Archive of SID

ارائه‌ی یک ابزار CAD برای طراحی و بهینه‌سازی مقایسه‌گرهای قفلدار

احسان یعقوبی[1]، سیدحمید ظهیری[2]
[1] کارشناسی ارشد برق-الکترونیک، دانشگاه بیرجند، Ehsan.Yaqubi@mail.com
[2] استاد دانشکده مهندسی برق و کامپیوتر، دانشگاه بیرجند

چکیده

در این مقاله، یک ابزار CAD بر پایه‌ی ارتباط بین نرم‌افزار شبیه‌ساز اسپایس و الگوریتم فراابتکاری بهینه‌سازی صفحات شیبدار (IPO)، ارائه می‌شود که هدف آن انجام یک طراحی بهینه برای مقایسه‌گرهای قفلدار است. این ابزار با استفاده از مدل‌های دقیق ترانزیستورها و در نظر گرفتن کلیهٔ عناصر پارازیتیک، توانسته است به جواب‌هایی نزدیک به واقعیت دست یابد. ابزار پیشنهادی چندین هدف را به طور همزمان کمینه می‌کند و در نهایت یک دسته از جواب‌های بهینه در قالب جبهه‌ی پرتوی دو به دو را ارائه می‌دهد. بنابراین، طراح می‌تواند با توجه به کاربرد خاص خود و درک همبستگی بین توابع هدف، بهترین طراحی را انتخاب کرده و در انتها، با اجرای شبیه‌سازی مونت کارلو و بررسی صحت طراحی در گوشه‌های PVT، جواب‌های پایانی را مورد ارزیابی قرار دهد.

کلیدواژه: *مقایسه‌گر قفلدار ، طراحی با رایانه، سایزبندی ترانزیستورها، الگوریتم‌های فراابتکاری.*

مقدمه

در سال‌های اخیر، مهندسین سعی در استفاده از رایانه به عنوان یک ابزار برای بهتر کردن طراحی مدار و راستی‌آزمایی آن می‌کنند [۱]-[۳]. در حالی‌که پیش‌ازاین، روند طراحی-بازبینی-بازطراحی آنقدر صورت می‌گرفت تا زمانی که طراح بر اساس تجربه و دانش به بهترین طراحی نزدیک می‌شد [۴], [۵].

به دلیل پیشرفت روز افزون مدارات و وجود حالت‌هایی نظیر کار در زیر ولتاژ آستانه و یا حساسیت بیش از حد به عوامل پارازیتیکی [۶]، ضرورت استفاده از مدل‌های دقیق بیش از پیش به چشم می‌خورد. از همین روی، تلفیق الگوریتم‌های فراابتکاری -که در کمترین زمان جواب‌هایی کارا و دقیق ارائه می‌دهند [۷]- با نرم‌افزارهای شبیه‌ساز –که مدل‌های دقیقی را بکار می‌گیرند- می‌تواند طراح را به جواب‌هایی عملی و کارا نزدیک کند. در همین راستا مطالعات گسترده ای بر روی ابزارهای CAD برای مدارهای آنالوگ و دیجیتال شکل گرفته است که برخی از آن‌ها با ذکر ویژگی‌ها و کاربردهایشان در جدول ۱ آورده شده‌اند. برخی از این ابزارها [۵], [۸], [۹]، از الگوریتم‌های تک‌هدفه استفاده کرده اند. به عبارت دقیق تر، یک مسئله چندهدفه را به یک مسئله تک‌هدفه‌ی تجمیعی تبدیل کرده‌اند که در آن تابع برازندگی اصلی، مجموعی از توابع دیگر است که با توجه به اهمیت، به هر کدام یک ضریب داده شده است (یعنی $F_T=af_1+bf_2+cf_3$ که a و b و c وزن ها، و f_1 و f_2 و f_3 پارامترهای هدف هستند). عیب اصلی این گونه روش‌ها، شهودی بودن وزن‌ها و حساس بودن پاسخ‌ها به تغییرات وزن‌ها است. از طرف دیگر، اگرچه بهترین مقادیر برای تابع F_T پیدا شده است، اما نمی‌توان گفت که متعاقبا بهترین مقادیر برای تک تک f_1 تا f_3 را می‌توان بدست آورد. دسته‌ی دیگر این ابزارها [۱] ,[۴] ,[۱۰] ,[۱۱]، اگرچه روش‌های مفید و بعضا برنامه‌های کاربردی[1]‌ای را ارائه کرده اند، اما آن‌طور که ما در ادبیات تحقیق دیده‌ایم به طراحی از روی جبهه پرتو اشاره نکرده‌اند. در اکثر این روش‌ها، در تابع برازندگی برای پارامترهای بهینه‌شونده قیودی در نظر گرفته می‌شود [۱۲]، اگرچه تعیین این قیود می‌تواند یک جواب در محدوده‌ی مورد نظر را به طراح ارائه دهد، اما نمی‌تواند طراح را از بده-بستانی که این پارامترها بر یکدیگر دارند آگاه کند. علاوه بر این، با توجه به اینکه تعیین این قیود به صورت شهودی و بر پایه دانش طراح صورت می‌گیرد، نمی‌توانند بهترین جواب‌های ممکن باشند. ما در ابزار پیشنهادی با ارائه تمام پاسخ‌ها در قالب جبهه پرتو، و ترسیم آن توانسته‌ایم علاوه بر باز گذاشتن دست طراح در انتخاب بهترین جواب، وابستگی بین پارامترهای مدار مورد طراحی را به نمایش بگذاریم.

در این مقاله، ما ابزاری را برای یافتن بهترین سایزبندی ترانزیستورهای مقایسه‌گرهای قفلدار ارائه می‌کنیم، به گونه‌ای که بتوان به حداکثر پتانسیل موجود در معماری مورد نظر (یعنی رسیدن به بهترین مقادیر در مشخصه‌ها از قبیل: توان مصرفی، سرعت، سایز و...)، دست یافت. اهمیت این موضوع از این جهت است که برخی از مشخصه‌ها با یکدیگر در تعارض هستند [۱۳] و یافتن ارتباط دقیق بین این پارامترها در مداری که از فیدبک مثبت بهره می‌گیرد، -به دلیل اینکه در شرایط ناپایداری قرار دارد و ناحیه‌کاری ترانزیستورها با اطمینان قابل تعیین نیست- کار دشواری است.

[1] Application

تاریخ دریافت: ۱۳۹۵/۰۷/۱۳ تاریخ پذیرش: ۱۳۹۵/۱۲/۱۸

www.SID.ir

Archive of SID

جدول ۱. ابزارهای بهینه‌سازی مدارها

مرجع	الگوریتم مورد استفاده	نرم افزار شبیه	مدارهای بهینه شده	مشخصه‌ها و پارامترهای بهینه شده	مونت کارلو
[۱]	الگوریتم تکاملی تفاضلی (CODE)	Hspice	مدارات تقویت‌کننده	ابعاد ترانزیستورها- بهره- حاشیه فاز- پهنای باند-توان	خیر
[۱۱]	الگوریتم ژنتیک گسسته	Hspice	مقایسه‌گر پیوسته	تاخیر- سوئینگ خروجی- آفست- بهره- ولتاژ مدمشترک و...	خیر
[۹]	PSO	Hspice	تقویت‌کننده چهارطبقه	توان و بهره	خیر
[۱۰]	ژنتیک نخبه گرای مرتب شده	Hspice - SpectreRF	میکسر- VCO- LNA	عدد نویز- توان – IIP2 – S11	بر روی جواب‌های پایانی
[۴]	PSO و MOPSO	ADS	LNA	بهره ولتاژ – مقاومت ورودی پارازیتیکی – فرکانس قطع	خیر
[۸]	الگوریتم‌های استراتژی‌های تکاملی (ES) و تبرید شبیه‌سازی شده(SA)	Hspice	OTA کسکود تا شده و OTA دو طبقه	پهنای باند، بهره، حاشیه فاز، توان، فضای اشغالی، بازدهی، انحراف استاندارد، بازدهی، سایز نمونه، زمان سنتز، درصد خطا	دارد
[۵]	الگوریتم بهینه‌سازی تکاملی-تفاضلی	Spectre ،Hspice	تقویت کننده قدرت کلاس E	بازدهی درین (DE)، بازدهی توان اضافه شده (PAE)	خیر

از ویژگی‌های CAD حاضر، باز بودن کد برنامه[1] و همچنین تابع برازندگی است، لذا طراحان می‌توانند قابلیت‌های جدیدی را به ابزار اضافه کنند (به عنوان مثال، اضافه کردن قسمتی به برنامه‌ی نوشته شده جهت انجام خودکار تحلیل‌های مونت کارلو برای گوشه‌های دما، ولتاژ و پروسس[2] بر روی جواب‌های پایانی؛ اضافه کردن یک بانک از تمام معماری‌های مقایسه‌گرهای ارائه شده و جستجو برای یافتن بهترین معماری برای کاربرد مورد نظر؛ بکارگیری یک الگوریتم ابتکاری ثانی برای یافتن بهترین الگوریتم جستجویی که در آن مسئله خاص بهترین جواب‌ها را ارائه می‌دهد).

در ادامه‌ی این مقاله ابتدا نحوه‌ی عملکرد مقایسه‌گرهای قفلدار را تشریح می‌کنیم. سپس، در بخش ۳ در مورد الگوریتم‌های ابتکاری به ویژه الگوریتم بهینه‌سازی صفحات شیبدار چندهدفه صحبت خواهیم کرد. در بخش ۴، با ارائه دو مثال به معرفی ابزار CAD پیشنهادی می‌پردازیم و در نهایت در بخش ۵ بحث و نتیجه‌گیری را ارائه خواهیم کرد.

نحوه‌ی عملکرد مقایسه‌گر قفلدار

اخیراً، زیربلوک مقایسه‌گر قفلدار به عنوان یک جزء اساسی در مبدل‌های داده مورد توجه محققان بسیاری است. این زیربلوک می‌تواند ویژگی‌های اساسی یک مبدل آنالوگ به دیجیتال نظیر سرعت، توان، درجه تفکیک و دقت را تحت تاثیر قرار دهد[۱۴]. مقایسه‌گر قفلدار رایج شکل ۱ به‌صورت گسترده‌ای در مبدل‌های آنالوگ به دیجیتال استفاده می‌شود. این مقایسه‌گر دارای امپدانس ورودی بالا و نوسان خروجی تمام‌مقیاس است، درحالی‌که توان مصرفی ایستا ندارد. عملکرد این مقایسه‌گر به این صورت است که، در مرحله ریست (زمانی که CLK=Low است) ترانزیستورهای ریست (Mr1-Mr2) گره‌های خروجی (Outp و Outn) را به VDD نزدیک می‌کنند. در مرحله مقایسه، (زمانی که CLK=High) ترانزیستورهای ریست خاموش می‌شوند و ترانزیستور دنباله (Mt) روشن می‌شود. گره‌های خروجی که قبلاً به VDD شارژ شده بودند، با نرخ‌های متفاوتی -که توسط ورودی‌های INN و INP تعیین می‌شوند- شروع به تخلیه‌ی بار خود می‌کنند. فرض کنید INP>INN است، بنابراین گره Outp با سرعت بیشتری نسبت به Outn بار خود را تخلیه می‌کند. زمانی که Outp به |VDD-|Vthp برسد، ترانزیستور M1 روشن شده و ترانزیستورهای طبقه قفل (M1 تا M4) عملِ بازتولید را انجام خواهند داد. بنابراین Outn به مقدار VDD رسیده و Outp به زمین تخلیه خواهد شد. درصورتی‌که INP<INN باشد عکس این عمل صورت خواهد گرفت.

عیب این مقایسه‌گر وجود چندین ترانزیستور بین منبع تغذیه و زمین است که این امر منبع ولتاژ بزرگی را می‌طلبد [۱۵]. به عبارت دیگر، در ابتدای مرحله‌ی تصمیم فقط ترانزیستورهای M_3 و M_4 در عمل فیدبک مثبت شرکت می‌کنند و تا زمانی که سطحِ ولتاژ یکی از گره‌های خروجی افتِ کافی نکند، ترانزیستور M_1 یا M_2 روشن نخواهند شد. بنابراین، در منابع ولتاژِ سطحِ پایین این افتِ ولتاژ تاثیر اندکی بر ولتاژ سورس-گیت ترانزیستورهای M_3 و M_4 خواهد گذاشت که

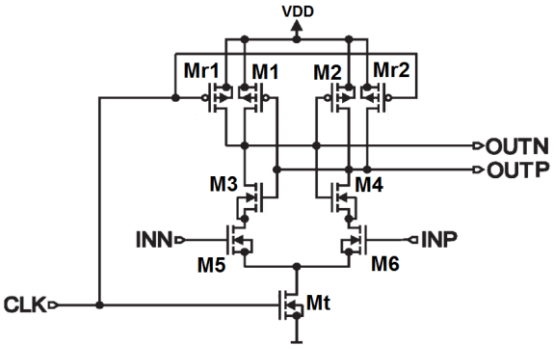
[1] Open source

[2] PVT corners

www.SID.ir

Archive of SID

متعاقباً، ولتاژ سورس-گیت M_1 و M_2 کوچک شده و سرعت مقایسه به شدت کم خواهد شد. عیب دیگر این مقایسه‌گر، وجودِ تنها یک جریان دنباله است که هر دو جریانِ طبقه قفل و طبقه‌ی ورودی را تعیین خواهد کرد. درحالی که، اگر طبقه‌ی تفاضلی (طبقه‌ی ورودی) در ناحیه‌ی زیر آستانه کار کند جریان کمی را مصرف خواهد کرد که این جریانِ کم، آفستِ ناچیزی در این طبقه ایجاد می‌کند. از طرف دیگر، طبقه قفل به جریان بالایی نیاز دارد تا بتواند با سرعت بالایی عمل باز تولید را انجام دهد. ضمنا، تا زمانی که ترانزیستور دنباله در ناحیه ترایود باشد، جریان آن به ولتاژ مد مشترک ورودی وابسته خواهد بود که این موضوع باعث تغییرات سرعت در عمل باز تولید خواهد شد [۱۶].



شکل۱. مقایسه‌گر قفلدار رایج [15]

برای رفع عیوب ذکر شده، ساختار مقایسه‌گر دودنباله برای اولین بار در [۱۶] و با ایده‌ی ایجاد استقلال میان جریان‌های طبقه قفل و طبقه‌ی ورودی، ارائه شد. پس از آن، مقالاتی مانند [۱۷]–[۲۶] با ارائه مقایسه‌گرهایی سعی در کاهش ولتاژ آفست، افزایش سرعت و کاهش نویز داشتند.

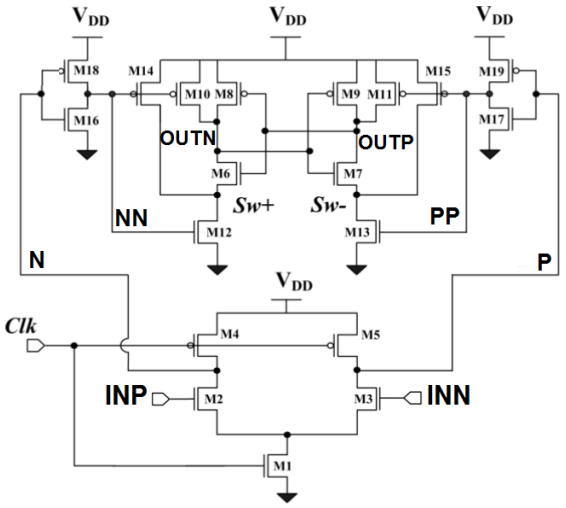
شکل۲ مقایسه‌گر پیش تقویت‌کننده‌ی بهینه‌شده در این مقاله را نشان می‌دهد. این مقایسه‌گر در دو مرحله ریست و مقایسه عمل می‌کند. در مرحله ریست (زمانی که CLK=low)، ترانزیستورهای M_5-M_4 روشن هستند. بنابراین، گره‌های P و N به V_{DD} شارژ می‌شوند که این باعث می‌شود تا ترانزیستورهای M_{17}-M_{16}، روشن شده و گره‌های NN و PP به زمین تخلیه شوند. در نهایت، ترانزیستورهای ریست (M_{10}، M_{11}، M_{14} و M_{15}) گره‌های خروجی (OUTN و OUTP) و گره‌های Sw را به VDD شارژ می‌کنند تا قفل خروجی ریست شود. در مرحله مقایسه (زمانی که CLK=high)، ترانزیستورهای ریست که در مرحله قبل بررسی شدند، همگی خاموش هستند. برای توضیح عملکرد مقایسه‌گر در این مرحله ، فرض کنید که INN>INP است، پس گره‌های N و P از VDD با سرعت‌های متفاوتی به زمین تخلیه می‌شوند (چون INN>INP است پس گره P با سرعت بیشتری تخلیه خواهد شد). بنابراین، معکوس‌کننده‌های M_{19}-M_{17} و M_{16}-M_{18}، باعث خواهند شد که ولتاژ گره PP نسبت به ولتاژ گره NN افزایش سریع‌تری داشته باشد، که این خود باعث می‌شود ولتاژ گره Sw- نسبت به Sw+ زودتر کاهش یابد، در نهایت، ولتاژ گره OUTP نسبت به OUTN کاهش سریع‌تری دارد، و پس از اینکه ترانزیستورهای قفل همگی روشن شوند، OUTP با سرعت بیشتری کاهش یافته و OUTN به VDD خواهد رسید.

الگوریتم‌های ابتکاری

به طور کلی الگوریتم‌های شبیه‌سازی به دو دسته الگوریتم‌های دقیق و الگوریتم‌های تقریبی تقسیم می‌شوند. الگوریتم‌های دقیق، در مواجهه با مسائل مشکل که دارای فضای جستجوی بزرگی هستند کارایی ندارند، درحالی که روش‌های تقریبی می‌توانند جواب‌های نزدیک به بهینه کلی را در کوتاه‌ترین زمان بیابند. الگوریتم‌های تقریبی را می‌توان در سه دسته‌ی ابتکاری (heuristic)، فراابتکاری (meta-heuristic) و فوق ابتکاری (hyper heuristic) [۲۸] قرار داد. الگوریتم‌های فراابتکاری با توجه به سه معیار،

- مبتنی بر یک جواب [۲۹] – [۳۲] و مبتنی بر جمعیت [۳۳]–[۳۶].
- الهام گرفته شده از طبیعت [۳۷] ,[۳۸] یا بدون الهام از طبیعت[۳۲].
- دارای حافظه [۳۵] یا بدون حافظه [۳۰].

دسته بندی می‌شوند.



شکل۲. ساختار مقایسه‌گر دارای پیش‌تقویت‌کننده‌ی بهینه‌شده در این مقاله، برگرفته از [۲۷].

الگوریتم بهینه‌سازی صفحات شیبدار که با نام IPO شناخته می‌شود، الهام گرفته از حرکت اجسام کروی بر روی سطوح شیبدار و بدون اصطکاک است. عوامل جستجو در این الگوریتم، توپ‌ها هستند. ایده اصلی، تمایل توپ‌ها به از دست دادن انرژی پتانسیل و رسیدن به کمترین ارتفاع است. میزان برازندگی توپ‌ها نسبت به ارتفاعی که به آن‌ها نسبت داده شده‌است

www.SID.ir

Archive of SID

سنجش می‌شود. مقادیر ارتفاع، تخمینی از انرژی پتانسیل توپ‌ها که در ارتفاعات مختلفی قرار دارند، بوده و با پایین آمدن این توپ‌ها انرژی پتانسیل به جنبشی تبدیل شده و توپ‌ها به سمت پایین شتاب می‌گیرند. در واقع توپ‌ها برای رسیدن به نقطه مینیمم تمایل به از دست دادن انرژی پتانسیل خود دارند. بنابراین، توپ‌ها برای جستجوی بهترین پاسخ در حال حرکت هستند و به تناسب، شتابی را به خود اختصاص می‌دهند [۳۷].

در الگوریتم چندهدفه‌ی بهینه‌سازی صفحات شیبدار که با نام اختصاری MOIPO شناخته می‌شود از روش بهینگی پرتو برای شناسایی پاسخ‌های غیرغالب استفاده شده و یک مخزن بیرونی، نگهداری این پاسخ‌ها را بر عهده دارد. ابتدا جمعیت اولیه با توجه به محدوده‌ی مشخص شده و به‌صورت تصادفی ایجاد می‌شود. سپس، برازندگی جمعیت محاسبه شده و از میان آن‌ها، بهترین توپ‌ها در یک مخزن بیرونی -شامل پاسخ‌های پرتو- نگهداری می‌شود. پس از آن، با توجه به الگوریتم IPO، موقعیت هر توپ برای تکرار بعدی الگوریتم بروزرسانی می‌شود. الگوریتم MOIPO شامل مراحل زیر است [۳۹].

۱- مشخص نمودن ابعاد مسئله (nvar) و توپ‌های موجود در فضای پاسخ (npop) و تعداد حجم مخزن (nRep).
۲- مقداردهی پارامترهای IPO که شامل مقداردهی به c_1، c_2، $shift_1$، $shift_2$، $scale_1$ و $scale_2$ می‌باشد.
۳- ایجاد جمعیت اولیه
۴- تنظیم سرعت و شتاب اولیه‌ی هر یک از توپ‌ها به مقدار صفر
۵- ارزیابی برازندگی هر یک از توپ‌ها توسط توابع برازندگی
۶- اعمال شروط بهینه‌سازی پرتو و ذخیره‌سازی موقعیت توپ‌های غیرغالب در یک مخزن و ایجاد ابرمکعب و مشخص کردن تعداد نقاط در هر مکعب
۷- ایجاد حلقه‌ی جستجوی for برای T مرتبه تکرار که شامل موارد زیر می‌شود:

- محاسبه‌ی شتاب، سرعت و موقعیت‌ها. بازگرداندن توپ‌هایی که از بازه‌ی فضای جستجو خارج شده‌اند به داخل فضای جستجو
- ارزیابی برازندگی هر یک از توپ‌ها توسط توابع برازندگی
- بروزرسانی محتویات مخزن که شامل قرار دادن همه‌ی توپ‌های غیرغالب شده‌ی فعلی در مخزن می‌شود. به‌طور همزمان هر توپِ غالب نشده در فرایند، از مخزن حذف می‌شود.
- کنترل حجم مخزن: از آنجایی که ظرفیت مخزن محدود می‌باشد، هر زمان که ظرفیت مخزن از حد نصاب گذشت، به هر ابرمکعب یک احتمال –که با تعداد توپ در هر ابرمکعب متناسب باشد- نسبت می‌دهیم. سپس، از طریق چرخ رولت یک ابرمکعب انتخاب می‌شود و به‌طور تصادفی نقطه‌ی مازاد حذف می‌گردد. این روند تا زمانی که حجم مخزن به حد نصاب برسد ادامه می‌یابد.
- پایان حلقه[۳۹].

طراحی سایزبندی ترانزیستورها با استفاده از الگوریتم‌های ابتکاری

برای انجام یک طراحی، ابتدا یک مدل دقیق از نظر فرکانسِ کاری و ناحیه‌ی کاری ترانزیستورها مورد نیاز است. پس از آن، می‌توان طراحی را با انجام محاسبات دستی و با اعمال تقریب‌هایی انجام داد. البته، با پیچیده‌تر شدن روز افزون مدارات و تشدید شدن برخی مشکلات، طراحی با رایانه بیشتر مورد استقبال قرار گرفته است. این مشکلات عبارتند از:

۱- حساس بودن مقایسه‌گرهای سرعت بالا به عناصر پارازیتیکی: درمقایسه‌گرهایی که در فرکانس‌های بالاتر از چند گیگاهرتز کار می‌کنند، خازن‌های پارازیتیکی باید در مدلِ ترانزیستورها لحاظ شوند تا تاثیرات آن‌ها بر میزانِ نویز، آفست و دیگر مشخصه‌ها بررسی شود. اضافه شدن این عناصر به مدل ترانزیستورها، محاسبات را بسیار پیچیده و چالش‌برانگیز می‌کند.
۲- دقیق نبودن روابط طراحی: برای پیدا کردن مهمترین عوامل دخیل، ناگزیر هستیم که در محاسبات از تقریب استفاده کنیم. اگرچه با این عمل، موثرترین پارامترها شناسایی می‌شوند، اما نمی‌توان با اطمینان اذعان نمود که به بهترین طراحی دست یافته‌ایم.
۳- نبودن مدل‌های دقیق از قطعات پسیو مجتمع: مدل‌های تقریبی، به روابط دارای تقریب منجر می‌شود که متعاقباً به نتایج دقیقی نخواهیم رسید.
۴- مشکل بودن محاسبات دستی: مدلسازی ترانزیستورها با توجه به ناحیه‌ی کاری آن‌ها و فرکانس کاری مدار، صورت می‌گیرد که این خود می‌تواند پیچیدگی محاسبات دستی را بیشتر کند.
۵- نیاز به سعی و خطای برای رسیدن به طراحی قابل قبول: پس از انجام طراحی دستی، طراح توسط شبیه‌سازی عملکرد مدار را بررسی می‌کند. در غالب اوقات، پس از طراحی دستی و بر اساس محاسبات انجام شده، برای رسیدن به جواب‌های بهتر باید به سعی و خطا پرداخت.
۶- وجود فیدبک مثبت و پاسخ‌های حالت گذار: انجام محاسبات دستی برای تحلیل عملکرد مقایسه‌گر در حالت‌های گذرا بسیار پیچیده و گاها ناممکن می‌شود.
۷- متغیر با زمان بودن نقاط کار ترانزیستورها: با توجه به اینکه مقایسه‌گر دائماً بین دو مرحله ریست و مقایسه در حال کار

www.SID.ir

Archive of SID

است، ناحیه‌کاری ترانزیستورها نیز در حال تغییر خواهد بود. لذا، محاسبات دستی فقط با فرض اینکه مدار به پایداری کامل رسیده است، دارای اعتبار خواهند بود.

از طرف دیگر، با توجه به اینکه شرکت‌های سازنده‌ی مدارات مجتمع، کتابخانه‌هایی دقیق را از المان‌های مداری ارائه کرده‌اند، می‌توان با استفاده از نرم‌افزارهای شبیه‌ساز، طراحی مقایسه‌گر را با اطمینان و بدون تقریب انجام داد. در ادامه، با استفاده از ایجاد ارتباط بین نرم‌افزار Hspice و الگوریتم‌های فراابتکاری سعی در رسیدن به یک طراحیِ بهینه برای مقایسه‌گر ولتاژ داریم. در این طراحی بهینه، سایزبندی ترانزیستورها را به گونه‌ای تعیین می‌کنیم که مشخصه‌های ضروری -از جمله توان مصرفی، سرعت و سایز مقایسه‌گر- به بهترین مقادیر خود نزدیک شوند.

فرایند پیوند یک الگوریتم فراابتکاری با شبیه‌ساز اسپایس با هدف طراحی مدار مقایسه‌گر ولتاژ

اساس کار الگوریتم‌های فراابتکاری کمینه (یا بیشینه) کردن یک یا چند تابع برازندگی[1] (تابع هدف[2]) است. هنگامی که هدف، کمینه (یا بیشینه) کردن یک تابع ریاضی باشد، تابع برازندگی، همان تابع ریاضی خواهد بود. اما در اینجا، ما از الگوریتم‌های فراابتکاری در بهینه‌سازی یک مدار استفاده خواهیم کرد. بنابراین تابع برازندگی نیاز به تعریف خواهد داشت.

در حالت کلی می‌توان گفت هر ابزار CAD از دو قسمت الگوریتم جستجو و نرم افزار شبیه‌ساز تشکیل شده است، بنابراین یکی از مهمترین مسائل در استفاده‌ی کارآمد از الگوریتم‌های فراابتکاری، تعریف یک تابع برازندگی مناسبِ مسئله است.

عوامل جستجو[3]: عوامل جستجو در الگوریتم‌های فراابتکاری با نام‌های متفاوتی نظیر ذرات (الگوریتم گروه ذرات)، کروموزوم (الگوریتم ژنتیک) ، آنتی بادی (الگوریتم ایمنی مصنوعی) ، توپ (الگوریتم صفحات شیبدار) و ... شناخته می‌شوند. ساختار عوامل جستجو برای دو مثال در جداولی آورده شده‌اند که در ادامه بررسی خواهند شد. ساختار عوامل جستجو، از پارامترهایی که می‌خواهیم برای آن‌ها مقادیر بهینه‌ای پیدا کنیم، تشکیل شده است. به عنوان مثال، پهنای ترانزیستورها، سطح منابع تغذیه، میزان نوسان ولتاژ خروجی و ولتاژهای بایاس، پارامترهایی هستند که می‌توانند ساختار عوامل جستجو را شکل دهند (در این مقاله فقط پهنای ترانزیستورها را در ساختار عوامل جستجوی وارد کرده ایم).

تابع برازندگی در ابزار پیشنهادی: هدف از تابع برازندگی، ارزیابی برازندگیِ هر یک از عوامل جستجو است، (بهترین عوامل جستجو، بهترین طراحی‌ها را در قالب جبهه پرتو بدست خواهند داد). لازم به ذکر است که تابع برازندگی برای هر مدار، مختص همان مدار است. بنابراین برای اینکه خواننده بتواند ابزار معرفی شده را برای بهینه‌سازی مدارات دلخواه بکار بگیرد، در بخش‌های آتی، دو مدار را از ابتدا تا تحلیل نتایج آن‌ها مورد بررسی قرار می‌دهیم.

برنامه الگوریتم‌های استفاده شده (NSGAII و MOIPO) که در ادامه مورد بحث قرار می‌گیرند به زبان Matlab نوشته شده اند. در فرایند پیوند، از نرم افزارهای Matlab 2011a و Hspice 2008 استفاده شده و مشخصات سیستم به مورد استفاده در جدول ۲ آورده شده است[4].

جدول۲. مشخصات سیستم استفاده شده برای بررسی عملکرد ابزار CAD پیشنهادی

قطعه یا نرم افزار	مشخصات
Processor	Intel® Core™i5-4210U CPU@ 1.70 GHz 2.40 GHz
RAM	6.00 GB (5.90 GB usable)
System Type	64-bit operating system, x64-based processor
Windows	Windows 8.1 pro

در ادامه، ابتدا پیش تنظیمات و ملاحظاتی که در هنگام طراحی باید رعایت شود را ذکر می‌کنیم. سپس فرایند طراحی و چگونگی تبادل اطلاعات بین یک الگوریتم فراابتکاری و اسپایس را به تفسیر بیان خواهیم کرد. سپس، دو مثال کاربردی را با استفاده از ابزار CAD پیشنهادی بهینه خواهیم کرد.

ملاحظات پیوند الگوریتم فراابتکاری و شبیه ساز اسپایس

برای ایجاد پیوند میان یک الگوریتم فراابتکاری و اسپایس و بدست آوردن نتایجی صحیح و کاربردی در کمترین زمان ممکن، رعایت چند نکته در بدنه‌ی برنامه نویسی تابع برازندگی ضروری است:

۱- پهنای ترانزیستورها، پارامترهایی هستند که قرار است بگونه‌ای انتخاب شوند تا توابع هدف، (توان، ولتاژ آفست، تاخیر، سایز و ...) به بهترین مقادیر خود برسند. بنابراین، این پارامترها باید در یک بازه‌ی عملی محدود شوند تا طراحی بدست آمده قابلیت ساخت داشته‌باشد.

۲- برای کاهش زمان طراحی، لازم است هر جفت از ترانزیستورهای طبقه‌ی تفاضلی به عنوان یک عامل جستجو در نظر گرفته شود. به عنوان مثال M_2-M_3 به عنوان یک توپ و M_{12}-M_{13} به عنوان توپ بعدی (به عنوان یک مثال، جدول ۳ را ببینید) انتخاب شوند.

۳- در برخی اوقات به دلیل ایجاد شدن نتلیست‌های غیرعملی از طرف الگوریتم فراابتکاری، شبیه‌ساز اسپایس اعلام

[1] Fitness Function
[2] Cost Function
[3] Search agent
[4] توابع برازندگی و الگوریتم‌های استفاده شده، برای دانلود در آدرس https://www.researchgate.net/profile/Ehsan_Yaqubi قرار داده شده است.

www.SID.ir

Archive of SID

خطا کرده[1] و فایل خروجی‌ای را ایجاد نخواهد کرد. در این موارد، اختصاص دادن یک برازندگیِ بسیار بد به نت‌لیستِ مربوطه، به الگوریتم جستجو کمک کند تا بتواند قسمت‌های بهتر فضا را پیمایش کند.

۴- در صورتی که بخواهیم در طراحی، ولتاژ آفست را نیز مد نظر قرار دهیم، می‌توانیم به این صورت عمل کنیم: ابتدا انحراف استاندارد پارامترهای ماسفت را به صورت زیر تعریف می‌کنیم.

```
.param Lmin='0.2u'
.param Awn='0.00373u'
.param Awp='0.00326u'
.param ALn='0.00458u'
.param ALp='0.00396u'
.param AVtn='0.03635u'
.param AVtp='0.04432u'
```

سپس برای هر یک از پارامترهای W، L و ولتاژ آستانه (مدل کردن ولتاژ آستانه به صورت شکل ۳ انجام می‌شود)، یک گوسی به شکل زیر تعریف می‌کنیم:

```
.param w1 = gauss('w01', 'Awn/sqrt(w01*Lmin)', 3)
.param L1 = gauss('Lmin', 'ALn/sqrt(w01*Lmin)', 3)
.param dVt1 = agauss('0', 'AVtn/sqrt(w01*Lmin)', 3)
```

که:

```
Vth1 gate1 node9 dc= dVt1
```

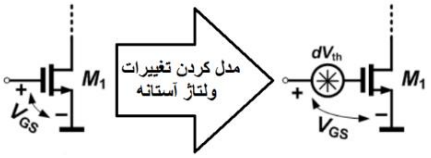
توجه داریم که w01 از سوی الگوریتم فراابتکاری مقدارگذاری می‌شود. در ادامه، یک سیگنال با شیب بسیار کم به عنوان ورودی به مقایسه‌گر اعمال می‌کنیم، به عنوان مثال،

```
VINp node2 gnd dc=700m
VINn node1 gnd pulse(600m 800m .1f 24n 2n 2n 24n)
```

سپس برای بدست آوردن آفست، کلاک را طوری تنظیم می‌کنیم که وقتی سیگنال تفاضلی ورودی به مقدار صفر رسید، لبه‌ی بالارونده‌ی کلاک اعمال شود. حال در زمانی که خروجی به نصف مقدار نهایی خود برسد، سیگنال تفاضلی ورودی را اندازه گیری می‌کنیم، این مقدار معرف ولتاژ آفست خواهد بود. برای این کار از دستورالعمل زیر در Hspice استفاده کرده‌ایم:

```
.measure tran Pdelay1 when v(outn,outp)='+400m' rise=1
```

در هنگام تحلیل فایل خروجی (در تابع برازندگی)، سیگنال خروجی را در مقابل سیگنال ورودی در هر شبیه‌سازی، برای اطمینان از درست کار کردن مقایسه‌گر بررسی می‌کنیم. به عبارت دیگر، وقتی سیگنال تفاضلیِ ورودی مثبت است، باید سیگنال تفاضلیِ خروجی نزدیک به $+V_{DD}$ باشد و زمانی که سیگنال تفاضلیِ ورودی منفی است، سیگنال تفاضلیِ خروجی باید نزدیک به $-V_{DD}$ باشد. درصورتی‌که این شرایط برقرار نباشد، باید به نت‌لیست مورد نظر یک برازندگی بد اختصاص داد تا الگوریتم بهینه ساز در نقاط بهتری از فضا به جستجو ادامه دهد.



شکل۳. مدل کردن تغییرات ولتاژ آستانه با هدف اجرای شبیه‌سازی مونت کارلو. dV_{th} یک منبع گوسی با میانگین صفر و انحراف استاندارد گزارش شده از سوی شرکت سازنده است.

فرایند طراحی و تبادل اطلاعات بین الگوریتم‌های فراابتکاری و اسپایس

نمودار روندنمای شکل۴ فرایند تبادل اطلاعات میان الگوریتم‌های فراابتکاری و اسپایس را نشان می‌دهد. خروجی‌های این فرایند می‌تواند یک یا چند جبهه‌ی پرتو (مثال‌هایی از این جبهه‌های پرتو در ادامه آمده است) باشد که نشان‌دهنده‌ی همبستگی بین توابع هدف (نظیر توان مصرفی، ولتاژ آفست، تاخیر، سایز، ولتاژ مدمشترک ورودی و ...) می‌باشد. ضمناً، هر نقطه در این جبهه‌های پرتو یک طراحی بهینه برای مقایسه‌گر را نشان می‌دهند.

نمودار روندنمای شکل۴ به پنج مرحله کلی تقسیم‌بندی می‌شود که عبارتند از: پیش آماده‌سازی ابزار CAD (تهیه نت‌لیست مدار)، تعریف مسئله، مقدمه‌سازی الگوریتم فراابتکاری، حلقه اصلی و چاپ نتایج. در مرحله تعریف مسئله، قیود و شرایط پارامترهای جستجو، تعداد متغیرهای جستجو (ابعاد مسئله) و محدوده‌ی تغییرات مجاز متغیرها (در اینجا پهنای ترانزیستورها به عنوان متغیرها هستند) مشخص می‌شوند، سپس عمل نوشتن در فایل نت لیست، اجرای اسپایس و خواندن فایل خروجی یکبار انجام می‌شود. پس از آن، در مرحله‌ی مقدمه‌سازی الگوریتم ابتکاری، پارامترهایی از قبیل جمعیت، تکرار، حجم مخزن و ... مقداردهی می‌شوند. در مرحله بعدی، وارد یک حلقه تکرار شونده خواهیم شد که در آن به طور مداوم نت‌لیست‌هایی به صورت تصادفی که متاثر از تکرارهای قبلی است، ایجاد شده (تفاوت این نت‌لیست‌ها در پهنای ترانزیستورهای آن‌ها است) و پس از شبیه‌سازی در اسپایس، از لحاظ توان مصرفی، تاخیر و سایزِ ترانزیستورها، مورد ارزیابی قرار می‌گیرند و جواب‌های بهتر در مخزن مرتب می‌شوند. در همین حین برای گریز از بهینه‌های محلی، نرخ اکتشاف[2] و استخراج[3] بازتنظیم می‌شوند. در نهایت، پس از ارضاء شدن شرط اتمام حلقه -که می‌تواند؛ رسیدن به تعداد تکرارهای تعیین شده، یا رسیدن به یک پاسخ مطلوب و از قبل تعیین شده، یا رسیدن به زمان از قبل تعیین شده، باشد-

[1] Job aborted
[2] Exploration
[3] Exploitation

www.SID.ir

Archive of SID

نتایج پایانی علاوه بر ذخیره در یک جدول، در قالب جبهه‌های پرتوی دو به دو، ترسیم می‌شوند.

طراحی سایزبندی مقایسه‌گر قفلدار ساده

به عنوان نمونه اول، با استفاده از ابزار پیشنهادی سعی در ارائه‌ی یک طراحی بهینه برای مقایسه‌گرِ شکل۱ داریم.

مقایسه‌گر مذکور، توسط پیوند Hspice (کتابخانه‌ی TSMC 180nm) سه بار با سیستم چندهدفه‌ی IPO با ۵۰ توپ (جمعیت) و ۱۸۰ تکرار، ۱۰۰ توپ و ۳۶۰ تکرار و ۱۵۰ توپ و ۵۰۰ تکرار بهینه‌سازی شد. تاخیر، توان مصرفی و سایز کلی ترانزیستورها (مجموعِ W-های ترانزیستورها) به ترتیب به عنوان اهدافِ اول، دوم و سوم انتخاب شدند. ساختار عوامل جستجو (توپ‌ها در IPO) نیز به صورت جدول ۳ می‌باشد. ضمناً، برای اینکه سایز ترانزیستورهای بدست آمده قابل ساخت باشد، محدوده‌ی جستجو برای پهنای ترانزیستورها بین $2*L_{min}$ تا $300*L_{min}$ درنظر گرفته شده‌است. هم‌چنین طول کانالِ تمامی ترانزیستورها یکسان و برابر با $L_{min}=180nm$ درنظر گرفته شد. در نهایت پس از بکارگیری ابزار پیشنهادی، جبهه‌های پرتوی شکل۵ بدست آمد. به دلیل اینکه بهینه‌سازی انجام شده، سه هدفه است، جبهه پرتو به صورت سه بعدی خواهد شد، اما به دلیل درک بهتر نوع وابستگی بین مشخصه‌های توان مصرفی، تاخیر و سایز کلی ترانزیستورها -که به عنوان توابع هدف انتخاب شده‌اند-، جبهه‌های پرتو به صورت دو به دو رسم شده‌اند.

با بررسی جبهه‌های پرتو به صورت دو به دو می‌توان رابطه‌ی بین مشخصه‌های هدف را تشخیص داد. در شکل ۵، با توجه به جبهه‌های پرتوی ستون اول (توان در مقابل تاخیر)، می‌توان دریافت که توان و تاخیر با یکدیگر بده-بستان دارند. یعنی در صورتی که بخواهیم یک مقایسه‌گر با سرعت بالا داشته باشیم باید توان مصرف کنیم، و برعکس آن؛ یعنی اگر بخواهیم یک مقایسه‌گر با توان مصرفی پایین داشته باشیم ناگزیر سرعت مقایسه‌گر کاهش پیدا خواهد کرد. بررسی ستون سوم (توان در مقابل سایز کلی ترانزیستورها) نشان می‌دهد که این دو مشخصه با یکدیگر یک رابطه خطی و مثبت دارند. یعنی در صورت افزایش سایز ترانزیستورها، توان مصرفی نیز به صورت خطی افزایش خواهد داشت [۴۰].

برای بررسی قابلیت اطمینان پاسخ‌های بدست آمده در جبهه‌های پرتو، بر روی برخی از پاسخ‌های پایانی یک شبیه‌سازی مونت کارلو با ۱۰۰۰ اجرا صورت گرفت، در حالی که $\Delta V_{in}=200mv$ و فرکانس کلاک 50 MHz در نظر گرفته شدند؛ نتایج بدست آمده از شبیه‌سازی‌های مونت کارلوی انجام شده در جدول ۴ آورده شده اند.

برای بررسی صحت طراحی در گوشه‌های PVT، پنج شبیه‌سازی TT، FF، SS، FS، SF[1] انجام شد، در حالی که $\Delta V_{in}=200mv$ و فرکانس کلاک 50 MHz در نظر گرفته شدند. نتایج این تجزیه و تحلیل در جدول ۵ آورده شده است.

طراحی سایزبندی ترانزیستورهای یک مقایسه‌گر دارای پیش تقویت‌کننده

مقایسه‌گر پیش تقویت‌کننده شکل۲ به عنوان مدار دوم برای طراحی انتخاب شد. در این طراحی، ساختار عوامل جستجو به صورت جدول۶ درنظر گرفته شدند. مقایسه‌گر مذکور، توسط پیوند Hspice (کتابخانه‌ی TSMC 180nm) و یک سیستم چندهدفه‌ی IPO با ۱۰۰ توپ و ۳۶۰ تکرار بهینه‌سازی شد. در تابع برازندگی برنامه‌نویسی‌شده برای این مدار، مشخصه‌های توان مصرفی، تاخیر و مجموع سایز ترانزیستورها به عنوان مشخصه‌های هدف تعیین شده‌اند. مانند مثال قبل، محدوده‌ی جستجو برای پهنای ترانزیستورها بین $2*L_{min}$ تا $300*L_{min}$ محدود شده‌است. هم‌چنین طول کانالِ تمامی ترانزیستورها یکسان و برابر با $L_{min}=180nm$ درنظر گرفته شد.

با بررسی ظاهری جبهه‌های پرتوی بدست آمده برای مقایسه‌گر دارای پیش تقویت‌کننده (شکل ۶)، مشاهده می‌شود که توان با سایز یک رابطه خطی مثبت داشته در حالی که، تاخیر با سایز، و توان با تاخیر، رابطه تقریبا خطی و منفی دارند.

جدول۳. ساختار عوامل جستجو (توپ‌ها در IPO) برای مقایسه‌گر قفلدار ساده

پارامترهای بهینه شونده	ساختار توپ
پهنای M_t	ت۱
پهنای M_1-M_2	ت۲
پهنای M_3-M_4	ت۳
پهنای M_5-M_6	ت۴
پهنای M_{r1}-M_{r2}	ت۵

جدول۴. ۱۰۰۰ اجرای شبیه سازی مونت کارلو بر روی دو مورد از پا سخ‌های جبهه پرتوی مربوط به مقایسه‌گر قفلدار ساده (پاسخ اول کمترین تاخیر و پاسخ دوم کمترین توان مصرفی را دارد)

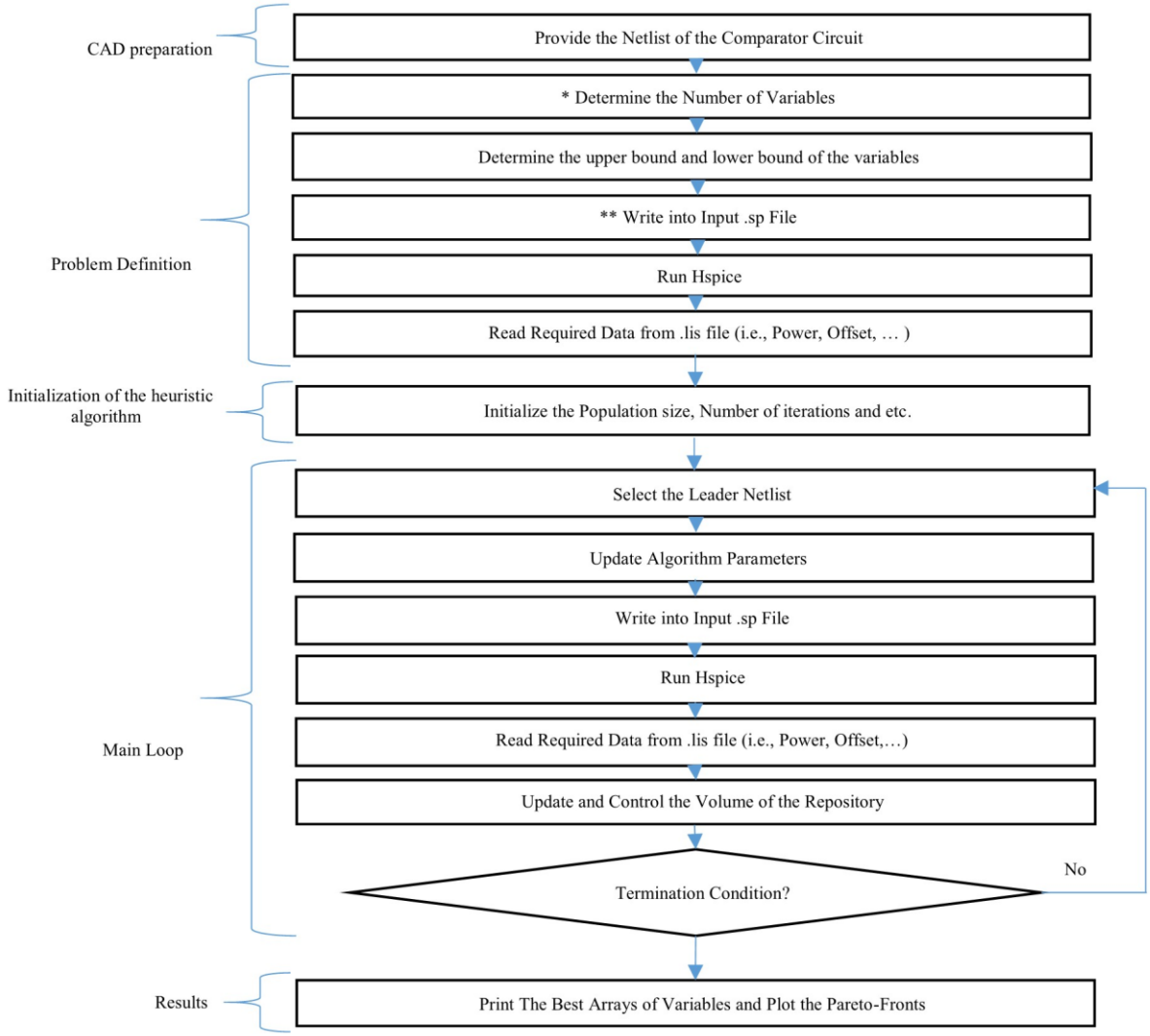
برخی از پاسخ‌های جبهه پرتو	توان مصرفی		تاخیر	
	۱-سیگما	میانگین	۱-سیگما	میانگین
$W_{1,2}$=9.09um $W_{3,4}$=12.57um $W_{5,6}$=29.53um $W_{r1,r2}$=0.36um W_t=20.54um	1.02e-8	1.51e-5	8.5e-15	7.0e-11
$W_{1,2}$=0.36um $W_{3,4}$=0.36um $W_{5,6}$=0.36um $W_{r1,r2}$=0.36um W_t=0.39um	7.9e-8	5.88e-7	9.0e-13	1.0e-10

[1] Typical Slow, T: Fast, S: F:

www.SID.ir

Archive of SID

در جبهه‌های پرتوی شکل ۶، برای بررسی صحت طراحی در گوشه‌های PVT . بر روی دو پاسخ (یعنی، دو مورد از طراحی‌های ارائه شده در جبهه پرتو)، پنج شبیه‌سازی TT، FF، SS، FS، SF انجام شد که نتایج آن در جدول ۷ آمده است. در ادامه برای بررسی قابلیت اطمینان این پاسخ‌ها، شبیه‌سازی مونت کارلو با ۱۰۰۰ اجرا انجام شد (جدول ۸). تنظیمات مدار در این شبیه‌سازی‌ها به این صورت بود که: ΔV_{in}=200mv و فرکانس کلاک 50 MHz در نظر گرفته شدند.



* در این مرحله تعداد متغیرهایی که می‌خواهیم برای آنها مقادیر بهینه‌ای پیدا کنیم، را وارد می‌کنیم.

** در این مرحله نت‌لیست مدار در مطلب خوانده شده و مقادیری تصادفی برای متغیرها، با توجه به محدوده تعیین شده تعیین می‌شود

• ضمنا، منظور از Variable، طول و پهنای کانال هر یک از ترانزیستورها، سطح منبع تغذیه و فرکانس کاری مدار است.

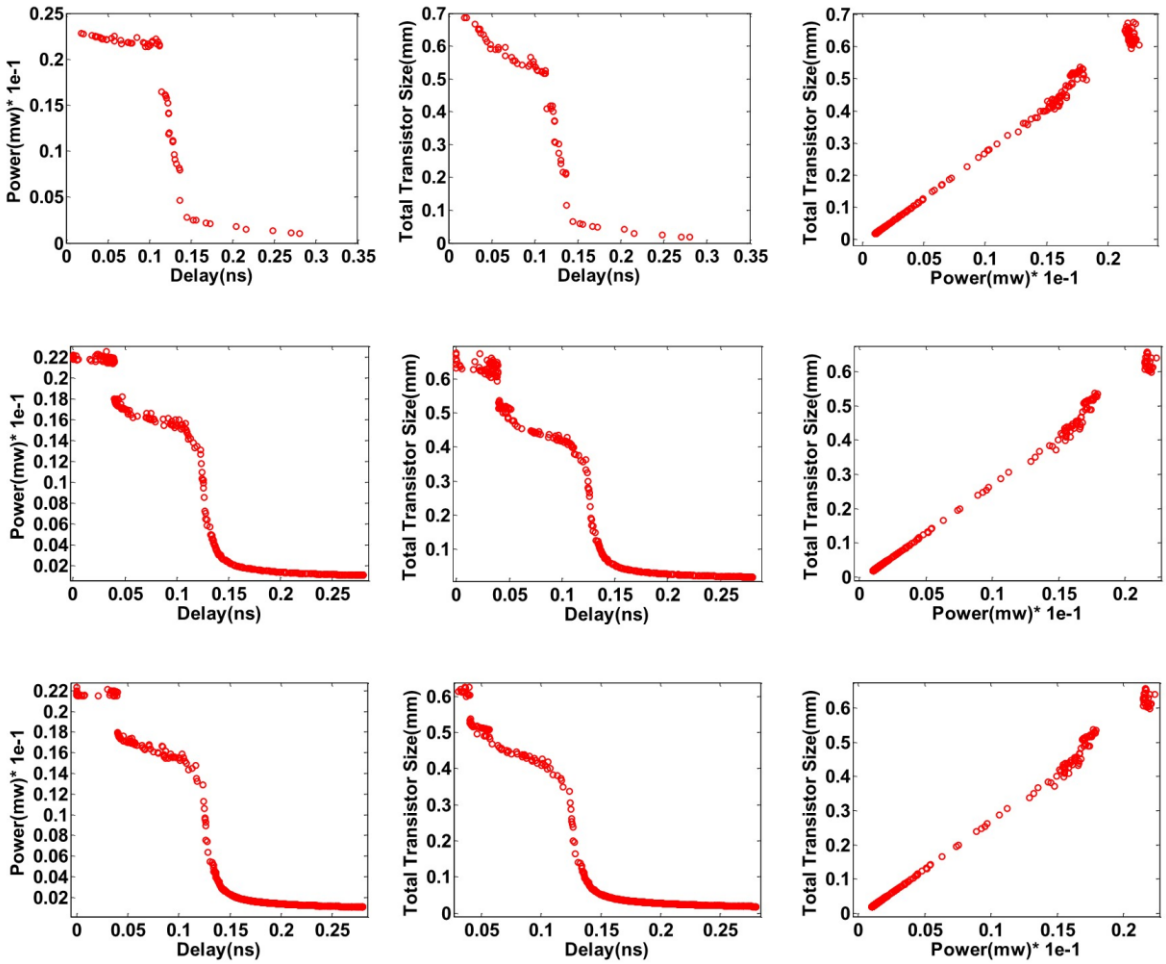
شکل۴. نمودار روندنمای پیوند میان Hspice و الگوریتم فراابتکاری

جدول۵. تجزیه و تحلیل گوشه‌های PVT برای مقایسه‌گر قفلدار ساده

SF		FS		SS		FF		TT		دو پاسخ منتخب از جبهه پرتو
تاخیر	توان	تاخیر	توان	تاخیر	توان	تاخیر	توان	تاخیر	توان	
2.20e-8	1.54e-5	2.13e-8	1.72e-5	2.24e-8	1.53e-5	2.10e-8	1.74e-5	2.15e-8	1.64e-5	$W_{1,2}$=9.09um $W_{3,4}$=12.57um $W_{5,6}$=29.53um $W_{r1,r2}$=0.36um W_t=20.54um
2.01e-8	4.34e-7	2.0e-8	4.71e-7	2.01e-8	4.11e-7	2.0e-8	4.95e-7	2.0e-8	4.49e-7	$W_{1,2}$=0.36um $W_{3,4}$=0.36um $W_{5,6}$=0.36um $W_{r1,r2}$=0.36um W_t=0.39um

www.SID.ir

Archive of SID



شکل۵. جبهه‌های پرتو بدست آمده از بهینه‌سازی مقایسه‌گر قفلدار ساده ردیف اول) برای سیستم IPO چند هدفه با جمعیت ۵۰ و ۱۸۰ تکرار. ردیف دوم) برای سیستم IPO چند هدفه با جمعیت ۱۰۰ و ۳۶۰ تکرار. ردیف سوم) برای سیستم IPO چند هدفه با جمعیت ۱۵۰ و ۵۰۰ تکرار. به دلیل اینکه بهینه سازی انجام شده، سه هدفه است، جبهه پرتو، به صورت سه بعدی خواهد شد، اما به دلیل درک بهتر نوع وابستگی مشخصه‌ها، جبهه‌های پرتو به صورت دو به دو رسم شده‌اند.

جدول۶. ساختار عوامل جستجو (توپ‌ها در IPO) برای مقایسه‌گر دارای پیش تقویت‌کننده

پارامترهای بهینه شونده	ساختار توپ
پهنای M_1	ت۱
پهنای M_2-M_3	ت۲
پهنای M_4-M_5	ت۳
پهنای M_6-M_7	ت۴
پهنای M_8-M_9	ت۵
پهنای M_{10}-M_{11}	ت۶
پهنای M_{12}-M_{13}	ت۷
پهنای M_{14}-M_{15}	ت۸
پهنای M_{16}-M_{17}	ت۹
پهنای M_{18}-M_{19}	ت۱۰

بحث

در ادامه به بیان چند نکته که در مورد نتایج بیان شده می‌پردازیم:

۱- به دلیل استفاده از نرم‌افزار شبیه‌ساز Hspice و کتابخانه‌ی TSMC ۱۸۰ نانومتر، هیچ‌گونه تقریبی در مدل ترانزیستورها وجود ندارد. البته، اثراتِ طراحی بردِ مدارچاپی و اثرات پدها در نظر گرفته نشده است.

۲- در بهینه‌سازیِ چندهدفه، یک پاسخ به عنوان بهترین پاسخ معرفی نمی‌شود. بنابراین در جواب‌های بدست آمده نمی‌توان یک پاسخ را به عنوان بهترین طراحی برای تمام کاربردها بیان نمود. به عبارت دیگر، ضروریات طراحی یک وسیله‌ی کاربردی[1]، تعیین‌کننده‌ی بهترین پاسخ از میان طراحی‌های پیشنهادی در جبهه‌ی پرتو خواهد بود.

۳- ضروریات طراحی، مشخص‌کننده‌ی تعداد و نوع توابع هدف هستند. به عبارتی، طراح باید با توجه به اینکه مقایسه‌گر را در چه کاربردی می‌خواهد، تعداد و نوع توابع هدف را انتخاب کند. البته باید در نظر گرفت که، در یک الگوریتم فرااِبتکاری با فرض یکسان بودن پارامترهایِ دیگر الگوریتم (مانند: تعداد تکرارها، میزان جمعیت اولیه و ...)، هر چه تعدادِ توابع هدف کمتر باشد، عملکرد الگوریتم بهتر خواهد شد.

[1] Application

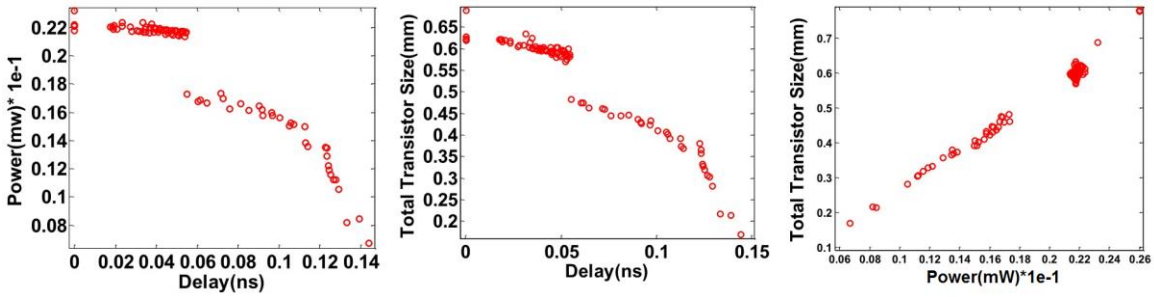
www.SID.ir

Archive of SID

۴- مشخصه‌هایی که می‌تواند در یک مقایسه‌گر با توجه به کاربرد اهمیت پیدا کند و به عنوان تابع هدف در نظر گرفته شود عبارتند از: سطح منبع تغذیه، نویز برگشتی، خازن گره خروجی، محدوده‌ی ولتاژ مدمشترک ورودی و ناپایداری.

یکی از مسائل اساسی در استفاده از ابزارهای طراحی با رایانه، توجه به میزان دقت مورد نیاز و طول زمان بهینه‌سازی نسبت به پیچیدگی مسئله است. بطور مثال، همانطور که در شکل ۵ دیده می‌شود، جبهه‌های پرتوی بدست آمده توانسته‌اند یک ذهنیت از رابطه میان پارامترهای توان، تاخیر و سایز را ایجاد کنند، لذا طراح می‌تواند به همین میزان دقت بسنده کرده و از اجرای مجدد بهینه‌سازی برای دست‌یابی به جبهه‌های پرتوی پیوسته‌تر خودداری کند. همان طور که مشاهده می‌شود، جبهه‌های پرتوی ردیف سوم در همین شکل، نسبت به ردیف دوم تغییرات چندانی ندارند؛ به عبارتی ابزار پیشنهادی در مواجهه با طراحی مقایسه‌گر قفلدار ساده با تنظیمات جمعیت ۱۰۰ و تکرار ۳۶۰ می‌تواند برطرف‌کننده‌ی نیازهای انجام یک طراحی بهینه و دقیق باشد. جدول ۹ تعداد شبیه‌سازی‌های صورت گرفته را (تعداد جمعیت و تکرار) به ازای مدت زمان اجرای بهینه‌سازی برای این مثال (مقایسه‌گر قفلدار ساده) نشان می‌دهد. از این جدول می‌توان فهمید که استفاده از الگوریتم‌های متفاوت می‌توانند زمان اجراهای متفاوتی داشته باشند. دلیل اصلی تفاوت زمان اجرای بین این الگوریتم‌ها، استفاده از مخزن (حافظه‌دار بودن) در الگوریتم MOIPO است. شکل ۷ جبهه‌های پرتوی بدست آمده برای NSGAII و MOIPO (هر دو در شرایط یکسان و با تنظیمات مشابه) را نشان می‌دهد. همان‌طور که مشاهده می‌شود، جبهه‌های پرتوی بدست آمده از الگوریتم فراابتکاری MOIPO پیوسته‌تر و در برخی از موقعیت‌ها علی‌الخصوص زمانی که تاخیر کمتر از ۰٫۱ نانوثانیه شده است، می‌باشد. از تحلیل همزمان جدول ۹ و شکل ۷، مشاهده می‌شود اگرچه زمان اجرای الگوریتم MOIPO بیشتر از NSGAII است، اما نتایج حاصله (جبهه‌های پرتو)، دارای پیوستگی بیشتر و مقادیر بهتری است.

در شبیه‌سازی مقایسه‌گر پیش تقویت‌کننده که نتایج آن در شکل ۶ نشان داده شده است، با توجه به اینکه پیچیدگی مسئله بیشتر شده، پیوستگی جبهه‌های پرتوی (با میزان یکسانی از توپ‌ها و تکرارها) نسبت به مثال اول کمتر است و طراح می‌تواند در صورت لزوم طراحی را با جمعیت و/یا تکرار بیشتر دوباره اجرا کند تا به نتایج مطلوب‌تری (جبهه پرتوی پیوسته‌تر) برسد.



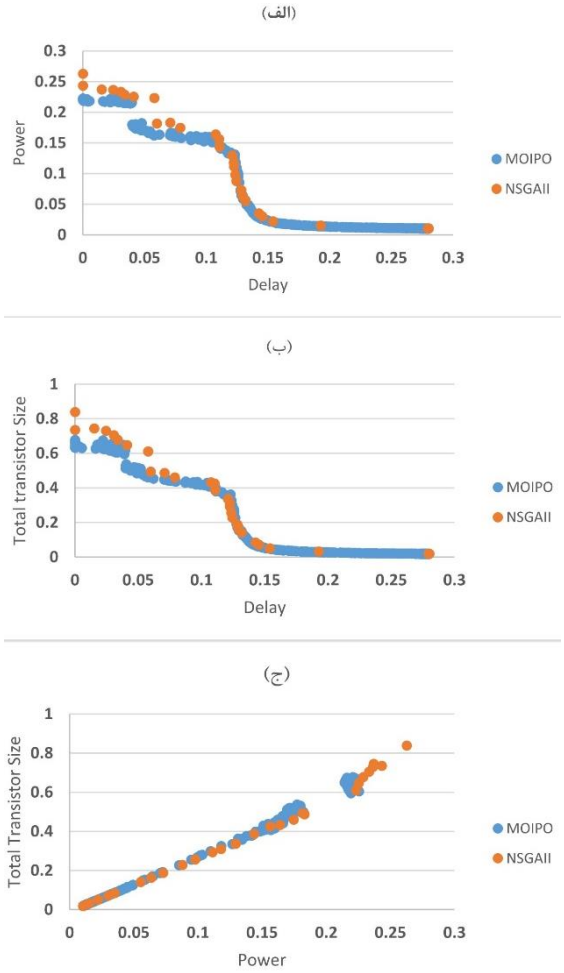
شکل۶. جبهه‌های پرتو بدست آمده از بهینه‌سازی مقایسه‌گر با پیش تقویت‌کننده برای سیستم IPO چند هدفه با جمعیت ۱۰۰ و ۳۶۰ تکرار. به دلیل اینکه بهینه‌سازی انجام شده، سه هدفه است، جبهه پرتو، به صورت سه بعدی خواهد شد، اما به دلیل درک بهتر نوع وابستگی مشخصه‌ها، جبهه‌های پرتو به صورت دو به دو رسم شده‌اند.

جدول۷. تجزیه و تحلیل گوشه‌های PVT برای مقایسه‌گر دارای پیش تقویت‌کننده

دو پاسخ منتخب از جبهه پرتو	TT		FF		SS		FS		SF	
	توان (W)	تاخیر (s)	توان (W)	تاخیر (s)	توان (W)	تاخیر (s)	توان (W)	تاخیر (s)	توان (W)	تاخیر (s)
W_1=0.40um $W_{2,3}$=0.40um $W_{4,5}$=20.04um $W_{6,7}$=38.08um $W_{8,9}$=7.52um $W_{10,11}$=28.06u $W_{12,13}$=43.76u $W_{14,15}$=19.26u $W_{16,17}$=28.82u $W_{18,19}$=28.22u	1.27e-4	3.01e-8	1.69e-4	3.01e-8	6.21e-5	3.02e-8	6.21e-5	3.02e-8	7.85e-5	3.01e-8
W_1=0.45um $W_{2,3}$=0.40um $W_{4,5}$=0.40um $W_{6,7}$=16.03um $W_{8,9}$=8.81um $W_{10,11}$=18.04u $W_{12,13}$=50.75u $W_{14,15}$=18.94u $W_{16,17}$=29.06u $W_{18,19}$=27.17u	1.33e-4	3.19e-8	1.90e-4	3.15e-8	6.02e-5	3.28e-8	1.22e-4	3.22e-8	1.09e-4	3.19e-8

www.SID.ir

Archive of SID



شـــکل۷. مقایســـه عملکرد الگوریتم NSGAII در مقابل MOIPO زمانی که جمعیت ۱۰۰ و تکرار ۳۶۰ در نظر گرفته شـده اسـت. (الف) مقایســه جبهه پرتوی توان و تاخیر (ب) مقایســه جبهه پرتوی تاخیر و مجموع ســایز ترانزیستورها (ج) مقایسه جبهه پرتوی توان و مجموع سایز ترانزیستورها.

نتیجه‌گیری

به دلایلی از جمله، حساس بودن مقایسه‌گرهای سرعت-بالا به عناصر پارازیتیکی، دقیق نبودن روابط طراحی، دقیق نبودن مدل‌های مداری مربوط به قطعات، مشکل بودن محاسبات دستی، نیاز به سعی و خطا در طراحی و متغیر با زمان بودن نقاط کار ترانزیستورها، نمی‌توان از پاسخ‌های ارائه‌شده توسط طراحی دستی اطمینان حاصل کرد. بنابراین، روشی پیشنهاد شد که با استفاده از نرم‌افزارِ شبیه‌ساز و سیستم بهینه‌سازی صفحات شیبدار، توانست محدودیت‌های ذکر شده را حذف کند. مزیت این CAD، ارائه یک یا چند جبهه پرتو است که طراح را از نوعِ همبستگیِ مشخصه‌ها آگاه کرده و طراحی را ساده می‌کند. استفاده از الگوریتم NSGAII و مقایسه آن با نتایج MOIPO، نشان داد که استفاده از الگوریتم‌های متفاوت، پاسخ‌های متفاوتی را ارائه می‌دهد.

جدول۸. ۱۰۰۰ اجرای شبیه‌سازی مونت کارلو بر روی دو پاسخ از جبهه پرتوی بدست آمده برای مقایسه‌گر پیش تقویت‌کننده

برخی از پاسخ‌های جبهه پرتو	توان مصرفی (W)		تاخیر (s)	
	۱-سیگما	میانگین	۱-سیگما	میانگین
W_1=0.40um $W_{2,3}$=0.40um $W_{4,5}$=20.04um $W_{6,7}$=38.08um $W_{8,9}$=7.52um $W_{10,11}$=28.06u $W_{12,13}$=43.76u $W_{14,15}$=19.26u $W_{16,17}$=28.82u $W_{18,19}$=28.22u	1.14e-6	9.45e-5	1.7e-13	3.02e-8
W_1=0.45um $W_{2,3}$=0.40um $W_{4,5}$=0.40um $W_{6,7}$=16.03um $W_{8,9}$=8.81um $W_{10,11}$=18.04u $W_{12,13}$=50.75u $W_{14,15}$=18.94u $W_{16,17}$=29.06u $W_{18,19}$=27.17u	9.3e-7	1.38e-4	6.69e-12	3.2e-8

جدول۹. میزان دقت به ازای زمان اجرای بهینه‌سازی در ابزار پیشنهادی

مدار بهینه شده	الگوریتم مورد استفاده	میزان دقت ابزار (جمعیت*تکرار)	زمان اجرا (دقیقه)
مقایسه‌گر قفلدار ساده	MOIPO	۹۰۰۰ شبیه‌سازی	۴۵
مقایسه‌گر قفلدار ساده	MOIPO	۳۶۰۰۰ شبیه‌سازی	۱۷۷
مقایسه‌گر قفلدار ساده	MOIPO	۷۵۰۰۰ شبیه‌سازی	۴۰۰
مقایسه‌گر قفلدار ساده	NSGAII	۹۰۰۰ شبیه‌سازی	۲۴
مقایسه‌گر قفلدار ساده	NSGAII	۳۶۰۰۰ شبیه‌سازی	۹۲
مقایسه‌گر قفلدار ساده	NSGAII	۷۵۰۰۰ شبیه‌سازی	۲۰۰

مراجع

[1] F. Olivera and A. Petraglia, A computer-aided approach for voltage reference circuit design, *2015 IEEE 6th Latin American Symposium on Circuits & Systems (LASCAS)*, Montevideo, 24-27 Feb. 2015, pp. 1–4.

[2] A. P. Chandrakasan and R. W. Brodersen, *Low power digital CMOS design*, 1st ed., Springer Science & Business Media, 2012, New York, pp. 409.

[3] J. Huijsing, R. J. van de Plassche, and W. Sansen, *Analog circuit design: operational amplifiers, analog to digital converters, analog computer aided design*, 1st ed., Springer Science & Business Media, 2013,

www.SID.ir

Archive of SID

[16] D. Schinkel, E. Mensink, E. Klumperink, and E. Tuijl, A double-tail latch-type voltage sense amplifier with 18ps setup+ hold time, in *IEEE International Solid-State Circuits Conference*, San Francisco, 11-15 Feb. 2007, pp. 314–315.
[17] H. Jeon and Y.-B. Kim, A CMOS low-power low-offset and high-speed fully dynamic latched comparator, in *23rd IEEE International SOC Conference*, Las Vegas, 27-29 Sep. 2010, pp. 285–288.
[18] M. Miyahara, Y. Asada, D. Paik, and A. Matsuzawa, A low-noise self-calibrating dynamic comparator for high-speed ADCs, in *IEEE Asian Solid-State Circuits Conference*, Fukuoka, 3-5 Nov. 2008, pp. 269–272.
[19] S. Babayan-Mashhadi and R. Lotfi, Analysis and design of a low-voltage low-power double-tail comparator , *IEEE Trans. Very Large Scale Integr. Syst.*, 2014, vol. 22, no. 2, pp. 343–352.
[20] M. Miyahara and A. Matsuzawa, A low-offset latched comparator using zero-static power dynamic offset cancellation technique, in *IEEE Asian Solid-State Circuits Conference*, Taipei, 16-18 Nov. 2009, pp. 233–236.
[21] J. Yang, X. Cheng, Y. Guo, Z. Zhang, and X. Zeng, A novel low-offset dynamic comparator for high-speed low-voltage pipeline ADC, in *10th IEEE International Conference on Solid-State and Integrated Circuit Technology*, Shanghai, 1-4 Nov. 2010, pp. 548 - 550.
[22] D.-S. Khosrov, A new offset cancelled latch comparator for high-speed, low-power ADCs, in *IEEE Asia Pacific Conference on Circuits and Systems*, Kuala Lumpur, 6-9 Dec. 2010, pp. 13–16.
[23] Y. Huang, H. Schleifer, and D. Killat, Design and analysis of novel dynamic latched comparator with reduced kickback noise for high-speed ADCs, in *European Conference on Circuit Theory and Design,* Dresden, 8-12 Sep. 2013, pp. 1–4.
[24] B.-J. Kuo, B.-W. Chen, and C.-M. Tsai, A 0.6 V, 1.3 GHz dynamic comparator with cross-coupled latches, in *International Symposium on VLSI Design, Automation and Test*, Hsinchu, 27-29 April 2015, pp. 1–4.
[25] A. Malik and P. Srivastava, New improved high speed low power double tail comparator design for 2.5 GHz input signal, in *IEEE Students' Technology Symposium (TechSym),* Kharagpur, 2 March 2014, pp. 205–209.
[26] Y. Lin, K. Doris, H. Hegt, and A. van Roermund, A dynamic latched comparator for low supply voltages down to 0.45 V in 65-nm CMOS, in *IEEE International Symposium on Circuits and Systems*, Tallinn, 18-20 April 2012, pp. 2737–2740.
[27] H. J. Jeon, Low-power high-speed low-offset fully dynamic CMOS latched comparator, Master thesis, Northeastern University, 2010, Dordrecht, pp. 452.
[4] M. Fakhfakh, Y. Cooren, A. Sallem, M. Loulou, and P. Siarry, Analog circuit design optimization through the particle swarm optimization technique, *Analog Integr. Circuits Signal Process.*, 2010, vol. 63, no. 1, pp. 71–82.
[5] J. Ramos, K. Francken, G. G. E. Gielen, and M. S. J. Steyaert, An efficient, fully parasitic-aware power amplifier design optimization tool, *IEEE Trans. Circuits Syst. I Regul. Pap.*, 2005, vol. 52, no. 8, pp. 1526–1534.
[6] P. E. Allen and D. R. Holberg, *CMOS analog circuit design*, 3rd ed., Oxford Univ. Press, 2011, USA, pp. 590.
[7] M. Dorigo, M. Birattari, C. Blum, M. Clerc, T. Stützle, and A. Winfield, *Ant Colony Optimization and Swarm Intelligence: 6th International Conference*, Springer, 2008, Brussels, pp. 416.
[8] E. Afacan, G. Berkol, G. Dundar, A. E. Pusane, and F. Baskaya, An analog circuit synthesis tool based on efficient and reliable yield estimation, *Microelectronics J.,* 2016, vol. 54, pp. 14–22,.
[9] J. Park, K. Choi, and D. J. Allstot, Parasitic-aware RF circuit design and optimization, *IEEE Trans. Circuits Syst. I Regul. Pap.*, 2004, vol. 51, no. 10, pp. 1953–1966.
[10] M. Chu and D. J. Allstot, Elitist nondominated sorting genetic algorithm based RF IC optimizer, *IEEE Trans. Circuits Syst. I Regul. Pap.*, 2005, vol. 52, no. 3, pp. 535–545.
[11] K. Suzuki, N. Takai, M. Kato, H. Seki, Y. Sugawara, and H. Kobayashi, Comparator circuits automation by combination of distributed genetic algorithm and HSPICE optimization, in *2015 IEEE 11th International Conference on ASIC (ASICON)*, Chengdu, 3-6 Nov. 2015, pp. 1–4.
[12] B. Liu, Y. Wang, Z. Yu, L. Liu, M. Li, Z. Wang, J. Lu, and F. V Fernández, Analog circuit optimization system based on hybrid evolutionary algorithms, *Integr. VLSI J.*, 2009, vol. 42, no. 2, pp. 137–148.
[13] M. Nomura, Y. Ikenaga, K. Takeda, Y. Nakazawa, Y. Aimoto, and Y. Hagihara, Delay and power monitoring schemes for minimizing power consumption by means of supply and threshold voltage control in active and standby modes, *IEEE J. Solid-State Circuits*, 2006, vol. 41, no. 4, pp. 805–814.
[14] B. Goll and H. Zimmermann, *Comparators in Nanometer CMOS Technology*, 1st ed., Springer, 2015, Berlin Heidelberg, pp. 250.
[15] B. Goll and H. Zimmermann, A comparator with reduced delay time in 65-nm CMOS for supply voltages down to 0.65 V, *IEEE Trans. Circuits Syst. II Express Briefs*, 2009, vol. 56, no. 11, pp. 810–814.

www.SID.ir

Archive of SID

Boston, pp. 64.
[28] E. K. Burke, M. Gendreau, M. Hyde, G. Kendall, G. Ochoa, E. Özcan, and R. Qu, Hyper-heuristics: A survey of the state of the art, *J. Oper. Res. Soc.*, 2013, vol. 64, no. 12, pp. 1695–1724.
[29] F. Glover and M. Laguna, *Tabu Search*∗, Springer, 2013, New York , pp. 3261-3362.
[30] E. Aarts, J. Korst, and W. Michiels, Simulated annealing, in *Search methodologies*, Springer, 2014, USA, pp. 265–285.
[31] M. G. C. Resende and C. C. Ribeiro, GRASP: greedy randomized adaptive search procedures, in *Search methodologies*, Springer, 2014, USA, pp. 287–312.
[32] P. Hansen and N. Mladenović, Variable neighborhood search, in *Search methodologies*, Springer, 2014, USA, pp. 313–337.
[33] B. Xing and W.-J. Gao, Imperialist competitive algorithm, in *Innovative Computational Intelligence: A Rough Guide to 134 Clever Algorithms*, Springer, 2014, Switzerland, pp. 203–209.
[34] X.-S. Yang and X. He, 'Firefly algorithm: recent advances and applications', *Int. J. Swarm Intell.*, 2013, vol. 1, no. 1, pp. 36–50.
[35] U. Aickelin, D. Dasgupta, and F. Gu, 'Artificial immune systems', in *Search Methodologies*, Springer, 2014, USA, pp. 187–211.
[36] A. Abdolalipour and A. Alibabaee, 'Harmony Search algorithm', *Int. J. Acad. Res. Appl. Sci.*, 2012, vol. 1, no. 3, pp. 13–16.
[37] M. H. Mozaffari, H. Abdy, and S.-H. ZAHIRI, 'IPO: AN INCLINED PLANES SYSTEM OPTIMIZATION ALGORITHM', *Comput. Informatics*, 2016, vol. 35, no. 1, pp. 222–240.
[38] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, GSA: a gravitational search algorithm, *Inf. Sci. (Ny).*, 2009, vol. 179, no. 13, pp. 2232–2248,.
[39] N. Sayyadi, Designing Optimum Neural Networks Using Multi-objective Optimization Algorithms,Master thesis, University of Birjand, 2014.
[40] E. Yaqubi, S. H. Zahiri, Optimum design of a double-tail latch comparator on power, speed, offset and size, *Analog Integr. Circuits Signal Process.*, (online) 2016, pp. 1–11.

www.SID.ir

Archive of SID

www.SID.ir