته ای از PSO هم برای افزایش سرعت همگرایی آن ارائه شده [30]. Kennedy نیز مدلی برای بهبود مسیر حرکت ذرات ارائه داده است [19]. او ابتدا تعداد c مرکز تجمع ذرات که تنوع ذرات، اطراف آنها زیاد است را تخمین می زند. ایده این روش این است که این مراکز باید نزدیک نقطه بهینه باشند و بنابراین می توان این مراکز را با $pbest$ و $gbest$ به

منظور بهبود مسیر جستجوی ذرات جایگزین کرد. نتایج نشان داده که این کار باعث افزایش سرعت همگرایی به سمت نقطه بهینه خواهد شد.

۳-۱۰- مواجه شدن با مسائل دینامیکی :

برای غلبه بر مشکلاتی که الگوریتم استاندارد PSO در مواجه شدن با مسائل دینامیکی با آنها دچار می شود نیز روشهایی ارائه شده است.

به طور کلی همه روشهای ارائه شده باعث می شوند که ذرات حافظه خود را از بهترین موقعیتهایی که در طول اجرای الگوریتم داشته اند به طریقی بازسازی کنند [31,32]. سئوالی که مطرح می شود این است که چه روندی را برای بازسازی حافظه ذرات باید پیش بگیریم. دو روش متفاوتی که نتایج خوبی هم دربرداشته اند [29] بدین قرار است که در یکی از آنها حافظه ذرات به صورت پریودیک^۱ بعد از تعدادی تکرار بازسازی می شود و در دیگری براساس اندازه تغییراتی که در محیط صورت گرفته بازسازی را انجام می دهد. در روش دوم^۲ ما از یک ذره نگهبان^۳ برای شناسائی تغییرات محیط استفاده می کنیم. بدین صورت که این ذره در موقعیت ثابت در تکرارهای متوالی ارزیابی می شود و ارزش آن در هر تکرارهای بعدی برای مشاهده اینکه چه تغییراتی و تا چه اندازه در محیط وجود داشته است مقایسه می شود. عیب

^۱ . Periodic resetting.

^۲ . Triggered resetting

^۳ . sentry particle

این روش در این است که ممکن است تغییراتی در هرکجا فضای جستجو و حتی نقاط دور از ذره نگهبان رخ دهد که دیگر این تغییرات شناسائی نخواهند شد.

روش دیگری که برای غلبه بر این مشکل و بازسازی حافظه ذرات پیشنهاد شده است [33] بدین

صورت است که مرحله ۳ الگوریتم بهترین جهانی را به صورت زیر تغییر می دهیم:

۳- مقایسه ارزش کنونی ذرات و بهترین تجربیاتشان و جایگزینی شرطی زیر :

- اگر $F(P_i) > pbest_i \times T$ آنگاه

$$pbest_i = F(P_i) \quad (\text{الف})$$

$$\bar{x} \ pbest_i = \bar{x}_i(t) \quad (\text{ب})$$

- و اگر $F(P_i) < pbest_i \times T$ آنگاه

$$pbest_i = T \times pbest_i$$

که در آن ضریب T مقداری بین صفر و یک دارد. الگوریتم بدین صورت است کار می کند که در تکرارهای متوالی از ارزش نقاط $pbest$ و $gbest$ ، با ضرب آن در ضریب T ، کاسته می شود تا در تکرارهای بعدی سایر ذرات شانس بیشتری برای جایگزینی با $pbest$ و $gbest$ را داشته باشند و در نتیجه الگوریتم تغییرات احتمالی دینامیکی محیط دنبال کند.

فصل چہارم :

سیستم ANC با الگوریتم

و فقی مبتنی جمعیت PSO

مقدمه

رایج ترین الگوریتمی که برای تغییر ضرایب فیلتر وقفی سیستم ANC وجود دارد، الگوریتم $LMS -x$ filtered می باشد. از مشکلات استفاده از این الگوریتم در سیستم ANC این است که باید تخمین دقیقی از تابع تبدیل مسیر ثانویه وجود داشته باشد [11]. همچنین روش $LMS -x$ filtered از روشهای مبتنی بر کاهش گرادیان می باشد که احتمال به دام افتادن آن در نقاط بهینه محلی وجود دارد. این مخصوصاً برای سیستمهای با خواص غیر خطی بسیار محتمل است.

در طراحی سیستم ANC عملی، دو مسئله را باید در مدل کردن مسیر ثانویه مد نظر داشته باشیم

که عبارتند از :

۱. مسیر ثانویه در عمل، متغیر با زمان می باشد و اگر خطای فاز بین مسیر ثانویه و مدل آن

بین $+90$ تا -90 درجه باشد، سیستم پایدار و در غیر اینصورت ناپایدار خواهد بود [11].

۲. مسیر ثانویه ممکن است خواص غیر خطی از خود نشان دهد که می بایست در مدل

کردن مد نظر قرار دهیم.

تکنیکهای متعددی برای مواجه شدن با مسئله اول ارائه شده است که از جمله مدل کردن بلادرنگ

مسیر ثانویه [34,35]، کنترل رباست [36] و تکنیکهایی که در آن مدلسازی مسیر ثانویه لازم نمی باشد

[37] را می توان نام برد.

برای مواجه شدن با خواص غیر خطی موجود در مسیر ثانویه می توان از کنترل غیر خطی که معمولترین روش پیاده سازی آن شبکه های عصبی می باشد، استفاده کرد. البته مشکل این روشها، سرعت همگرایی کم و پیچیدگی محاسباتی بالای آنها می باشد که در ادامه به آن پرداخته خواهد شد.

الگوریتم ارائه شده در این پایان نامه، مشکلات ناشی از مدل کردن مسیر ثانویه و همچنین پیچیدگی محاسباتی و بدام افتادن در می نیمم محلی که الگوریتم $LMS - x$ filtered با آن مواجه است را ندارد. در روش ارائه شده، از الگوریتم هوشمند PSO به جای الگوریتم LMS، برای تغییر ضرایب فیلتر وفقی استفاده شده است. الگوریتم PSO از روشهای بهینه سازی هوشمند و مبتنی بر جمعیت می باشد که در مسایل بهینه سازی پیچیده منجر به نتایج بسیار خوبی شده است .

مسئله حذف نویز به روش فعال را می توان یک مسئله بهینه سازی با هدف حداقل کردن نویز حاصل از اثر متقابل دو منبع نویز اولیه (نویز محیط) و ثانویه (نویز ساختگی) که خطای باقیمانده نامیده می شود، دانست. بنابراین می توان از الگوریتم PSO برای تغییر وفقی ضرایب فیلتر، به منظور حداقل کردن نویز باقیمانده استفاده کرد.

۴-۱- سیستم ANC ، با بکارگیری الگوریتم وفقی مبتنی بر جمعیت PSO

در این پایان نامه، هر دو نوع سیستم های ANC خطی و غیرخطی را پیشنهاد می دهیم. در سیستم خطی از فیلتر FIR، و در سیستم غیرخطی از شبکه عصبی MLP برای پیاده سازی استفاده می نمائیم.

از آنجائی که بکارگیری الگوریتم وفقی PSO برای کنترلر خطی و غیر خطی کاملاً مشابه می باشد، توضیح الگوریتم PSO برای این سیستم ها به موازات هم انجام می گردد.

روند کلی حل مسئله سیستم ANC خطی به روش PSO به این صورت است که یک فضای M بعدی را در نظر می گیریم. M تعداد ضرایب پارامترهای وفقی برای حذف نویز می باشد که در حالت خطی ضرایب فیلتر FIR و در حالت غیر خطی، وزنه‌های شبکه می باشد که قرار است برای عملکرد مناسب سیستم ANC بهینه شوند. در ابتدا یک جمعیت p ذره ای به صورت نقاطی تصادفی در این فضا انتخاب می کنیم. مکان هر ذره در فضای M بعدی، ضرایب یک فیلتر حذف نویز را تعیین می کند. در جمعیت p نقطه ای به دنبال نقطه ای که معیار حذف نویز را بهینه کند، می گردیم. با یافتن نقطه بهینه در جمعیت انتخاب شده، ذرات جمعیت را در تکرار بعد الگوریتم، به نحوی جابجا می کنیم تا جمعیت را به سمت بهینه واقعی سوق دهیم .

برای یافتن نقطه بهینه در جمعیت انتخاب شده، لازم است که در هر تکرار الگوریتم، ارزش هر یک از ذرات را با توجه به یک تابع معیار که عملکرد سیستم را به ازای هر یک از ذرات توصیف کند، داشته باشیم. بنابراین یک مسئله مهم، تعیین تابع معیار است که قرار است ارزش هر ذره را با آن بسنجیم.

به طور معمول در مسائل بهینه سازی عددی، تابع هدف با بکارگیری تمام داده های ورودی تعریف می شود. ولی در مسئله حذف نویز به روش فعال، اولاً ما تمام نمونه های نویز را در هر لحظه در اختیار

نداریم و ثانیاً اگر بخواهیم خطای حاصل از هر ذره را به ازای تعداد زیادی نمونه های نویز ورودی محاسبه کنیم، همگرایی الگوریتم از لحاظ زمانی بسیار کند خواهد بود.

اگر بخواهیم الگوریتم بهینه سازی PSO مانند الگوریتم x -LMS filtered با رسیدن هر نمونه نویز ورودی، ضرایب فیلتر وقفی را تغییر دهد (تابع هدف را بر اساس خطای حاصل از ورود هر یک نمونه از نویز ورودی تعریف کنیم) و در واقع یک تکرار الگوریتم PSO را با ورود هر نمونه نویز ورودی داشته باشیم، با یک مشکل مواجه خواهیم بود. مشکل از اینجا ناشی می شود که منبع نویز ثانویه در سیستم ANC تک کاناله تنها یک بلندگو می باشد و از طرفی خروجی هر فیلتر (هر ذره) برای ارزیابی ارزش هر ذره باید به منبع نویز ثانویه که بلندگو می باشد وارد شود. بنابراین با رسیدن هر نمونه نویز ورودی، تنها ارزش یک ذره را می توان سنجید و در حالت عادی نمی توان یک تکرار الگوریتم PSO را مانند روش LMS با ورود یک نمونه نویز ورودی داشته باشیم.

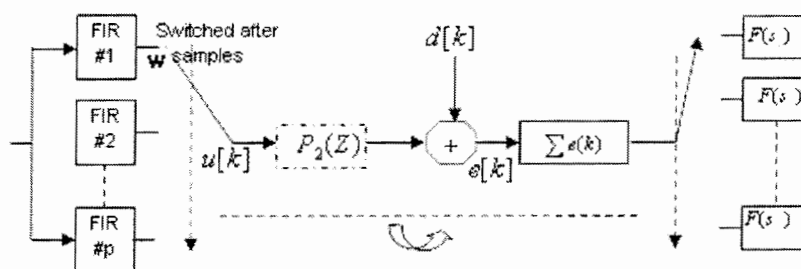
به منظور اینکه با رسیدن هر نمونه نویز ورودی، بتوانیم ارزش فیلترهای متناظر با ذرات جمعیت را بسنجیم، می توانیم یک فضای مجازی، جدا از فضای صوتی تعریف کنیم. در فضای مجازی از تخمین مدل مسیر ثانویه که شامل بلندگو هم هست، استفاده می کنیم و بنابراین می توان با رسیدن هر نمونه نویز ورودی ارزش تمام ذراتی جمعیت را در این فضا سنجید .

فضای صوتی نیز، شامل مدل واقعی سیستم خواهد بود و با توجه به نتایج حاصل در فضای مجازی ، مناسبترین فیلتر را جایگزین فیلتر فضای صوتی خواهیم کرد و با رسیدن هر نمونه نویز ورودی یک تکرار

الگوریتم PSO را خواهیم داشت. مشکل این روش این است که به تخمین مسیر ثانویه نیاز دارد و این در حالیکه می خواهیم تخمین مسیر ثانویه را در روش ارائه شده، حذف نماییم .

بدین منظور، روشی پیشنهاد می کنیم که تخمین مسیر ثانویه را نیاز نداشته باشیم. در روش پیشنهادی، هر ذره، در یک بازه W تائی از نمونه های سیگنال مرجع، به عنوان فیلتر سیستم ANC انتخاب می شود و تابع معیار را نیز بر اساس نویز باقیمانده سیستم ANC، تولید شده در بازه مذکور تعریف می کنیم.

بعد از اینکه ارزش هر ذره جمعیت در بازه ی زمانی متوالی از نمونه های سیگنال نویز مرجع سنجیده شد، الگوریتم PSO ذرات دیگر جمعیت را ارزیابی می کند. شکل زیر بلوک دیاگرام چگونگی سوئیچ کردن روی هر یک از ذراتی جمعیت PSO و یافتن معیار ارزش مربوط به هر ذره (فیلتر) را نشان می دهد:



شکل ۴-۱- بلوک دیاگرام سیستم ANC تک کاناله با استفاده از الگوریتم PSO

بدین ترتیب ملاحظه می شود که بر خلاف الگوریتم مبتنی بر کاهش گرادیان، که با رسیدن هر نمونه از سیگنال نویز ورودی یک تکرار الگوریتم را داشتیم، در روش PSO، یک تکرار الگوریتم با رسیدن چندین نمونه از سیگنال نویز ورودی صورت می گیرد. به عنوان مثال، اگر تعداد ذرات جمعیت را P بگیریم و تعداد نمونه های ورودی که هر ذره جمعیت به عنوان فیلتر سیستم ANC به کار گرفته می شود را W در نظر بگیریم، هر تکرار الگوریتم PSO بعد از هر $W \times P$ نمونه از سیگنال نویز ورودی صورت خواهد گرفت.

الگوریتم کامل این روش برای یک سیستم ANC خطی و غیر خطی به صورت زیر است:

رابطه ورودی و خروجی نمونه n ام فیلتر خطی FIR با توجه به ضرایب فیلتر، مطابق رابطه زیر

تعریف می شود:

$$y[n] = \sum_{k=0}^{M-1} C[k]x[n-k] \quad (1-4)$$

(M تعداد ضرایب فیلتر ، $y[n]$ نمونه n ام سیگنال خروجی ، $C[k]$ ضریب k ام فیلتر و

$x[n-k]$ ، نمونه $n-k$ ام سیگنال ورودی می باشد) .

و برای شبکه عصبی این رابطه با استفاده از روابط ۲-۲۷ و ۲-۳۰ تعریف می شود.

I - ابتدا جمعیت اولیه شامل ذراتی s_1, s_2, \dots, s_p به طور تصادفی شکل می گیرد. موقعیت هر

ذره را در تکرار i ام الگوریتم PSO ، $\vec{x}_i(t)$ نامگذاری می کنیم. این موقعیت، ضرایب فیلتر متناظر

با ذره را مشخص می کند .

$$\vec{x}_i(0) = [Ci(0), Ci(1), \dots, Ci(M-1)] \quad 1 \leq i \leq p \quad (2-4)$$

۲- در مرحله بعد، ارزش هر ذره با توجه به موقعیت آن محاسبه می شود. ارزش هر ذره جمعیت، برابر با عکس مجموع مربعات خطای حذف نویز برای W نمونه سیگنال ورودی در نظر گرفته می شود :

$$F(s_i) = 1 / \sum_{n=0}^{W-1} e_i^2(n) \quad (3-4)$$

$e_i(n)$ ، سیگنال خطای حذف نویز در لحظه n ام در صورت استفاده از فیلتر متناظر با ذره i ام جمعیت PSO می باشد .

۳- در این مرحله مقایسه ای بین ارزش هر ذره و بهترین تجربه گذشته اش ($pbest_i$) صورت می گیرد و در شرایط زیر جایگزینی صورت می گیرد. ($xpbest_i$ ، موقعیت بهترین تجربه گذشته ذره i ام می باشد) .

- اگر $F(s_i) > pbest_i$ آنگاه :

$$pbest_i = F(s_i) \quad (\text{الف})$$

$$xpbest_i = \vec{x}_i(t) \quad (\text{ب})$$

۴- در این مرحله ارزش کنونی هر ذره با بهترین نتیجه قبلی کلیه ذراتی جمعیت ($gbest$) مقایسه می شود و صورت شرایط زیر جایگزینی صورت می گیرد. ($xgbest$ ، موقعیت بهترین نتیجه قبلی کلیه ذراتی جمعیت می باشد)

- اگر $F(s_i) > gbest$ آنگاه :

$$gbest = F(s_i) \quad (\text{الف})$$

$$xgbest = \bar{x}_i(t) \quad (\text{ب})$$

۵- برای هر ذره جمعیت، بردار سرعت طبق رابطه (۴-۴) تغییر می کند و طبق رابطه (۴-۵) ، هر ذره به مکان جدیدش نقل مکان می کند .

$$\vec{v}_i(t) = \vec{v}_i(t-1) + \rho_1 (\vec{x} gbest_i - \vec{x}_i(t)) + \rho_2 (\vec{x} gbest - \vec{x}_i(t)) \quad (4-4)$$

$$\bar{x}_i(t) = \bar{x}_i(t-1) + \vec{v}_i(t) \quad (5-4)$$

۶- به مرحله ۲ می رویم و الگوریتم تکرار می گردد .

با توجه به الگوریتم ذکر شده می توان دید که بر خلاف دروשהای گرادیانی مذکور، روش ارائه شده برای تغییر پارترهای کنترلر به مدل مسیر ثانویه نیازی ندارد.

با پیاده سازی الگوریتم استاندارد PSO، متوجه شدیم الگوریتم بعد از یافتن فیلتر مناسب (که نقطه بهینه الگوریتم PSO خواهد بود)، تمام ذرات جمعیت را با سرعتهای خیلی کم در نزدیکی نقطه بهینه قرار می دهد. در این شرایط، اگر ناگهان ماهیت نویز ورودی مانند فرکانس یا توان نویز تغییر قابل ملاحظه ای کند، الگوریتم نمی تواند این تغییرات را دنبال کند.

برای رفع این مشکل، ابتدا لازم است که الگوریتم تشخیص دهد که شرایط محیطی و به تبع آن نقطه بهینه عوض شده است. برای تشخیص عوض شدن نقطه بهینه، بهترین ذره جمعیت را در هر تکرار، به عنوان ذره نگهبان برچسب می زنیم. بدین منظور، الگوریتم در هر تکرار، ارزش بهترین ذره خود را با ارزش بهترین ذره در تکرار قبلی مقایسه می کند. از آنجائیکه بعد از یافتن نقطه بهینه و قبل از تغییر ناگهانی در ماهیت منبع نویز اولیه، سرعت ذرات خیلی کم است و همگی اطراف نقطه بهینه هستند، می توانیم بهترین ذره در هر تکرار را نقطه بهینه تلقی کنیم که در تکرارهای متوالی هیچگاه ارزش آن با ارزش تکرارهای قبلی خود تغییر چندانی نخواهد کرد. ولی به محض اینکه مشخصات منبع نویز عوض شود، موقعیت نقطه بهینه در فضای جواب عوض شده و بنابراین ارزش بهترین ذره در تکرار بعد از این تغییرات، به شدت کمتر از ارزش تکرار قبلی آن خواهد بود.

بدین ترتیب اگر در تکرارهای متوالی ارزش بهترین ذره در هر تکرار، از ارزش بهترین ذره در تکرار قبلی خود کمتر نباشد (یا تفاوت بسیار ناچیزی داشته باشند) می توانیم ادعا کنیم که تغییر قابل توجه در ماهیت نویز ورودی رخ نداده است، ولی به محض اینکه این تفاوت زیاد شده می توانیم ادعا کنیم که موقعیت نقطه بهینه در فضا عوض شده است. بعد از اینکه تغییرات ایجاد شده در محیط را با استفاده از ذره نگهبان جمعیت تشخیص دادیم، باید ذرات را به نحوی در فضای اطراف پراکنده کنیم که به نقطه بهینه جدید همگرا شوند. در این رابطه دو کار را به طور همزمان انجام می دهیم. اول اینکه سرعت ذرات را به قدری زیاد می کنیم که بتوانند فضای اطراف را برای یافتن نقطه بهینه جدید جستجو کنند و از

اطراف نقطه بهینه قبلی پراکنده شوند و دوم اینکه برای عملکرد درست الگوریتم به هنگام تشخیص عوض شدن نقطه بهینه، باید موقعیت های $xgbest$ و $xpbest_i$ با مقادیر جدیدی جایگزین شوند تا بالاخره خود را به نقطه بهینه برسانند و دلیل آن این است که هرچند با افزایش سرعت ذرات آنها می توانند فضای اطراف را برای یافتن نقاطی با ارزش بالا جستجو کنند، ولی چون نقطه بهینه قبلی نسبت به شرایط قبلی ارزش بسیار بالائی داشته، باز هم نقاطی که ذراتی جمعیت بعد از تغییر شرایط می یابند ممکن است هیچگاه ارزش بالاتر از آن نداشته باشند. پس ضمن اینکه سرعت ذراتی جمعیت را با تشخیص اینکه نقطه بهینه عوض شده است باید افزایش دهیم، بایستی ارزش بالای نقطه بهینه قبلی را هم به طریقی کم کنیم تا ارزش ذراتی جمعیت در طی جستجوی فضای جواب، قابل مقایسه با آن و بلکه بهتر از آن شود.

در الگوریتم تغییر یافته PSO، ضریب Δ را به عنوان معیاری برای بازبینی ارزش ذره نگهبان، به

صورت زیر تعریف می کنیم:

$$\Delta = \begin{cases} 0 & F(gbest(t)) \geq F(gbest(t-1)) \\ \frac{F(gbest(t)) - F(gbest(t-1))}{F(gbest(t))} & F(gbest(t)) < F(gbest(t-1)) \end{cases} \quad (6-4)$$

رابطه بالا نشان می دهد که هرگاه میزان خطای حاصل از بهترین ذره جمعیت در تکرار حاضر، از

این خطا در تکرار قبلی بیشتر شد (ارزش بهترین ذره کمتر شد)، Δ بسته به این اختلاف ارزش، مقداری

بین صفر و یک را خواهد گرفت. در غیر این صورت، یعنی هرگاه تغییر قابل ملاحظه ای در نویز ورودی رخ

نداد (نقطه بهینه تغییر چندانی نداشت)، مقدار Δ صفر و بنابراین مقدار ضریب وزن اینرسی نیز ثابت و برابر φ_0 خواهد بود.

بدین ترتیب اگر Δ مقداری بزرگتر از صفر داشت، می توان ادعا کرد که مشخصات منبع نویز تغییر قابل ملاحظه ای داشته است و بایستی برای جستجوی نقطه بهینه جدید، سرعت ذرات را افزایش دهیم و همزمان ارزش $gbest$ و $pbest_i$ را کم کنیم. برای رسیدن به این منظور، کافی است در هر تکرار الگوریتم، رابطه زیر را چک کنیم :

$$\text{if } \Delta > 0 \quad (a) \quad \begin{cases} gbest = gbest \times T \\ pbest_i = pbest_i \times T \end{cases} \quad (7-4)$$

$$(b) \quad \varphi = \varphi_0 + \eta\Delta$$

η و φ_0 مقادیری ثابت می باشند. بنابراین اگر شرایط منبع نویز عوض شود ($\Delta > 0$)، سرعت ذرات به اندازه $\eta\Delta$ افزایش می یابد و ارزش بهترین ذرات در شرایط قبلی نیز با ضرب شدن در ضریب T کم می شود که الگوریتم بتواند نقطه بهینه جدید را ردیابی کند. در غیر این صورت (عوض نشدن مشخصات منبع نویز) سرعت ذرات مقداری ثابت و برابر φ_0 خواهد داشت و ارزش بهترین ذرات نیز هیچ تغییری نمی کند. به کمک ضریب α در رابطه (7-4)، که مقدار آن بزرگتر از یک انتخاب می گردد، می توان از واگرائی الگوریتم در شرایطی که Δ تغییرات ناگهانی می کند، جلوگیری نمود. از آنجائی که با عوض شدن نقطه بهینه، مقدار باارزش ذرات مجموعه $F(gbest(t))$ خیلی کمتر از $F(gbest(t))$ (بالاترین ارزش ذرات در شرایط همگرائی) می شود، α را می توان مقداری نسبتاً بزرگ انتخاب کرد.

با تغییرات ایجاد شده در الگوریتم، هرگاه نقطه بهینه به دلیل تغییرات محیطی عوض شود، ضریب وزن اینرسی تنها در یک تکرار الگوریتم، به اندازه $\eta\Delta$ افزایش می یابد تا ذرات سرعت کافی را برای

جستجوی نقطه بهینه جدید پیدا کنند. همچنین ارزش $gbest$ و $pbest_i$ با ضرب شدن در ضریب T کم می شود تا در تکرارهای بعدی الگوریتم، جایگزینیهای لازم در موقعیتهای $xgbest$ و $xpbest_i$ ، برای رسیدن به نقطه بهینه صورت گیرد و بنابراین الگوریتم به راحتی نقطه بهینه جدید را ردیابی خواهد کرد.

فصل پنجم: نتایج شبیه سازی

۵-۱- شبیه سازی سیستم ANC خطی :

در این بخش، نتایج حاصل از روش پیشنهادی، با نتایج روش filtered-x LMS که بر روی سیستم خطی مورد مطالعه در مرجع [38] اعمال شده اند، مقایسه می شود.

در مرجع [38]، تابع تبدیل مسیر اولیه، از منبع نویز اولیه تا میکروفون تشخیص خطا عبارت است

از:

$$P_1(z) = 0.8z^{-9} + 0.6z^{-10} - 0.2z^{-11} - 0.5z^{-12} - 0.1z^{-13} + 0.4z^{-14} - 0.05z^{-15}$$

و تابع تبدیل مسیر ثانویه، از منبع نویز ثانویه تا میکروفون تشخیص خطا عبارت است از :

$$P_2(z) = z^{-5} + 2.5z^{-6} + 1.7z^{-7} + 0.15z^{-8} - 0.4825z^{-9} - 0.1862z^{-10}$$

نویز اولیه سینوسی با توان ۰،۵ می باشد که با نویز سفید با توان ۰،۰۰۱۵ آغشته شده است.

فرکانس نمونه برداری سیگنال نویز ۲۰۰۰Hz می باشد.

۵-۱-۱- نتایج حاصل از روش filtered-x LMS برای سیستم ANC خطی با کنترلر خطی FIR :

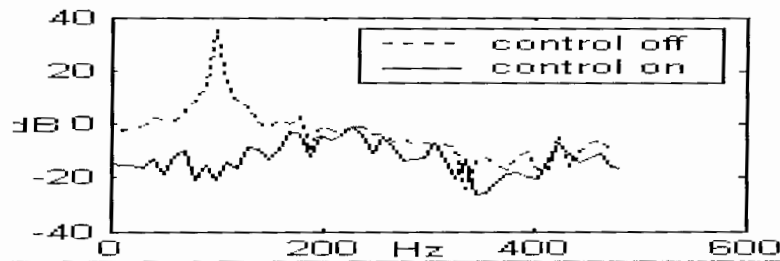
برای شبیه سازی روش filtered-x LMS بر روی سیستم ANC خطی، ضرایب اولیه فیلتر FIR را

به صورت تصادفی و در بازه [۰، ۱] انتخاب کرده ایم. تعداد ضرایب فیلتر وفقی را نیز ۲۰ گرفته ایم.

بزرگترین ضریب همگرایی که به ازای آن الگوریتم filtered-x LMS پایدار است را با سعی و خطا، در

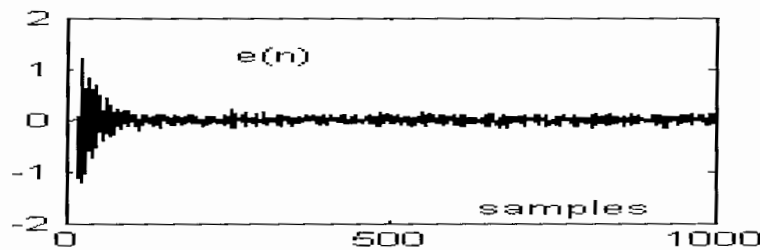
تمام رنج فرکانسی ۵۰ تا ۵۰۰ هرتز نویز، برابر ۰،۰۰۵ بدست آورده ایم.

شکل‌های ۱-۵ و ۲-۵ نتایج حوزه زمان و فرکانس را به ترتیب برای نویز با فرکانسهای ۱۰۰Hz و ۲۰۰Hz نشان می دهد. ملاحظه می شود که الگوریتم LMS برای نویز با فرکانس ۱۰۰Hz پس از حدود ۱۵۰ تکرار همگرا می شود و دانسیته توان نویز به میزان حدوداً ۵۰dB کاهش می یابد. برای فرکانس ۲۰۰Hz همگرایی پس از حدود ۴۰۰ تکرار اتفاق می افتد و کاهش دانسیته توان حدود ۵۰dB می باشد.

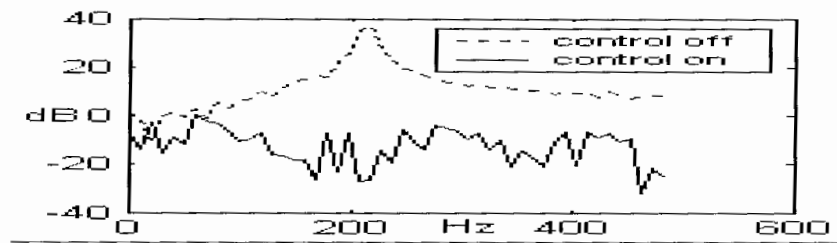


شکل ۱-۵(a) - سیگنال خطای باقیمانده در حوزه فرکانس، برای نویز اولیه ۱۰۰Hz، قبل و بعد از اعمال کنترلر به روش filtered-

x LMS

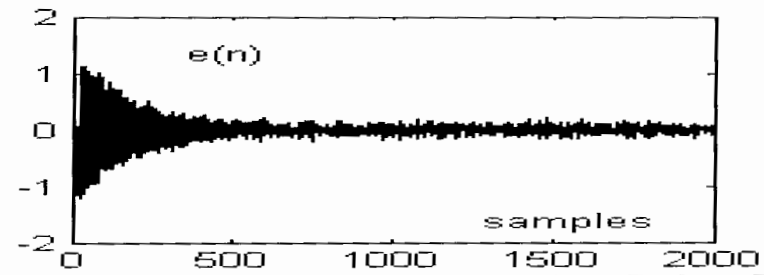


شکل ۱-۵(b) - سیگنال خطای باقیمانده در حوزه زمان برای نویز اولیه ۱۰۰Hz با توان ۰,۰۵، با بکارگیری روش filtered-x LMS



شکل 2-5 (a) - سیگنال خطای باقیمانده در حوزه فرکانس برای نویز اولیه ۲۰۰ Hz، قبل و بعد از اعمال کنترلر به روش filtered-x

LMS

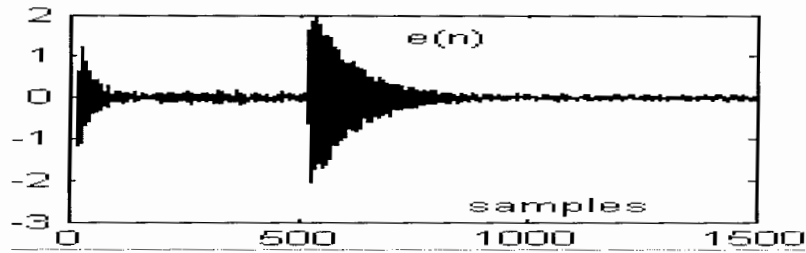


شکل 2-5 (b) - سیگنال خطای باقیمانده برای نویز اولیه ۲۰۰ Hz، با بکارگیری روش filtered-x LMS

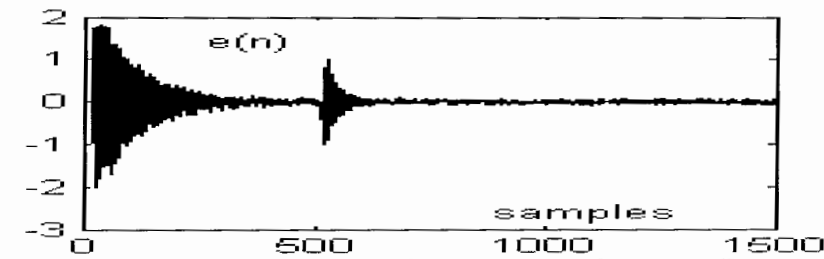
در شکل‌های ۳-۵ و ۴-۵ عملکرد الگوریتم filtered-x LMS با ضریب همگرایی ثابت در تمام فرکانسها، به ازای تغییر ناگهانی فرکانس از ۱۰۰ Hz به ۲۰۰ Hz و برعکس، نشان داده شده است. با مقایسه این نتایج، با نتایج شکل‌های ۱-۵ و ۲-۵، مشاهده می شود که سرعت همگرایی دوباره الگوریتم LMS پس از تغییر در فرکانس، تنها به فرکانس جدید نویز وابسته می باشد و نه به میزان تغییر در فرکانس.

الگوریتم LMS به ازای تغییرات فرکانس در رنج فرکانسی مورد نظر در سیستم ANC مقاوم می

باشد.

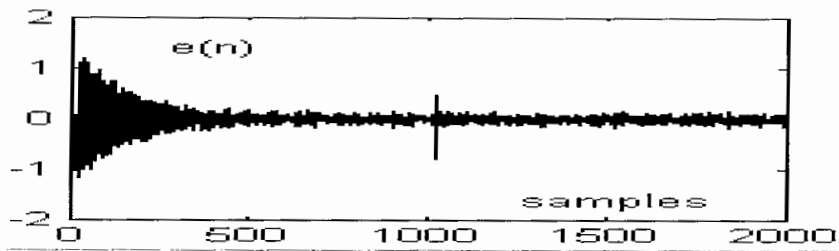


شکل ۳-۵- سیگنال خطای باقیمانده با تغییر ناگهانی فرکانس نویز از ۱۰۰Hz به ۲۰۰Hz با بکارگیری روش filtered-x LMS

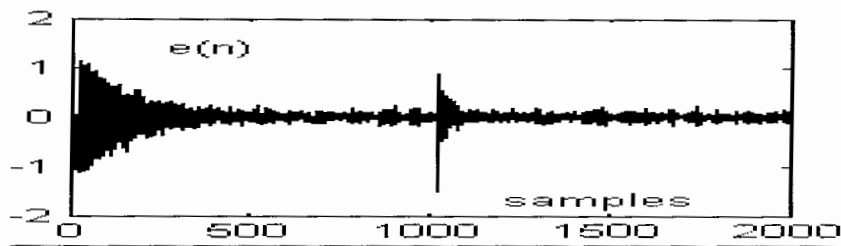


شکل ۴-۵- سیگنال خطای باقیمانده با تغییر ناگهانی فرکانس نویز از ۲۰۰Hz به ۱۰۰Hz با بکارگیری روش filtered-x LMS

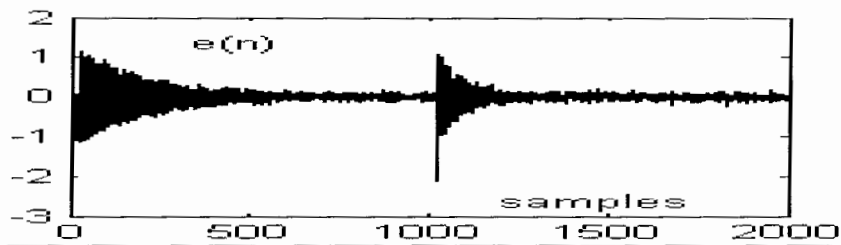
شکل‌های ۵-۵ تا ۵-۸، عملکرد الگوریتم filtered-x LMS را به ازای تغییر در توان نویز ورودی نشان می‌دهد. با مشاهده نتایج می‌توان دید که برای نویز مورد نظر، به ازای افزایش توان نویز تا ۸ برابر توان اولیه، نقطه بهینه تغییر نخواهد کرد و به ازای افزایش توان بیشتر از این حد و حداکثر افزایش توان آن به میزان ۴۰ برابر توان نویز اولیه، الگوریتم نقطه بهینه جدید را می‌یابد و نتیجتاً همگرا می‌شود ولی اگر تغییرات توان نویز را از این حد افزایش دهیم، الگوریتم واگرا خواهد شد.



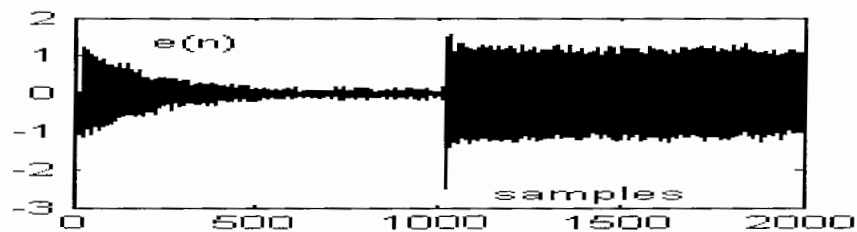
شکل ۵-۵- سیگنال خطای باقیمانده با افزایش توان نویز (از ۰.۵ به ۲) با بکارگیری روش filtered-x LMS



شکل ۵-۶- سیگنال خطای باقیمانده با افزایش توان نویز (از ۰.۵ به ۴.۵) با بکارگیری روش filtered-x LMS



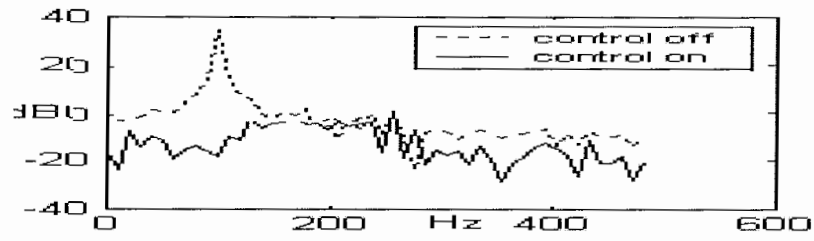
شکل ۵-۷- سیگنال خطای باقیمانده با افزایش توان نویز (از ۰.۵ به ۸) و با بکارگیری روش filtered-x LMS



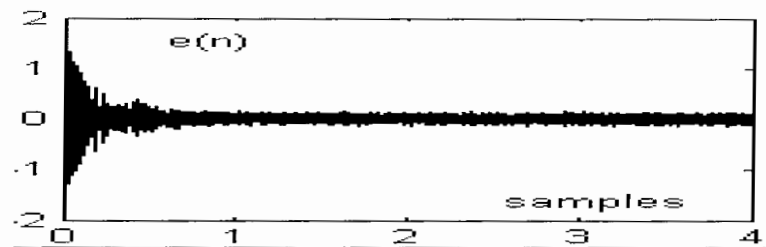
شکل ۵-۸- سیگنال خطای باقیمانده با افزایش توان نویز (از ۰.۵ به ۲۰) و با بکارگیری روش filtered-x LMS

۵-۱-۲- نتایج حاصل از روش PSO برای سیستم ANC خطی با کنترلر خطی FIR :

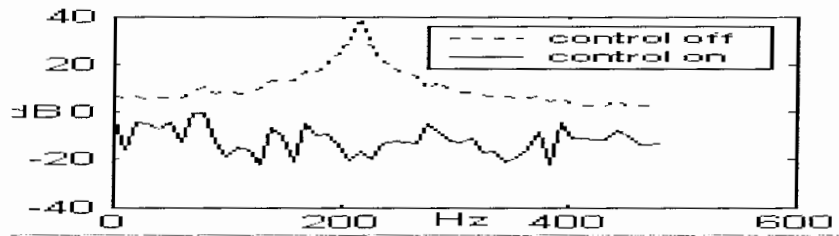
برای شبیه سازی روش PSO، ضرایب اولیه فیلتر را به طور تصادفی و در بازه $[0, 0.1]$ انتخاب کرده ایم. تعداد اعضای جمعیت را ۸، تعداد ضرایب فیلتر وقتی را نیز ۱۰، ضرایب C_1 و C_2 را به ترتیب ۱ و ۲، و ضریب φ_0 مربوط به وزن اینرسی را برابر ۰.۷۵، طول پنجره L را برابر ۴۰ انتخاب کرده ایم. شکل‌های ۵-۹ و ۵-۱۰، نتایج حوزه نمونه ها و فرکانس را به ترتیب برای نویز با فرکانسهای ۱۰۰ Hz و ۲۰۰ Hz نشان می دهند. ملاحظه می شود که الگوریتم PSO برای نویز با فرکانسهای ۱۰۰ Hz و ۲۰۰ Hz پس از حدود ۵۰۰۰ نمونه سیگنال مرجع همگرا می شود که در مقایسه با روش filtered-x LMS سرعت همگرایی آن خیلی کمتر است. از آنجائی که هر تکرار الگوریتم PSO پس از $L \times P$ (تعداد ضرایب فیلتر وقتی و P تعداد ذرات جمعیت می باشد) نمونه سیگنال مرجع صورت می گیرد، پس می توان نتیجه گرفت که الگوریتم PSO پس از ۱۵ تا ۲۰ تکرار الگوریتم که معادل همان ۵۰۰۰ نمونه سیگنال مرجع می باشد، همگرا می شود. همچنین نتایج نشان می دهد که دانسیته توان نویز برای فرکانسهای مورد نظر، به میزان حدوداً ۵۰ dB کاهش می یابد که نشان دهنده این است که الگوریتم PSO از لحاظ قدرت حذف نویز مشابه روش filtered-x LMS می باشد.



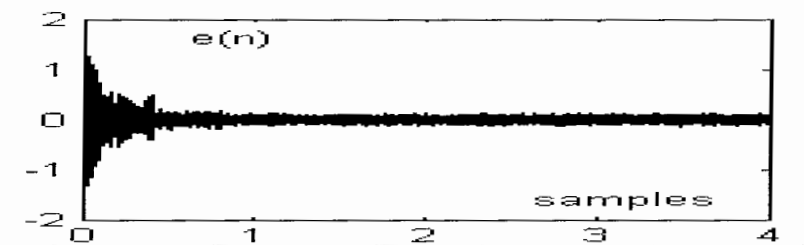
شکل ۹-۵ (a) - سیگنال خطای باقیمانده در حوزه فرکانس برای نویز اولیه ۱۰۰Hz قبل و بعد از اعمال کنترلر به روش PSO



شکل ۹-۵ (b) - سیگنال خطای باقیمانده در حوزه زمان، برای نویز اولیه ۱۰۰Hz با توان ۰،۰۵، با بکارگیری روش PSO



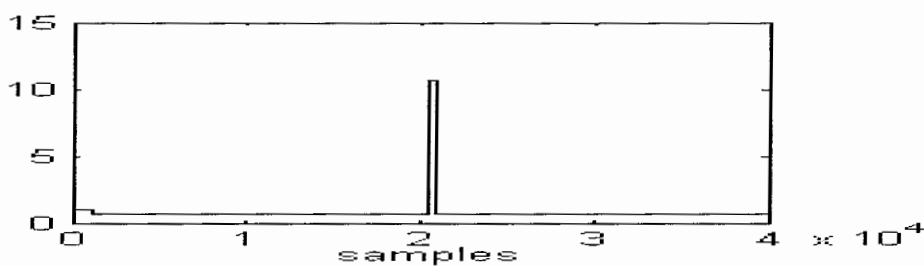
شکل ۹-۵ (a) - سیگنال خطای باقیمانده در حوزه فرکانس برای نویز اولیه ۲۰۰Hz قبل و بعد از اعمال کنترلر به روش PSO



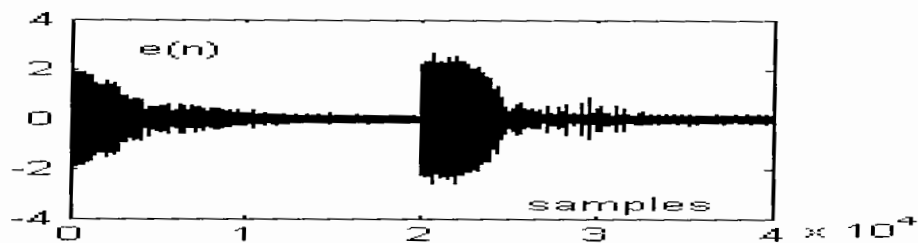
شکل ۹-۵ (b) - سیگنال خطای باقیمانده در حوزه زمان، برای نویز اولیه ۲۰۰Hz با توان ۰،۰۵، با بکارگیری روش PSO

در شکل‌های ۵-۱۱ و ۵-۱۲ عملکرد الگوریتم PSO، به ازای تغییر ناگهانی فرکانس از ۱۰۰Hz به ۲۰۰Hz و برعکس، نشان داده شده است. قسمت (a) شکلها، تغییرات ضریب وزن اینرسی را در طول اجرای الگوریتم PSO نشان می دهند. مشاهده می شود که این ضریب تنها در یک تکرار الگوریتم و آن هم هنگامی که فرکانس نویز تغییر یافته، یک افزایش ناگهانی دارد و در غیر این صورت ثابت می باشد. این افزایش ناگهانی همانطور که قبلا ذکر شد باعث پراکندگی بیشتر ذرات به دلیل افزایش سرعت آنها می شود که شرایط را برای جستجوی نقطه بهینه جدید آماده می کند.

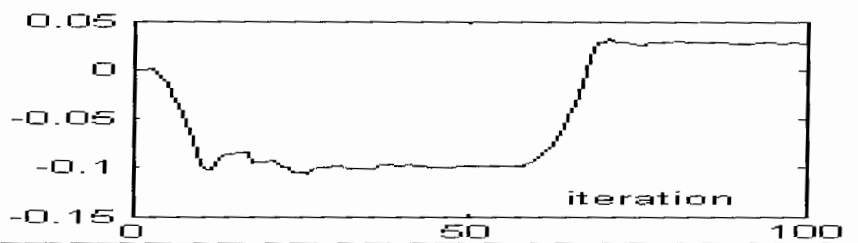
قسمتهای (c) و (d) شکل‌های ۵-۱۱ و ۵-۱۲ نیز تغییرات دو تا از ضرایب فیلتر FIR را در طول اجرای الگوریتم PSO نمایش می دهند. از روی تغییرات این دو ضریب نیز می توان مشاهده کرد که پس از تغییر ناگهانی فرکانس نویز، الگوریتم نقطه بهینه جدید را بعد از تعدادی تکرار می یابد. الگوریتم PSO نیز مانند filtered-x LMS، به ازای تغییرات فرکانس در رنج فرکانسی مورد نظر در سیستم ANC مقاوم می باشد.



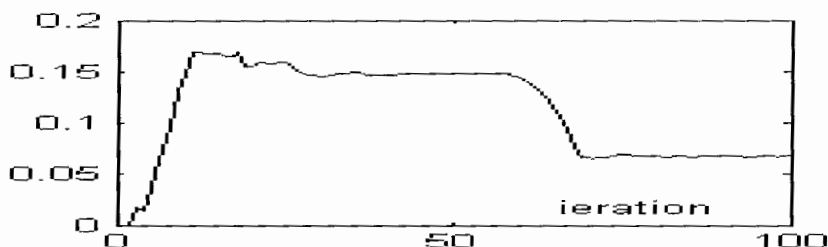
شکل ۵-۱۱(a) - تغییرات ضریب وزن اینرسی به ازای تغییر ناگهانی فرکانس نویز، با بکارگیری روش PSO



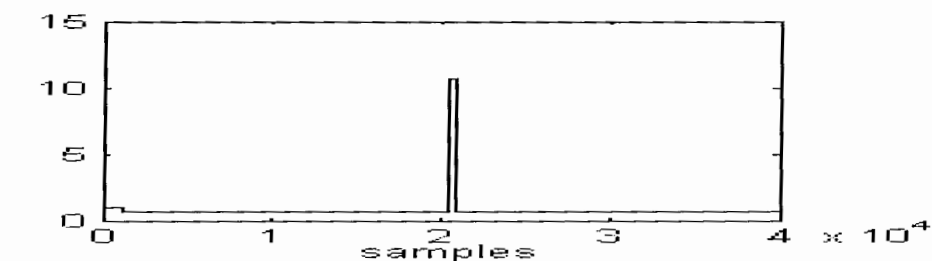
شکل ۵-۱۱(b) - سیگنال خطای باقیمانده با تغییر ناگهانی تغییر ناگهانی فرکانس نویز، با بکارگیری روش PSO



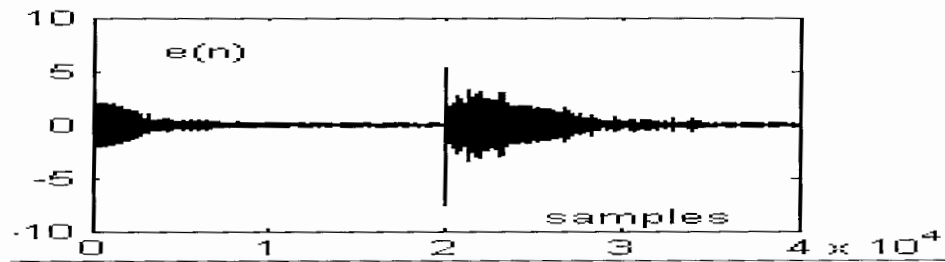
شکل ۵-۱۱(c) - منحنی تغییرات یکی از ضرایب فیلتر با تغییر ناگهانی در فرکانس نویز و با بکارگیری روش PSO



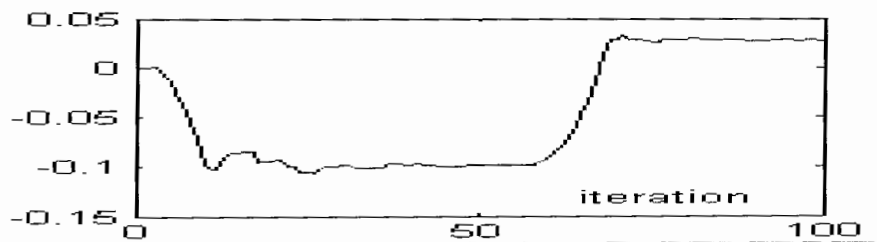
شکل ۵-۱۱(d) - منحنی تغییرات یکی از ضرایب فیلتر با تغییر ناگهانی در فرکانس نویز و با بکارگیری روش PSO



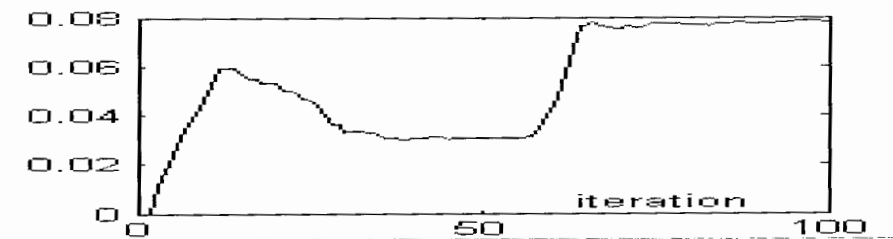
شکل ۵-۱۲(a) - تغییرات ضریب وزن اینرسی به ازای تغییر ناگهانی فرکانس نویز، با بکارگیری روش PSO



شکل ۵-۱۲ (b) - سیگنال خطای باقیمانده با تغییر تغییر ناگهانی فرکانس نویز با بکارگیری روش PSO



شکل ۵-۱۲ (c) - منحنی تغییرات یکی از ضرایب فیلتر با تغییر ناگهانی در فرکانس نویز و با بکارگیری روش PSO



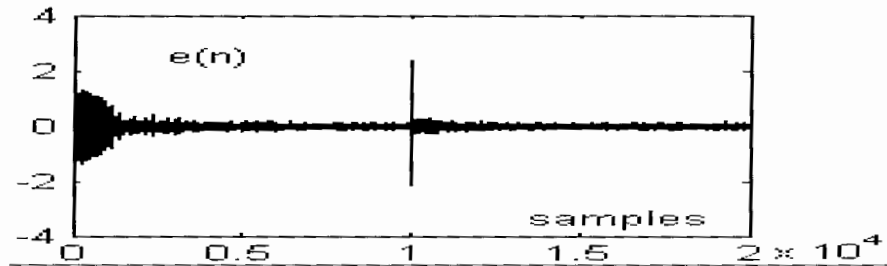
شکل ۵-۱۲ (d) - منحنی تغییرات یکی از ضرایب فیلتر با تغییر ناگهانی در فرکانس نویز و با بکارگیری روش PSO

شکل‌های ۵-۱۳ و ۵-۱۴ عملکرد الگوریتم filtered-x LMS را به ازای تغییر در توان نویز ورودی نشان می‌دهد. با مشاهده نتایج شکل ۵-۱۳ می‌توان دید که برای نویز مورد نظر، به ازای افزایش توان نویز تا ۲۵ برابر توان اولیه، نقطه بهینه تغییر نخواهد کرد. شکل ۵-۱۳ همچنین تغییرات ضریب وزن اینرسی و همچنین یکی از ضرایب فیلتر FIR را در طول اجرای الگوریتم PSO نشان می‌دهد. ملاحظه

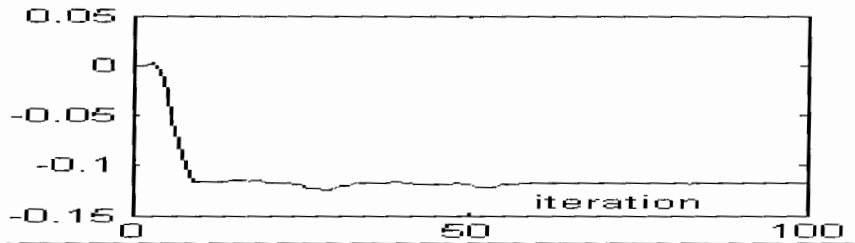
می شود که در این حالت ضریب فیلتر و همچنین ضریب وزن اینرسی هر دو ثابتند که تایید می کنند هیچ تغییری در نقطه بهینه نداشته ایم.

شکل ۵-۱۴ نشان می دهد که به ازای افزایش توان نویز به میزان ۵۰ برابر نویز اولیه، نقطه بهینه تغییر می کند و نتیجتاً همانطور که از روی نتایج مشخص است، ضریب وزن اینرسی یک افزایش ناگهانی برای افزایش سرعت ذرات جمعیت، جهت یافتن نقطه بهینه جدید خواهد داشت.

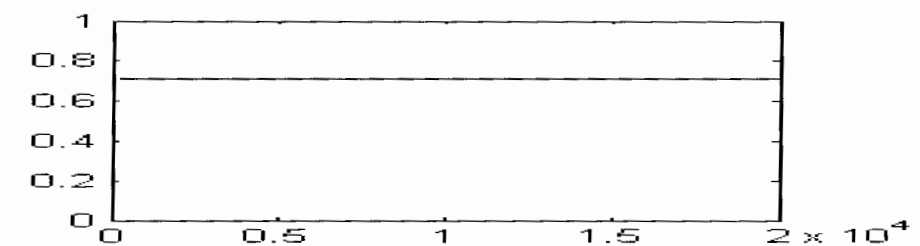
نتایج مربوط به تغییر در توان نویز نشان می دهند که الگوریتم PSO نسبت به تغییرات توان نویز، از الگوریتم filtered-x LMS مقاوم تر می باشد.



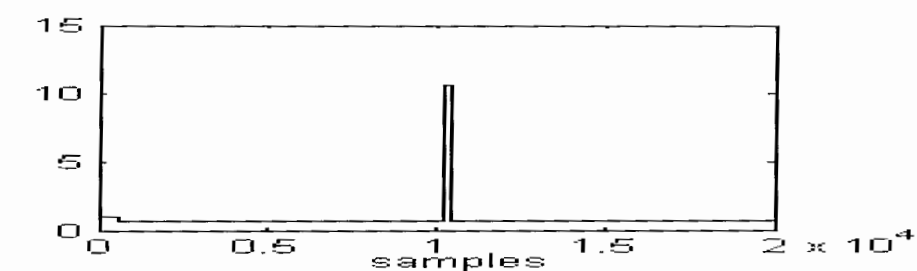
شکل ۵-۱۳ (a) - سیگنال خطای باقیمانده با افزایش ناگهانی در توان آن (از ۰.۵ به ۱۲.۵) و با بکارگیری روش PSO



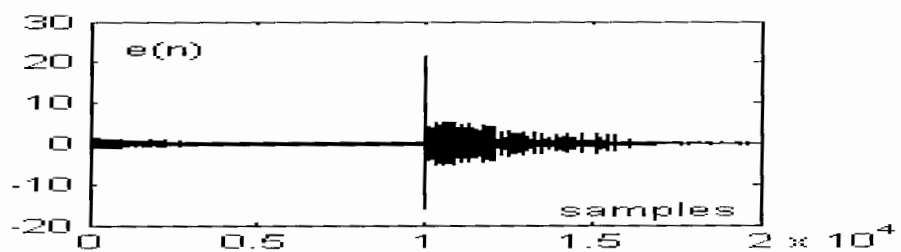
شکل ۵-۱۳ (c) - تغییرات یکی از ضرایب فیلتر با تغییر ناگهانی توان نویز (از ۰.۵ به ۱۲.۵) و با بکارگیری روش PSO



شکل ۵-۱۳ (b) - تغییرات ضریب وزن اینرسی به ازای تغییر توان نویز (از ۰.۵ به ۱۲.۵) و با بکارگیری روش PSO



شکل ۵-۱۴ (a) - تغییرات ضریب وزن اینرسی به ازای تغییر ناگهانی توان نویز (از ۰.۵ به ۵۰) با بکارگیری روش PSO



شکل ۵-۱۴ (b) - سیگنال خطای باقیمانده با تغییر ناگهانی توان نویز (از ۰.۵ به ۵۰)، با بکارگیری روش PSO

۵-۲- شبیه سازی سیستم ANC غیرخطی :

در این بخش، نتایج حاصل از بکارگیری روش PSO، با نتایج یک روش مبتنی بر کاهش گرادیان

که بر پایه filtered-x می باشد مقایسه می شود. روشهای فوق بر روی سیستم غیرخطی مورد مطالعه در

مرجع [11]، در حالی که کنترلر مورد استفاده در هر دو مورد غیر خطی و از نوع شبکه عصبی می باشد، اعمال شده اند.

همچنین نتایج حاصل از بکارگیری هر دو روش، برای سیستم غیر خطی مورد ذکر، در حالی که از یک کنترلر خطی FIR در سیستم ANC بهره گرفته شده، آورده شده است و با حالتی که کنترلر غیر خطی بکار گرفته شده مقایسه صورت گرفته است.

در مرجع [11]، تابع تبدیل مسیر اولیه، از منبع نویز اولیه تا میکروفون تشخیص خطا به صورت تابع غیر خطی زیر داده شده است:

$$d(k+1) = x(k-4) - 0.3x(k-5) + 0.2x(k-6) + 0.9x^2(k-4)$$

و تابع تبدیل مسیر ثانویه، از منبع نویز ثانویه تا میکروفون تشخیص خطا، خطی و به صورت زیر داده شده است:

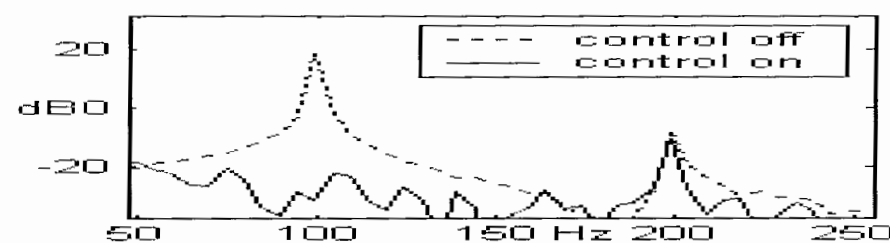
$$y(k+1) = u(k-1) + 0.5u(k-2)$$

نویز اولیه، نویز سینوسی است که با نویز سفید آغشته شده است. فرکانس نمونه برداری سیگنال نویز ۲۰۰۰Hz می باشد.

۵-۲-۱- نتایج حاصل از الگوریتم مبتنی بر کاهش گرادیان^۱ و بر پایه filtered-x برای سیستم ANC غیرخطی با کنترلر شبکه عصبی:

شکل‌های ۵-۱۵ و ۵-۱۶ نتایج حوزه نمونه‌ها و فرکانس را به ترتیب برای نویز با فرکانسهای ۱۰۰Hz و ۲۰۰Hz نشان می‌دهد. ملاحظه می‌شود که الگوریتم مبتنی بر کاهش گرادیان و بر پایه filtered-x، برای نویز با فرکانس ۱۰۰Hz پس از حدود ۱۵۰۰ تکرار همگرا می‌شود. دانسیته توان نویز در فرکانس اصلی نویز به میزان حدوداً ۴۲dB و در هارمونیک فرکانسی که در اثر خاصیت غیرخطی مسیر اولیه بوجود می‌آید به میزان تقریباً ۳dB کاهش می‌یابد.

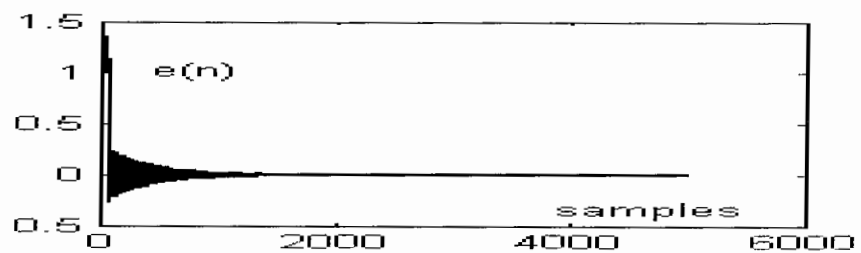
برای فرکانس ۲۰۰Hz همگرایی پس از حدود ۲۲۰۰ تکرار اتفاق می‌افتد، دانسیته توان نویز در فرکانس اصلی نویز به میزان حدوداً ۴۰dB و در هارمونیک فرکانسی که در اثر خاصیت غیرخطی مسیر اولیه بوجود می‌آید به میزان تقریباً ۵dB کاهش می‌یابد.



شکل ۵-۱۵(a) - سیگنال خطای باقیمانده در حوزه فرکانس برای سیستم ANC غیرخطی و نویز اولیه ۱۰۰Hz، قبل و بعد

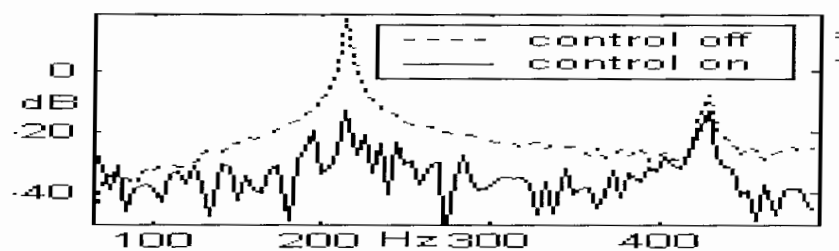
از اعمال روش filtered-x LMS برای کنترلر غیر خطی

^۱. steepest descent algorithms



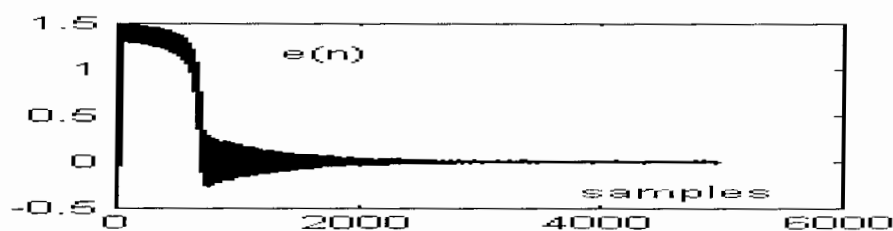
شکل ۵-۱۵ (b) - سیگنال خطای باقیمانده برای سیستم ANC غیر خطی و نویز اولیه ۱۰۰Hz، با بکارگیری روش

filtered-x LMS برای کنترلر غیر خطی



شکل ۵-۱۶ (a) - سیگنال خطای باقیمانده در حوزه فرکانس برای سیستم ANC غیر خطی و نویز اولیه ۲۰۰Hz، قبل و بعد

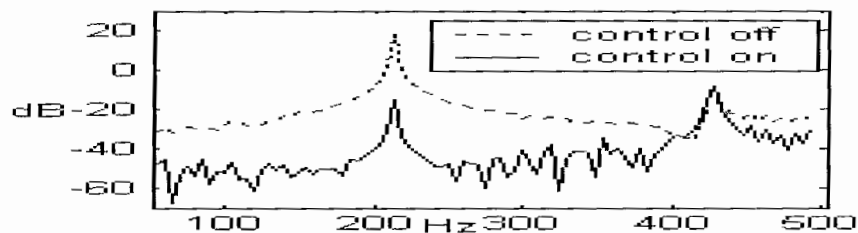
از اعمال روش filtered-x LMS برای کنترلر غیر خطی



شکل ۵-۱۶ (b) - سیگنال خطای باقیمانده برای سیستم ANC غیر خطی و نویز اولیه ۲۰۰Hz، با بکارگیری روش

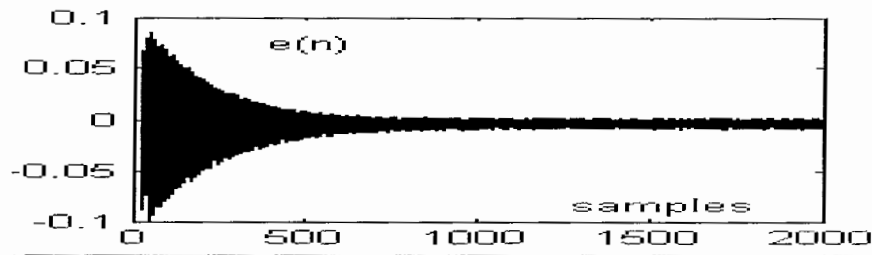
filtered-x LMS برای کنترلر غیر خطی

نتایج روش filtered-x LMS برای نویز ۲۰۰Hz که بر روی کنترلر خطی FIR اعمال شده است نیز برای سیستم غیر خطی مورد بحث، جهت مقایسه با نتایج کنترلر غیر خطی، در شکل ۵-۱۷ آورده شده است. با بکارگیری کنترلر خطی بر مبنای الگوریتم filtered-x LMS برای سیستم غیر خطی، همگرایی پس از حدود ۸۰۰ تکرار اتفاق می افتد، دانسیته توان نویز در فرکانس اصلی نویز به میزان حدودا ۴۰dB و در هارمونیک فرکانسی که در اثر خاصیت غیر خطی مسیر اولیه بوجود می آید هیچ کاهش نویزی را نخواهیم داشت. ملاحظه می شود که در روش filtered-x LMS، بکارگیری کنترلر غیر خطی برای سیستم مورد مطالعه، در هارمونیک فرکانسی نویز، نتیجه بهتری را از کنترلر خطی می دهد ولی از لحاظ سرعت همگرایی کندتر از آن می باشد.



شکل ۵-۱۷ (a) سیگنال خطای باقیمانده در حوزه فرکانس برای سیستم ANC غیر خطی و نویز اولیه ۲۰۰Hz، قبل و بعد از اعمال

روش filtered-x LMS برای کنترلر خطی FIR



شکل ۵-۱۷ (b) - سیگنال خطای باقیمانده برای سیستم ANC غیر خطی و نویز اولیه ۲۰۰Hz، با بکارگیری روش filtered-x

LMS برای کنترلر خطی FIR

۵-۲-۲- نتایج حاصل از الگوریتم PSO برای سیستم ANC غیرخطی با کنترلر شبکه عصبی :

شکل‌های ۵-۱۸ و ۵-۱۹ نتایج حوزه نمونه ها و فرکانس را به ترتیب برای نویز با فرکانسهای

۱۰۰Hz و ۲۰۰Hz نشان می دهد. ملاحظه می شود که الگوریتم PSO، برای نویز با فرکانس ۱۰۰Hz

پس از حدود ۵۰۰۰ تکرار همگرا می شود. دانسیته توان نویز در فرکانس اصلی نویز به میزان حدوداً

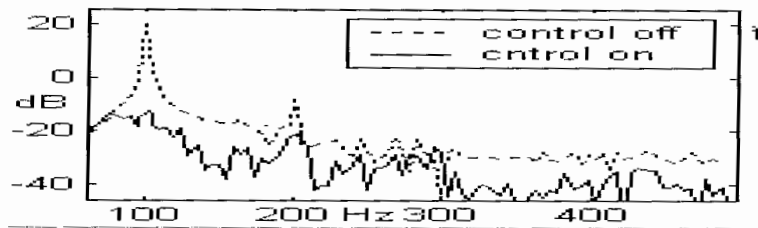
۴۰dB و در هارمونیک فرکانسی که در اثر خاصیت غیر خطی مسیر اولیه بوجود می آید به میزان تقریباً

۱۵dB کاهش می یابد.

برای فرکانس ۲۰۰Hz، همگرایی پس از حدود ۵۰۰۰ تکرار اتفاق می افتد، دانسیته توان نویز در

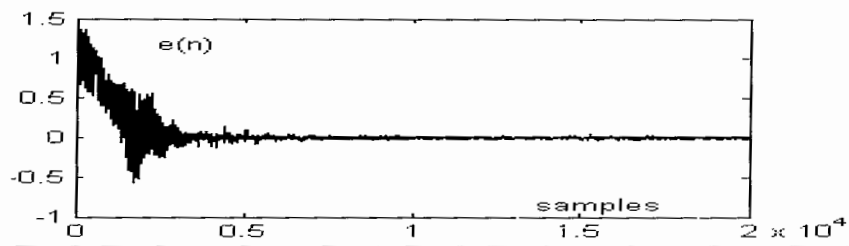
فرکانس اصلی نویز به میزان حدوداً ۴۰dB و در هارمونیک فرکانسی که در اثر خاصیت غیر خطی مسیر

اولیه بوجود می آید به میزان تقریباً ۱۵dB کاهش می یابد.



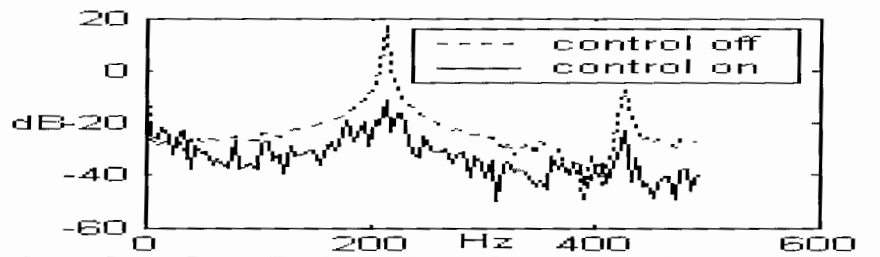
شکل ۵-۱۸ (a) - سیگنال خطای باقیمانده در حوزه فرکانس، برای سیستم ANC غیر خطی و نویز اولیه ۱۰۰Hz، قبل و بعد از اعمال

روش PSO برای کنترلر غیر خطی



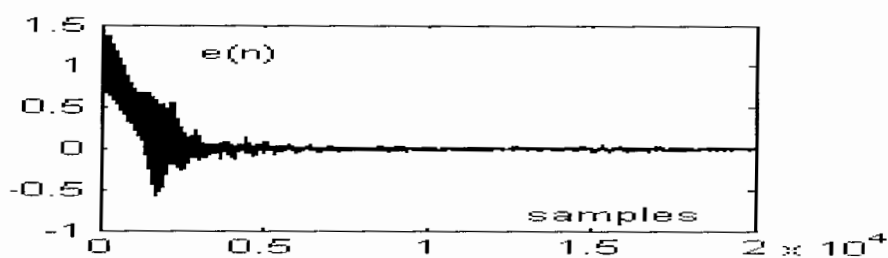
شکل ۵-۱۸ (b) - سیگنال خطای باقیمانده برای سیستم ANC غیر خطی و نویز اولیه ۱۰۰Hz، با بکارگیری روش PSO برای

کنترلر غیر خطی



شکل ۵-۱۹ (a) - سیگنال خطای باقیمانده در حوزه فرکانس برای سیستم ANC غیر خطی و نویز اولیه ۲۰۰Hz، قبل و بعد

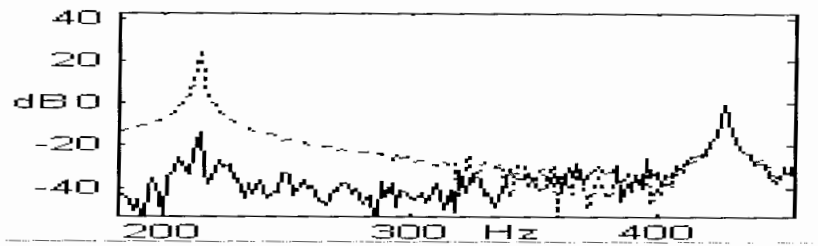
از اعمال روش PSO برای کنترلر غیر خطی



شکل ۵-۱۹ (b) - سیگنال خطای باقیمانده برای سیستم ANC غیر خطی و نویز اولیه ۲۰۰Hz، با بکارگیری روش PSO برای

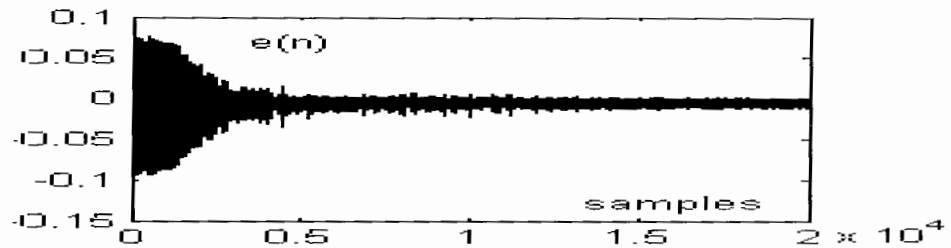
کنترلر غیر خطی

نتایج روش PSO برای نویز ۲۰۰Hz که بر روی کنترلر خطی FIR اعمال شده است نیز برای سیستم غیر خطی مورد بحث، جهت مقایسه با نتایج کنترلر غیر خطی، در شکل ۵-۲۰ آورده شده است. با بکارگیری کنترلر خطی بر مبنای الگوریتم PSO برای سیستم غیر خطی، همگرایی پس از حدود ۵۰۰۰ تکرار اتفاق می افتد، دانسیته توان نویز در فرکانس اصلی نویز به میزان حدوداً ۴۰dB و در هارمونیک فرکانسی که در اثر خاصیت غیر خطی مسیر اولیه بوجود می آید هیچ کاهش نویزی را نخواهیم داشت. ملاحظه می شود که در روش PSO، بکارگیری کنترلر غیر خطی برای سیستم مورد مطالعه، در هارمونیک فرکانسی نویز، نتیجه بهتری را از کنترلر خطی می دهد و از لحاظ سرعت همگرایی نیز تفاوتی با کنترلر خطی نخواهد داشت.



شکل ۵-۲۰ (a) - سیگنال خطای باقیمانده در حوزه فرکانس برای سیستم ANC غیر خطی و نویز اولیه ۲۰۰ Hz، قبل و بعد

از اعمال روش PSO برای کنترلر خطی FIR



شکل ۵-۲۰ (b) - سیگنال خطای باقیمانده برای سیستم ANC غیر خطی و نویز اولیه ۲۰۰ Hz، با بکارگیری روش PSO برای کنترلر

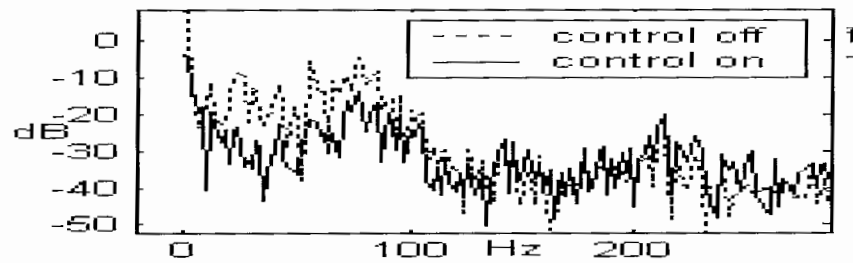
خطی FIR

در نهایت با مقایسه نتایج حاصل از روش PSO و روش مبتنی بر کاهش گرادیانی مبتنی و بر پایه-filtered-x بر روی کنترلر غیر خطی، مشاهده می شود که روش پیشنهادی در هارمونیک فرکانسی نویز نتیجه بهتری را از روش گرادیانی filtered-x می دهد. همچنین با مقایسه سرعت همگرایی این دو روش به هنگام پیاده سازی آنها روی کنترلر غیر خطی، ملاحظه می شود که هر چند سرعت همگرایی روش گرادیانی filtered-x بهتر از PSO می باشد، ولی تفاوت فاحشی که در اختلاف سرعت همگرایی آنها به هنگام پیاده سازی این دو روش بر روی کنترلر خطی داشت، در این حالت وجود ندارد. به عنوان مثال با مشاهده نتایج در فرکانس ۲۰۰ Hz می توان دید که سرعت همگرایی با روش گرادیانی filtered-x حدود

۲۲۰۰ تکرار می باشد و برای روش PSO همگرایی بعد از حدود ۵۰۰۰ تکرار اتفاق می افتد. این در حالی است که به هنگام پیاده سازی روش filtered-x LMS بر روی سیستم خطی در فرکانس مورد نظر، الگوریتم پس از ۴۰۰ تکرار همگرا می شد و PSO پس از ۵۰۰۰ تکرار.

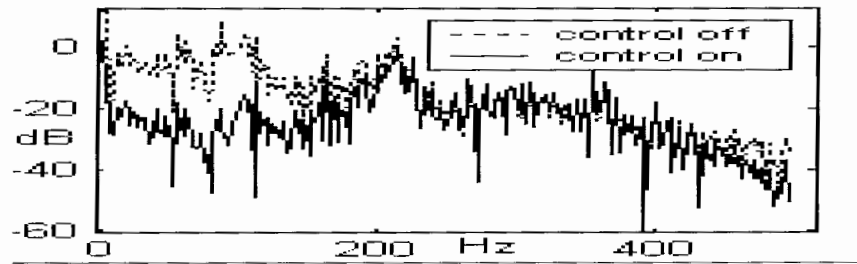
۵-۳- شبیه سازی روش پیشنهادی برای حذف نویز واقعی حاصل از ماشین تراش و سشوار

در انتها برای بررسی عملکرد روش پیشنهادی در مورد نویزهای واقعی، این روش بر روی دو نویز واقعی که از یک ماشین تراش و یک سشوار بدست آمده اند، اعمال شده است. نتایج آنها در شکل‌های ۵-۲۱ و ۵-۲۲ آورده شده است. ملاحظه می شود که روش پیشنهادی در فرکانسهایی که دامنه نویز زیاد است، حدود ۲۰dB کاهش نویز را داریم.



شکل ۵-۲۱ - سیگنال خطای باقیمانده در حوزه فرکانس برای نویز واقعی حاصل از یک ماشین تراش، قبل و بعد

از اعمال روش PSO برای کنترلر خطی FIR



شکل ۵-۲۲ - سیگنال خطای باقیمانده در حوزه فرکانس برای نویز واقعی حاصل از یک سشوار، قبل و بعد

از اعمال روش PSO برای کنترلر خطی FIR

۵-۴- مقایسه روشهای filtered-x LMS و PSO پیشنهادی از لحاظ پیچیدگی محاسباتی :

پیچیدگی محاسباتی هر الگوریتم، مربوط به تعداد جمع کننده ها و ضرب کننده هائی است که برای پیاده سازی الگوریتم، نیاز می باشد. در این قسمت، به طور جداگانه به مقایسه پیچیدگی محاسباتی الگوریتم پیشنهادی با روش filtered-x LMS برای کنترلر خطی و همچنین با روشهای گرادینتی برای کنترلر غیر خطی، می پردازیم.

۵-۴-۱- مقایسه دو روش، در یک سیستم ANC با مدل خطی :

برای محاسبه تعداد جمع کننده ها و تعداد ضرب کننده های مورد نیاز در یک تکرار الگوریتم filtered-x LMS، باید به رابطه (۲-۱۳) برای محاسبه خروجی کنترلر FIR در هر تکرار الگوریتم، و همچنین روابط (۲-۱۸) و (۲-۲۰) برای تعیین ضرایب کنترلر وقتی در هر تکرار، مراجعه کرد. برای روشنتر شدن مطلب، روابط مربوطه را بازنویسی می کنیم.

$$y'[k] = \sum_{L=0}^{M-1} b_i[k]x[k-i] \quad (۲-۱۳)$$

$$V[z] \Delta P_2(z) X(z) \quad (۲-۱۸)$$

$$b_i[k+1] = b_i[k] + \eta e[k]v[k-i] \quad (۲-۲۰)$$

رابطه (۲-۱۳)، از آنجائی که مرتبه کنترلر M می باشد، به تعداد M ضرب کننده و M-1 جمع کننده نیازمند است.

رابطه (۲-۱۸)، از آنجائی که مرتبه فیلتر FIR که برای مدل کردن مسیر ثانویه استفاده شده است، L می باشد، به تعداد L ضرب کننده و L-1 جمع کننده نیازمند است. و در نهایت رابطه (۲-۲۰) به تعداد M جمع کننده (برای هر ضریب کنترلر یک جمع کننده) و M+1 ضرب کننده نیازمند می باشد. پس با در نظر گرفتن سه رابطه مذکور، الگوریتم filtered-x LMS به تعداد 2M+L+1 ضرب کننده و 2M+L-2 جمع کننده در هر تکرار الگوریتم خود، نیازمند می باشد.

برای محاسبه تعداد جمع کننده و ضرب کننده مورد نیاز روش PSO پیشنهادی، باید به روابط مورد نیاز برای تغییر موقعیت ذرات، یعنی رابطه تغییر سرعت ذرات (۴-۴)، و تغییر موقعیت آنها (۴-۵) اشاره کرد. از طرفی رابطه مربوط به محاسبه خروجی فیلتر FIR را در (۴-۱) داریم. برای روشنتر شدن موضوع، این روابط را دوباره می نویسیم.

$$y[n] = \sum_{k=0}^{M-1} C[k]x[n-k] \quad (۴-۱)$$

$$\vec{v}_i(t) = \vec{v}_i(t-1) + \rho_1 (\vec{x} \text{ gbest}_i - \vec{x}_i(t)) + \rho_2 (\vec{x} \text{ gbest} - \vec{x}_i(t)) \quad (۴-۲)$$

$$\vec{x}_i(t) = \vec{x}_i(t-1) + \vec{v}_i(t) \quad (۴-۳)$$

واضح است که برای رابطه (۴-۱)، تعداد جمع کننده و ضرب کننده مورد نیاز، به تعداد ذرات جمعیت وابستگی ندارد و برای هر نمونه نوبت اولیه، تعداد M ضرب کننده و M-1 جمع کننده نیازمندیم.

برای رابطه (۴-۵)، به ازای هر ذره جمعیت و در یک تکرار الگوریتم PSO (و نه به ازای هر نمونه نوبت اولیه)، به تعداد ابعاد موقعیت ذره، $\vec{x}_i(t)$ ، که همان ابعاد فیلتر FIR، یعنی M می باشد، جمع کننده نیازمندیم. برای رابطه (۴-۴) نیز به ازای هر ذره جمعیت و در یک تکرار الگوریتم PSO، نیاز به M جمع کننده برای محاسبه هر کدام از جملات دوم و سوم داریم. همچنین به تعداد 2M جمع کننده برای جمع

کردن این دو جمله با هم و جمع آنها با سرعت قبلی ذره نیازمندیم. نتیجتاً در هر تکرار الگوریتم، برای محاسبه رابطه مذکور به $4M$ جمع کننده برای تشکیل بردار سرعت هر ذره نیازمندیم. اگر P تعداد ذرات جمعیت باشد، می توان نتیجه گرفت که برای برآوردن دو رابطه مذکور، به تعداد $5MP$ جمع کننده (MP جمع برای رابطه $(\Delta-4)$ و $4MP$ برای رابطه $(4-4)$) نیازمندیم.

ولی از آنجائی که هر تکرار الگوریتم پیشنهادی به ازای هر $W \times P$ نمونه نویز اولیه به سیستم ANC صورت می گیرد (W طول پنجره ای ایست که برای سنجش ارزش اعضای جمعیت استفاده می شود)، برای هر $W \times P$ نمونه نویز ورودی، به تعداد $5MP$ جمع کننده نیازمندیم و نتیجتاً به طور متوسط برای

هر نمونه نویز اولیه، برای روابط $(4-4)$ و $(\Delta-4)$ به $5 \frac{PM}{WP} = 5 \frac{M}{W}$ جمع کننده نیازمندیم.

با محاسبات فوق، به طورمتوسط در الگوریتم پیشنهادی، به $1 - \Delta \frac{M}{W} + M$ جمع کننده و M ضرب کننده نیاز داریم.

در جدول زیر مقایسه های بین تعداد جمع کننده و ضرب کننده مورد نیاز با هر نمونه نویز اولیه، برای دو روش مذکور صورت گرفته است.

Operator	MUL	ADD
Filtered-x LMS	$2M+L+1$	$2M+L-2$
Modified PSO	M	$M + \Delta \frac{M}{W} - 1$

جدول ۵-۱- مقایسه روش پیشنهادی با روش filtered-x LMS از لحاظ پیچیدگی محاسباتی

با مشاهده جدول ۵-۱ ملاحظه می شود که تعداد ضرب کننده های روش پیشنهادی بسیار کمتر از روش filtered-x LMS می باشد. همچنین از آنجائی که $5 \frac{M}{W}$ مقدار کوچکی (کمتر از $0.2M$) می باشد، تعداد جمع کننده های مورد نیاز روش پیشنهادی بسیار کمتر از روش filtered-x LMS می باشد و

نتیجتاً روش پیشنهادی پیچیدگی محاسباتی کمتری نسبت به روش filtered-x LMS برخوردار می باشد.

۵-۴-۱- مقایسه پیچیدگی محاسباتی دو روش، در سیستم ANC غیر خطی :

در حالتی که از کنترلر غیر خطی استفاده می شود، الگوریتمهای گرادینانی از پیچیدگی محاسباتی خیلی بیشتری نسبت به حالتی که برای کنترلر خطی بکار می روند، برخوردارند که بکارگیری آنها را در سیستمهای واقعی با مشکل مواجه کرده است. در مورد روش گرادینانی مورد استفاده در این پایان نامه، با مراجعه به روابط (۲-۳۳) و (۲-۳۴) می توان به پیچیدگی محاسباتی بسیار بالای آن پی برد. این در حالی است که بکارگیری روش پیشنهادی برای کنترلر غیر خطی از پیچیدگی محاسباتی برابر با کنترلر خطی برخوردار است، با این تفاوت که در کنترلر غیر خطی، L تعداد ضرایب شبکه عصبی خواهد بود. از مقایسه صورت گرفته بین پیچیدگی محاسباتی دو روش، نتیجه می شود که روش پیشنهادی در پیاده سازی زمان واقعی نسبت به روشهای گرادینانی که دارای پیچیدگی محاسباتی بسیار بالا می باشند (مخصوصاً در سیستمهای غیر خطی)، بسیار مناسبتر می باشد.

فصل ششم :

نتیجه گیری و پیشنهادات

نتیجه گیری :

در این پایان نامه، یک الگوریتم وفقی جدید مبتنی بر اجتماع ذرات (PSO) برای سیستم تک کاناله حذف نویز به روش فعال (ANC) ارائه شد. الگوریتم پیشنهاد شده به سیستم ANC اعمال گردید. در این رابطه سیستم ANC را یکبار به صورت خطی و بار دیگر به صورت غیرخطی مدل نمودیم. جهت ارزیابی روش پیشنهاد شده در سیستمهای خطی و غیرخطی ANC، الگوریتم متداول گرادایانی filtered-x LMS را پیاده نموده، دو روش را از لحاظ قدرت حذف نویز، سرعت همگرایی، مقاوم بودن در برابر تغییرات محیطی، پیچیدگی محاسباتی و سایر موارد قابل ذکر، مقایسه نموده ایم.

• قدرت حذف نویز

الگوریتم پیشنهادی از لحاظ قدرت حذف نویز در سیستم خطی مشابه روش filtered-x LMS می باشد و با مقایسه نتایج حاصل از روش PSO و روش filtered-x Backpropagation بر روی کنترلر غیر خطی، مشاهده می شود که روش پیشنهادی هارمونیکای فرکانس اصلی نویز را بهتر از روش گرادایانی filtered-x کاهش می دهد. نتیجتاً روش پیشنهادی از لحاظ قدرت حذف نویز، در سیستمهای غیرخطی بر روشهای گرادایانی برتری دارد.

• سرعت همگرایی

از لحاظ سرعت همگرایی، روش پیشنهادی به میزان قابل توجهی کندتر از روش filtered-x LMS می باشد. همچنین با مقایسه سرعت همگرایی این دو روش به هنگام پیاده سازی آنها روی کنترلر غیر خطی، ملاحظه می شود که هر چند سرعت همگرایی روش گرادایانی filtered-x بهتر از PSO می باشد، ولی تفاوت فاحشی که در اختلاف سرعت همگرایی آنها به هنگام پیاده سازی این دو روش بر

روی کنترلر خطی داشت، در این حالت وجود ندارد. نتیجتاً روش پیشنهادی از لحاظ سرعت همگرایی، ضعیف تر از روشهای گرادینانی عمل می کند.

• مقاوم بودن

نتایج شبیه سازی نشان داد که روش پیشنهادی نسبت به تغییرات توان منبع نویز ورودی مقاوم تر از روش filtered-x LMS عمل می کند. این نتایج همچنین نشان می دهند که هر دو روش نسبت به تغییرات فرکانس اصلی نویز ورودی مقاوم می باشند.

• پیچیدگی محاسباتی

از لحاظ پیچیدگی محاسباتی، روش پیشنهادی هیچ وابستگی به خطی یا غیر خطی بودن کنترلر سیستم ANC ندارد و تنها به تعداد ضرایب کنترلر بکار گرفته شده وابسته می باشد. همچنین از آنجائی که روش پیشنهادی تخمین مسیر ثانویه را نیاز ندارد، پیچیدگی محاسباتی آن از تعداد ضرایب مدل مسیر ثانویه مستقل می باشد.

با مقایسه های صورت گرفته در فصل پنجم، ملاحظه می شود که برای روش پیشنهادی، پیچیدگی محاسباتی به طور فاحشی کمتر از روشهای گرادینانی می باشد. این مزیت، مخصوصاً در سیستمهای غیر خطی که کاربرد روشهای گرادینانی در آنها به دلیل پیچیدگی محاسباتی با مشکل مواجه است، سودمند واقع می شود. لذا پیاده سازی روش PSO با توجه به سادگی محاسباتی، به صورت زمان واقعی امکان پذیر می باشد.

• سایر موارد

بزرگترین مزیت روش پیشنهادی این است که، با بکارگیری الگوریتم، نیازی به تخمین مسیر ثانویه بین بلندگوی موجود برای تولید نویز ساختگی و میکروفون اندازه گیری خطا وجود ندارد. یک مشکل تخمین زدن مسیر ثانویه این است که اگر اختلاف فاز بین مدل واقعی و مدل تخمین زده شده آن بیش از 90 درجه باشد باعث واگرایی الگوریتم خواهد شد و در صورت استفاده از الگوریتم پیشنهادی دیگر نگران اینکه همواره تخمین دقیق و سریعی از مسیر ثانویه داشته باشیم، نیستیم.

یک مزیت دیگر الگوریتم پیشنهادی که بر پایه الگوریتم هوشمند PSO استوار است، این است که الگوریتم وقتی مورد نظر را می توان بدون کوچکترین تغییری برای هر نوع کنترلی به کار برد، و این در حالی است که روشهای گرادیانی رایج برای هر نوع کنترلی، الگوریتم وقتی خاص آن نوع کنترلر را به کار می گیرند.

مزیت دیگری که میتوان برای روش پیشنهادی ذکر کرد این است که این روش بر خلاف روشهای گرادیانی که امکان به دام افتادن در یک نقطه بهینه محلی را دارند، (مخصوصاً هنگامی که فیلتر بکار گرفته شده به عنوان کنترلر، نامناسب و با درجه کمتر از حد مورد نیاز در نظر گرفته شده باشد)، الگوریتم PSO از نقاط بهینه محلی فرار می کند.

البته الگوریتمهای گرادیانی رایج، مزایایی نسبت به روش پیشنهادی دارند، از جمله اینکه تعداد پارامترهایی تنظیم شونده آنها کمتر از الگوریتم PSO می باشد.

پیشنهادات

همانطور که ملاحظه شد، مهمترین عیب روش پیشنهادی سرعت همگرایی کم آن می باشد. دلیل اصلی آن هم این است که برای جمعیت P تائی از ذرات، ارزش هر ذره از جمعیت، در طول یک پنجره L تائی از نمونه های سیگنال نویز اولیه صورت می گیرد. بنابراین هر تکرار الگوریتم، پس از هر $L \times P$ نمونه

سیگنال نویز اولیه صورت می پذیرد، در حالی که برای الگوریتمهای متداول LMS، با هر نمونه نویز ورودی، یک تکرار الگوریتم انجام می شود. برای رفع این مشکل می توان تعداد اعضای جمعیت P و طول پنجره L را تا حد امکان کوچک انتخاب کرد.

به نظر می رسد که چنانچه با ورود هر نمونه نویز اولیه، الگوریتم PSO یک بار تکرار شود، روش پیشنهادی از سرعت همگرایی بالایی برخوردار گردد و در اینصورت می توان تعداد ذرات جمعیت را افزایش داد که در سیستمهای با خواص غیر خطی شدیدتر امکان به دام افتادن در می نیمم های محلی را کاهش می دهد. همچنین در این صورت می توان از الگوریتمهای ترکیبی با سایر الگوریتمهای هوشمند، مانند الگوریتم ژنتیک استفاده کرد که در این صورت از خصوصیات مفید هر دو الگوریتم برای فرار از نقطه بهینه محلی سود می بریم.

پیشنهاد می گردد برای کاهش پیچیدگی محاسباتی، از الگوریتم بهترین محلی به جای الگوریتم بهترین جهانی استفاده شود. در این صورت، به جای استفاده از رابطه (۳-۳) برای تغییر سرعت ذرات، از رابطه (۲-۳) استفاده می شود که به جای $4MP$ جمع کننده به $2MP$ جمع کننده نیاز دارد.

- [١] Peterson, W.H., and Northwood, T. D., Noise Raised Blood Pressure Without Impairing Auditory Sensitivity, *Science*, vol. 211, pp. 1450-1452, 1981.
- [٢] Donnerstein, E., and Wilson, D. W., Effects of Noise and Perceived Control on Ongoing and Subsequent Aggressive Behavior, *Journal of Personality and Social Psychology*, vol. 34., pp. 774 781, 1976.
- [٣] James F. Abbott , Acoustical Design Criteria For Active Noise-Control Systems , 1993 *IEEE*
- [٤] LJohn C.canfield , Active Disturbance Cancellation In Nonlinear Dynamical Systems Using Neural Networks, *Ph.D. Thesis, University Of New Hampshire, December, 2003*
- [٥] P. Leug, Process of Silencing Sound Oscillations, *U.S. Patent No. 2,043,416, 1936.*
- [٦] S.J.Elliot and P.A. Nelson , Active Noise Control . *IEEE signal processing magazine*, 1993
- [٧] C. Mosquera, J.A. Gomez, F. Perez, M. Sobreira , Adaptive IIR filters for active noise control . sixth international congress on sound and vibration , 5-8 July 1999, Copenhagen, Denmark
- [٨] X. Sun and D.-S. Chen , A new infinite impulse response filter-based adaptive algorithm for active noise. *Journal of Sound and Vibration* (2002)
- [٩] LJohn C.canfield , Active Disturbance Cancellation In Nonlinear Dynamical Systems Using Neural Networks. *Ph.D. Thesis, University Of New Hampshire, December, 2003*
- [١٠] Martin Bouchard, Bruno Paillard, and Chon Tan Le Dinh , Improved Training of Neural Networks for the Nonlinear Active Control of Sound and Vibration. *IEEE TRANSACTIONS ON NEURAL NETWORKS*, VOL. 10, NO. 2, MARCH 1999
- [١١] Qi-Zhi Zhang, Woon-Seng Gan , Active noise control using a simplified fuzzy neural network. *Journal of Sound and Vibration* 272 (2004) 437-449

-
- [١٢] S. Haykin, Adaptive Filter Theory, 2nd Ed., Prentice-Hall, Englewood Cliffs, NJ, 1991.
- [١٣] Kuo, S. M., and Morgan, D. R., Active Noise Control Systems. New York: John Wiley & Sons, Inc., 1996
- [١٤] L. Tan, J. Jiang, Adaptive volterra filters for active control of nonlinear noise processes, *IEEE Transactions on Signal Processing* 49 (2001) 1667–1676.
- [١٥] J. Kennedy, RC Eberhart, Particle Swarm Optimization, *Proceeding of the IEEE International Conference on Neural Networks, Vol. 4, pp 1942-1948, 1995.*
- [١٦] J Kennedy, The Behavior of Particles, in VW Porto, N Saravanan, D Waagen (eds), *Proceeding of the 7th International Conference on Evolutionary Programming, pp 581-589, 1998.*
- [١٧] J Kennedy, The Particle Swarm: Social Adaptation of Knowledge, *Proceeding of 1997 International Conference on Evolutionary Computing, IEEE Press. Pp 303 308, 1997.*
- [١٨] F. van den Bergh, Particle Swarm Weight Initialization in Multi-layer Perceptron Artificial Neural Networks, *Proceeding of the International Conference on Artificial Intelligence, pp 41-45, 1999.*
- [١٩] J. Kennedy, and R. C. Eberhart, Swarm Intelligence, *ISBN 1-55860-595 9, Academic Press (2001).*
- [٢٠] Maurice Clerc, The Swarm and the Queen, Towards a Deterministic and Adaptive Particle Swarm Optimization, Proc. 1999 Congress on Evolutionary computation, Washington DC, pp 1951-1957.
- [٢١] Shi, Y. H., Eberhart, R. C., Parameter Selection in Particle Swarm Optimization, The 7th Annual Conference on Evolutionary Programming, San Diego, USA, 1998.
- [٢٢] Y. Shi, and R. C. Eberhart, _Empirical Study of Particle Swarm Optimization , Proceedings of the 1999 Congress of Evolutionary Computation, vol. 3, IEEE Press, (1999): 1945_1950.

- [٢٣] M. Clerc, J. Kennedy, The Particle Swarm: Explosion, Stability, and Convergence in a Multi Dimensional Complex Space, *IEEE Transactions on Evolutionary Computation*, Vol. 6, pp 58-73, 2002.
- [٢٤] PJ Angeline, Using Selection to Improve Particle Swarm Optimization, *Proceeding of the IEEE International Joint Conference on Neural Networks*, pp 84-89, 1999.
- [٢٥] M Lovbjerg, TK Rasmussen, T Krink, Hybrid Particle Swarm Optimizer with Breeding and Subpopulations, *Proceeding of the Genetic and Evolutionary Computation Conference, San Francisco, 2001*.
- [٢٦] PN Suganthan, Particle Swarm Optimizer with Neighborhood Operator, *Proceeding of the IEEE Congress on Evolutionary Computation, 1999*, pp 1958-1961.
- [٢٧] T. Krink, J. Vesterstrøm, J. Riget., Particle Swarm Optimization with Spatial Particle Extension, *To appear in: Proceedings of the Congress on Evolutionary Computation 2002 (CEC-2002)*.
- [٢٨] J. Riget, J. Vesterstrøm, controlling diversity in particle swarm Optimization, *In preparation, (2002)*.
- [٢٩] Jakob Vesterstrøm and Jacques Riget, Particle Swarms Extensions for improved local, multi-modal, and dynamic search in numerical optimization, Master's Thesis, May 2002
- [٣٠] Vesterstrøm, J. Riget, and T. Krink, Division of Labor in Particle Swarm Optimization, *To appear in: Proceedings of the Congress on Evolutionary Computation 2002 (CEC-2002)*.
- [٣١] R. C. Eberhart, Y. Shi, Tracking and Optimizing Dynamic Systems with Particle Swarms, *Proceedings of congress on evolutionary computation 2001*, Seoul, Korea, IEEE service center, Piscataway, NJ, (2001): 94-97.
- [٣٢] A. Carlisle, G. Dozier, Tracking Changing Extrema with Particle Swarm Optimizer, Auburn University Technical Report CSSE01-08, (2001).
- [٣٣] X. Cui, C. T. Hardin², R. K. Ragade², T. E. Potok and A. S. Elmaghraby, Tracking non-Stationary Optimal Solution by Particle Swarm Optimizer, *Proceedings of the Sixth International Conference on Software Engineering*,

*Artificial Intelligence, Networking and Parallel/Distributed Computing and
First ACIS International Workshop on Self-Assembling Wireless Networks
(2005)*

- [٣٤] Qi-Zhi Zhang, Woon-Seng Gan , A model predictive algorithm for active noise control with online secondary path modeling. *Journal of Sound and Vibration* 270 (2004) 1056–1066
- [٣٥] X.J. Qiu, C.H. Hansen, A modified filtered-x LMS Algorithm for active control of periodic noise with on-line cancellation path modeling. *Journal of Low Frequency Noise, Vibration and Active Control* 19 (2000) 35–46.
- [٣٦] W.K. Tseng, BRafaely, S.J. Elliott, Performance limits and real -time implementation of a virtual microphone active headrest. *Active2002, ISVR, Southampton, UK, 2002*, pp. 1231–1242.
- [٣٧] Y. Maeda, T. Yoshida, An active noise control without estimation of secondary path–ANC using simultaneous perturbation. *Active1999, Florida, USA, 1999*, pp. 985–994.
- [٣٨] Qi-Zhi Zhanga, Woon-Seng Ganb, A model predictive algorithm for active noise control with online secondary path modeling. *Journal of Sound and Vibration* 270 (2004) 1056–1066

Abstract

In This thesis we present a new adaptive algorithm, based on Particle Swarm Optimization (PSO) for Active Noise Cancellation (ANC) problem. We consider both linear and non-linear models for ANC system. The proposed algorithm is used for determining the filter coefficients of FIR filter (in linear model) or weights of neural network (in non linear model). These filters are used as controller, in order to reduce the noise in silence zone.

In contrast to the existing gradient descent-type algorithms, the proposed method does not require estimating the secondary path transfer function. This not only improves stability of it, but also reduces the computational complexity of the method.

We modified PSO algorithm to adopt it to ANC system, when a significant changes for input noise characteristic is occurred. After detecting sudden change for input noise characteristics, we increase the velocity of particles in PSO algorithm as well as reset their global memories to find new optimal coefficients for the ANC filter.

The proposed method is more robust than the existing gradient descent-type algorithms, against significant changes in the input noise power.

Keywords: PSO, ANC, FIR filter, Neural Network, Optimization