

کاهش سطح گلبرگ‌های فرعی در فشردهسازی پالس رادار برای آشکارسازی اهداف همسان با استفاده از شبکه عصبی

علیرضا حکم‌آبادی^۱ آرزو ذاکری^۲ احمد رضا شرافت^۳

۱- استادیار- گروه مهندسی برق - واحد علی آباد کتول - دانشگاه آزاد اسلامی - علی آباد کتول - ایران

hokmabadi@aliabadiau.ac.ir

۲- استادیار- گروه مهندسی برق - واحد علی آباد کتول - دانشگاه آزاد اسلامی - علی آباد کتول - ایران

zakeri@aliabadiau.ac.ir

۳- استاد- دانشکده مهندسی برق و کامپیوتر- دانشگاه تربیت مدرس- تهران- ایران

sharafat@isc.iranet.net

چکیده: در این مقاله از یک شبکه عصبی چند لایه برای فشردهسازی پالس رادار که ورودی آن برخلاف روش‌های پیشین، کد پلی فاز است، استفاده می‌کنیم. مزیت کدهای پلی فاز نسبت به کدهای باینری، داشتن سطح گلبرگ‌های فرعی کمتر و تولرانس داپلر بسیار بهتر است. بعلاوه، برخلاف کدهای باینری، برای هر طول کد، کدهای پلی فاز متنوع وجود دارد که این موضوع باعث سودمندی بیشتر شبکه پیشنهادی می‌شود. برای آموزش شبکه نیز از روش بهینه‌سازی Levenberg-Marquardt استفاده کردہ‌ایم که سرعت همگرایی شبکه را تا حد قابل توجهی افزایش می‌دهد. پس از ۱۰۰ بار آموزش شبکه مورد اشاره، سطح گلبرگ‌های فرعی تا حد خیره‌کننده‌ای کاهش می‌یابد. به عنوان نمونه، در مورد کد P3 ($N=30$) این مقدار برابر $dB_{238/66}$ و در مورد کد P4 ($N=45$) این مقدار برابر $dB_{240/51}$ است. این طرح برای SNR بسیار پایین نیز عملکرد بسیار خوبی دارد و همچنین در مقایسه با سایر روش‌های موجود تولرانس داپلر بسیار بالاتری را در خروجی شبکه ارائه می‌دهد.

کلمات کلیدی: بهینه‌سازی Levenberg-Marquardt، تولرانس داپلر، شبکه عصبی چند لایه، فشردهسازی پالس، گلبرگ‌های فرعی.

تاریخ ارسال مقاله: ۱۳۹۱/۱۰/۱۹

تاریخ پذیرش مشروط مقاله: ۱۳۹۲/۱۱/۱۲

تاریخ پذیرش مقاله: ۱۳۹۵/۸/۱۶

نام نویسنده‌ی مسئول: دکتر علیرضا حکم‌آبادی
نشانی نویسنده‌ی مسئول: گروه مهندسی برق - واحد علی آباد کتول - دانشگاه آزاد اسلامی - علی آباد کتول - ایران



۱- مقدمه

ادامه مقاله به شرح زیر است؛ در بخش دوم، مدل سیستم و ساختار شبکه عصبی پیشنهادی معرفی شده است. نحوه آموزش شبکه عصبی در بخش سوم بطور کامل بیان شده است. پس از آن در بخش چهارم به تحلیل نتایج حاصل از شبیه‌سازی‌ها و بررسی عملکرد شبکه در مقابله با نویز، داپلر و قابلیت آشکارسازی اهداف مجاور هم پرداخته شده است. جمع‌بندی مقاله در بخش پنجم ارائه شده است.

۲- مدل سیستم و ساختار شبکه عصبی

در حالت ایده‌آل، سیگنال خروجی فیلتر فشرده‌ساز باید به شکل تابع ضربه باشد، اما در عمل، گلبرگ‌های فرعی ناخواسته‌ای در خروجی فیلتر منطبق تولید می‌شود. هدف اصلی از به‌کارگیری شبکه عصبی به عنوان فیلتر فشرده‌ساز، شکل‌دهی سیگنال خروجی به صورت مطلوب است. فرض کنید دنباله سیگنال ارسالی به صورت $s^T = [s_1, s_2, \dots, s_N]^T$ و دنباله سیگنال دریافتی به صورت $r^T = [r_1, r_2, \dots, r_N]^T$ باشد که N طول دنباله و علامت (\cdot) معرف عمل گر تراهنده است. در این صورت سیگنال خروجی فیلتر منطبق را می‌توان با رابطه ماتریسی زیر نوشت

$$\mathbf{y} = \mathbf{R} \mathbf{s}^* \quad (1)$$

که $\mathbf{y} = [y_{-N+1}, \dots, y_0, y_1, \dots, y_{N-1}]^T$ بردار سیگنال خروجی فیلتر منطبق و

$$\mathbf{R} = \begin{bmatrix} r_1 & 0 & 0 & \dots & 0 & 0 \\ r_2 & r_1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ r_{N-1} & r_{N-2} & r_{N-3} & \dots & r_1 & 0 \\ r_N & r_{N-1} & r_{N-2} & \dots & r_2 & r_1 \\ 0 & r_N & r_{N-1} & \dots & r_3 & r_2 \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & r_N & r_{N-1} \\ 0 & 0 & 0 & \dots & 0 & r_N \end{bmatrix} \quad (2)$$

ماتریسی متشکل از نسخه‌های شیفت یافته دنباله دریافتی، و $(\cdot)^*$ معرف عمل گر مزدوج مختلط است.

با توجه به روابط (۱) و (۲)، مسئله کاهش سطح گلبرگ‌های فرعی در فشرده‌سازی پالس را می‌توان به صورت یک نگاشت غیرخطی از فضای ورودی‌ها (سطرهای ماتریس \mathbf{R}) به فضای خروجی (تابع ضربه) در نظر گرفت. بدین منظور یک شبکه عصبی چند لایه در نظر گرفته و از روش بهینه‌سازی Levenberg-Marquardt برای آموزش شبکه استفاده کرده‌ایم. برخلاف [۱۳]-[۱۰]، کدهای مورد استفاده در این شبکه کدهای پلی فاز بوده و محدودیتی در نوع و طول آنها نداریم. در واقع مهمترین ویژگی کدهای باینری پیچیدگی بسیار کم آنها است، زیرا برای پیاده‌سازی آنها نیازی به ضرب کننده نیست. در مقابل، سطح گلبرگ‌های فرعی این کدها نسبتاً بزرگ و تولرانس داپلر آنها بسیار کوچک است. برای رفع این نقصه می‌توان از کدهای پلی فاز استفاده

امروزه در سیستم‌های مدرن رادار برای دستیابی به دقت و تفکیک مناسب در برد مورد نظر و همچنین افزایش نسبت سیگنال به نویز (SNR)، بدون افزایش حداکثر توان سیگنال، به طور وسیعی از روش‌های فشرده‌سازی پالس استفاده می‌شود [۱]-[۲]. اما متأسفانه این کار سبب بوجود آمدن گلبرگ‌های فرعی ناخواسته‌ای در توان خروجی برای ناحیه‌هایی که مورد نظر نیستند، خواهد شد. حضور این گلبرگ‌های فرعی نامطلوب است، زیرا در مواردی باعث ایجاد هشدارهای غلط شده و یا ممکن است گلبرگ‌های فرعی هدفی با SNR بزرگ‌تر سبب پوشانده شدن هدف‌های کوچک‌تر مجاور شوند.

برای کاهش سطح گلبرگ‌های فرعی روش‌های مختلفی پیشنهاد شده است که از مهمترین آنها می‌توان به استفاده از یک شبکه وزن‌دهی پس از فیلتر منطبق [۳]-[۴]، طراحی فیلترهای غیرمنطبق [۵]-[۶]، فشرده‌سازی تطبیقی پالس [۷]-[۹] و استفاده از شبکه عصبی [۱۰]-[۱۳] اشاره کرد. استفاده از شبکه عصبی در فشرده‌سازی پالس رادار در مواردی سودمند است که دامنه سیگنال دریافتی از اهداف مختلف تفاوت زیادی با یکدیگر نداشته باشد، در غیر اینصورت ممکن است بعضی از اهداف آشکارسازی نشوند. برای مواجه با چنین حالاتی در [۹] روش فشرده‌سازی تطبیقی پالس بهبود یافته را ارائه کرده‌ایم که سطح گلبرگ‌های فرعی را تا حد قابل توجهی کاهش داده، تولرانس داپلر بالایی داشته و در کنار آن کوچک‌ترین هدفها را نیز آشکارسازی می‌کند. برای مواردی که دامنه سیگنال دریافتی از اهداف مختلف تفاوت زیادی با یکدیگر ندارند، استفاده از شبکه عصبی می‌تواند نتایج بهتری حاصل کند. شبکه عصبی استفاده شده در [۱۰]-[۱۳] برای فشرده‌سازی پالس، یک شبکه چند لایه پس انتشار خطأ است. این روش در مقایسه با روش‌های سنتی عملکرد بسیار خوبی داشته و سطح گلبرگ‌های فرعی را تا حد قابل توجهی کاهش می‌دهد. اما متأسفانه سرعت همگرایی آن بسیار پایین بوده و برای آموزش شبکه به عدد دوره‌های تکرار زیادی نیاز است. علاوه‌به این روش نسبت به اعوجاج حاصل از شیفت فاز داپلر بسیار حساس بوده و تولرانس داپلر خوبی ندارد.

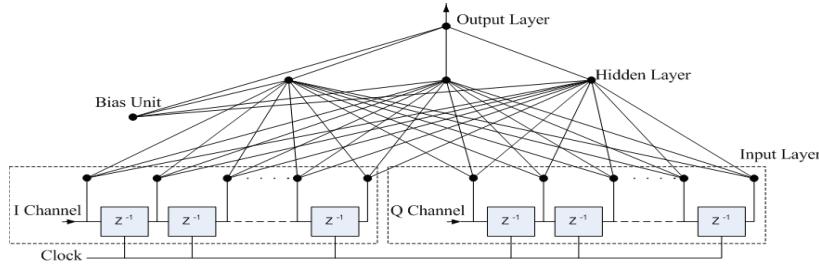
در این مقاله طرح شبکه عصبی [۱۳]-[۱۰] را به گونه‌ای تغییر داده‌ایم که بتوان از کدهای پلی فاز در ورودی شبکه استفاده کرد. بدین ترتیب می‌توان تولرانس داپلر شبکه را تا حد قابل توجهی افزایش داد. بعلاوه، برای آموزش شبکه از روش بهینه‌سازی Levenberg-Marquardt [۱۴] استفاده کرده‌ایم که سرعت همگرایی شبکه را بسیار بالا می‌برد. در این روش برخلاف روش متداول پس انتشار خطأ، بسته به شیب منحنی خطأ طول گام‌ها تغییر کرده و در نتیجه سرعت همگرایی به شدت افزایش خواهد یافت. نتایج حاصل از شبیه‌سازی نیز عملکرد بسیار موفق روش پیشنهادی را تأیید می‌کند.

این کد نسبت به کد P3 در مقابل محدودیت پهنای باند مصون تر است.

شبکه عصبی ارائه شده از سه لایه تشکیل شده که تعداد نورون های لایه ورودی، دو برابر تعداد المان های کد ارسالی ($2N$)، تعداد نورون های لایه میانی، برابر سه و تعداد نورون های لایه خروجی، برابر یک است. بعلاوه، برای بهتر شدن عملکرد شبکه، یک نورون بایاس به هر یک از لایه های میانی و خروجی اضافه کردند.تابع آستانه نیز سیگموئید است.

همانطور که در شکل (۱) نشان داده است، برای به کارگیری کدهای پلی فاز در ورودی شبکه، نیمی از نورون های لایه ورودی برای عبور مؤلفه های هم فاز سیگنال (کانال I) و نیمی دیگر برای عبور مؤلفه های فاز متقابل سیگنال (کانال Q) در نظر گرفته شده است. از طرف دیگر، در سیستم های مدرن رادار، گیرنده به صورت دیجیتالی پیاده سازی شده و کانال های I و Q به طور مستقل از هم محاسبه می شوند. بنابراین نیازی به محاسبه مجدد آنها نیست. تعداد نورون های لایه میانی نیز با توجه به نتیجه شبیه سازی ها، برابر سه انتخاب شده است که بهترین گزینه برای داشتن حداکثر کارآیی و حداقل پیچیدگی است.

الگوهای آموزشی شبکه، سطرهای ماتریس R بعلاوه یک بردار تمام صفر که به معنای نداشتن ورودی است، خواهد بود. خروجی مطلوب تابع ضربه بوده که برای الگوی کد صحیح برابر یک و برای سایر الگوها برابر صفر است. برای آموزش شبکه نیز تعداد ۱۰۰ دوره تکرار در نظر گرفته ایم.



شکل (۱): ساختار شبکه عصبی پیشنهادی

$$e(\mathbf{w}) = E[(f(\mathbf{x}; \mathbf{w}) - f_{opt})^2] \quad (5)$$

که f_{opt} مقدار مطلوب و $E[\cdot]$ بیان گر مقدار متوسط است. با فرض خطی بودن سیستم، تابع خطابه صورت رابطه مرتبه دوم زیر است

$$e(\mathbf{w}) = a + 2\mathbf{b}^T \mathbf{w} + \mathbf{w}^T \mathbf{C} \mathbf{w} \quad (6)$$

که ضریب a بردار \mathbf{b} و ماتریس \mathbf{C} به مقدار متوسط الگوهای ورودی و خروجی وابسته اند. با حداقل کردن خطابه، مقدار وزن ها به صورت زیر بدست می آید

$$\nabla e(\mathbf{w}) = -\mathbf{C}^{-1} \mathbf{b} \quad (7)$$

که ∇ معرف عملگر گرادیان است. مشاهده می شود که در یک گام می توان به سمت نقطه کمینه منحنی خطابه پرش کرد. اما در مورد توابع

نمود. در این کدها به خاطر استفاده از مقادیر مختلف فاز (غیر باینری)، سطح گلبرگ های فرعی کاهش و تولرنس داپلر افزایش می یابد. در این میان، دسته ای از کدهای پلی فاز نظیر کدهای Frank, P1, P2, P3 و P4 وجود دارد که به کدهای پلی فاز Chirlike معروف اند. کدهای P1 و P2 (نسخه های اصلاح شده کد Frank) تنها برای طول های مربع یعنی $N = L^2$ قابل کاربرد بوده و از بسیاری جهات شبیه هم هستند. اما بخلاف آنها کدهای P3 و P4 برای تمامی طول کدها قابل کاربردند. بعلاوه، این کدها از فاز شکل موج FM خطی مشتق شده اند و نسبت به کدهای Frank, P1 و P2 که از فاز شکل موج FM پله ای مشتق شده اند، تولرنس داپلر بیشتری دارند [۱۵، ۱۶]. فاز این کدها با استفاده از روابط زیر بدست می آید

$$P3: \varphi_n = \frac{\pi}{N}(n-1)^2 \quad \text{for } 1 \leq n \leq N \quad (3)$$

$$P4: \varphi_n = \frac{\pi}{N}(n-1)^2 - \pi(n-1) \quad \text{for } 1 \leq n \leq N \quad (4)$$

به طور مفهومی، کد P3 با آوردن سیگنال LFM به باند پایه توسط یک نوسان ساز محلی و نمونه برداری از آن با نرخ نایکوئیست تولید می شود [۱۶]. بیشترین تغییر فاز بین المان های این کد در بخش میانی آن اتفاق می افتد. به همین جهت این کد در مقابل محدودیت پهنای باند آسیب پذیر است. کد P4 نیز همانند کد P3 تولید می شود، اما با این تفاوت که در این کد برای غلبه بر محدودیت پهنای باند، فرکانس نوسان ساز محلی تغییر می کند. با این راه کار بیشترین تغییر فاز بین المان های این کد در کناره های آن صورت می گیرد و در نتیجه

۳- آموزش شبکه عصبی

مهمنترین مشکل شبکه متداول پس انتشار خطابه، سرعت همگرایی بسیار پایین آن است. در این شبکه، با یک گام ثابت به سمت کمینه محلی خطابه حرکت می کنیم، در حالی که استفاده از طول گام های متغیر بسته به شبیه منحنی خطابه، باعث افزایش سرعت همگرایی شبکه خواهد شد. برای حل این مسئله در این مقاله از روش بهینه سازی Levenberg-Marquardt برای آموزش شبکه بهره گرفته ایم.

مدل کلی سیستم را به صورت $f(\mathbf{x}; \mathbf{w})$ در نظر می گیریم. هنگام آموزش شبکه، بردار ورودی \mathbf{x} ثابت و بردار وزن \mathbf{w} متغیر است. تابع خطابه e به صورت زیر تعریف می شود

دوم و بالعکس، اگر λ بزرگ باشد، رابطه اصلاح وزن به سمت روش پس انتشار خطا تغییر وضعیت می‌دهد. مقدار λ نیز بر اساس تغییرات

خطای e به صورت زیر تنظیم می‌شود:

- وزن ها را مطابق رابطه (13) اصلاح کرده و خطای حاصل از بردار

وزن جدید را محاسبه می‌کنیم.

- اگر خطای افزایش یافته، اصلاح وزن ها را نپذیرفته و با تغییر λ به یک عدد بزرگ‌تر (مثلًا 10 برابر) مجددًا اصلاح وزن ها را انجام داده و خطای را محاسبه می‌کنیم. این فرآیند تا زمانی که خطای

بزرگ‌تر از مقدار قبلی شود، تکرار می‌شود.

- اگر خطای کاهش یافته، اصلاح وزن ها مورد تایید بوده و حال با تغییر λ به یک عدد کوچک‌تر (مثلًا 10 برابر) سعی می‌کنیم سرعت همگرایی شبکه را افزایش دهیم. با محاسبه مجدد خطای

مراحل فوق تکرار شده تا نهایتاً به یک جواب پایدار بررسیم.

به طور شهودی، فرآیند اصلاح مقدار λ را می‌توان به صورت زیر بیان کرد: اگر پس از اصلاح وزن، خطای افزایش یافته، واضح است که تقریب مرتبه دوم مناسب نبوده و در مجاورت نقطه کمینه خطای قرار نداریم. بنابراین باید λ را افزایش داد تا تأثیر روش پس انتشار خطا بیشتر شود. بر عکس، اگر خطای کاهش یافته، تقریب مرتبه دوم مناسب بوده و برای آنکه بیشتر به نقطه کمینه خطای نزدیک شویم، لازم است λ را کاهش داده و تأثیر ماتریس Hessian را بیشتر کنیم.

برای افزایش هر چه بیشتر سرعت همگرایی در [۱۴] نشان داده شده که هنگامی که λ بزرگ است و از روش پس انتشار خطا استفاده می‌کنیم، به نحو مؤثرتری می‌توان از ماتریس Hessian تخمین زده شده بهره گرفت. بدین منظور ماتریس یکه در رابطه اصلاح وزن Levenberg Marquardt با ماتریس قطری Hessian جایگزین گردیده و رابطه

(13) به صورت زیر اصلاح می‌شود

$$\mathbf{w}_{i+1} = \mathbf{w}_i - (\mathbf{H} - \lambda \text{diag}[\mathbf{H}])^{-1} \mathbf{d} \quad (14)$$

در عمل، این رابطه اصلاح وزن که به روش بهینه‌سازی Levenberg Marquardt معروف است، برای ساختارهایی از شبکه عصبی با تعداد پارامترهای کم (تعداد وزن‌ها)، بسیار خوب عمل کرده و سرعت همگرایی را به طور قابل ملاحظه‌ای بالا می‌برد.

۴- نتایج شبیه‌سازی

بر اساس مطالعی که در بخش 2 به آنها اشاره شد، از کدهای پلی فار به عنوان ورودی شبکه استفاده می‌کنیم. کدهایی که در این مقاله برای شبیه‌سازی در نظر گرفته‌ایم، کد P3 [۱۶] با طول $N=30$ و کد P4 [۱۶] با طول $N=45$ است. برای آموزش شبکه نیز تنها 100 دوره تکرار در نظر گرفته‌ایم. شکل‌های (2) و (3) ، خروجی شبکه را پس از آموزش نشان می‌دهند. می‌بینیم که با وجود تعداد بسیار کم دوره‌های آموزشی، سطح گلبرگ‌های فرعی تا حد خیره‌کننده‌ای کاهش یافته است. برای نمونه، در مورد کد P3 ($N=30$), سطح بزرگ‌ترین گلبرگ

غیرخطی، تابع خطای بسیار پیچیده‌تر بوده و نمی‌توان مانند بالا عمل کرد. با این وجود، در مجاورت نقطه کمینه منحنی خطای می‌توان مدل سیستم را به صورت خطی تقریب زد و از روش بالا استفاده کرد. در این روش اگر تقریب خوبی زده باشیم، حتی اگر بار اول هم نتوانیم دقیقاً به نقطه کمینه منحنی خطای برسیم، نسبت به روش پس انتشار خطای بسیار سریع‌تر همگرا خواهیم شد.

برای تقریب زدن تابع خطای به شکل مرتبه دوم، تابع جدید $\hat{f}(\mathbf{x}; \mathbf{w})$ را که تقریب خطی $f(\mathbf{x}; \mathbf{w})$ در همسایگی مقدار وزن معین \mathbf{w}_0 است، به صورت زیر تعریف می‌کنیم

$$\hat{f}(\mathbf{x}; \mathbf{w}) = f(\mathbf{x}; \mathbf{w}_0) + (\mathbf{w} - \mathbf{w}_0)^T \nabla f(\mathbf{x}; \mathbf{w}_0) \quad (8)$$

با جایگزینی \hat{f} به جای f در رابطه (5) و به دست آوردن خطای \hat{e} ، به روابط زیر می‌رسیم

$$\nabla \hat{e}(\mathbf{w}) = 2 \mathbf{H}(\mathbf{w} - \mathbf{w}_0) + 2 \mathbf{d} \quad (9)$$

$$\nabla \hat{e}(\mathbf{w}) = 0 \Rightarrow \mathbf{w}_{opt} = -\mathbf{H}^{-1} \mathbf{d} + \mathbf{w}_0 \quad (10)$$

که $\mathbf{d} = E[(f(\mathbf{x}; \mathbf{w}_0) - f_{opt}) \nabla f(\mathbf{x}; \mathbf{w}_0)]$ گرادیان متوسط خطای \hat{e} است. این تقریب، تنها زمانی صحیح است که f خطی Hessian باشد. بنابراین از رابطه (10) تنها در ناحیه‌هایی که برای f یک تقریب خطی قابل قبول وجود دارد، استفاده می‌کنیم.

با توجه به تعریف \mathbf{d} ، رابطه اصلاح وزن در روش پس انتشار خطای را می‌توان به صورت زیر نوشت

$$\mathbf{w}_{i+1} = \mathbf{w}_i - \mu \mathbf{d} \quad (11)$$

که μ عددی ثابت بین صفر و یک است. از طرف دیگر، رابطه اصلاح وزن بدست آمده از تقریب مرتبه دوم به صورت زیر است

$$\mathbf{w}_{i+1} = \mathbf{w}_i - \mathbf{H}^{-1} \mathbf{d} \quad (12)$$

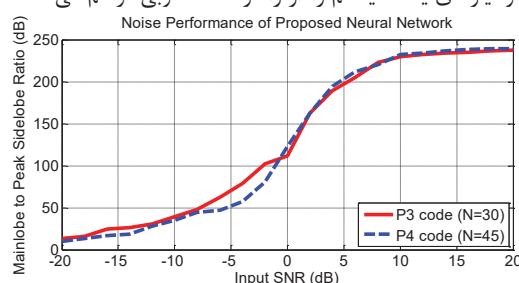
مسلمًا تقریب مرتبه دوم در همه نقاط بهتر عمل نمی‌کند، چون در بدست آوردن آن برای f تقریب خطی در نظر گرفتیم که این تقریب تنها در همسایگی نقطه کمینه خطای قابل قبول است. برای حل این مسئله در [۱۴] ترکیبی از این دو حالت کرانه‌ای (روابط (11) و (12)) در نظر گرفته شده است. در این الگوریتم، ابتدا از روش پس انتشار خطای استفاده کرده تا به نقطه کمینه خطای نزدیک شویم. سپس به تدریج رابطه اصلاح وزن را به سوی تقریب مرتبه دوم تغییر وضعیت داده تا سریع‌تر به نقطه کمینه خطای برسیم. بدین ترتیب با تغییر میزان خطای در هر مرحله می‌توان میزان نزدیکی به کمینه خطای حدس زد. در این روش رابطه اصلاح وزن به صورت زیر است

$$\mathbf{w}_{i+1} = \mathbf{w}_i - (\mathbf{H} - \lambda \mathbf{I})^{-1} \mathbf{d} \quad (13)$$

که \mathbf{I} ماتریس یکه و λ ضریبی متغیر است و میزان تأثیر دو حالت پس انتشار خطای و تقریب مرتبه دوم را تعیین می‌کند. با توجه به این رابطه، اگر λ کوچک باشد، رابطه اصلاح وزن به سمت تقریب مرتبه



برای بررسی عملکرد شبکه عصبی در مقابل نویز، SNR سیگنال ورودی را از -20 dB تا 20 dB تغییر داده و نسبت گلبرگ اصلی به بزرگترین گلبرگ فرعی (MPSR) سیگنال خروجی را محاسبه می‌کنیم. در شکل (۴)، منحنی‌های حاصل از تغییرات MPSR در مقابل SNR ورودی در مورد این دو کد رسم شده است. می‌بینیم که طرح پیشنهادی برخلاف سایر روش‌ها در SNR پایین نیز کارآیی فوق العاده‌ای دارد، به طوری که در $\text{SNR} = -13$ dB مقدار MPSR برای هر دو کد تقریباً برابر 30 است که این مقدار برای غالب کاربردهای رادار کفایت می‌کند.علاوه می‌بینیم که با افزایش SNR، عملکرد شبکه به نحو چشمگیری افزایش می‌یابد، به عنوان مثال در SNRهای -8 ، 0 و 15 dB مقدار MPSR برای هر دو کد به ترتیب بزرگ‌تر از 50 ، 100 و 200 است. بنابراین می‌توان چنین نتیجه گرفت که شبکه ارائه شده علاوه بر SNRهای بزرگ، در SNRهای پایین نیز عملکرد بسیار مؤثر داشته و نیازهای یک سیستم رادار را در حد مطلوبی فراهم می‌کند.



شکل (۴): عملکرد شبکه در مقابل نویز

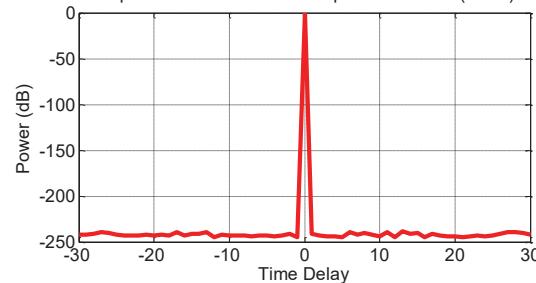
۴- عملکرد شبکه در مقابل اثر داپلر

یکی از چالش‌های مهمی که در طراحی فیلتر فشرده‌ساز مطرح است، رفتار سیستم در مواجهه با اهداف متحرک است. در چنین سناریوهایی وجود سرعت نسبی معمولاً بزرگ بین هدف و رادار، سبب به وجود آمدن اثر داپلر و تغییر فرکانس سیگنال برگشتی خواهد شد [۱۷]. در این بخش اثر داپلر را در شبیه‌سازی لحاظ کرده و رفتار شبکه عصبی پیشنهادی را در مواجهه با آن بررسی کرده‌ایم. باید توجه داشت که در آموزش شبکه تنها از سطرهای ماتریس \mathbf{R} استفاده شده و هیچ‌گونه تغییری ناشی از نویز و داپلر در آموزش شبکه استفاده نشده است. نتایج حاصله به صورتتابع ابهام نرمالیزه که یک تابع دوبعدی وابسته به زمان و فرکانس داپلر (x, f) است، بوده و بیان‌گر تغییرات خروجی فیلتر فشرده‌ساز به ازاء مقادیر مختلف تاخیر زمانی و داپلر است. شکل‌های (۵) و (۶) به ترتیب مربوط به تابع ابهام نرمالیزه کدهای P3 ($N=30$) و P4 ($N=45$) می‌باشند. می‌بینیم که با وجود طولانی بودن کدها، شبکه پیشنهادی در برخورد با پیدیده داپلر بسیار موفق

فرعی (PSL^*) برابر $\text{dB} -238/66$ و در مورد کد P4 ($N=45$), برابر $\text{dB} -240/51$ است.

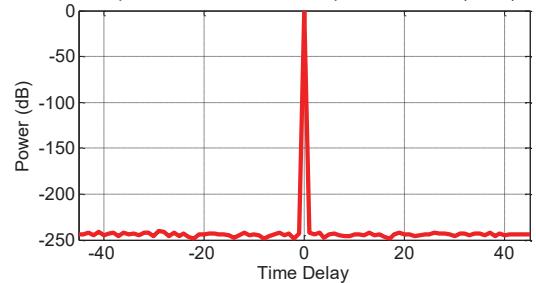
در شبکه پیشنهادی، تنها مجاز به استفاده از کدهای پلی‌فاز هستیم، در حالی که در ساختار پیشنهاد شده [۱۰]-[۱۲] تنها کدهای باینری قابل کاربرد است. بنابراین نمی‌توان عملکرد این روش‌ها را در مورد کدهای مشابه با یکدیگر مقایسه نمود. با این وجود، برای آنکه بتوانیم عملکرد بسیار موفق الگوریتم آموزشی Levenberg-Marquardt را نشان دهیم، همان ساختار شبکه عصبی [۱۳]-[۱۰] را در نظر گرفته و از الگوریتم Levenberg-Marquardt برای اصلاح وزن‌ها استفاده می‌کنیم. در این حالت تنها پس از گذشت 100 دوره تکرار، مقدار PSL خروجی شبکه عصبی در مورد کد Barker ($N=13$) به 500 - $236/16$ رسید. این در حالی است که در روش [۱۰] پس از 500 دوره تکرار، مقدار PSL برابر با $42/73$ dB خواهد بود. در مورد کد Levenberg- m -Sequence ($N=31$) نیز با استفاده از الگوریتم- Marquardt، تنها پس از گذشت 100 دوره تکرار مقدار PSL برابر $224/87$ dB خواهد بود، در حالی که در [۱۳] پس از گذشت 10^4 دوره تکرار، مقدار PSL از 62 dB تجاوز نخواهد کرد. بنابراین مشاهده می‌شود که استفاده از روش بهینه‌سازی Levenberg- Marquardt مشکل کند شدن سرعت همگرایی شبکه عصبی را به نحو شایسته‌ای حل کرده است.

Proposed Neural Network Output of P3 Code ($N=30$)

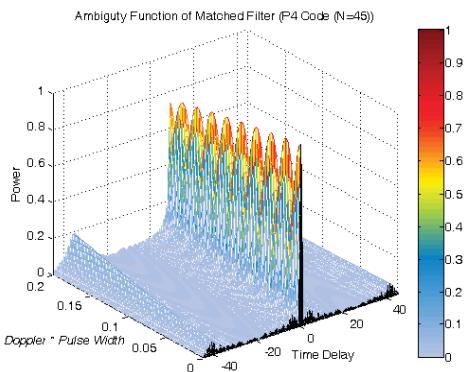


شکل (۲): شکل موج سیگنال خروجی شبکه با استفاده از کد P3 ($N=30$)

Proposed Neural Network Output of P4 Code ($N=45$)



شکل (۳): شکل موج سیگنال خروجی شبکه با استفاده از کد P4 ($N=45$)

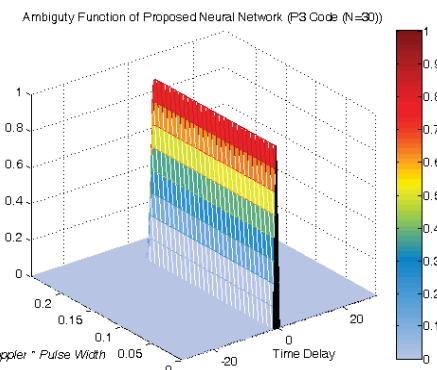


شکل (۸): تابع ابهام نرمالیزه فیلتر منطبق در مورد کد P4 (N=۴۵)

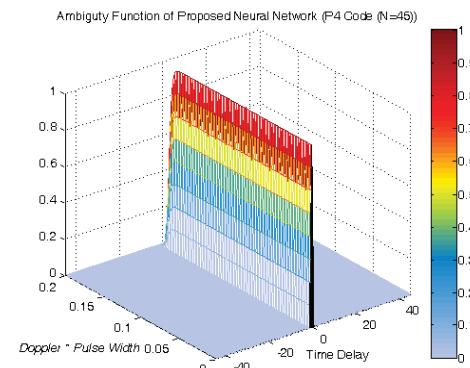
برای مقایسه دقیق‌تر، در شکل‌های (۹) و (۱۰) منحنی افت اندازه گلبرگ اصلی این کدها در مقابل تغییرات داپلر (به صورت نرمالیزه شده) رسم شده است. می‌بینیم که افت اندازه گلبرگ اصلی ناشی از شبکه پیشنهادی بسیار ناچیز بوده و حتی برای داپلرهای بزرگ نیز این مقدار تغییر چندانی نخواهد کرد، در حالی که افت اندازه گلبرگ اصلی سیگنال خروجی فیلتر منطبق با افزودن داپلر با شیب ملایمی کاهش می‌یابد. البته همان طور که در شکل‌ها نیز مشهود است، چنانچه مقدار فرکانس داپلر بیش از حد بزرگ شود شبکه پیشنهادی دیگر قادر به شناسایی هدف نخواهد بود. اما این مسئله نگران کننده نیست؛ چراکه در سناریوهای واقعی با چنین سرعت‌هایی روبرو نخواهیم شد. به عنوان نمونه با توجه به شکل‌های (۹) و (۱۰) عملکرد شبکه پیشنهادی به ازاء مقادیر حاصل ضرب داپلر در عرض پالس (Doppler * Pulse Width) بزرگ‌تر از $0.02 \mu\text{s}$ مختل می‌شود؛ این مقدار داپلر در یک سیستم راداری معمول با مشخصات $f_0 = 9 \text{ GHz}$ و $\tau = 2 \mu\text{s}$ (فرکانس مرکزی) و (عرض پالس) معادل با سرعت 1600 m/s بود که سرعت بسیار بزرگی است.

در شکل‌های (۱۱) و (۱۲) نیز منحنی مقدار PSL این کدها در مقابل تغییرات داپلر (به صورت نرمالیزه شده) رسم شده است. می‌بینیم که برخلاف فیلتر منطبق، حتی برای داپلرهای بسیار بزرگ نیز مقدار PSL کماکان نزدیک صفر (-200 dB) باقی می‌ماند. به عبارت دیگر ساختار پیشنهادی با استفاده از کدهای پلی فاز در ورودی شبکه عصبی بسیار کارآمد بوده و تولرانس داپلر بالایی دارد. در شبیه‌سازی‌های انجام شده با استفاده از دیگر کدهای پلی فاز و با طول‌های مختلف نیز نتایج مشابهی بدست آمده است که صحت عملکرد ساختار پیشنهادی را تأیید می‌کنند.

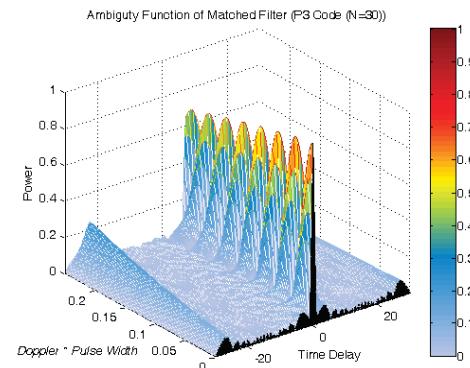
عمل کرده و تولرانس داپلر بسیار خوبی دارد. همچنین برای مقایسه، در شکل‌های (۷) و (۸) تابع ابهام نرمالیزه شده فیلتر منطبق نیز رسم شده است. می‌بینیم که تابع ابهام حاصل از شبکه پیشنهادی، تولرانس داپلر و تزویج برد داپلر مشابه با تابع ابهام حاصل از فیلتر منطبق داشته، درحالی که سطح گلبرگ‌های فرعی تقریباً در تمام سطح تابع ابهام نزدیک صفر است.



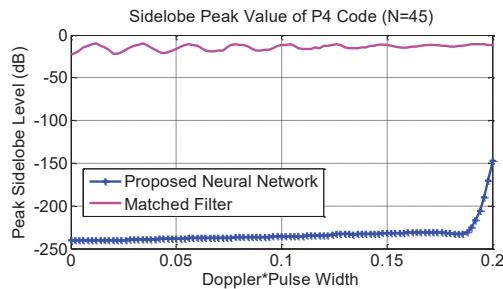
شکل (۵): تابع ابهام نرمالیزه شبکه عصبی در مورد کد P3 (N=۳۰)



شکل (۶): تابع ابهام نرمالیزه شبکه عصبی در مورد کد P4 (N=۴۵)



شکل (۷): تابع ابهام نرمالیزه فیلتر منطبق در مورد کد P3 (N=۳۰)

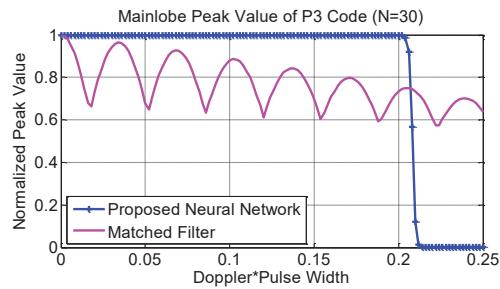


شکل (۱۲): منحنی مقدار PSL کد P4 (N=۴۵) در مقابل تغییرات داپلر (به صورت نرمالیزه شده)

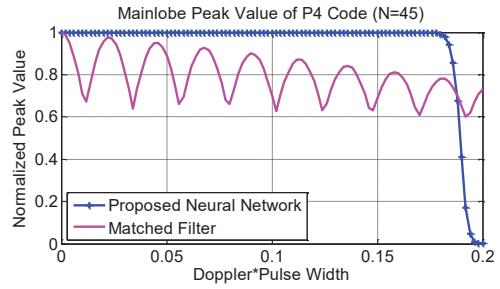
۳-۴- قابلیت آشکارسازی اهداف مجاور هم و تولرانس قابل قبول دامنه برای سیگنال‌های دریافتی

در این قسمت برای شبیه‌سازی سیگنال حاوی دو موج معنکس شده از اهداف مستقل که با یکدیگر همپوشانی دارند، دو کد مشابه با طول N را نسبت به یکدیگر L بیت تأخیر داده ($L \in N \wedge L < N$) و با نسبت دامنه‌های متفاوت با یکدیگر جمع می‌کنیم. سپس از این سیگنال به عنوان ورودی شبکه عصبی استفاده خواهیم کرد. بدین ترتیب می‌توان با تغییر مقدار تأخیر و نسبت دامنه‌ها، قابلیت تفکیک و تولرانس قابل قبول دامنه برای سیگنال‌های دریافتی را بررسی کرد.

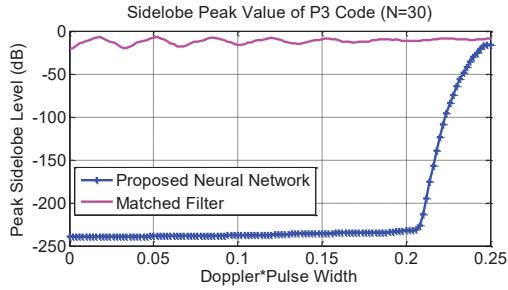
نتایج بدست آمده برای حالت‌های مختلف تأخیر و نسبت دامنه در مورد کد P3 (N=۳۰) و P4 (N=۴۵) در جدول (۱) آورده شده است. در این جدول نسبت دامنه ورودی‌ها به صورت نسبت دامنه کد اول به دامنه کد تأخیر یافته تعریف می‌شود. نسبت دامنه‌های خروجی نیز به صورت نسبت اندازه گلبرگ‌های اصلی در سیگنال خروجی شبکه تعریف می‌شود.



شکل (۹): منحنی افت اندازه گلبرگ اصلی کد P3 (N=۳۰) در مقابل تغییرات داپلر (به صورت نرمالیزه شده)



شکل (۱۰): منحنی افت اندازه گلبرگ اصلی کد P4 (N=۴۵) در مقابل تغییرات داپلر (به صورت نرمالیزه شده)



شکل (۱۱): منحنی مقدار PSL کد P3 (N=۳۰) در مقابل تغییرات داپلر (به صورت نرمالیزه شده)

جدول (۱): نتایج بدست آمده از شبیه‌سازی برای حالت‌های مختلف تأخیر و نسبت دامنه.

مقدار تأخیر	نسبت دامنه‌های خروجی ورودی (N=۴۵) P4	PSMR [dB]	
		(N=۳۰) P3	(N=۴۵) P4
۲ بیت تأخیر	۱	-۲/۳×10 ^{-۱۰}	-۲۷۷/۲۶
	۲	-۳۶/۶۰	-۲۰۲/۶۰
	۳	-۱۳۵/۸۴	-۱۰۴/۸۳
	۴	-۱۷۷/۷۴	-۶۲/۱۵
	۵	-۲۰۱/۱۰۶	-۳۸/۹۶
۳ بیت تأخیر	۱	-۱/۹×10 ^{-۱۴}	-۲۳۲/۳۹
	۲	-۲/۸×10 ^{-۸}	-۲۳۹/۱۰
	۳	-۱۵/۸۹	-۲۲۳/۷۷
	۴	-۱۲۸/۱۵	-۱۱۱/۷۷
	۵	-۱۷۶/۳۸	-۶۳/۶۷
۴ بیت تأخیر	۱	-۴/۴×10 ^{-۱۱}	-۲۳۳/۶۸
	۲	-۴۱/۴۴	-۱۹۸/۹۶
	۳	-۱۷۶/۸۱	-۶۳/۸۸
	۴	-۲۱۷/۷۹	-۲۳/۱۰
	۵	-۲۳۱/۳۱	-۹/۶۷

- Codes," Journal of Advanced Electrical and Computer Engineering, no. 1 pp. 22-31, 2014.
- [4] Nunn, Carroll J., and Gregory E., "Polyphase pulse compression codes with optimal peak and integrated sidelobes," IEEE Transactions on Aerospace and Electronic Systems, vol. 45, no. 2, 2009.
- [5] Kulpa, Janusz, S. "Mismatched filter for range sidelobes suppression of pseudo-noise signals," In IEEE Signal Processing Symposium (SPSympo), pp. 1-4, 2015.
- [6] Rabaste, Olivier, and Savy, L., "Mismatched filter optimization for radar applications using quadratically constrained quadratic programs," IEEE Transactions on Aerospace and Electronic Systems, vol. 51, no. 4, pp. 3107-3122, 2015.
- [7] S. D. Blunt, and K. Gerlach, "Adaptive pulse compression via MMSE estimation," IEEE Trans. Aero. and Electron. Sys., vol. 42, no. 2, Apr. 2006.
- [8] Henke, Dakota, McCormick, P., Shannon D., and Higgins, D., "Practical aspects of optimal mismatch filtering and adaptive pulse compression for FM waveforms," In IEEE Radar Conference (RadarCon), pp. 1149-1155, 2015.
- [٩] علیرضا حکم‌آبادی، احمد رضا شرافت، و آرزو ذاکری، "بهبود فشرده‌سازی تطبیقی پالس با در نظر گرفتن اثر داپلر در سامانه رادار،" هفدهمین کنفرانس مهندسی برق ایران، دانشگاه علم و صنعت ایران، اردیبهشت ۱۳۸۸.
- [10] Padaki, Aditya, V., and George, K., "Improving performance in neural network based pulse compression for binary and polyphase codes," 12th International IEEE Conference on Computer Modelling and Simulation (UKSim), pp. 278-283, 2010.
- [11] Baghel, V., and Panda, G., "Development of an efficient hybrid model for range sidelobe suppression in pulse compression radar." Aerospace Science and Technology, vol. 27, no. 1, pp. 156-162, 2013.
- [12] Duh, F. B., Juang, C. F., Lin, C. T., "A neural fuzzy network approach to radar pulse compression," IEEE Geosci. Remote Sens. Lett., vol. 1 no. 1, pp. 15–20, 2004.
- [13] Khairnar, D. G., Merchant, S. N., and Desai, U. B., "Radial basis function neural network for pulse radar detection," IET Radar Sonar Navig. vol. 1, no. 1, pp. 8–17, 2007.
- [14] L. S. H. Ngia, and J. Sjoberg, "Efficient training of neural nets for nonlinear adaptive filtering using a recursive Levenberg-Marquardt algorithm," IEEE Trans. Sig. Proc., vol. 48, pp. 1915-1927, Jul. 2000.
- [15] Jamil, Momin, Zepernick, H., and Pettersson, M. I., "Performance assessment of polyphase pulse compression codes," IEEE 10th International Symposium on Spread Spectrum Techniques and Applications, pp. 166-172, 2008.
- [16] Blunt, Shannon D., Cook, M., Jakabosky, J., Graaf, J. D., and Perrins, E., "Polyphase-coded FM (PCFM) radar waveforms, part I: implementation," IEEE Transactions on Aerospace and Electronic Systems vol. 50, no. 3, pp. 2218-2229, 2014.
- [١٧] محمد علائی، و حمیدرضا امین‌داور، "روشی جدید در بازناسایی خودکار اهداف متحرک زمینی با استفاده از رادارهای مراقبت زمینی پالس داپلر"، مجله انجمن مهندسین برق و الکترونیک ایران- شماره دوم، پائیز و زمستان ۱۳۸۸

مقدار PSMR₅ را نیز به صورت نسبت اندازه بزرگ‌ترین گلبرگ فرعی به اندازه گلبرگ اصلی هدف کوچک‌تر تعریف کرده‌ایم. با توجه به مقادیر جدول (۱) می‌بینیم که برای نسبت دامنه ورودی ۵ و تأخیرهای مختلف، به سختی می‌توان سیگنال کوچک‌تر را تشخیص داد. با شبیه‌سازی حالت‌های دیگر نیز می‌توان به نتایج مشابهی رسید. بنابراین بطور کلی می‌توان چنین نتیجه گرفت که این شبکه عصبی همانند [۱۰]-[۱۳]، تنها قادر به تشخیص سیگنال‌های نزدیک هم با نسبت دامنه کمتر از ۵ است.

۵- نتیجه‌گیری:

در این مقاله از یک شبکه عصبی چند لایه برای فشرده‌سازی پالس رادار استفاده کردیم که ورودی آن برخلاف روش پیشین، کد پلی فاز است. علت استفاده از کدهای پلی فاز به جای کدهای باینری در ورودی شبکه، بالا بودن تولرانس داپلر آنها است. بعلاوه، برخلاف کدهای باینری، برای هر طول کد، کدهای پلی فاز متنوعی وجود دارد که این مسئله باعث سودمندی بیشتر شبکه پیشنهادی می‌شود. به منظور افزایش سرعت همگرایی شبکه نیز از روش بهینه‌سازی Levenberg-Marquardt برای آموزش شبکه استفاده کردیم. در این روش برخلاف الگوریتم متداول پس انتشار خطأ که با یک گام ثابت به سمت کمینه محلی حرکت می‌کنیم، بسته به شبکه منحنی خطأ طول گام‌ها را تغییر داده و در نتیجه سرعت همگرایی به شدت افزایش می‌یابد. استفاده از روش پیشنهادی نتایج بسیار خوبی را حاصل می‌کند. تنها پس از ۱۰۰ بار آموزش، سطح گلبرگ‌های فرعی تقریباً به صفر رسیده است. این روش در SNR بسیار پایین نیز کارآیی قابل توجهی از خود نشان می‌دهد. بعلاوه، پس از بررسی اثر داپلر بر روی عملکرد شبکه و استخراج تابع ابهام آن، می‌بینیم که این طرح همانند فیلتر منطبق، تولرانس داپلر بالایی دارد.

به منظور بررسی عملکرد شبکه عصبی در آشکارسازی اهداف مجاور هم با نسبت دامنه‌های مختلف، شبیه‌سازی‌هایی انجام شد که مشخص گردید که این شبکه عصبی همانند [۱۰]-[۱۳]، تنها قادر به تشخیص سیگنال‌های نزدیک هم با نسبت دامنه کمتر از ۵ است. بنابراین شاید بتوان تنها نقطه ضعف این روش را زیاد بالا نبودن تولرانس دامنه برای سیگنال‌های دریافتی دانست. البته در غالب موارد عملی چنین مشکلی پیش نخواهد آمد و شبکه عصبی ارائه شده بسیار موفق عمل خواهد کرد.

مراجع

- M. Phillips, Roger, Bell, G., Storz, G., and Lilburn, L. "Frequency Correction for Pulse Compression Radar," U.S. Patent Application 15/004,752, filed January 22, 2016.
- Richards, M.A., Fundamentals of Radar Signal Processing, 2ed., McGraw-Hill Education, 2013.
- Ranganath, Praveen, and Rao, S., "Effect of Pulse Shaping on Autocorrelation Function of Barker and Frank Phase



[۱۸] معین احمدی، و کمال محمدپور، " تشخیص مدولاسیون فاصله تکرار پالس های راداری" ، مجله انجمن مهندسین برق و الکترونیک ایران، سال ششم، شماره دوم، پائیز و زمستان ۱۳۸۸.

زیرنویس‌ها

^۱ Frequency Modulation

^۲ Linear Frequency Modulation

^۳ Peak Sidelobe Level

^۴ Mainlobe to Peak Sidelobe Ratio

^۵ Peak Sidelobe to Mainlobe Ratio

