

مدل سازی رفتار گذرای وابسته به زمان مدار حلقه قفل فاز دیجیتالی به کمک شبکه‌ی عصبی واحد بازگشتی گیتی

سیده فاطمه موسوی قوام‌آبادی، سید علیرضا صدرالسادات و علی مفتخرزاده

چکیده: امروزه شبکه‌های عصبی یک ابزار قدرتمند برای مدل‌سازی مدارهای پیچیده و غیرخطی می‌باشند. در این مقاله، مدل‌سازی رفتار گذرای وابسته به زمان مدار حلقه قفل فاز دیجیتالی به کمک دو مدل شبکه‌ی عصبی که وابسته به زمان هستند، انجام شده است. مدل‌سازی با استفاده از شبکه عصبی RNN با چالش محو گرادیان در مرحله آموزش مواجه می‌باشد و از نظر سرعت و دقت در مدل‌سازی عملکرد مناسبی ندارد. برای دستیابی به مقادیر مطلوب خطای آموزش و آزمون، از شبکه عصبی GRU برای مدل‌سازی استفاده می‌شود. این شبکه به دلیل وجود دو گیت به‌روزرسانی و بازنشانی، قادر می‌باشد مشکل محو شدن گرادیان را برطرف کرده و مدل‌سازی قابل قبولی ارائه دهد. مقایسه نتایج این دو شبکه نشان می‌دهد که روش مبتنی بر ساختار شبکه عصبی گیتی، توانایی و قابلیت بهتری در مدل‌سازی رفتار مدارهای غیرخطی دارد. در پایان، جهت ارزیابی منصفانه عملکرد مدل‌ها، مقایسه‌ای نیز با ساختار LSTM انجام شده که نتایج آن نشان می‌دهد شبکه GRU از نظر دقت مدل‌سازی و سرعت آموزش عملکرد بهتری نسبت به LSTM نیز دارد.

کلیدواژه: مدار DPLL، مدل‌سازی رفتار دینامیکی مدارات غیرخطی، محو گرادیان، شبکه‌های عصبی GRU.

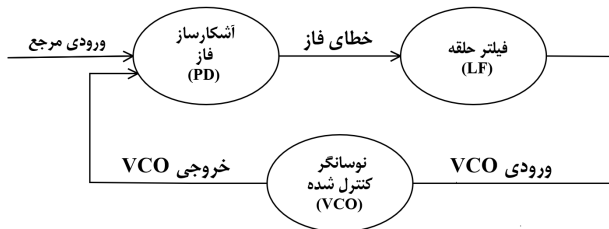
۱- مقدمه

با توجه به پیشرفت روز افزون صنعت نیمه‌هادی و پیچیده شدن مدارات الکترونیکی، امروزه در مراحل طراحی و ساخت قطعات نیمه‌هادی استفاده از رایانه‌ها اجتناب‌ناپذیر است. به استفاده از فناوری رایانه در طراحی مبتنی بر رایانه (CAD) اطلاق می‌شود و در زمینه‌های مختلف مانند صنعت تولید قطعات و افزاره‌های الکترونیکی^۱ استفاده می‌شود. گزارش‌ها نشان می‌دهند که رویکرد CAD مبتنی بر هوش مصنوعی^۲ اولین بار برای مدل‌سازی، شبیه‌سازی و بهینه‌سازی مدارهای میکروویو معرفی شد و به سایر مدارهای الکترونیکی خصوصاً مدارهای غیرخطی تعمیم داده شد [۱]. مدل‌های CAD مبتنی بر شبکه‌های عصبی مصنوعی

این مقاله در تاریخ ۳۱ شهریور ماه ۱۴۰۳ دریافت و در تاریخ ۶ مرداد ماه ۱۴۰۴ بازنگری شد.
سیده فاطمه موسوی قوام‌آبادی، دانشکده مهندسی برق، دانشگاه یزد، یزد، ایران، (email: f.mousavi@stu.yazd.ac.ir)
سید علیرضا صدرالسادات، دانشکده مهندسی کامپیوتر، دانشگاه یزد، یزد، ایران، (email: alireza.sadr@yazd.ac.ir)
علی مفتخرزاده (نویسنده مسئول)، دانشکده مهندسی برق، دانشگاه یزد، یزد، ایران، (email: moftakharzadeh@yazd.ac.ir)

1. Computer-Aided Design
2. Electronic Devices
3. Artificial Intelligence

4. Artificial Neural Network
5. Digital Phase-Locked Loops



شکل ۱: بلوک دیاگرام ساختار اصلی PLL.

ردیابی کند و با ترکیب یک تقسیم کننده فرکانس، یک PLL می‌تواند فرکانس پایداری که مضربی از فرکانس ورودی است تولید کند [۱۷] تا [۱۹]. همان‌طور که ساختار پایه مدار در بلوک دیاگرام شکل ۱ نشان داده شده است، PLL از یک آشکارساز فاز (PD)^۷، یک فیلتر حلقه (LF)^۸ و یک نوسان‌ساز کنترل‌شونده با ولتاژ (VCO)^۹ تشکیل شده است. در شکل ۲-الف شماتیک مداری بلوک دیاگرام شکل ۱ نشان داده شده است. در شکل ۲-ب، مدار داخلی آشکارساز فاز نشان داده شده است. این آشکارساز، فاز سیگنال ورودی را با فاز سیگنال خروجی VCO مقایسه می‌کند و سیگنال خطا تولید می‌کند. مدار شکل ۲-ج یعنی، فیلتر حلقه سیگنال خطا را فیلتر می‌کند و مدار شکل ۲-د، VCO را کنترل کرده که سیگنال خروجی تولید شود [۲۰].

قفل سریع و مصرف کم انرژی مهم‌ترین پارامتر در DPLL است [۲۱]. مدار DPLL مشابه PLL از سه جزء اصلی تشکیل شده است. عملکرد مدار DPLL به این صورت است که در ورودی آشکارساز فاز XOR، فاز دو سیگنال ورودی مرجع و خروجی VCO با یکدیگر مقایسه شده و بر مبنای اختلاف فاز این سیگنال یک سیگنال خطای فاز PWM تولید می‌شود که با عبور این سیگنال از فیلتر پایین‌گذر فعال یک سیگنال DC به عنوان ورودی کنترلی VCO ایجاد می‌شود که در حقیقت متناسب با اختلاف فاز دو سیگنال ابتدایی است. در نهایت با تغییر سیگنال کنترلی ورودی VCO، فرکانس سیگنال خروجی VCO به گونه‌ای تغییر خواهد کرد تا اختلاف فاز صفر شود و فرکانس سیگنال VCO با ورودی مرجع برابر شود.

با توجه به شکل ۲، مدار حلقه قفل فاز دیجیتالی از سه بلوک تشکیل شده است و مدار داخلی هر بلوک نشان داده شده است. شبیه‌سازی رفتار دینامیکی مدارات الکترونیکی غیرخطی با شبکه‌های عصبی نسبت به مدارهای خطی به مراتب سخت‌تر است، در این راستا در این پژوهش تلاش شده است تا مدل کارآمدتری را برای یک مدار الکترونیکی پرکاربرد ارائه شود.

۳- شبکه‌های عصبی مصنوعی

شبکه‌های عصبی مصنوعی، یک سیستم پردازش اطلاعات است که از روش‌های یادگیری ماشین^{۱۰} الهام گرفته است و ایده اصلی آن از شبکه‌های عصبی بیولوژیکی گرفته شده است. این شبکه‌ها از عملکردی شبیه به مغز انسان برای پردازش اطلاعات استفاده می‌کنند. طبق قضیه تقریب عمومی^{۱۱}، شبکه‌های عصبی پیشرو^{۱۲} با یک لایه پنهان قادرند هر

بر شبکه عصبی، به ویژه شبکه‌های عصبی بازگشتی^۱ (RNN)، در این زمینه عملکرد خوبی دارند. این شبکه‌ها می‌توانند با استفاده از داده‌های ورودی-خروجی آموزش ببینند و نتایج گام‌های زمانی قبلی را در نظر بگیرند که به افزایش کارایی آن‌ها کمک می‌کند. همچنین قضیه تقریب جهانی برای این ساختار بیان می‌کند که RNN می‌تواند هر رابطه غیرخطی بین ورودی‌ها و خروجی‌ها را مدل‌سازی کند [۱۳].

علی‌رغم قابلیت بالای این نوع شبکه‌های عصبی بازگشتی در یادگیری دینامیک مدارهای غیرخطی، پس از گذشت زمان سیگنال خطای انتشاری آن در فاز آموزش شبکه عصبی، دچار امحای^۲ می‌شود. امحای گرادیان با افزایش تعداد لایه‌ها که باعث عمیق شدن در طول لایه‌ها می‌باشد، به وجود می‌آید. محو شدن گرادیان باعث کاهش سرعت به‌روزرسانی وزن‌ها یا متوقف شدن فرآیند آموزش می‌شود. به همین دلیل، RNN‌های ساده که ظرفیت آن‌ها برای مدل‌سازی مدارهای غیرخطی محدود است، مورد استفاده قرار می‌گیرند [۱۴].

شبکه عصبی واحد بازگشتی گیتی^۳ (GRU)، یک نوع شبکه عصبی بازگشتی است که ساختار آن بسیار مشابه شبکه حافظه طولانی کوتاه مدت^۴ (LSTM) می‌باشد و به گونه‌ای طراحی شده است که مشکل امحای را حل می‌کند و به منظور دستیابی به دقت بهتر و سرعت بیشتر برای زمان آموزش و آزمون استفاده می‌شود. شبکه LSTM با گیت‌های بیشتر و ساختار پیچیده‌تر دارای پارامترهای بیشتری است که می‌تواند منجر به دقت بالاتر در موارد خاص شود، اما خطر بیش‌برازش^۵ و زمان آموزش طولانی‌تر را نیز به همراه دارد [۱۵]. مدل‌های RNN و GRU به دلیل ساختار نسبتاً ساده و توانایی آن‌ها در مدل‌سازی وابستگی‌های زمانی و رفتار گذرای داده‌ها برای مسئله مدل‌سازی مدار حلقه قفل فاز دیجیتالی انتخاب شدند.

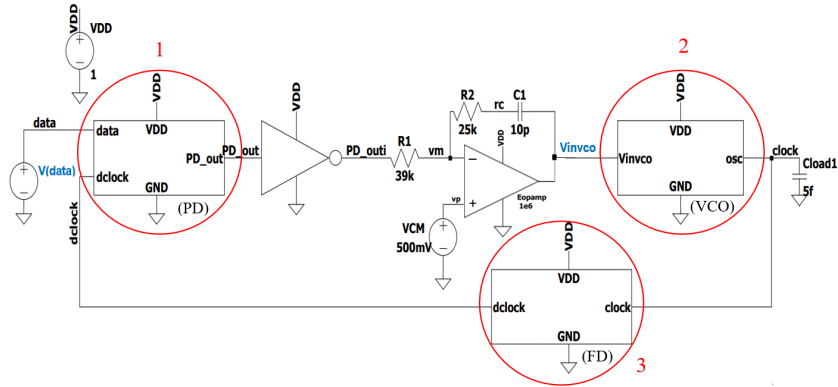
مدار DPLL، زمان شبیه‌سازی بالایی دارد، بنابراین هدف ارائه ماکرومدلی است که علاوه بر دقت مناسبی مشابه با SPICE، دارای سرعت به مراتب بالاتری نسبت به SPICE باشد. در بخش دوم به بررسی ساختار مدار DPLL پرداخته شده است. سپس در بخش سوم به توصیف کلی شبکه‌های عصبی مصنوعی اشاره شده است و بررسی شبکه‌ی عصبی بازگشتی و شبکه‌ی پیشنهادی برای مدل‌سازی مدار در بخش چهارم ارائه خواهند شد. سپس در بخش پنجم به مدل‌سازی مدار DPLL با سه شبکه RNN، GRU و LSTM پرداخته شده است و نتایج حاصل از آموزش مدل با سه شبکه، بررسی و مقایسه شده است.

۲- مدار حلقه قفل فاز دیجیتالی

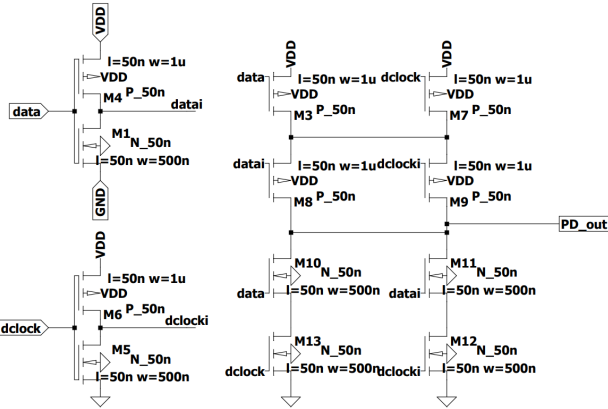
مدار حلقه قفل فاز^۶ (PLL) یک بلوک مدار بسیار پرکاربرد است که به طور گسترده در فرکانس‌های رادیویی استفاده می‌شود [۱۶]. مدار PLL، یک سیستم کنترلی است که یک سیگنال خروجی تولید می‌کند و فاز آن نسبت به فاز سیگنال ورودی ثابت است. نگه داشتن فاز ورودی و خروجی در lockstep، مستلزم فرکانس‌های ورودی و خروجی یکسان است، بنابراین یک حلقه قفل فاز همچنین می‌تواند فرکانس ورودی را

7. Phase Detector
8. Ring Filter
9. Voltage Controlled Oscillator
10. Machine Learning
11. Universal Approximation Theorem
12. Feed Forward

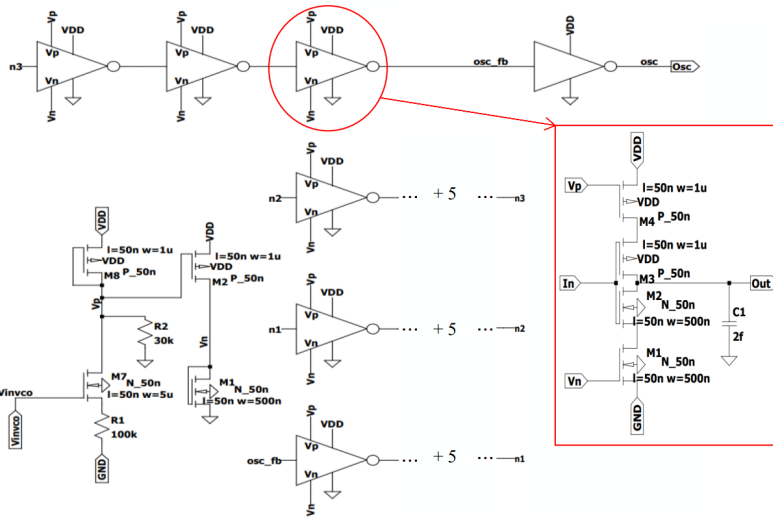
1. Recurrent Neural Network
2. Vanishing
3. Gated Recurrent Unit
4. Long-Short Term Memory
5. Overfitting
6. Phase-Locked Loops



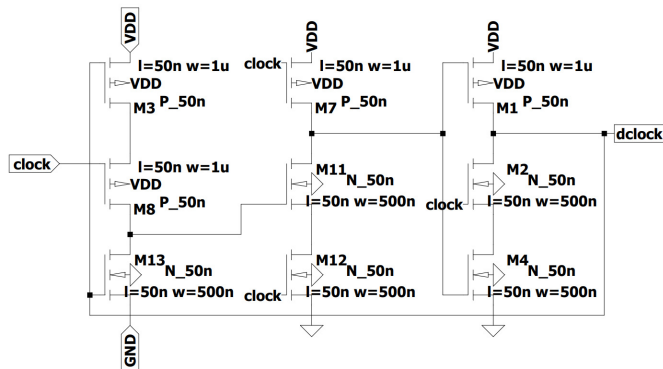
(الف)



(ب)

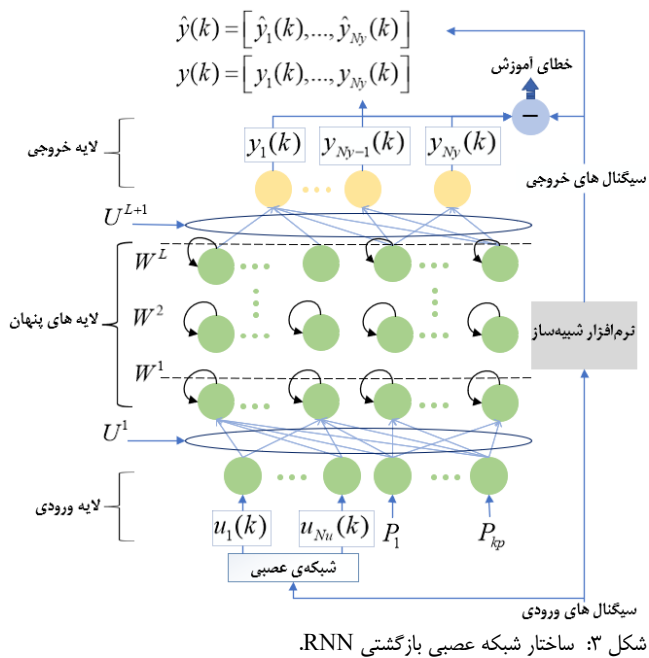


(ج)



(د)

شکل ۲: ساختار مدار DPLL، (الف) مدار DPLL، (ب) بلوک شماره یک مدار آشکار ساز فاز (PD)، (ج) بلوک شماره دو مدار نوسان ساز (VCO) و (د) بلوک شماره سه مدار تقسیم کننده فرکانس (FD).



شکل ۳: ساختار شبکه عصبی بازگشتی RNN.

با ماتریس $P = [P_1 P_2 \dots P_{K_p}]^T$ به صورت نشان داده می‌شوند، که در آن K_p تعداد پارامترها است. همچنین خروجی RNN را می‌توان به صورت (۳) بیان کرد

$$\hat{y}(t) = \psi(X(t), u(t), t) \quad (3)$$

که در آن تابع، رابطه غیرخطی بین سیگنال‌های ورودی و سیگنال‌های خروجی مدار اصلی را بر اساس شبکه عصبی مصنوعی عمیق نشان می‌دهد.

ساختار RNN پیشنهادی باید برای ایجاد یک مدل مناسب که رفتار دینامیکی یک مدار غیرخطی را نشان می‌دهد، آموزش داده شود. برای این کار از مجموعه‌ای از شکل موج‌های ورودی-خروجی تحت عنوان داده‌های آموزش استفاده می‌شود. برای تولید داده‌های آموزش از ابزارهای شبیه‌سازی متداول مانند SPICE استفاده می‌شود [۲۳].

شبکه‌های RNN بر روش‌های پیشین از نظر دقت و زمان برتری دارند. این شبکه‌ها قابلیت بالایی در یادگیری رفتار دینامیکی مدارهای غیرخطی دارند. با این حال، فاز آموزش آن، پروسه زمان‌بری می‌باشد و وابستگی بسیاری به پارامترهای روش آموزش دارد. بدین صورت که در طی مرحله آموزش، نیاز است تا از تابع خطا نسبت به وزن‌های اتصالات مشتق گرفته شود. به دلیل این‌که خروجی در هر لحظه به خروجی لحظه‌های پیشین، وابسته است، عملیات پس‌انتشار در طول زمان عمیق می‌شود و مقدار گرادیان به سمت صفر میل می‌کند، به عبارتی دیگر امحا گرادیان اتفاق می‌افتد. این امر باعث می‌شود که در روند آموزش اختلال ایجاد شود و به روزرسانی وزن‌ها به کندی صورت پذیرد و یا متوقف گردد. در نتیجه، زمان زیادی برای آموزش صرف می‌شود. بنابراین، ارائه مدل کارآمدتری در زمینه مدل‌سازی مدارهای غیرخطی ضروری است.

۲-۳ شبکه GRU

شبکه GRU در سال ۲۰۱۴ توسط چو و همکاران ارائه شد [۲۴]. این شبکه از خانواده شبکه‌های عصبی بازگشتی می‌باشد که محبوبیت و کارایی بالایی دارد. ساختار شبکه عصبی GRU از دو گیت تشکیل شده

تابع غیرخطی را مدل کنند [۱۳]. نورون‌ها همان عناصر پردازشگر می‌باشند، نورون‌ها داده‌های ورودی را از نورون‌های موجود و یا از خارج شبکه دریافت می‌کنند و با عملیات پردازشی که انجام می‌دهند خروجی را تعیین می‌کنند و به خروجی یا سایر نورون‌های موجود در شبکه انتقال می‌دهند. نورون‌ها برای تبیین خروجی از توابع مختلفی استفاده می‌کنند که از پر کاربردترین آن‌ها می‌توان به تابع سیگموئید^۱ یا لجستیک، تابع فعال‌سازی خطی^۲، تانژانت هایپربولیک، آرک تانژانت و تابع رلو^۳ اشاره کرد. شبکه‌های عصبی دارای ساختارهای مختلفی هستند که نوع ساختار آن‌ها با توجه به مسئله‌ای که در حال حاضر مورد بررسی قرار دارد، انتخاب می‌شود. استفاده از مدل‌های شبکه‌های عصبی نیازی به دانستن جزئیات داخلی مدار و قطعات ندارد، این ویژگی یک مزیت بسیار مهم است، زیرا در مدل‌هایی که به معماری داخلی مدار وابسته هستند، با افزایش پیچیدگی مدار، محاسبات پیچیده‌تری نیاز است و زمان لازم برای مدل‌سازی نیز بیشتر می‌شود. در این پژوهش به بررسی ساختاری دو نوع از این شبکه‌ها و بررسی عملکرد این دو ساختار در مدل‌سازی مدار DPLL پرداخته شده است. همچنین، به منظور مقایسه منصفانه، شبکه LSTM نیز با شرایط مشابه پیاده‌سازی و تحلیل شده است و در نهایت به مقایسه نتایج پرداخته شده است. در ادامه ساختار شبکه‌های عصبی بازگشتی مورد بحث قرار می‌گیرد.

۳-۱ شبکه RNN

شبکه‌های عصبی بازگشتی، که دارای بازخوردهای محلی و سراسری هستند، اغلب برای داده‌هایی که ترتیب دارند و بحث زمان در آن‌ها مطرح است، استفاده می‌شوند و به عنوان پرکاربردترین نوع شبکه‌های عصبی محسوب می‌شوند. این شبکه‌ها انواع مختلفی دارند و در زمینه‌های مختلفی از جمله مدل‌سازی رفتاری مدارات غیرخطی کاربرد دارند. خواص یک مدار غیرخطی اصلی را می‌توان به طور کلی به عنوان یک سیستم غیرخطی در قالب متغیرهای حالت مطابق با (۱) و (۲) توصیف کرد [۲۲].

$$\dot{X}(t) = \varphi(X(t), u(t), p, t) \quad (1)$$

$$\hat{y}(t) = \psi(X(t), u(t), t) \quad (2)$$

که در آن X بردار متغیرهای حالت، u بردار سیگنال‌های ورودی، \hat{y} بردار سیگنال‌های خروجی و p بردار پارامترهای مدار و \dot{X} مشتق زمانی X است. هدف RNN، تقریب زدن فرمول غیر خطی دیفرانسیلی با سرعت بالاتر و پیچیدگی کمتر می‌باشد.

شکل ۳، ساختار کلی RNN را نشان می‌دهد. این ساختار از یک لایه ورودی، یک لایه خروجی و چندین لایه پنهان تشکیل شده است. در این شکل $u(k)$ و $\hat{y}(k)$ به ترتیب سیگنال‌های ورودی و خروجی مدار اصلی و $y(k)$ سیگنال خروجی RNN می‌باشد. بردار ورودی به لایه پیشخور شامل سیگنال‌های ورودی و پارامترهای مدار است.

سیگنال‌های آموزشی از مدار اصلی تولید می‌شوند، در این شکل، بردارهای سیگنال ورودی و خروجی در زمان جاری به ترتیب $u(k)$ و $y(k)$ نامیده می‌شوند که در آن $u(k) = [u_1(k), u_2(k), \dots, u_{N_u}(k)]$ و $y(k) = [y_1(k), y_2(k), \dots, y_{N_y}(k)]$ به ترتیب تعداد سیگنال‌های ورودی و خروجی هستند. پارامترهای طراحی در مدار اصلی

1. Sigmoid
2. Linear Activation Function
3. Relu

جدول ۱: تعداد داده‌های تولید شده برای مدار DPLL در نرم افزار LTSpice

تعداد فایل داده	نوع داده تولید شده
۴۱ فایل داده	داده‌های آموزش شبکه عصبی
۱۶ فایل داده	تولید داده‌های آزمون شبکه عصبی

پنهان قبلی نگه داشته می‌شود و به لحظه فعلی منتقل می‌شود. وقتی یک سیگنال زمانی ورودی داده می‌شود $X = [x_1, x_2, \dots, x_t, \dots, x_T]$ با طول T ، GRU آخرین حالت پنهان h_t را از طریق یک تبدیل غیرخطی به عنوان خروجی عبور می‌دهد [۲۶].

$$o = \sigma(W_o h_t + b_o) \quad (\lambda)$$

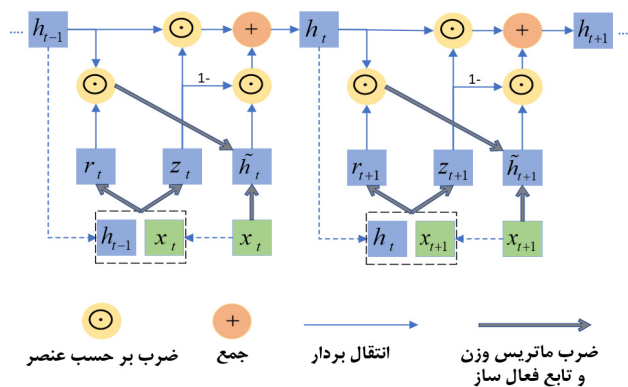
با توجه به (λ) ، $W_o \in \mathbb{R}^{d_o \times d_h}$ ماتریس‌های وزن لایه خروجی می‌باشند. شبکه عصبی GRU می‌تواند با مشکل ناپدید شدن گرادینان در RNN کنار بیاید، بنابراین برای مدل‌سازی فرآیندهای وابسته به زمان طولانی مناسب‌تر می‌باشد.

۴- مدل‌سازی مدار DPLL با سه شبکه عصبی منتخب

در این قسمت به شبیه‌سازی رفتار مدار غیرخطی DPLL با سه شبکه-ی عصبی RNN، GRU و LSTM پرداخته شده است. در فرآیند آموزش شبکه عصبی، داده‌ها نقش بسیار مهمی ایفا می‌کنند. داده‌های ورودی مدل به صورت شبیه‌سازی شده و تحت شرایط کنترل شده تهیه شده‌اند و شامل نویز یا اختلالات احتمالی نمی‌باشند. مدار انتخابی دارای یک ورودی و دو خروجی می‌باشد که از یک خروجی آن برای مدل‌سازی استفاده می‌شود. برای استخراج داده، یک ورودی $V(\text{data})$ و یک خروجی $V(\text{vinvco})$ در نظر گرفته شده است و هر بار پارامترها در محدوده مشخصی تنظیم می‌شوند. داده‌های مورد نیاز از طریق نرم افزار شبیه‌سازی LTSpice استخراج شده است.

همان گونه که جدول ۱ نشان می‌دهد نسبت داده‌های آموزش و آزمون بدین صورت است که داده‌های آموزش باید بیشتر از داده‌های آزمون باشند تا فرآیند آموزش با کمترین میزان خطا صورت گیرد. داده‌های تولید شده توسط نرم‌افزار، نتیجه‌ی تغییرات زمان صعود/ نزول پالس ورودی و ظرفیت خازن خروجی مدار DPLL هستند. سپس داده‌ها برای مدل‌سازی به شبکه مدنظر داده می‌شود و شبیه‌سازی انجام می‌گیرد.

در فرآیند آموزش مدل‌های بازگشتی نظیر RNN، GRU و LSTM، بهینه‌ساز Adam^۳ به عنوان یکی از روش‌های پیشرفته و تطبیقی گرادینان به خدمت گرفته شده است. این الگوریتم با تنظیم خودکار نرخ یادگیری برای هر پارامتر، موجب افزایش سرعت همگرایی مدل، کاهش نوسانات گرادینان در طول فرآیند آموزش و کاهش مشکلاتی از قبیل محوشدگی یا انفجار^۴ گرادینان می‌گردد. این ویژگی‌ها به بهبود پایداری و اثربخشی فرایند یادگیری در شبکه‌های بازگشتی منجر می‌شوند و عملکرد مدل‌ها را به شکل قابل توجهی ارتقاء می‌دهند [۲۷]. همچنین، به دلیل ماهیت توالی‌محور و بازگشتی این مدل‌ها و اندازه نسبتاً کوچک شبکه‌ها، از تکنیک نرمال‌سازی دسته‌ای^۵ استفاده نشده است. گرچه این روش در شبکه‌های عمیق معمولی کاربرد فراوان دارد، در ساختارهای بازگشتی



شکل ۴: ساختار شبکه عصبی واحد بازگشتی گیتی.

است: گیت به‌روزرسانی^۱ و گیت بازنشانی^۲. شبکه عصبی GRU ساده‌شده شبکه LSTM می‌باشد و پارامترها و گیت‌های کمتری دارد [۲۵]. شبکه GRU با توجه به ساختاری که دارد، دارای ویژگی حفظ اطلاعات برای طولانی‌مدت می‌باشد که با استفاده از دو گیت انجام می‌شود.

بخش‌های به‌روزرسانی h_t در شبکه GRU به شرح زیر است:

۱- گیت بازنشانی r_t و گیت به‌روزرسانی z_t

$$z_t = \sigma(W_{zh} h_{t-1} + W_{zx} x_t + b_z) \quad (4)$$

$$r_t = \sigma(W_{rh} h_{t-1} + W_{rx} x_t + b_r) \quad (5)$$

تابع فعال‌ساز این شبکه تابع سیگموئید σ است و محدوده خروجی آن بین $[0, 1]$ است.

۲- بردار فعال‌ساز:

$$\tilde{h}_t = \tanh(W_{\tilde{h}h} (r_t \odot h_{t-1}) + W_{\tilde{h}x} x_t + b_{\tilde{h}}) \quad (6)$$

در معادلات بالا $W_{zh}, W_{rh}, W_{\tilde{h}h} \in \mathbb{R}^{d_h \times d_h}$ ماتریس‌های وزن لایه پنهان به لایه پنهان هستند، $W_{zx}, W_{rx}, W_{\tilde{h}x} \in \mathbb{R}^{d_h \times d_x}$ ماتریس‌های وزن لایه ورودی به لایه پنهان و $b_z, b_r, b_{\tilde{h}}, b_o \in \mathbb{R}^{d_h}$ بایاس می‌باشند.

گیت \tilde{h}_t از گیت بازنشانی r_t برای کنترل جریان وضعیت پنهان قبلی h_{t-1} که حاوی اطلاعات گذشته است، استفاده می‌کند، اگر گیت بازنشانی تقریباً صفر باشد، حالت پنهان قبلی حذف خواهد شد. بنابراین گیت بازنشانی مکانیزمی را برای حذف حالت‌های پنهان قبلی که به آینده مرتبط نیستند فراهم می‌کند، یعنی گیت بازنشانی تعیین می‌کند که چه مقدار اطلاعات در گذشته فراموش شده است.

۳- حالت پنهان:

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \tilde{h}_t \quad (7)$$

که در آن r_t, z_t, x_t, h_t به ترتیب گیت بازنشانی، گیت به‌روزرسانی، بردار ورودی و بردار خروجی می‌باشند.

شکل ۴، ساختار شبکه GRU را نشان می‌دهد، شبکه GRU از یک گیت بازنشانی استفاده می‌کند و گیت را به روز می‌کند، و می‌تواند قضاوت کند که آیا اطلاعات در حالت پنهان قبلی مفید می‌باشد یا خیر؟ سپس اطلاعات مفید را نگه می‌دارد و اطلاعات بی‌فایده را حذف می‌کند. حالت پنهان h_t از گیت به‌روزرسانی z_t برای به‌روزرسانی حالت پنهان قبلی h_{t-1} و گیت \tilde{h}_t می‌باشد. اگر گیت به‌روزرسانی تقریباً ۱ باشد، حالت

3. Adam Optimizer
4. Exploding
5. Batch Normalization

1. Update
2. Reset

جدول ۲: مقایسه نتایج خطای آموزش و آزمون به دست آمده از شبکه عصبی بازگشتی RNN با تعداد نورون‌های متغیر.

پارامترها	خطای آزمون	خطای آموزش	نورون
۱۲۸۱	۰٫۱۵۸۰۱۵۰۲۳	۰٫۱۵۶۵۶۰۰۵۵	۲۰
۲۸۲۱	۰٫۱۶۵۴۶۳۰۴۱	۰٫۱۶۵۱۶۸۵۹۷	۳۰
۴۹۶۱	۰٫۱۶۵۷۶۴۸۶۳	۰٫	۴۰

جدول ۳: مقایسه نتایج خطای آموزش و آزمون به دست آمده از شبکه عصبی بازگشتی RNN با تعداد لایه‌های متغیر.

پارامترها	خطای آزمون	خطای آموزش	تعداد لایه
۴۶۱	۰٫۱۶۵۱۶۶۴۸۹	۰٫۱۶۴۶۸۲۰۵۷	۱
۱۲۸۱	۰٫۱۵۸۰۱۵۰۲۳	۰٫۱۵۶۵۶۰۰۵۵	۲
۲۱۰۱	۰٫۱۹۲۱۶۵۷۴۱	۰٫۱۹۳۶۷۸۹۰۴	۳

جدول ۴: مقایسه نتایج خطای آموزش و آزمون به دست آمده از شبکه عصبی بازگشتی RNN با نرخ یادگیری متغیر.

پارامترها	خطای آزمون	خطای آموزش	نرخ یادگیری
۱۲۸۱	۰٫۲۰۸۶۶۹۸۷۳	۰٫۲۰۵۹۳۴۷۷۹	۰٫۰۰۱
۱۲۸۱	۰٫۱۲۷۳۱۳۹۷۸	۰٫۱۲۷۷۵۷۷۸۶	۰٫۰۱
۱۲۸۱	۰٫۱۸۲۱۶۴۷۸۰	۰٫۱۸۱۹۰۹۹۰۵	۰٫۱

مؤثری برای یافتن ترکیب بهینه پارامترها هستند و می‌توانند در مطالعات آتی برای بهبود دقت مدل و تنظیم دقیق‌تر هاپیرپارامترها مورد استفاده قرار گیرند. در این شبیه‌سازی، چهار پارامتر که به ترتیب شامل تعداد نورون‌ها (N)، تعداد لایه‌ها (L)، نرخ یادگیری (L_r) و دوره‌ها (E) هستند، متغیر قرار داده شده و هر بار کد اجرا می‌شود تا نتایج خطای آموزش و خطای آزمون به دست آیند. در هر مرحله از شبیه‌سازی، پارامتر متغیر به عنوان مقدار مجهول در نظر گرفته شده است. در این فرآیند، تابع سیگموئید و تانژانت هاپیربولیک به عنوان تابع فعال‌ساز انتخاب شده و اندازه دسته برابر با ۳۲ تا انتهای مدل‌سازی برای هر دو شبکه RNN و GRU تنظیم شده است. برای مثال، اگر L برابر با ۳ باشد، به معنای وجود ۳ لایه است.

با توجه به جدول ۲، تعداد نورون‌ها متغیر قرار داده شده است و تنظیم پارامترها به صورت (2000 و 0.01 و N) می‌باشند. با تنظیم مقدار متغیر و سپس مدل‌سازی مدار DPLL در شبکه RNN، در نهایت خطای آموزش و آزمون به دست آمده است و هدف دستیابی به کمترین مقدار خطا است.

پس از تغییر تعداد نورون‌ها و عدم دستیابی به خطای قابل قبول در آموزش و آزمون، تعداد لایه‌ها متغیر قرار داده شده است. با توجه به نتایج حاصل از مرحله قبل، تعداد نورون برابر با ۲۰ خطای بهتری ایجاد می‌کند و در نتیجه در این مرحله تعداد نورون‌ها ثابت و برابر با ۲۰ انتخاب شده است و تعداد لایه‌ها تغییر می‌کند. سپس با تنظیم پارامترها به صورت (2000 ، 0.01 ، L و 20)، مقادیر خطای آموزش و آزمون محاسبه می‌شود. همان‌طور که در جدول ۳ مشاهده می‌شود، خطای آموزش و خطای آزمون با دو لایه، مقادیر کمتری دارا هستند، بنابراین در تنظیم پارامترهای قسمت‌های بعد، تعداد لایه‌ها مقدار دو در نظر گرفته شده است. بررسی نتایج به دست آمده از جداول ۲ و ۳ نشان می‌دهد با متغیر قرار دادن تعداد نورون‌ها و تعداد لایه‌ها، خطاهای به دست آمده بسیار زیاد و نزدیک به یکدیگر می‌باشند و قابل قبول نیستند، بنابراین شبکه عصبی بازگشتی RNN مدل‌سازی مناسبی ارائه نداده است و گرادیان محو شده است.

در ادامه پس از تعیین تعداد لایه‌ها و ثابت شدن آن، نرخ یادگیری به صورت متغیر تعیین می‌شود. کاهش خطای آموزش و آزمون نشان‌دهنده بهبود آموزش شبکه است، بنابراین تنظیم نرخ یادگیری به گونه‌ای انجام می‌شود که خطای کمتری حاصل شود و تنظیم پارامترها به صورت (2000 ، L_r ، 20 و 20) می‌باشد و نتایج در جدول ۴ نشان داده شده است. نرخ یادگیری یکی از پارامترهای کلیدی در فرآیند آموزش مدل‌های یادگیری عمیق به شمار می‌رود. در این پژوهش، مقدار بهینه نرخ یادگیری برای هر مدل به صورت تجربی و از طریق جستجوی پارامتری^۵ تعیین شده است. این فرآیند با آزمایش مقادیر مختلف نرخ یادگیری و ارزیابی عملکرد مدل در مجموعه‌های آموزش و آزمون انجام گرفت. هدف از این

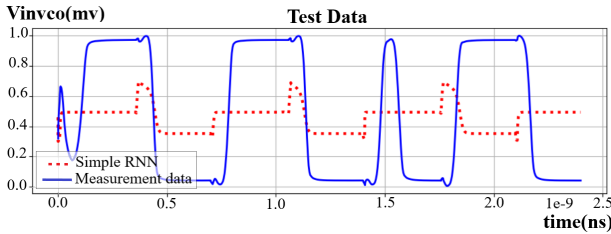
نظیر RNN و GRU، به طور معمول تکنیک‌هایی همچون نرمال‌سازی لایه^۱ یا روش‌های مشابه که درون سلول‌های بازگشتی پیاده‌سازی می‌شوند، کارآمدتر و مناسب‌تر هستند. علت این امر نیاز به نرمال‌سازی داده‌ها در طول توالی و حفظ وابستگی‌های زمانی است که در نرمال‌سازی دسته‌ای به خوبی قابل تأمین نیست. این روش‌های جایگزین همچنین به بهبود کنترل مشکلات مرتبط با محوشدگی گرادیان کمک شایانی می‌کنند. استفاده از بهینه‌ساز Adam همراه با ساختارهای بازگشتی بهینه، در کنار پرهیز از تکنیک‌هایی که با ماهیت زمانی مدل سازگاری ندارند، منجر به افزایش پایداری، سرعت همگرایی بالاتر و بهبود دقت در فرآیند آموزش مدل‌های RNN، GRU و LSTM شده است.

ساختار شبکه (شامل تعداد لایه‌ها و تعداد نورون‌ها در هر لایه) با بهره‌گیری از فرآیند جستجوی تجربی و تحلیل حساسیت پارامترهای معماری تعیین شده است. در این فرآیند، مدل با ترکیبات مختلف ساختاری آموزش داده شد و ساختاری که کمترین مقدار خطا و بالاترین قابلیت تعمیم‌پذیری^۲ را داشت، به عنوان ساختار بهینه انتخاب گردید. این رویکرد به کنترل پیچیدگی مدل، کاهش خطر بیش‌برازش و حفظ تعادل میان پیچیدگی و عملکرد مدل کمک کرده است.

۴-۱ مدل‌سازی با شبکه RNN

ابتدا مدل‌سازی با شبکه RNN انجام شده است که نتایج این شبیه‌سازی به صورت جدول ۲ ارائه شده است. در این پژوهش، تنظیم هاپیرپارامترها به صورت تجربی و دستی انجام شده است. به دلیل محدودیت‌های زمانی و منابع محاسباتی، از روش‌های خودکار مانند جستجوی شبکه‌ای^۳ یا بهینه‌سازی بیزین^۴ استفاده نشده است. فرآیند تنظیم به صورت مرحله‌ای و با آزمایش مقادیر مختلف هاپیرپارامترها انجام شده و انتخاب مقادیر مناسب، بر اساس تحلیل عملکرد مدل بر روی داده‌های آموزش و آزمون صورت گرفته است. این روش تجربی امکان تعیین بهترین ترکیب پارامترها برای دستیابی به کمترین میزان خطا و بالاترین تعمیم‌پذیری را فراهم کرده است. با این حال، روش‌های بهینه‌سازی خودکار مانند جستجوی شبکه‌ای و بهینه‌سازی بیزین ابزارهای

1. Layer Normalization
2. Generalization
3. Grid Search
4. Bayesian Optimization



شکل ۵: نتیجه شبیه‌سازی مدار مدل شده به کمک شبکه‌ی عصبی بازگشتی RNN با نرخ یادگیری ۰/۰۱ در حالت آزمون.

جدول ۵: مقایسه نتایج خطای آموزش و آزمون به دست آمده از شبکه عصبی بازگشتی RNN با دوره‌های متغیر.

دوره‌ها	خطای آموزش	خطای آزمون	پارامترها
۱۰۰۰	۰/۲۰۳۳۶۵۰۷۶	۰/۲۱۶۵۰۱۰۳۴	۱۲۸۱
۲۰۰۰	۰/۲۰۲۱۶۵۴۳۴	۰/۲۱۵۱۱۰۵۶۵	۱۲۸۱
۳۰۰۰	۰/۲۰۴۹۳۴۴۶۷	۰/۲۱۵۳۹۰۱۳۹	۱۲۸۱

بعد از تعیین تعداد نورون‌ها، تعداد لایه‌ها به صورت متغیر قرار داده شده است. پارامترها به صورت $(L, ۰/۰۱, ۲۰۰۰)$ و (۴۰) تنظیم و شبیه‌سازی انجام شده است. نتایج شبیه‌سازی در جدول ۷ ارائه شده است. در مدل‌سازی با شبکه عصبی GRU، با استفاده از دو لایه کمترین خطاهای آموزش و آزمون به دست آمده است. بنابراین، تعداد لایه‌ها در شبیه‌سازی به دو ثابت شده و در مرحله بعد نرخ یادگیری به عنوان یک پارامتر متغیر در نظر گرفته شده است. بر اساس جدول ۸، با تنظیم پارامترها به صورت $(L, ۰/۰۱, ۲, ۴۰)$ ، نرخ یادگیری ۰/۰۱ بهترین عملکرد را با کمترین خطا ارائه داده است.

در قسمت بعد، مقدار دوره‌ها برای شبکه GRU تنظیم شده است و سایر مقادیر به ترتیب $(E, ۰/۰۱, ۲, ۴۰)$ تعیین شده‌اند. نتایج این بررسی در جدول ۹ نمایش داده شده است.

نتایج جدول ۹ نشان می‌دهد که با تعداد دوره‌ها برابر با ۶۰۰۰ کمترین خطای آموزش و آزمون که به دست آمد، خطای قابل قبولی را ارائه داده است و نتیجه این شبیه‌سازی در شکل ۷ آورده شده است. مشاهده می‌شود که شبکه عصبی واحد بازگشتی گیتی و داده‌های اندازه‌گیری به‌طور کامل یکدیگر را پوشش داده و خطای کمتری را نشان می‌دهند.

با توجه به نتایج به دست آمده، شبکه GRU در مدل‌سازی مدار محوشدگی گرادیان را حل کند. این شبکه قادر به پردازش توالی‌های طولانی و حفظ اطلاعات بلندمدت است، در حالی که RNN در این زمینه ناکام است و به دلیل زمان‌های طولانی و پیچیدگی داده‌ها نمی‌تواند به خوبی آموزش ببیند و گرادیان به سمت صفر میل می‌کند.

۳-۴ مدل‌سازی با شبکه LSTM

مدل‌سازی با شبکه LSTM پس از اجرای شبیه‌سازی با شبکه GRU انجام پذیرفته است. در این مرحله نیز، تنظیم پارامترهای مدل با هدف دستیابی به عملکرد بهینه صورت گرفته و فرآیند آموزش و آزمون با تعداد نورون‌های متغیر اجرا شده است. مشابه مراحل پیشین، نوع تابع فعال‌ساز و اندازه دسته مطابق تنظیمات بهینه در مراحل قبلی در نظر گرفته شده‌اند. در جدول ۱۰، نتایج حاصل از تغییر تعداد نورون‌ها در لایه بازگشتی شبکه LSTM ارائه شده است. پارامترها برای هر مدل به صورت $(۲۰۰۰, ۰/۰۱)$

جدول ۴: مقایسه نتایج خطای آموزش و آزمون به دست آمده از شبکه عصبی بازگشتی RNN با نرخ یادگیری متغیر.

پارامترها	خطای آزمون	خطای آموزش	نرخ یادگیری
۱۲۸۱	۰/۲۰۸۶۹۸۸۷۳	۰/۲۰۵۹۳۴۷۷۹	۰/۰۰۱
۱۲۸۱	۰/۱۲۳۱۳۹۷۸	۰/۱۲۷۷۵۷۷۸۶	۰/۰۱
۱۲۸۱	۰/۱۸۲۱۶۴۷۸۰	۰/۱۸۱۹۰۹۹۰۵	۰/۱

ارزیابی، یافتن مقداری از نرخ یادگیری بود که منجر به کمترین مقدار تابع هزینه^۱ در هر دو مجموعه آموزش و آزمون گردد. نرخ یادگیری انتخاب‌شده موجب همگرایی سریع و پایدار مدل به سمت کمینه تابع هزینه شد، بدون آن‌که باعث نوسانات شدید یا ناپایداری در روند بهینه‌سازی گردد. این رویکرد تضمین می‌کند که فرآیند آموزش با بیشترین کارایی و پایداری ممکن انجام شود و مدل نهایی از تعمیم‌پذیری مطلوبی در مواجهه با داده‌های جدید برخوردار باشد.

در شکل ۵، کمترین مقدار خطای آموزش و آزمون نشان داده شده است. محور هدف (منحنی آبی‌رنگ) نشان‌دهنده شکل موج مدار بر اساس داده‌های آموزش و آزمون است، در حالی که خط قرمز مقدار پیش‌بینی شده برای شبیه‌سازی مدار را نشان می‌دهد. با توجه به نتایج، مقدار هدف و پیش‌بینی شده به خوبی هم‌پوشانی ندارند و این امر نشان‌دهنده عدم رضایت‌بخش بودن نتایج است، زیرا ایده‌آل این است که این دو خط یکدیگر را به‌طور کامل پوشش دهند و خطای کمتری را ارائه دهند.

پس از تعیین نرخ یادگیری، در مرحله بعد تعداد دوره‌ها به صورت متغیر در نظر گرفته شده است و پارامترها به صورت $(E, ۰/۰۱, ۲, ۲۰)$ تنظیم شده است. نتایج شبیه‌سازی در جدول ۵ ارائه شده است.

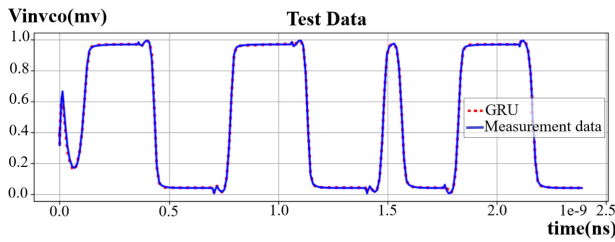
مدل‌سازی مدار غیرخطی با استفاده از شبکه عصبی RNN نتایج رضایت‌بخشی نداشته است و تغییر در تعداد نورون‌ها، لایه‌ها، نرخ یادگیری و دوره‌ها تأثیر مثبتی بر کیفیت مدل‌سازی نداشته است. شبکه‌های عصبی بازگشتی قادر به پردازش توالی‌های طولانی نیستند و اطلاعات بلندمدت در اثر محو شدن گرادیان از بین می‌رود. این پدیده باعث می‌شود که گرادیان به سمت صفر میل کند و فرآیند به‌روزرسانی وزن‌ها به کندی انجام شود یا متوقف گردد، که در نتیجه آموزش شبکه به خوبی انجام نمی‌گیرد.

در نهایت، برای حل مشکل محوشدگی گرادیان به سراغ شبکه‌های عصبی GRU رفته و مدل‌سازی مدار با این شبکه انجام شده است.

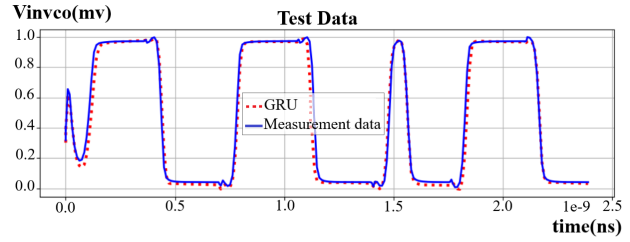
۴-۲ مدل‌سازی با شبکه GRU

مدل‌سازی با شبکه GRU پس از شبکه RNN انجام گرفته است، برای این شبکه هم نیز پارامترها تنظیم می‌شوند و شبیه‌سازی اجرا می‌شود. در ابتدا، تعداد نورون‌ها متغیر قرار داده شده است. با توجه به توضیحات قبل، نوع تابع فعال‌ساز و مقدار اندازه دسته به صورت بهینه تنظیم شده است. در جدول ۶ تعداد نورون‌ها متغیر قرار داده شده است و پارامترها به صورت $(۲۰۰۰, ۰/۰۱, ۲, N)$ تعیین شده‌اند و خطاهای آموزش و آزمون محاسبه شده است.

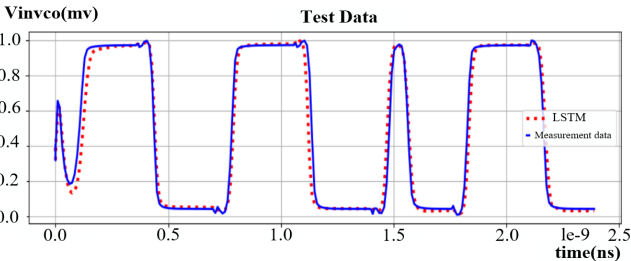
در شکل ۶، نتایج خطای آموزش و آزمون با تعداد متغیر نورون‌ها نمایش داده شده است. نتایج نشان می‌دهد که مدل‌سازی مدار با شبکه GRU می‌تواند خطای قابل قبولی ارائه دهد. در ادامه برای بهبود خطای آموزش و آزمون، پارامترهای دیگر تغییر و تنظیم شده‌اند.



شکل ۷: نتیجه شبیه‌سازی مدار مدل شده به کمک شبکه‌ی عصبی بازگشتی GRU با تعداد ۶۰۰۰ دوره در حالت آزمون.



شکل ۸: نتیجه شبیه‌سازی مدار مدل شده به کمک شبکه‌ی عصبی بازگشتی GRU با تعداد ۴۰ نرون در حالت آزمون.



شکل ۹: نتیجه شبیه‌سازی مدار مدل شده به کمک شبکه‌ی عصبی بازگشتی LSTM با تعداد ۴۰ نرون در حالت آزمون.

جدول ۶: مقایسه نتایج خطای آموزش و آزمون به دست آمده از شبکه عصبی واحد بازگشتی GRU با تعداد نرون‌های متغیر.

پارامترها	خطای آزمون	خطای آموزش	نرون
۱۰۶۱	۰/۰۰۳۸۸۶۸۹۶	۰/۰۰۳۴۳۳۸۱۰	۲۰
۲۳۵۱	۰/۰۰۳۰۹۱۵۴۴	۰/۰۰۲۵۹۸۶۹۸	۳۰
۳۹۲۱	۰/۰۰۲۳۶۲۶۵۵	۰/۰۰۱۹۹۸۱۳۰	۴۰
۶۱۱۱	۰/۰۰۳۲۲۲۴۴۸	۰/۰۰۲۹۲۰۶۳۳	۵۰

جدول ۷: مقایسه نتایج خطای آموزش و آزمون به دست آمده از شبکه عصبی واحد بازگشتی GRU با تعداد لایه‌های متغیر.

پارامترها	خطای آزمون	خطای آموزش	تعداد لایه
۵۲۰۱	۰/۰۰۲۳۰۷۶۷۹	۰/۰۰۲۲۷۷۶۱۷	۱
۳۹۲۱	۰/۰۰۲۳۶۲۶۵۵	۰/۰۰۱۹۹۸۱۳۰	۲
۳۰۱۱	۰/۰۰۴۴۵۵۸۵۶۱	۰/۰۰۳۲۹۸۱۸۶	۳

جدول ۸: مقایسه نتایج خطای آموزش و آزمون به دست آمده از شبکه عصبی واحد بازگشتی GRU با نرخ یادگیری متغیر.

پارامترها	خطای آزمون	خطای آموزش	نرخ یادگیری
۳۹۲۱	۰/۰۰۴۱۲۴۵۰۹	۰/۰۰۳۴۹۵۹۰۳	۰/۰۰۱
۳۹۲۱	۰/۰۰۲۳۶۲۶۵۵	۰/۰۰۱۹۹۸۱۳۰	۰/۰۱
۳۹۲۱	۰/۱۸۰۷۹۲۲۹۱	۰/۱۸۰۶۰۳۶۶۷	۰/۱

جدول ۹: مقایسه نتایج خطای آموزش و آزمون به دست آمده از شبکه عصبی واحد بازگشتی GRU با دوره‌های متغیر.

پارامترها	خطای آزمون	خطای آموزش	دوره‌ها
۳۹۲۱	۰/۰۰۴۳۲۲۷۸۴۰	۰/۰۰۴۳۴۲۵۴۳۵	۱۵۰۰
۳۹۲۱	۰/۰۰۲۳۶۲۶۵۵	۰/۰۰۱۹۹۸۱۳۰	۲۰۰۰
۳۹۲۱	۰/۰۰۲۲۵۶۹۶۸	۰/۰۰۱۹۲۵۰۶۹	۲۵۰۰
۳۹۲۱	۰/۰۰۱۳۹۰۸۷۷	۰/۰۰۱۳۰۱۲۹۸	۳۰۰۰
۳۹۲۱	۰/۰۰۰۷۴۲۳۲۴	۰/۰۰۰۶۷۹۹۷۷	۴۰۰۰
۳۹۲۱	۰/۰۰۰۲۹۵۹۲۳	۰/۰۰۰۲۱۰۹۵۴	۵۰۰۰
۳۹۲۱	۰/۰۰۰۱۵۲۱۲۱	۰/۰۰۰۰۹۳۲۷۹	۶۰۰۰

جدول ۱۰: مقایسه نتایج خطای آموزش و آزمون به دست آمده از شبکه عصبی واحد بازگشتی LSTM با تعداد نرون‌های متغیر.

پارامترها	خطای آزمون	خطای آموزش	نرون
۵۱۶۱	۰/۰۰۳۱۲۸۴۱۷	۰/۰۰۳۰۱۴۳۸۳	۲۰
۱۱۱۹۱	۰/۰۰۳۶۵۲۳۴۱	۰/۰۰۳۴۸۱۵۵۴	۳۰
۱۹۷۲۱	۰/۰۰۳۴۷۹۳۱۶	۰/۰۰۲۸۹۹۷۶۷	۴۰
۳۰۶۵۱	۰/۰۰۳۶۶۱۶۱۸	۰/۰۰۳۰۳۷۰۱۸	۵۰

جدول ۱۱: مقایسه نتایج خطای آموزش و آزمون به دست آمده از شبکه عصبی واحد بازگشتی LSTM با تعداد لایه‌های متغیر.

پارامترها	خطای آزمون	خطای آموزش	تعداد لایه
۶۷۶۱	۰/۰۰۳۱۸۲۲۹۵	۰/۰۰۳۰۱۸۶۴۱	۱
۱۹۷۲۱	۰/۰۰۳۸۷۱۰۶۸	۰/۰۰۳۵۸۰۱۳۶۸	۲
۳۲۶۸۱	۰/۰۰۳۸۸۵۷۴۹	۰/۰۰۳۱۴۲۵۹۶	۳

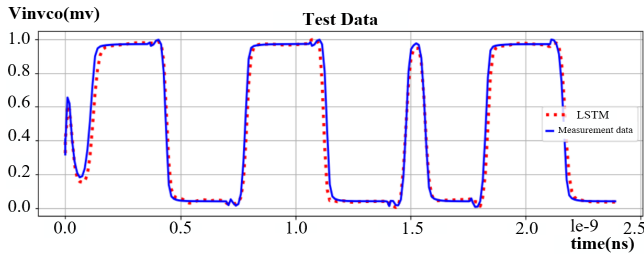
بیشتر، در برخی تنظیمات دچار بیش‌برازش شده و عملکرد آن در داده‌های آزمون نسبت به GRU ضعیف‌تر است. بنابراین در ادامه، به‌منظور کاهش بیش‌برازش و بهبود دقت در داده‌های آزمون، سایر پارامترهای مدل نیز مورد بازتنظیم قرار گرفته‌اند.

پس از تعیین تعداد نرون‌ها، به‌منظور بررسی تأثیر عمق شبکه، تعداد لایه‌ها در ساختار شبکه LSTM به‌صورت متغیر در نظر گرفته شده است. سایر پارامترها به‌صورت (۲۰۰۰، ۰/۰۱، L، ۴۰) تنظیم شده و شبیه‌سازی برای مقادیر مختلف L انجام گردیده است. نتایج حاصل از این شبیه‌سازی در جدول ۱۱ ارائه شده است.

بررسی نتایج نشان می‌دهد که ساختار تک‌لایه در مقایسه با ساختارهای چندلایه، کمترین خطای آموزش و آزمون را به همراه دارد. با این حال، به‌منظور انجام مقایسه‌ای منصفانه و یکسان میان سه ساختار RNN، GRU و LSTM، در ادامه فرآیند مدل‌سازی، ساختار دو لایه برای شبکه LSTM نیز انتخاب شده است؛ این تصمیم بر مبنای تطابق شرایط مدل‌سازی سه شبکه و یکسان‌سازی تعداد لایه‌ها در فرآیند آموزش و آزمون اتخاذ گردیده است. شایان ذکر است که استفاده از

۲، N) مقداردهی شده‌اند و خطاهای آموزش و آزمون برای هر حالت گزارش گردیده است. همان‌طور که مشاهده می‌شود، کاهش قابل توجهی در خطای آموزش در مقایسه با RNN مشاهده می‌شود، اما روند خطای آزمون نسبت به GRU یکنواخت نبوده و در برخی نقاط افزایش یافته است.

در شکل ۸، نتایج خطای آموزش و آزمون در برابر تعداد نرون‌ها ترسیم شده است. با وجود دستیابی به دقت بالاتر در آموزش، نوسان خطای آزمون نشان می‌دهد که شبکه LSTM به دلیل تعداد پارامترهای



شکل ۹: نتیجه شبیه‌سازی مدار مدل شده به کمک شبکه‌ی عصبی بازگشتی LSTM با تعداد دوره در حالت آزمون.

جدول ۱۴: مقایسه نتایج خطای آموزش و آزمون با سه شبکه RNN، GRU و LSTM در مدل‌سازی مدار DPLL.

نوع شبکه	خطای آموزش	خطای آزمون
RNN	۰٫۱۶۱۶۶۶۰۰۹	۰٫۱۶۱۹۶۸۲۸۳۰
GRU	۰٫۰۰۰۹۳۲۷۹	۰٫۰۰۰۱۵۲۱۲۱
LSTM	۰٫۰۰۰۹۷۳۰۹۸	۰٫۰۰۱۰۰۸۷۷۳

LSTM با بهره‌گیری از سه گیت مجزا شامل گیت‌های ورودی، خروجی و فراموشی، توانایی نگهداری اطلاعات بلندمدت و مدل‌سازی وابستگی‌های زمانی عمیق را داراست. این ساختار پیچیده باعث افزایش ظرفیت مدل در یادگیری روابط پیچیده و غیرخطی می‌شود و در برخی کاربردهای حساس، دقت بالاتری فراهم می‌آورد. با این حال، همین پیچیدگی ساختاری منجر به افزایش تعداد پارامترها و در نتیجه مصرف منابع محاسباتی بالاتر و زمان آموزش طولانی‌تر می‌شود. علاوه بر آن، در شرایطی که داده‌های آموزشی محدود یا نویزی باشند، مدل LSTM به دلیل ظرفیت یادگیری بالاتر، مستعد بیش‌برازش است؛ به عبارتی مدل ممکن است داده‌های آموزش را به‌خوبی یاد بگیرد ولی عملکرد ضعیف‌تری روی داده‌های آزمون داشته باشد. در مقابل، GRU با ساختار ساده‌تر و تعداد پارامتر کمتر، نه تنها نیاز به منابع محاسباتی پایین‌تری دارد، بلکه با کاهش احتمال بیش‌برازش، به‌ویژه در مسائل با داده‌های محدود، عملکرد پایدارتری در فاز آزمون ارائه می‌دهد. این ویژگی باعث می‌شود که GRU در بسیاری از کاربردهای مهندسی از جمله مدل‌سازی دینامیک مدارهای غیرخطی، که هم دقت و هم تعمیم‌پذیری اهمیت دارند، انتخابی مناسب‌تر از LSTM باشد. از سوی دیگر، عملکرد شبکه ساده‌تر، RNN نیز در این مقایسه آورده شده است. با وجود ساختار پایه‌ای RNN که فاقد گیت‌های کنترلی درونی است، این مدل در یادگیری وابستگی‌های بلندمدت ناتوان‌تر بوده و معمولاً با مشکل ناپایداری گرادیان مواجه می‌شود. همان‌گونه که در جدول ۱۴ مشاهده می‌شود، خطای آموزش و آزمون در مدل RNN به مراتب بالاتر از دو معماری دیگر است. این نتایج بیانگر آن است که در مدل‌سازی مدار DPLL که شامل وابستگی‌های زمانی و دینامیک پیچیده است، استفاده از معماری‌های پیشرفته‌تر مانند GRU و LSTM می‌تواند منجر به دقت بالاتر و همگرایی بهتر مدل شود.

۵- نتیجه‌گیری

در این پژوهش، شبیه‌سازی رفتار دینامیکی مدارات غیرخطی با استفاده از شبکه‌های عصبی، به‌ویژه شبکه‌های عصبی بازگشتی (RNN، GRU و LSTM) مورد بررسی قرار گرفته است. نتایج مقایسه نشان می‌دهد که ساختار GRU به‌گونه‌ای طراحی شده است که می‌تواند مشکل امحا را برطرف کند. نتایج نشان داد که برای شبیه‌سازی مدار غیرخطی DPLL،

جدول ۱۲: مقایسه نتایج خطای آموزش و آزمون به دست آمده از شبکه عصبی واحد بازگشتی LSTM با نرخ یادگیری متغیر.

پارامترها	خطای آزمون	خطای آموزش	نرخ یادگیری
۱۹۷۲۱	۰٫۰۰۱۹۴۲۶۷۹	۰٫۰۰۱۶۶۹۵۵۳	۰٫۰۰۱
۱۹۷۲۱	۰٫۰۰۲۸۵۸۲۶۱	۰٫۰۰۲۶۳۷۱۵۸	۰٫۰۱
۱۹۷۲۱	۰٫۱۲۵۸۳۸۸۳۴	۰٫۱۲۵۱۸۵۸۷۲	۰٫۱

جدول ۱۳: مقایسه نتایج خطای آموزش و آزمون به دست آمده از شبکه عصبی واحد بازگشتی LSTM با دوره‌های متغیر.

پارامترها	خطای آزمون	خطای آموزش	دوره‌ها
۱۹۷۲۱	۰٫۰۰۱۵۷۸۲۱۶	۰٫۰۰۱۶۵۹۷۶۸	۱۵۰۰
۱۹۷۲۱	۰٫۰۰۳۸۷۱۰۶۸	۰٫۰۰۳۵۸۰۱۳۶	۲۰۰۰
۱۹۷۲۱	۰٫۰۰۴۳۷۵۶۳۰	۰٫۰۰۳۵۳۵۸۰۱	۲۵۰۰
۱۹۷۲۱	۰٫۰۰۳۰۲۹۴۱۷	۰٫۰۰۲۷۸۹۵۸۹	۳۰۰۰
۱۹۷۲۱	۰٫۰۰۱۵۳۳۴۷۲	۰٫۰۰۱۳۸۸۵۳۳	۴۰۰۰
۱۹۷۲۱	۰٫۰۰۰۱۹۰۵۸۳	۰٫۰۰۰۱۶۹۱۲۹۱	۵۰۰۰
۱۹۷۲۱	۰٫۰۰۱۰۰۸۷۷۳	۰٫۰۰۰۹۷۳۰۹۸	۶۰۰۰

لایه‌های بیشتر در LSTM منجر به افزایش تعداد پارامترها و زمان آموزش شده و در عین حال، افزایش خطای آزمون در ساختارهای عمیق‌تر نیز مشاهده گردیده است که می‌تواند به دلیل بروز پدیده‌ی بیش‌برازش باشد. بنابراین، تعداد لایه‌ها در شبیه‌سازی به دو ثابت شده و در مرحله بعد نرخ یادگیری به عنوان یک پارامتر متغیر در نظر گرفته شده است. با توجه به جدول ۱۲، نتایج نشان می‌دهند که این مدل نیز به شدت تحت تأثیر نرخ یادگیری قرار دارد؛ به‌طوری‌که با کاهش نرخ یادگیری از ۰٫۱ به ۰٫۰۰۱، میزان خطای آموزش و آزمون به‌طور قابل توجهی کاهش یافته است. با این حال، در شرایط برابر و با نرخ یادگیری ۰٫۰۰۱، مدل LSTM عملکرد مناسبی از خود نشان داده و خطاهای آموزش و آزمون قابل قبولی را ارائه داده است. این نتایج تأکید می‌کنند که تنظیم دقیق نرخ یادگیری نقش مؤثری در بهبود همگرایی و دقت پیش‌بینی مدل دارد، به‌ویژه زمانی که هدف مقایسه عملکرد مدل‌ها در شرایط یکسان باشد. در ادامه، مقدار دوره‌ها برای شبکه LSTM تنظیم شده است و سایر مقادیر به ترتیب $(E, 0.01, 2, 40)$ تعیین شده‌اند. نتایج این بررسی در جدول ۱۳ نمایش داده شده است.

نتایج به‌دست‌آمده از شبکه LSTM با دوره‌های آموزشی متغیر نشان می‌دهد که افزایش تعداد دوره‌ها به کاهش خطای آموزش و آزمون منجر شده است. با این حال، در دوره‌های میانی نوساناتی در خطای آزمون مشاهده می‌شود که می‌تواند به عوامل تنظیمی مانند Dropout یا وزن‌دهی اولیه نسبت داده شود. از دوره ۳۰۰۰ به بعد، مدل همگرایی بهتری داشته و عملکرد آن تثبیت شده است. شکل ۹ افزایش جزئی خطای آزمون در دوره ۶۰۰۰ احتمال آغاز بیش‌برازش را نشان می‌دهد. این روند با رفتار طبیعی شبکه‌های بازگشتی سازگار بوده است. جدول ۱۴ مقایسه نهایی بین این سه شبکه در شرایطی که پارامترها یکسان تنظیم شده‌اند و به صورت (۶۰۰۰ و ۰٫۰۱ و ۲ و ۲۰) می‌باشد را ارائه می‌دهد.

در مقایسه سه شبکه، معماری‌های GRU و LSTM به‌عنوان دو ساختار پیشرفته از شبکه‌های عصبی بازگشتی، تفاوت‌های ساختاری آن‌ها منجر به رفتار متفاوتی در مسائل مدل‌سازی توالی‌محور شده است. شبکه

- [17] L. G. B. Rolim, D. R. da Costa, and M. Aredes, "Analysis and software implementation of a robust synchronizing PLL circuit based on the pq theory," *IEEE Trans. on Industrial Electronics*, vol. 53, no. 6, pp. 1919-1926, Dec. 2006.
- [18] S. Al-Araji, Z. M. Hussain, and M. Al-Qutairi, *Digital Phase Locked Loops, Architecture and Applications*, Springer, Berlin, 2006.
- [19] R. J. Baker, *CMOS: Circuit Design, Layout, and Simulation*, John Wiley & Sons, 2019.
- [20] D. Banerjee, *PLL Performance, Simulation and Design*, Dog Ear Publishing, 2006.
- [21] R. I. Abdo, A. A. Khalaf, and H. F. A. Hamed, "A power efficient and fast locking CMOS design of all-digital phase-locked loop," *Journal of Advanced Engineering Trends*, vol. 42, no. 1, pp. 197-203, Jan. 2022.
- [22] Y. Fang, M. C. Yagoub, F. Wang, and Q. -J. Zhang, "A new macromodeling approach for nonlinear microwave circuits based on recurrent neural networks," *IEEE Trans. on Microwave Theory and Techniques*, vol. 48, no. 12, pp. 2335-2344, Dec. 2000.
- [23] Z. Naghibi, S. A. Sadrossadat, and S. Safari, "Time-domain modeling of nonlinear circuits using deep recurrent neural network technique," *AEU-International Journal of Electronics and Communications*, vol. 100, pp. 66-74, Feb. 2019.
- [24] J. Chung, *Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling*, arXiv preprint arXiv:1412.3555, 2014.
- [25] R. Fu, Z. Zhang, and L. Li, "Using LSTM and GRU neural network methods for traffic flow prediction," in *Proc. 31st Youth Academic Annual Conf. of Chinese Association of Automation*, pp. 324-328, Wuhan, China, 11-13 Nov. 2016.
- [26] J. Yuan and Y. Tian, "An intelligent fault diagnosis method using GRU neural network towards sequential data in dynamic processes," *Processes*, vol. 7, no. 3, Article ID: 152, 2019.
- [27] T. Robert, M. Safaryan, I. V. Modoranu, and D. Alistarh, *LDAdam: Adaptive Optimization from Low-Dimensional Gradient Statistics*, arXiv preprint arXiv:2410.16103, Oct. 21, 2024.

سیده فاطمه موسوی قوام‌آبادی تحصیلات خود را در مقطع کارشناسی مهندسی برق- الکترونیک در سال ۱۴۰۲ از دانشگاه آزاد اسلامی واحد یزد به پایان رسانده است و هم‌اکنون دانشجوی ارشد مهندسی برق- سیستم‌های الکترونیک و دیجیتال دانشگاه یزد می‌باشد. زمینه تحقیقاتی مورد علاقه ایشان شامل موضوعاتی مانند طراحی دیجیتال، مدل‌سازی مبتنی بر شبکه‌های عصبی می‌باشد.

سید علیرضا صدرالسادات در سال ۱۳۸۶ مدرک کارشناسی خود را در رشته سخت افزار از دانشگاه تهران و در سال ۱۳۸۹ مدرک کارشناسی ارشد خود را در رشته مهندسی برق و کامپیوتر از دانشگاه واترلو کانادا و در سال ۱۳۹۴ مدرک دکترای خود را در رشته مهندسی برق و کامپیوتر از دانشگاه کارلتون کانادا دریافت نمود. از سال ۱۳۹۵ الی ۱۳۹۷ ایشان به عنوان عضو هیأت علمی دانشکده برق و کامپیوتر دانشگاه تهران مشغول به فعالیت بود. دکتر صدرالسادات از سال ۱۳۹۷ الی ۱۳۹۸ به عنوان مسئول پروژه بهینه سازی مدارات VLSI دانشگاه تورونتو کانادا مشغول به کار بوده و پس از آن از سال ۱۳۹۹ الی ۱۴۰۱ به عنوان رئیس بخش هوش مصنوعی دانشگاه یزد بوده است. زمینه‌های علمی مورد علاقه نام‌برده متنوع بوده و عبارتند از: بهینه‌سازی و مدل‌سازی مبتنی بر شبکه‌های عصبی مولفه‌ها به قطعات و مدارهای خطی/غیرخطی، یادگیری عمیق، طراحی به کمک رایانه، طراحی ادغام در مقیاس بسیار بزرگ، طراحی احتمالاتی و حداکثر کردن بازده.

علی مفتخرزاده در سال ۱۳۸۳ مدرک کارشناسی مهندسی برق- الکترونیک را از دانشگاه خواجه نصیرالدین طوسی و در سال ۱۳۸۵ مدرک کارشناسی ارشد مهندسی برق- الکترونیک را از دانشگاه صنعتی شریف دریافت نمود. پس از آن به دوره دکترای مهندسی برق- الکترونیک در دانشگاه صنعتی شریف وارد گردید و در سال ۱۳۹۰ موفق به اخذ درجه دکترا در مهندسی برق- الکترونیک از دانشگاه مذکور گردید. دکتر مفتخرزاده از سال ۱۳۹۱ تا کنون در دانشگاه یزد مشغول به فعالیت بوده و در حال حاضر رییس بخش الکترونیک و کنترل و عضو هیأت علمی دانشکده مهندسی برق دانشگاه یزد می‌باشد. زمینه‌های تحقیقاتی مورد علاقه ایشان شامل موضوعاتی مانند طراحی سیستم دیجیتال، DSP، پردازش تصویر و مدل‌سازی ماکرو مدارهای خطی/ غیرخطی است.

مدل ارائه‌شده در زمان آموزش و آزمایش سریع‌تر می‌باشد و از دقت بسیار بالاتری نسبت به RNN برخوردار است و مشکل محوشدگی گرادیان را حل کرده است. با این حال، ساختار پیچیده LSTM موجب افزایش ظرفیت یادگیری آن شده و در شرایطی مانند حجم محدود داده یا نویز بالا، احتمال بروز پدیده بیش‌برازش در این مدل افزایش می‌یابد. لذا نتیجه می‌شود که مدل‌سازی با روش پیشنهادی می‌تواند در زمینه شبیه‌سازی رفتار مدارات غیرخطی مؤثر باشد.

مراجع

- [1] V. K. Devabhaktuni, M. C. Yagoub, Y. Fang, J. Xu, and Q. J. Zhang, "Neural networks for microwave modeling: Model development issues and nonlinear modeling techniques," *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 11, no. 1, pp. 4-21, 2001.
- [2] Q. -J. Zhang and K. C. Gupta, *Neural Networks for RF and Microwave Design (Book+ Neuromodeler Disk)*. Artech House, Inc., 2000.
- [3] W. Liu, W. Na, L. Zhu, and Q.-J. Zhang, "A review of neural network based techniques for nonlinear microwave device modeling," in *Proc. IEEE MTT-S Int. Conf. on Numerical Electromagnetic and Multiphysics Modeling and Optimization*, 2 pp., Beijing, China, 27-29 Jul. 2016.
- [4] A. Rawat, R. Yadav, and S. Shrivastava, "Neural network applications in smart antenna arrays: A review," *AEU-International Journal of Electronics and Communications*, vol. 66, no. 11, pp. 903-912, Nov. 2012.
- [5] S. A. Sadrossadat, P. Gunupudi, and Q. -J. Zhang, "Nonlinear electronic/photonic component modeling using adjoint state-space dynamic neural network technique," *IEEE Trans. on Components, Packaging and Manufacturing Technology*, vol. 5, no. 11, pp. 1679-1693, Nov. 2015.
- [6] W. Na, F. Feng, C. Zhang, and Q. -J. Zhang, "A unified automated parametric modeling algorithm using knowledge-based neural network and 11 optimization," *IEEE Trans. on Microwave Theory and Techniques*, vol. 65, no. 3, pp. 729-745, Mar. 2016.
- [7] H. Zhang, Y. Jing, and P. Zhou, "Machine learning-based device modeling and performance optimization for FinFETs," *IEEE Trans. on Circuits and Systems II: Express Briefs*, vol. 70, no. 4, pp. 1589-1585, Apr. 2022.
- [8] H. Kabir, et al., "Smart modeling of microwave devices," *IEEE Microwave Magazine*, vol. 11, no. 3, pp. 105-118, 2010.
- [9] B. Mutyry, et al., "Macromodeling of nonlinear transistor-level receiver circuits," *IEEE Trans. on Advanced Packaging*, vol. 29, no. 1, pp. 55-66, Feb. 2006.
- [10] M. Noohi, A. Mirvakili, and S. A. Sadrossadat, "Modeling and implementation of nonlinear boost converter using local feedback deep recurrent neural network for voltage balancing in energy harvesting applications," *International Journal of Circuit Theory and Applications*, vol. 49, no. 12, pp. 4231-4247, Dec. 2021.
- [11] R. Dinesh and R. Marimuthu, "An analysis of ADPLL applications in various fields," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 18, no. 2, pp. 856-866, May 2020.
- [12] P. K. Hanumolu, G. -Y. Wei, U. -K. Moon, and K. Mayaram, "Digitally-enhanced phase-locking circuits," in *Proc. IEEE Custom Integrated Circuits Conf.*, pp. 361-368, San Jose, CA, USA, 16-19 Sept. 2007.
- [13] M. Moradi, S. A. Sadrossadat, and V. Derhami, "Long short-term memory neural networks for modeling nonlinear electronic components," *IEEE Trans. on Components, Packaging and Manufacturing Technology*, vol. 11, no. 5, pp. 840-847, May 2021.
- [14] M. Azodi, S. A. Sadrossadat, and Y. Savaria, "Nonlinear circuit macromodeling using new heterogeneous-layered deep clockwork recurrent neural network," *IEEE Access*, pp. 89506-89519, 2024.
- [15] T. Kandadi and G. Shankarlingam, *Drawbacks of LSTM Algorithm: A Case Study*, Available at SSRN, no. 5080605, Jan. 2025, <http://dx.doi.org/10.2139/ssrn.5080605>.
- [16] H. Ibrahim et al., "Radio frequency energy harvesting technologies: A comprehensive review on designing, methodologies, and potential applications," *Sensors*, vol. 22, no. 11, Article ID: 4144, 2022.